Proficy™
Logic Developer - PLC

GETTING STARTED

Version 5.50
September 2005
GFK-1918G

Disclaimer of Warranties and Liability

The information contained in this manual is believed to be accurate and reliable. However, GE Fanuc Automation Americas, Inc. assumes no responsibilities for any errors, omissions or inaccuracies whatsoever. Without limiting the foregoing, GE Fanuc Automation Americas, Inc. disclaims any and all warranties, expressed or implied, including the warranty of merchantability and fitness for a particular purpose, with respect to the information contained in this manual and the equipment or software described herein. The entire risk as to the quality and performance of such information, equipment and software, is upon the buyer or user. GE Fanuc Automation Americas, Inc. shall not be liable for any damages, including special or consequential damages, arising out of the use of such information, equipment and software, even if GE Fanuc Automation Americas, Inc. has been advised in advance of the possibility of such damages. The use of the information contained in the manual and the software described herein is subject to GE Fanuc Automation Americas, Inc. standard license agreement, which must be executed by the buyer or user before the use of such information, equipment or software.

Notice

GE Fanuc Automation Americas, Inc. reserves the right to make improvements to the products described in this publication at any time and without notice.

We want to hear from you. If you have any comments, questions, or suggestions about our documentation, send them to the following e-mail address:

doc@gefanuc.com

Contents

1 Welcome

Congratulations on your purchase of Proficy™ Logic Developer - PLC, the PACSystems™ Controller and GE Fanuc PLC programming component of Proficy™ Machine Edition automation software.[1]

This software package provides all the tools necessary to create powerful control applications. Logic Developer - PLC provides a way to configure your PLC hardware or remote I/O, create and edit logic, upload and download projects, and monitor and debug the execution of control programs. Projects can be imported from Logicmaster, VersaPro, and CimplicityControl folders.

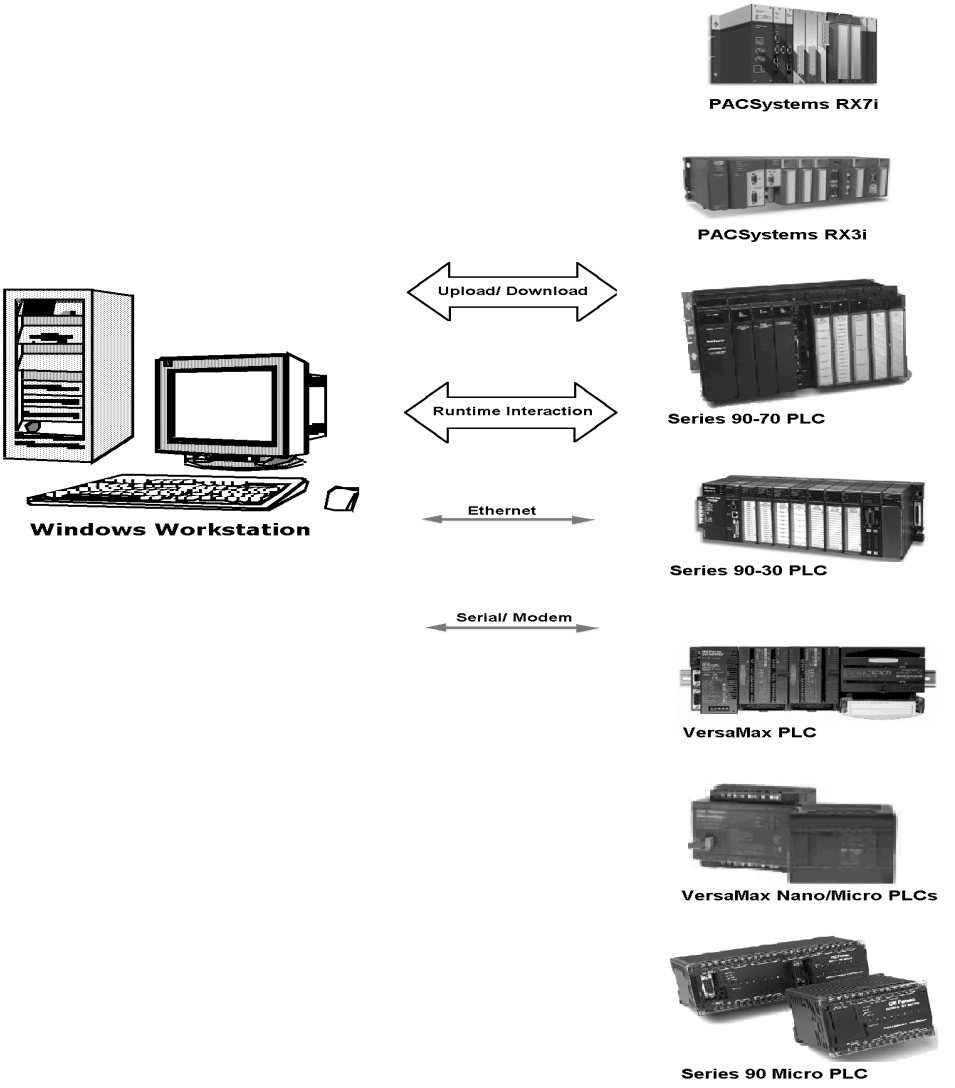
Hosted in the Machine Edition environment, Logic Developer - PLC takes advantage of a powerful set of common programming tools (see page 17). The same tools can be applied to Logic Developer - PC (PC Control), Logic Developer - State, View and Motion components, providing a single programming environment. The Machine Edition environment unites and organizes components, providing data sharing and networked operation.

The following features are included in this version of Logic Developer - PLC:

- Hardware Configuration. See page 49.
- LD Editor. See page 83.
- FBD Editor. See page 97.
- IL Editor. See page 108.
- ST Editor. See page 115.
- C Blocks. See page 123.
- C Programs. See page 125.
- PLC Motion Editor. See page 133.
- Local Logic Editor. See page 137.
- CAM Editor. See page 142.

1. "PLC" is used throughout this manual to include PACSystems™ Controllers, Series 90™ PLCs, and VersaMax™ PLCs.

Proficy™ Logic Developer - PLC Environment



SYSTEM REQUIREMENTS

To use Logic Developer - PLC and its tools, you require the following:

- Windows® XP Professional (Service Pack 1 recommended.)

  - or -

  Windows 2000 Professional (Service Pack 3 recommended.)

  - or -

  Windows NT version 4.0 with service pack 6.0a or later

  Note: Windows XP Tablet PC Edition is not supported.
- Internet Explorer™ version 5.5 with Service Pack 2 Q810847 or later, or Internet Explorer version 6.0 Service Pack 1 or later.

  Note: You must install Internet Explorer before installing Machine Edition.)
- TCP/IP Network protocol-based computer (if you use an Ethernet connection).
- Requirements for processor speed and memory are as follows:
  - Minimum: 1 GHz Pentium-based processor with 256 MB RAM
  - Recommended: 2 GHz Pentium-based processor with 512 MB RAM
  - **Note:** For projects with more than 75,000 variables or variable elements, at least 1 GB of RAM is *strongly* recommended.
- 150-750 MB free hard disk space depending on the installed products.
- 110 MB additional hard disk space, plus 40 MB during installation, for Microsoft .NET Framework 1.1. This space is not needed if .NET Framework 1.1 is already installed.

  Tip: If the Microsoft .NET Framework is not yet installed, it is automatically included during installation of Machine Edition.

  Note: After installation of Machine Edition and the .NET Framework is complete, do not attempt to install an older or beta version of the .NET Framework. This will cause the newest version of the .NET Framework to fail.
- 200 MB hard disk space for sample projects (optional).
- Additional hard disk space for projects and temporary files.

INSTALLATION

For last-minute information, release notes, and supported hardware lists for Machine Edition products, see the Important Product Information (IPI) document on the CD. There are several ways to view this document

- When installing Machine Edition, select **Important Product Information** on the initial Launcher screen.
- When running Machine Edition, from the Help menu, choose **Important Product Information**.

If you have a previous version of Machine Edition installed on your computer, you must uninstall it before installing the latest version. You can do so by selecting the "Uninstall CIMPLICITY Machine Edition" option from the CIMPLICITY Machine Edition Start menu group. All of your existing projects, settings, and authorizations are preserved following an uninstall operation.

To install Logic Developer - PLC

1. Insert the Machine Edition CD into your CD-ROM drive.

   By default, the setup program automatically starts. If the setup program does not automatically start, run *Setup.exe* in the root directory of the CD.
2. Click **Install** to start the install process.
3. Follow the instructions as they appear on the screen.

   Note: See the IPI Release Notes for information on troubleshooting installation problems.

PRODUCT AUTHORIZATION

Before you can start developing projects in with Logic Developer - PLC, you must authorize the software. If you do not do so, you will have unrestricted use of the software's features only for a short trial period. The authorization process takes only a few moments and enables you to take advantage of any product support for which you qualify.

There are two types of authorization available: *software key* authorization and *hardware key* authorization. Machine Edition products can use a mixture of authorization types on a single computer.

To see which products you are currently authorized for, on the Machine Edition Help menu, point to **Product Authorization** and choose **Authorize Software**. This displays the Product Authorization dialog box. If you have a hardware key plugged in a USB port, you can select the Show Hardware Keys check box to view the authorizations provided by that key.

Hardware Key Authorization

Hardware key authorization requires a special USB hardware key. This key contains settings that allow use of specific Machine Edition products. Hardware keys can be used to authorize Machine Edition development and runtime software.

Hardware key authorization is available only for Windows 2000 or Windows XP and requires a free USB port on your computer.

To authorize Machine Edition products with a Hardware key

1. Locate a free USB port on your computer.

   Typically, the USB port is located on the front or back of your tower case, or in the side of a laptop computer. In some cases, a USB port can be found on your computer's monitor.

2. Plug the hardware key into the USB port.

   The first time you plug the key into a USB port, a progress dialog box indicates that Windows has detected new hardware and is updating its system settings. After the dialog box disappears, the settings stored in the hardware key are active and you can use the Machine Edition products that it authorizes.

   You must leave the hardware key in the port while using Machine Edition. If you remove the key from the port, authorization for those products vanishes, though you can still use products previously authorized by means of a software key.

To move authorization to another computer, simply remove the hardware key from the source computer and plug it into a USB port on the destination computer.

Software Key

When authorizing Machine Edition products with a software key, you need to contact us by telephone, fax, or e-mail. If you want to authorize your software by telephone, note that authorization personnel are available only during regular business hours (between 6 a.m. and 6 p.m. MST).

Software key authorization is specific to a single computer. If you want to work with Machine Edition on a different computer, you must move the authorization to that second computer (see page 7).

To authorize a copy of Machine Edition with a software key

1. Have your serial numbers ready.
   The serial numbers can be found on the License Key sheet that came with your product.
2. Run the Product Authorization program from the **Start** menu > **Programs** > **GE Fanuc** > **Proficy Machine Edition** > **Product Authorization**.
   The Product Authorization dialog box appears.
3. Click **Add**.
4. In the dialog box that appears, select the medium with which you are authorizing: **Internet**, **Phone/Fax/E-mail**, or **Floppy Disk Transfer**. Click **Next**.
   If you choose the Internet option, follow the instructions on the web site.
   If you choose the Phone/Fax/E-mail option, proceed to step 5.
   If you choose the Floppy Disk Transfer option, ensure you have an authorization disk before proceeding.
5. Fill in the fields in the dialog box.
   Fields that are identified with an asterisk (*) must be filled in.
6. If authorizing by:
   - **Phone**, click **Phone/Fax** and call the number on the screen to receive one or more new key codes.
   - **Fax**, click **Phone/Fax.** In the dialog box that appears, click **Print FAX**. Fax the Product Authorization Request to us, using our fax number on the printout. We will reply by fax with your new key code(s).
   - **E-mail**, click **Send E-mail**. In the dialog box that appears, click **Authorize** to e-mail us. We will reply by e-mail with your new key code(s).

Product Authorization is complete after you enter the new key code and it has been accepted. Depending on the products you have purchased, you may need to run the Product Authorization program a number of times.

To move the authorization to another computer

You can run the software only on the computer that has Product Authorization. To develop your projects on a different computer, you need to complete the following steps to move the authorization from one computer to another.

1. Install Logic Developer - PLC on the computer that the authorization will be moved to. Run the Product Authorization program: **Start menu** > **Programs** > **GE Fanuc** > **Proficy Machine Edition** > **Product Authorization**.
   The Product Authorization dialog box appears.
2. Click **Move**, and then click **OK**.
   There is a Target Site Code on the top right hand side of the screen. Write down this site code carefully. It must be correct for the move to work. You need the Target Site Code when you move the authorized software from the source computer.
3. Click **Authorize by disk**.
   At this point, you need to go to the source computer that has the authorized software, and move the authorization to a disk.
4. From the source computer, run the Product Authorization program.
5. Click **Move** and then click **OK**. Enter the Target Site Code that you wrote down from Step 2 and click **Next**. Verify that the site code is correct and click **OK**.
6. Insert a blank formatted floppy disk into the floppy drive and click **Next**.
   The authorization code will be moved to the disk and a dialog box should appear telling you it was successful.
7. Click **OK**.
8. Go back to the computer to which you are moving the authorization and insert the floppy disk.
   The screen that is asking for an authorization disk should be displayed.
9. Click **Next**.
10. Click **Finish**.
    A screen should appear telling you the move was successful.
11. Click **OK**.

    The authorization has now been moved to the new computer.

TECHNICAL SUPPORT

Support is available to registered users at no charge for 90 days after purchase. A support agreement can be purchased from your local GE Fanuc distributor if extended support is required.

If problems arise that cannot be solved by using the information in your product manual, online Help system, or the Proficy™ GlobalCare knowledge base, contact us by telephone, fax, or e-mail.

When contacting us, call from a telephone near your computer and have your Machine Edition software running. Have the following information handy to help us assist you as quickly as possible:

- The serial number from your installation License Key sheet, and the Product name and version number from the **Help**>**About** dialog box.
- The brand and model of any hardware in your system.
- Operating system and version number.
- The steps you performed prior to the problem occurring.

North America

Support Hotline: 1-800-GEFANUC (1-800-433-2682) **Fax:** (780) 420-2049
Internet: http://www.globalcare.gefanuc.com **E-mail:** support@gefanuc.com
Comments about our manuals and help: doc@gefanuc.com
Mailing Address: GE Fanuc
2700 Oxford Tower, 10235 - 101 St.
Edmonton, AB, Canada, T5J 3G1

South America

Telephone: +58 (261) 760 2862 **Fax:** +58 (261) 765 0909
Internet: http://www.gefanuc.com (visit our Portuguese web site at www.gefanuc.com.br)
E-mail: luis.wilhelm@gefanuc.com
Mailing Address: GE Fanuc Automation Latin America
Calle 120 con Av. 17, Los Haticos -GE Turbimeca
Maracaibo, Venezuela

Europe, Middle East, and Asia

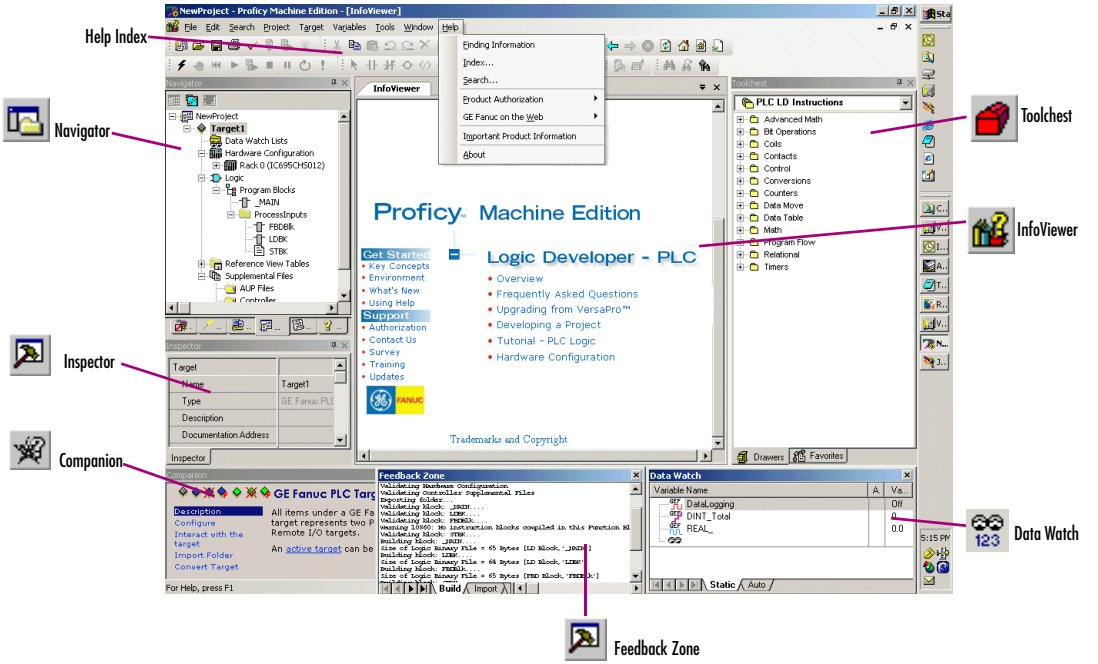
Telephone: 00 800 1 GE FANUC **Fax:**
81-3-5405-7555 (Japan) 81-3-5405-7550 (Japan)
0086-21-32224555 x200 (China) 0086-21-62793066 (China)
Internet: For up-to-date contact information, visit www.gefanuc-europe.com and click on "Offices and Services".
For up-to-date technical information visit www.gefanuc.com/support.
E-mail: support.emea@gefanuc.com

2 Proficy Machine Edition

Proficy Machine Edition offers you a complete solution for the development of automation applications, in one package. Machine Edition features an integrated development environment and tools that enable you to spend more time building applications and less time learning the software. All Machine Edition products are fully integrated into the environment and interact with each other.

- They share the same set of tools providing a consistent interface throughout the development process.
- They feature full drag-and-drop capabilities between tools and editors.
- They feature a true scalable solution. You have the choice of what type of PLC your projects run on.

The following illustrates a Machine Edition project:



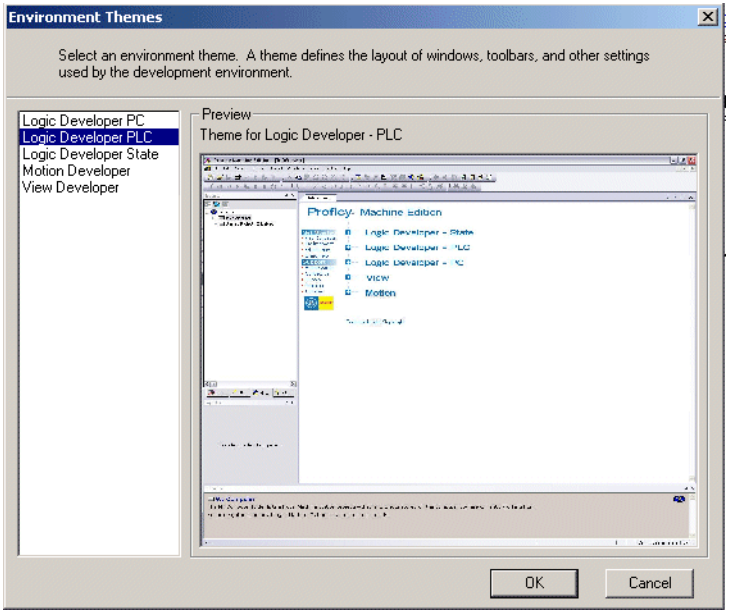
QUICK START

Machine Edition makes it easy to get started developing a project.

To start Machine Edition

1. Click Start, point to **Programs**, point to **GE Fanuc**, point to **Proficy Machine Edition**, and then choose **Proficy Machine Edition**.

   After Machine Edition initializes, the Environment Themes dialog box appears.

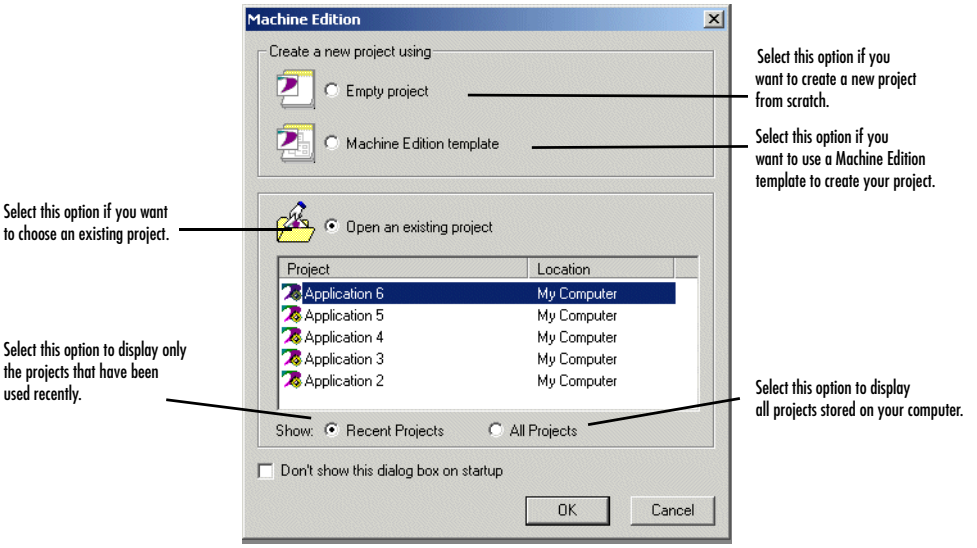


   Note: The Environment Themes dialog box appears automatically the first time you start Machine Edition. To change the theme later, choose **Apply Theme** from the **Windows** menu.

2. Choose the Logic Developer - PLC theme.
3. Click **OK**.

   When you open a project, the appearance of your Machine Edition screen matches the preview displayed in the Environment Themes dialog box.

The Machine Edition dialog box appears.



4. Select the appropriate option to open a project. The **Open an existing project** option is selected by default.

   Notes:

   - If you select either the **Empty project** option or the **Machine Edition template** option, the New Project dialog box appears and you can continue creating a new project (see page 15).
   - If you select the **Open an existing project** option, you can also select either the **Recent Projects** option or the **All Projects** option. The **Recent Projects** option is selected by default.

5. If you selected the **Open an existing project** option, select from the list the project that you want to open.
6. If you want, select the **Don't show this dialog box on startup** option.
7. Click **OK**.

   Your project opens in the Machine Edition environment that you specified in the Environment Themes dialog box.

PROJECTS

You can create and edit Machine Edition projects by using products such as View, Logic Developer - PC, Motion, Logic Developer - State, and Logic Developer - PLC. These products share Machine Edition tools to provide a high level of integration between the various parts of your project.

You can import folders created with Logicmaster, CimplicityControl, or VersaPro.

With Logic Developer - PLC, you can build multiple projects to suit your specific requirements.

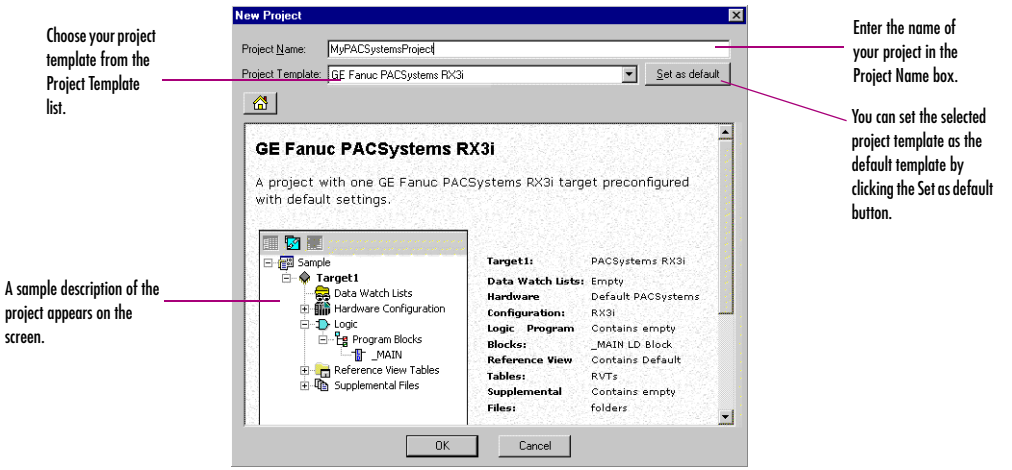
To create a new project using a template

Before creating a project, there are some things you should know:

- The primary components your project will contain.
- The GE Fanuc controller your project will run on.

1. From the **File** menu, choose **New Project**, or click on the File toolbar.

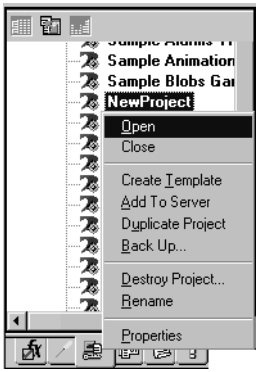
   The New Project dialog box appears.



2. Choose a Project Template that suits your needs.
3. Enter a descriptive Project Name.
4. Click **OK**.

   Your project opens in the Machine Edition environment.

To open an existing project for editing



Navigator: Manager tab

1. Open the Navigator and select the Manager tab.

   A list of projects appears.

2. Do one of the following:

   - Right-click a project and choose **Open**.

     - or -

   - Double-click a project.

   Your project loads and is ready for editing.

> **Want to know more?** In the online help index, look up "Projects".

To import a folder

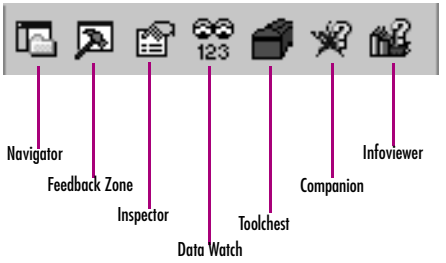
1. Open the Navigator and select the Project tab.
2. Select the target that you want to import the folder into.
3. Right-click the target, point to **Import**, and choose the folder type.
4. In the dialog box that appears, navigate to and double-click the folder you want to import.

> **Want to know more?** In the online help index, look up "importing CimplicityControl folder", "importing Logicmaster folder", or "importing VersaPro folder".

TOOLS

Project development is supported by the Machine Edition tools. Each tool is opened by means of a button on the Tools toolbar.

A description of each tool's function is provided below.



Navigator

The Navigator is a docking tool window containing a set of tabs. Each tab displays information about your development system in a hierarchical tree structure similar to Windows Explorer. The available tabs depend on which Machine Edition products you have installed and what kind of work you are developing or managing. The Project tab shows the overall organization of your application.

Feedback Zone

The Feedback Zone window is a docking window used to display several types of output information generated by Machine Edition components. This interactive window uses category tabs to organize the output generated from the Machine Edition products you have installed.

Inspector

The Inspector lists the properties and current settings for a selected object or element. You can edit these properties directly in the Inspector. When you select several objects, the Inspector window lists the properties common to all of them. The Inspector window provides a simple method of viewing and setting properties for all Machine Edition objects.

Data Watch

The Data Watch tool is a run-time debugging tool that enables you to monitor and edit the values of variables. This tool is useful while working online to a target. With the Data Watch tool, you can monitor individual variables or user-defined lists of variables. Data Watch lists can be imported, exported, or saved with a project.

Toolchest

The Toolchest is a powerful storehouse of objects you can add to your project. You can drag most items directly from the Toolchest into Machine Edition editors. You can choose from predefined objects or create your own reusable fxClasses. The Toolchest adds true object-oriented capability to Machine Edition.

Companion

The Companion provides useful tips and information while you work. While the Companion is open, it tracks your moves and displays help on whatever item is currently selected in the Machine Edition environment. It is context-sensitive and displays a description of whatever you click on the Machine Edition screen.

InfoViewer

The InfoViewer is an embedded Web browser used mainly to display the following

- Machine Edition help.
- Machine Edition reports.
- The documentation associated with a project or target.

If you are familiar with Internet Explorer or Netscape Navigator, then you are already familiar with the basic InfoViewer interface. Like the Companion, the InfoViewer is context-sensitive. Press F1 to get help on any item you select in the Machine Edition environment.

A table of contents is found in the InfoView tab of the Navigator.

Want to know more? In the online help index, look up "Tools: an Overview".

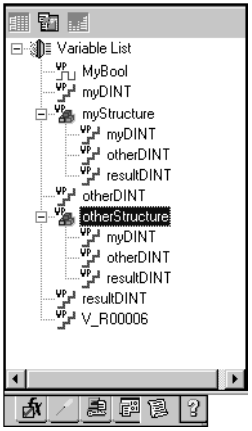
Using docking markers

If the Show Docking Markers option is enabled, you can use docking markers to help dock a tool window to a desired location.

As you drag a dockable window, a series of docking markers appear, indicating valid docking locations. Docking markers appear as a series of blue arrows. Move the mouse over a docking marker to preview how the window would be docked when using that marker. Release the mouse button while hovering over a marker to use that location.

	These markers dock tools to the edge of an existing tool or to the Machine Edition Environment window.
	This marker "stacks" the tool overtop an existing tool window, forming a series of tab-docked windows. You can then switch between tab-docked tools by clicking the tabs that appear at the bottom of the tab-docked windows.

VARIABLES



Navigator: Variables tab. This tab displays all the variables used in a project.

A variable is a named storage space for data. That is, the variable name represents a memory location in the target PLC. The way in which the variable represents a memory location is determined by the value of the variable's Ref Address property. A variable's Ref Address property can be set to any of the following:

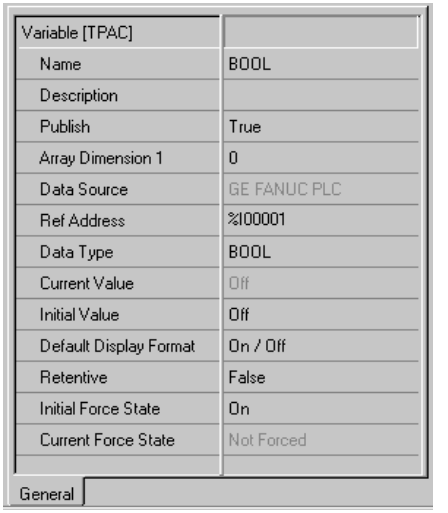
- **A reference address**, for example, %R00001. This is the GE Fanuc PLC memory location that contains the variable's value. This can be an I/O register or an internal memory register on the PLC.
- **A blank**. On a PACSystems, if you leave a variable's Ref Address property blank, the variable is a symbolic variable. Machine Edition handles the mapping for you in a special portion of PACSystems user memory space. On Series 90 and VersaMax PLCs, you cannot leave the Ref Address property blank.
- **An I/O variable address, for example %IX0.6.0.1**. (PACSystems only.) This represents the PACSystems Hardware Configuration terminal that contains an I/O variable's value. For example, this can be a physical discrete or analog I/O point on a module or Genius device, a discrete or analog status returned by the module, or global data.
- **A name**. This makes the current variable an alias variable of the name, that is, a variable that points to the same memory location as the name. The following names may be valid:
  - The name of a parameterized LD block parameter (provided the alias variable is local to the same block)
  - The name and index of an array element, for example, MyComplexArray[255,3]
  - The name of a structure element, for example, MyStructure.MyElement
  - The name of another variable that is not used as an alias variable.

Want to know more? In the online help index, look up "Reference Address (GE Fanuc PLCs)", "Symbolic Variables", "I/O Variables", and "Alias Variables".

Arrays and compound structure variables are supported by Machine Edition. Variable definitions can be imported from and exported to a variety of file types. You can edit your variables in a spreadsheet and then import them. Variables used on PACSystems targets must be externally published in order to be viewed in Proficy View.

Want to know more? In the online help index, look up "Arrays in GE Fanuc PLC targets", "Structure Data Types", "Importing variables from a file", and "Publishing PACSystems variables".

The Ref Address and other properties of a variable, such as Data Type, are configured in the Inspector. The following is an illustration of the Inspector displaying a typical set of variable properties.

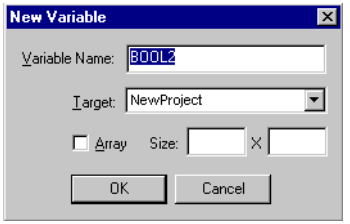


To create a variable

New Variable menu

1. In the Variables tab of the Navigator, right-click **Variable List**, point to **New Variable** and then choose the data type of the variable.

   The New Variable dialog box appears.



2. Type a name for the variable.

   Variable names can range from 1 through 32 characters, begin with a letter or the $ character, contain upper or lower case letters, use numbers between zero and nine, and use the underscore character ("_").

3. Select the target the variable will be used in.
4. If the new variable is an array, select the Array check box and enter the size(s).
5. Click **OK**.

   The new variable appears in the list in the Variables tab.

To map a variable to PLC memory or alias a variable

There are two ways to map a variable to PLC memory or alias any variable.

First method:

1. In the Variables tab of the Navigator, right-click a variable and choose **Properties**.

   The Inspector displays the properties of the variable.

2. In the **Ref Address** field, enter a reference address to map the variable or enter the name of another existing variable, a parameterized LD block parameter, an array element, or a structure element to alias the variable.
   - Spell out the exact reference address, for example, %R00123 or 123R (in either case this maps the variable to %R00123), or enter only the memory area, for example %R. This maps the variable to the next available address in that memory area. For example, if %R00122 is the last address used by a 16-bit variable, entering %R maps the variable to %R00123.
   - Spell out the name of another variable, or of a parameterized LD block parameter, for example, MyVarWithALongName, or Abc. Or provide the array's name and the element's number, for example, MyArray[5]. Or provide the structure element's name, for example, MyStructure.MyElement.

Notes (PACSystems only)

- If you leave the Ref Address property of a variable blank, the variable is a symbolic variable.
- You can enter an I/O variable address to make the variable an I/O variable, but it is easier to map an I/O variable in the Terminals tab of a module. See "I/O variables", on page 60.

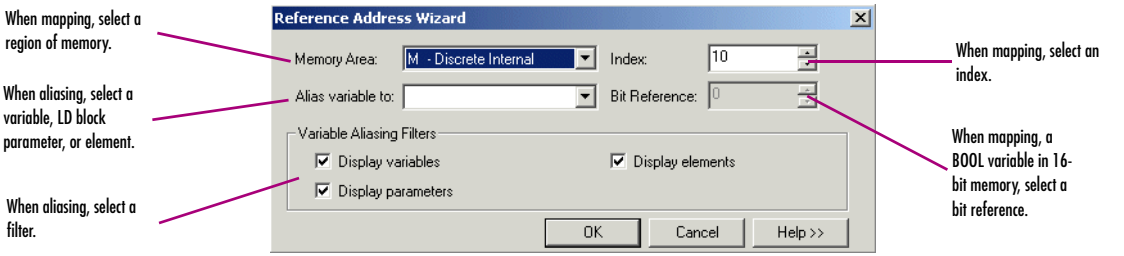
Second method:

1. In the Variables tab of the Navigator, right-click a variable and choose **Properties**.

   The Inspector displays the properties of the variable.

2. Click the **Ref Address** field and then click the ••• button.

   The Reference Address Wizard appears.



3. Do one of the following:

   To map a variable to PLC memory

   a. From the **Memory Area** list, choose a region of PLC memory.

   b. In the **Index** box, enter an index from the beginning of the region.

   c. Select a bit reference.

   The **Bit Reference** box is available only if you're mapping a BOOL variable to 16-bit memory area on a PACSystems.

   d. Click **OK**.

   Your variable is mapped to the reference address.

   - or -

   To alias a variable

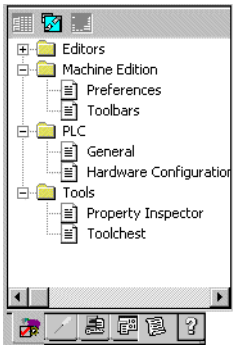
   a. In the **Variable Aliasing Filters** section, select the appropriate check boxes for the filters you want to use.

   b. In the **Alias variable to** box, select what you want to alias the variable to.

   c. Click **OK**.

Want to know more? In the online help index, look up "Bit references in non-BOOL variables".

OPTIONS



Navigator: Options tab

The Options tab of the Navigator contains option and preference settings. Options are organized into folders and pages. Click ⊞ or ⊟ to expand or collapse folders. Examples:

Editors > Ladder > View > Coil Justification: The default column in which coils are placed; also the column where the right power rail resides. The default is 10.

Machine Edition > Preferences > Visual Style: The visual style used for the Machine Edition environment, including the appearance of toolbars, tool windows, and menus. You can choose from several visual styles based on different versions of Microsoft products and the Windows operating system: Classic (based on Windows 2000), XP, 2003, and 2005.

PLC > General > Duplicate Addresses: Indicates whether to prevent, warn about, or ignore mapping two variables of the same data size (1, 8, 16, or 32 bits) and length (array size) to the same reference address.

PLC > Hardware Configuration > New Reference Assignment: The way in which default reference addresses are assigned when modules are added to the Hardware Configuration.

To set options and preferences

1. In the Options tab of the Navigator, expand an options folder and right-click a page within the folder to display the options in the Inspector.
2. In the Inspector, edit the option's settings as needed.

   TIP: When you click an option, the Companion automatically displays help on that option. If the Companion is not already open, press SHIFT + F11 to open it.

To reset an entire page of options to their default settings:

1. In the Options tab of the Navigator, expand an options folder.
2. Right-click one of the options pages, and then choose **Reset**.

   TIP: To reset only one option, look up its default value in the Companion and set the option to that value.

MACHINE EDITION HELP

Machine Edition includes a comprehensive online help system that enables you to access specific help topics while working with Machine Edition.

Use the InfoViewer or Companion to access help.

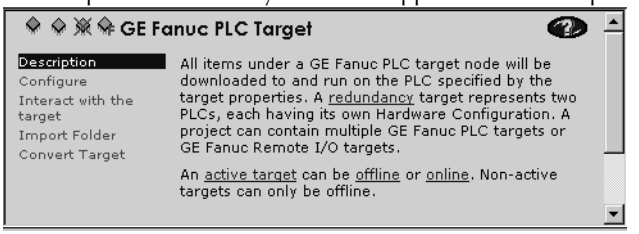
Companion Help

The Companion is a Machine Edition help system that provides useful tips and information while you work. While the Companion is open, it displays help on whatever item is currently selected in the Machine Edition environment.

To use Companion help

1. Ensure that the Companion is open. To open it, press Shift+F11.
2. Click any item in Machine Edition, outside the Companion.

   A description of the item you clicked appears in the Companion.



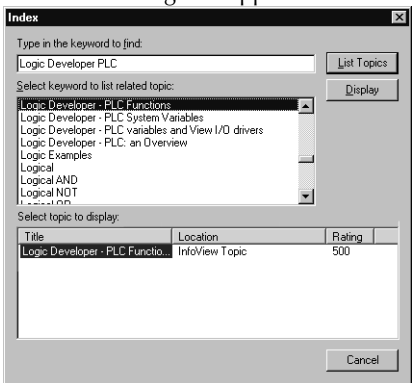
InfoViewer Help

The InfoViewer, another Machine Edition help system, provides detailed information. The InfoViewer has its own toolbar for navigating the help system, a Table of Contents (in the InfoView tab of the Navigator), an index, and a full-text search engine. InfoViewer help is context-sensitive. Click an item on the screen and press F1 to display the appropriate topic in the InfoViewer.

To use the Help index

1. From the Help menu, choose **Index**.

   The Index dialog box appears.



2. Enter a keyword in the top text box or select one in the middle text box.
3. Click **List Topics**.

   A list of topics appears in the bottom text box. The topics are sorted by their rating or likelihood of containing the correct information.

4. In the bottom text box, select a topic to display.
5. Click **Display**.

   The selected Help topic is displayed in the InfoViewer.

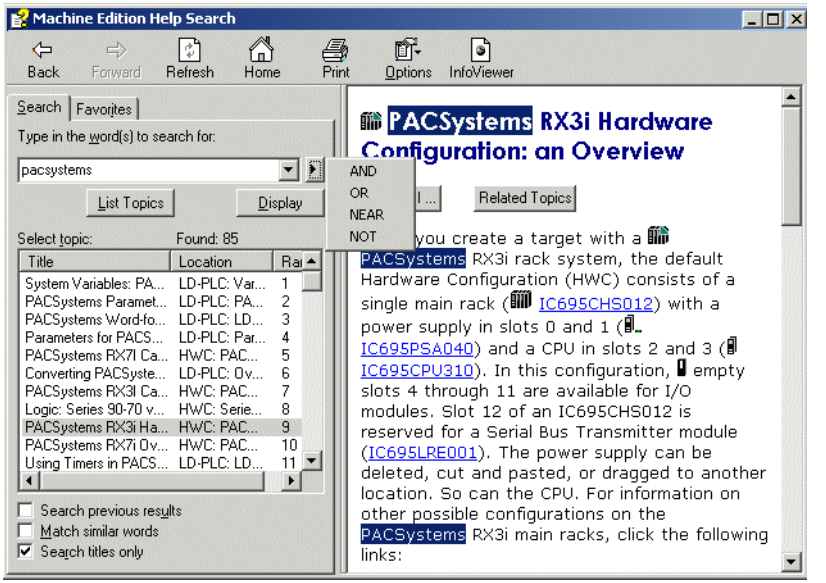
   TIP: To narrow a search, click the Location heading. The list of topics will be sorted by location and then by rating.

To use the full-text search

You can use the standard HTML Help Viewer to perform a full-text search of the Help. Advanced boolean search expressions can be applied.

1. From the Help menu, choose **Search**.

   The Machine Edition Help Search dialog box appears.



2. Enter a search word or phrase in the top text box. SURROUND A PHRASE WITH QUOTATION MARKS.
3. Click ▸ to perform a boolean search; that is, use AND, OR, NEAR, or NOT to create a boolean search expression.
4. Click **List Topics**.

   A list of topics appears in the Select topic box. The topics are sorted by their rating or likelihood of containing the search term, terms, or phrase.
5. In the Select topic box, select a topic to display.
6. Click **Display**.

   The selected Help topic is displayed in the Help Viewer.
7. (Optional.) Click the InfoViewer button to display the current topic in the InfoViewer.

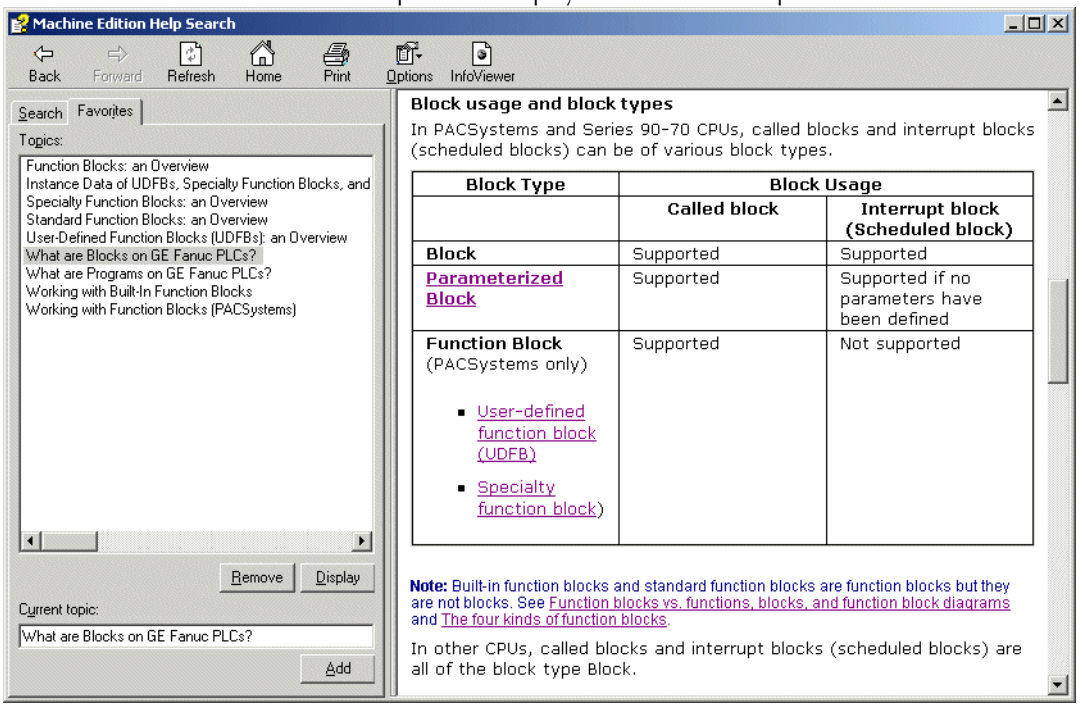
Tips:

- To narrow the search results, click the Location heading. The listed topics are sorted by location and then by rating.
- When using the HTML Help Viewer window, you may get better results if you select the "Search titles only" check box and/or clear the "Match similar words" check box.

To bookmark favorite topics

1. After using the HTML Viewer to perform a full-text search, select a topic you want to add to your list of favorite topics and then click the **Favorites** tab.

   The selected topic title is displayed in the Current topic text box.



2. Click **Add**.

   The topic is added to the Topics list.

3. (Optional.) Click the InfoViewer button to display the current topic in the infoViewer.
4. To display a favorite topic in the Help Viewer, select it and click **Display**.
5. To remove a topic from the Favorites list, select it and click **Remove**.

To look up topics in the Help Table of Contents

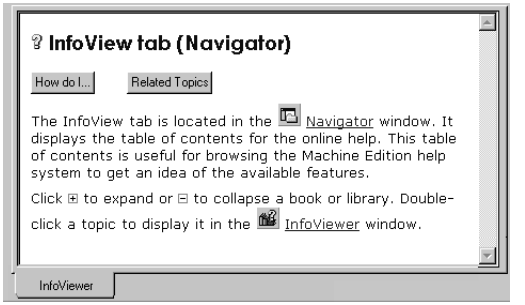
Navigator: InfoView Tab

1. Click the InfoView tab of the Navigator.

   A table of contents for the entire help system appears.

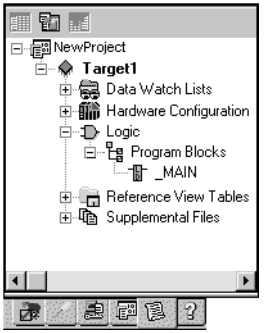
2. Expand Libraries and Books to locate a topic of interest
3. Double-click a topic.

   The topic is displayed in the InfoViewer.



3

GE Fanuc Targets



Navigator: New PACSystems Project

A target represents a run-time destination of the program or programs you develop with Logic Developer - PLC. Each target contains all of the components associated with that target. Logic Developer - PLC supports the following GE Fanuc families of PACSystems controllers, Series 90 PLCs, and VersaMax PLCs:

- PACSystems RX7i
- PACSystems RX3i
- Series 90-70 PLC
- Series 90-30 PLC
- VersaMax PLC
- VersaMax Nano/Micro PLC
- Series 90 Micro PLC

and the following GE Fanuc remote I/O interface targets:

- GE Fanuc Series 90-30 Ethernet Network Interface Unit
- GE Fanuc Series 90-70 Genius Remote I/O Scanner
- GE Fanuc VersaMax Ethernet
- GE Fanuc VersaMax Genius
- GE Fanuc VersaMax Profibus

ADDING, CONFIGURING AND CONVERTING TARGETS

Adding Targets

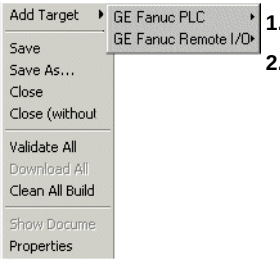
Typically a target is present in a project when you use a template to create the project (see page 15). A project can contain multiple targets. Before grouping targets into a project or isolating each target in its own project, consider the following:

- How large are the targets? For example, a target that contains 200,000 variables should probably be by itself in a project.
- Are the targets related? For example, a PACSystems with a relatively small program is connected to twenty GE Fanuc Remote I/Os (RIOs). Size is not an issue because RIO targets use little memory in a project. Grouping all the targets into one project is appealing because it makes navigation much easier from one target to another than if the PACSystems and RIOs are isolated in 21 separate projects. For another example, if you have various targets that do not communicate with one another or do not serve a common purpose, there may be little value in grouping them in one project.

One target is required for each PLC or remote I/O adapter your project accesses, except when you are using CPU redundancy. In this case, one target contains a Primary Hardware Configuration and a Secondary Hardware Configuration, which correspond to the primary PLC and the redundancy PLC, respectively. For more information, see page 62.

Existing targets can be converted from one PLC family to another. For more information, see page 32.

To add a target to a project



Project right-click menu

1. In the Project tab of the Navigator, right-click the Project node and point to **Add Target**.
2. Point to **GE Fanuc PLC** or **GE Fanuc Remote I/O** and then respectively choose the GE Fanuc PLC family or the Remote I/O.

A new target appears in the Project tab of the Navigator. When expanded, the new target contains a default Hardware Configuration, logic blocks, and miscellaneous components.

Configuring PLC Targets

The properties of a target specify the PLC family, the communication connections between your computer and the PLC, and various other settings. All properties are edited in the Inspector. The following table describes a PLC target's properties:

Name	Edit the name for your target in this field.
Type	The type of target is set by default to GE Fanuc PLC. Here the expression "GE Fanuc PLC" includes PACSystems controllers.
Description	Enter a description of your project in this field. The maximum number of characters is 255.
Documentation Address	Enter the URL where your project documentation is stored.
Family	By changing the PLC type in this field, you initiate a target conversion. **Caution: Changes are irreversible.**
PLC Target Name	The name of the target as stored on the PLC.
Update Rate (ms)	Set the rate at which the screen is updated while online to the target.
Sweep Time (ms)	The sweep time of the PLC when online. This value is also displayed on the status bar. (Read-only.)
PLC Status	The online/offline, run/stop status of the PLC. (Read-only.)
Online Project Synchronization	When Enabled, Machine Edition performs project synchronization checks with an online PLC.
Physical Port	Choose the type of connection to the PLC (Ethernet, COM, or modem).
IP Address	(Ethernet protocol only.) Set the IP address of the PLC.
⊞ Additional Configuration	Group of properties used for the detailed configuration of your communication connection.
⊞ Modem Communication	Group of properties to configure the modem communications link between your computer and a remote GE Fanuc target.

Other properties are available depending on the value of the Family property and the CPU model.

To configure a PLC target

1. In the Project tab of the Navigator, right-click a target and choose **Properties**.

   The properties of the target appear in the Inspector.

2. In the Inspector, configure the properties as needed.

Converting Targets

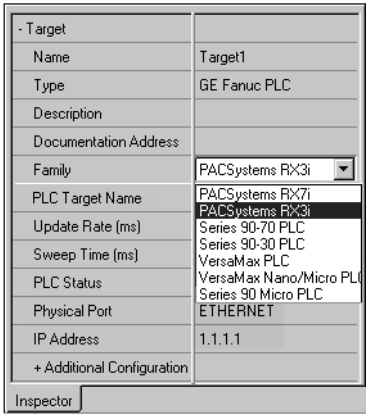
With Logic Developer - PLC, you can convert targets from one GE Fanuc PLC family type to another. For example, you can take the Hardware Configuration and logic written for a Series 90-70 PLC and convert them for use on a PACSystems RX7i. However, target conversions are irreversible; when logic blocks are deleted during a conversion, they cannot be restored. It is recommended that you make a backup of your project before converting a target. There may also be logic execution differences when converting an application from one GE Fanuc PLC family type to another, especially if a Series 90-70 is involved in the conversion. You must validate the application execution prior to deployment into a production environment.

To convert a target

1. In the Project tab of the Navigator, right-click a target and choose **Properties**.

   The Inspector displays the target's properties.

2. In the Inspector, click **Family.**
3. From the list, choose the new GE Fanuc PLC Family you want to convert the target to.



There are two types of target conversions.

A **basic** conversion does the following:

- Strips all the configured modules from the original Hardware Configuration (HWC) and sets up the destination family's default HWC, with a single power supply and a CPU.
- Deletes or adds target components, and logic programs or blocks.
- Updates the system variables, including the fault locating references if applicable.

An **enhanced** conversion does the following:

- Replaces a power supply with the destination family's default power supply.
- Retains the settings of CPU parameters common to the original and destination CPUs when the settings are supported in the destination family; otherwise, replaces them with the default settings.
- Retains all the original modules that are supported by the destination family. Retains all the module parameter settings that are supported by the destination family; otherwise, the default settings are used.
- Ethernet module settings of non-supported Ethernet modules are used to configure the destination's Ethernet daughterboard or default Ethernet modules.
- Converts unsupported expansion racks to the nearest equivalents.
- Deletes or adds target components, and logic programs or blocks.
- Updates the system variables, including the fault locating references if applicable.
- Displays a conversion report in the InfoViewer.

Caution: The capabilities of a destination target may be different from those of the original target. Carefully examine the conversion report when it is available. Validate the project and test it thoroughly before deploying it in production.

Note: GE Fanuc PLC hardware must be configured before it is operational. For detailed information on hardware configuration, see page 55.

Want to know more? In the online help index, look up "Target Conversions".

CONFIGURING COMMUNICATION

For Logic Developer - PLC to communicate with a target PLC, a connection must be properly configured. The properties of a target are adjustable to accommodate your connection(s).

To configure an Ethernet, modem, or serial connection with any GE Fanuc PLC

1. In the Project tab of the Navigator, right-click a target and choose **Properties**.

   The Inspector displays the target's properties.
2. In the Inspector, set the **Physical Port** property to Ethernet or a COM port (or a modem if one is installed).
3. If the **Physical Port** is Ethernet, enter the **IP Address** of the target PLC.
4. Double-click **Additional Configuration** to access the detailed settings for your connection.

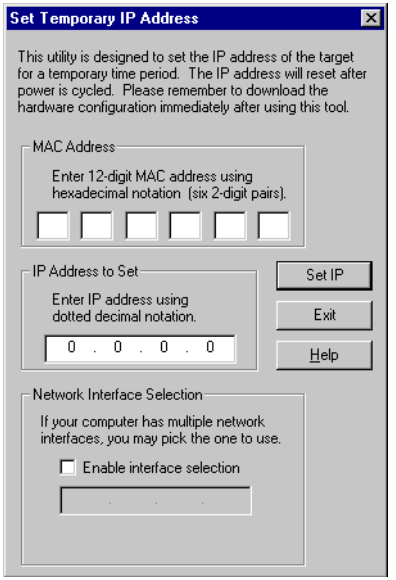
   Note: An IP address must be set in the PLC before an Ethernet connection can be established.

> **Want to know more?** In the online help index, look up "configuring communications with GE Fanuc PLC target".

To set a temporary IP address for a PACSystems PLC target or VersaMax Ethernet NIU target

1. In the Utilities tab of the Navigator, double-click Set Temporary IP Address.

The Set Temporary IP Address dialog box appears.



2. In the MAC Address field, specify the MAC address for the target.
3. In the IP Address to Set field, specify the temporary IP address you want to set on the target.
4. If required, select the Network Interface Selection check box and specify the network interface the target is located on.
5. Click the Set IP button.

   The IP address of the specified PLC or device is temporarily set to the indicated address.

   Warning: The IP address set by the Set Temporary IP Address utility is not retained through a power cycle. If you want to set a permanent IP address, you must download the Hardware Configuration to the PLC.

To set a permanent IP address for a PACSystems

1. In the Project tab of the Navigator, right-click the target and choose **Properties**.

   The Inspector displays the target's properties.
2. In the IP Address property, enter an IP address.
3. Expand the Hardware Configuration and expand the main rack.
4. Do one of the following:

   - For a PACSystems RX7i, expand the CPU and double-click the Ethernet subnode.

     - or -

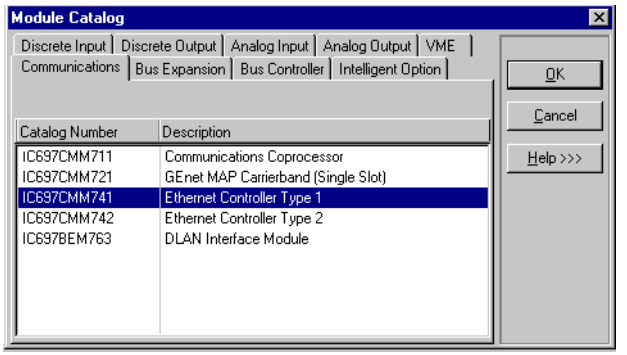
   - For a PACSystems RX3i, double-click every IC695ETM001 Ethernet module and repeat step 5 for each one of them.
5. In the Settings tab that appears in the Parameter editor, set the IP Address, the Subnet Mask, and the Gateway IP Address.
6. Download the Hardware Configuration to the PACSystems RX7i.

> **Want to know more?** In the online help index, look up "Configuring communications with GE Fanuc PLC target" and "Set Temporary IP Address utility".

To set an IP address for Series 90-70 PLCs

1. In the Project tab of the Navigator, expand the Hardware Configuration.
2. Expand the main rack and double-click the slot where you want to add a communications module.

   The Module Catalog appears.



3. Choose the Communications tab and select the communications module you want to add.

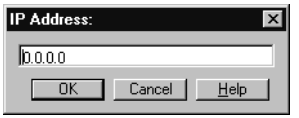
   Select either IC697CMM741 or IC697CMM742.

4. Click **OK**.

   The Parameter editor displays the configuration settings for the module.

5. In the Settings tab, double-click the **IP Address** parameter.

   The IP Address dialog box appears.



6. Enter the IP Address and click **OK**.
7. Download the hardware configuration. For more information, see page 41.

To set an IP address for a Series 90-30 CPU364, CPU374, NIU004, or a VersaMax CPUE05

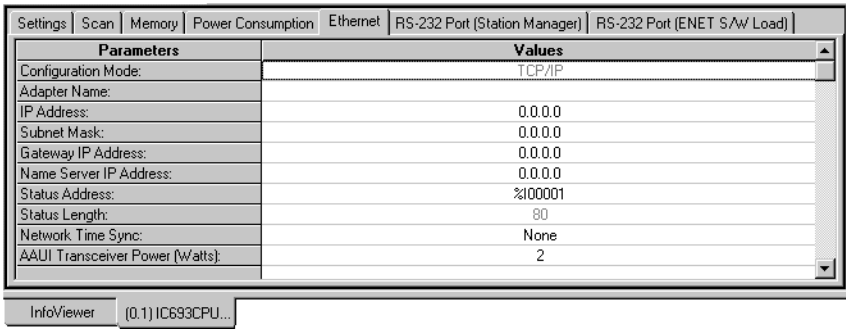
Note: An Ethernet adapter is built into CPU364, CPU374, NIU004, and CPUE05. For other Series 90-30 CPUs, Ethernet connections are supported through an optional expansion slot on the rack.

1. Obtain an IP address, perhaps from your network administrator.
2. In the Project tab of the Navigator, expand the Hardware Configuration and expand the main rack.
3. Right-click the slot containing the CPU364, CPU374, NIU004, or CPUE05 and choose **Configure**.

   The Parameter editor appears.

4. In the Parameter editor, click the Ethernet tab.

   The Parameter editor displays the contents of the Ethernet tab.



5. Double- click the **IP address** parameter.

   The IP Address dialog box appears

6. Enter the IP address in the box and click **OK**.
7. Download the hardware configuration. For more information, see page 41.

To set an IP address for other Series 90-30 CPUs (not CPU364 or CPU374)

1. In the Project tab of the Navigator, expand the Hardware Configuration.
2. Right-click the slot containing an Ethernet Interface Module (IC693CMM321) and choose **Configure**.

   The Parameter editor displays the configuration settings for the module.
3. In the Settings tab, double-click the **IP Address** field.

   The IP Address dialog box appears.
4. Enter the IP address and click **OK**.
5. Download the hardware configuration. For more information, see page 41.

To download an IP address via a PLC serial port

1. Configure a CPU or Ethernet communications module with an IP address using Hardware Configuration.
2. Right-click the target and choose **Properties**.

   The Inspector displays the target's properties.
3. In the Physical Port property, select a serial connection.
4. Right-click the target and choose **Go Online**.
5. Right-click the target and choose **Download to PLC**.

   The Download to PLC dialog box appears.
6. Select **Hardware Configuration** and click **OK**.

   The IP address is assigned to the PLC.
7. Right-click the target and choose **Go Offline**.
8. Right-click the target and choose **Properties**.
9. In the Physical Port property, select **Ethernet**.

   The next time you go online, Machine Edition will use an Ethernet connection with the specified IP address.

INTERACTING WITH A PLC

Communicating with a PACSystems, Series 90, or VersaMax is necessary to perform such operations as storing and loading programs or monitoring data values and PLC status information. You can connect to a PLC from Logic Developer - PLC over a serial, Ethernet, or modem connection, depending on the PLC's capabilities.

All interactions with a target are available from the target's right-click menu.

Validating a Target

Validating your target detects syntax and configuration errors on the target. Error messages are generated for each error and displayed in the Feedback Zone. A target containing errors cannot be downloaded. A target is automatically validated before a download operation.

To validate a target

- In the Project tab of the Navigator, right-click a target and choose **Validate**.

  Logic Developer - PLC checks all items under the target for syntax errors. Any errors detected are noted in the Build tab of the Feedback Zone.

  Tip: Double-click an error message to locate the noted error in your project. The appropriate editor or tool opens automatically with the item in question selected. Press the F4 key to locate the next error or warning in your project. Tips showing you how to proceed are displayed in the Companion. To open the Companion, press SHIFT + F11.

Offline, Online: monitor mode, programmer mode

When offline from a PACSystems, Series 90, or VersaMax, there is no ongoing communication between the PLC and your development computer. A physical communication link is not required as long as you only edit logic; it is required only when you want to communicate with the PLC.

The only PLC operations you can perform while offline are to go online or to set up the temporary IP address (see page 34).

When online with a target PLC, a communication link exists and is active and you have an ongoing real-time communication with the PLC.

When online in monitor mode, you can monitor the PLC while it is executing. You cannot edit logic on your computer except State Logic. You cannot change any

values on the PLC. Depending on your level access on the PLC and your Change Management permission levels, you can upload from the PLC.

When online in programmer mode, you can make changes on your computer and the PLC and can monitor the PLC while it is executing. You can edit any type of PLC logic on your computer. Depending on your level access on the PLC and your Change Management permission levels, you can upload from the PLC, control the PLC while it is executing, and change values on the PLC, including downloading to the PLC. You can edit LD logic while online; you are prompted to download your word-for-word changes. For more information on word-for-word changes, see page 91.

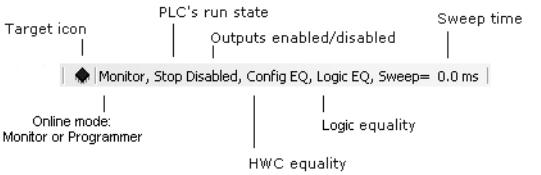
When online in either mode, if the project stored on the PLC is equal to the current project in Logic Developer - PLC, the LD editor displays a graphical representation of LD logic as it executes.

Want to know more? In the online help index, look up "Offline (Logic Developer - PLC)".

To go online to a GE Fanuc PLC

1. **In the Project tab of the Navigator, ensure that all the target PLCs are offline.**
   You can be online to only one target at a time. When a target is offline, its icon is grey.
2. **If the name of the target you want to go online to does not appear in bold characters, right-click the target and choose Set as Active Target.**
   Bold characters indicate that this is the active target.
3. **Do one of the following:**
   - Right-click the active target and choose **Go Online**.
     - or -
   - Click the thunderbolt on the Online toolbar:

   Logic Developer - PLC connects your project to the GE Fanuc PLC. The online status is indicated by the target icon in the Project tab and on the status bar.



Note: When online to a GE Fanuc PLC, the target icon in the Project tab of the Navigator appears as (equal), (not equal), or (stop faulted).

To change the online mode

- In the Project tab of the Navigator, right-click the target PLC, point to **Online Commands**, and choose **Set Programmer Mode** or **Set Monitor Mode**.

To go offline from a GE Fanuc PLC

- Do one of the following:
  - In the Project tab of the Navigator, right-click a target PLC and choose **Go Offline**.

    - or -
  - Click the thunderbolt on the Online toolbar: 

  Logic Developer - PLC goes offline.

> **Want to know more?** In the online help index, look up "going online to GE Fanuc PLCs", "setting online programmer mode", "setting online monitor mode", and "going offline from GE Fanuc PLCs".

Upload/Download

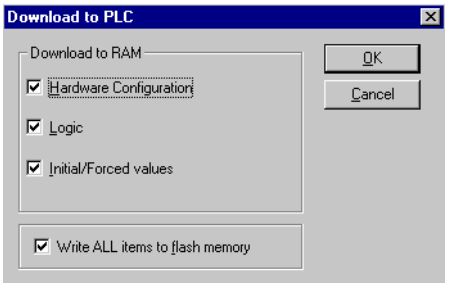
The download process builds and validates all run-time files necessary for a target to perform its role in a completed project. The compiled project is then transferred to the target hardware over the communication connection previously configured. For more information, see page 34.

The upload process acquires a project from the active GE Fanuc PLC target and transfers it to Logic Developer - PLC for editing.

To download to a GE Fanuc PLC

1. Ensure you are online in programmer mode to the target PLC.
2. In the Project tab of the Navigator, right-click the target to which you want to download files and choose **Download to PLC**.

   The Download to PLC dialog box appears.



Note: If the PLC is running, you can download only logic that is not equal to the PLC's current logic and the Download to PLC dialog box does not appear. On a PACSystems, some source files that do not affect equality are also downloaded.

3. Choose the items you want to download and click **OK**.

   Note: Only one project can be downloaded to a target at a time. If you download to a target PLC that already has a project on it, the existing project is overwritten.

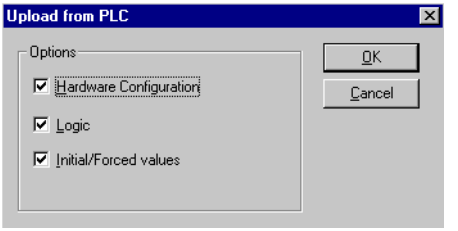
   For each target that you download, Machine Edition performs a validation. Any errors that occur are displayed in the Build tab of the Feedback Zone. If there are no errors, Machine Edition builds and sends all the necessary run-time files to the PLC.

Want to know more? In the online help index, look up "downloading to GE Fanuc PLCs".

To upload files from a GE Fanuc PLC

1. Ensure you are online to the target PLC.
2. In the Project tab of the Navigator, right-click the target PLC from which you want to upload information and choose **Upload from PLC**.

   The Upload from PLC dialog box appears.



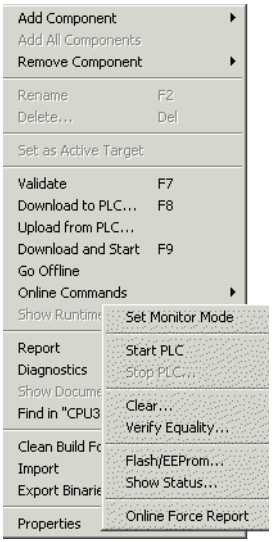
3. Choose the item(s) you want to upload and click **OK**.

   The selected items are uploaded to Logic Developer - PLC. If you already had a version of the PLC's project open, the uploaded project merges with the existing project. Because variable names are not stored on non-PACSystems GE Fanuc PLCs, if you upload to an empty target, all variables are assigned default names. For example, %R00001 is named R00001.

Want to know more? In the online help index, look up "uploading from GE Fanuc PLCs".

Run/Stop

When you are online in programmer mode, you can set a target GE Fanuc PLC to Run or Stop mode. In Stop mode, you can choose to enable or disable the outputs.



To start a GE Fanuc PLC

- Do one of the following:
  - In the Project tab of the Navigator, right-click a target, point to **Online Commands**, and then choose **Start PLC**.

    - or -
  - Click on the Online toolbar:

    The target PLC begins executing its program.

    Note: If you are starting a PACSystems or Series 90-70 PLC, you can choose to have outputs enabled or disabled.

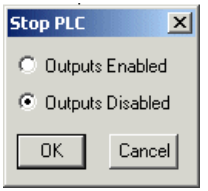
To stop a GE Fanuc PLC

1. Do one of the following:
   - In the Project tab of the Navigator, right-click a target, point to **Online Commands**, and choose **Stop PLC**.

     - or -
   - Click on the Online toolbar:

The Stop PLC dialog box appears, prompting you to enable or disable the PLC's outputs.



2. Select an option.
3. Click **OK**.

The target PLC stops executing its program.

Want to know more? In the online help index, look up "starting GE Fanuc PLCs" and "stopping GE Fanuc PLCs".

Fault Tables

The PLC and I/O Fault Tables display fault information logged by the CPU or modules in the PLC. This information is used to determine if there are problems with the PLC hardware or software running in the PLC's CPU.

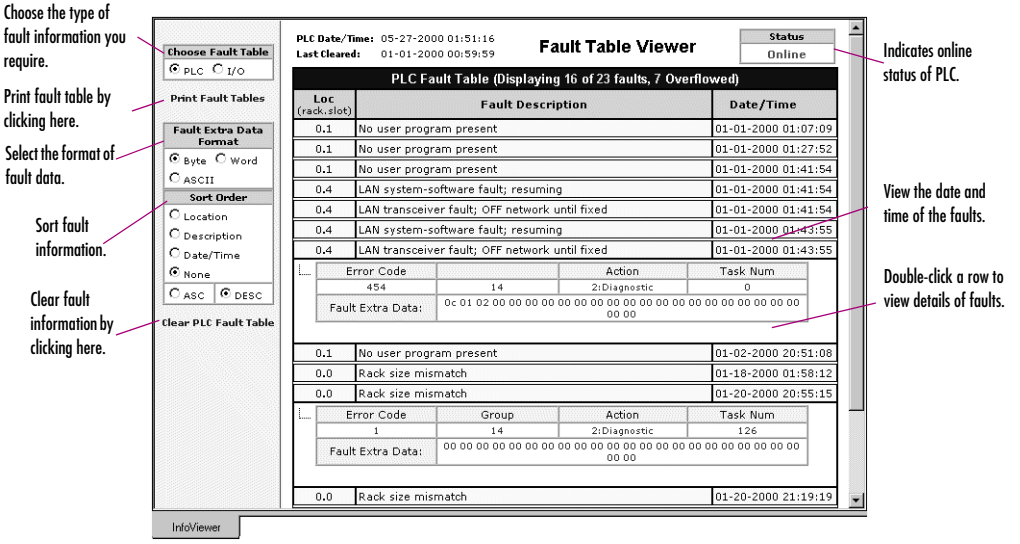
To view the fault table reports

- Do one of the following:
  - In the Project tab of the Navigator, double-click the target you want a report on, or right-click it and choose **Diagnostics**.

    - or -
  - In the status bar, at the bottom of the Machine Edition window, double-click the target.

Note: To view the PLC and I/O fault tables, your computer must be online to the PLC. To clear faults, you must be in online programmer mode.

The fault table appears in the InfoViewer window.



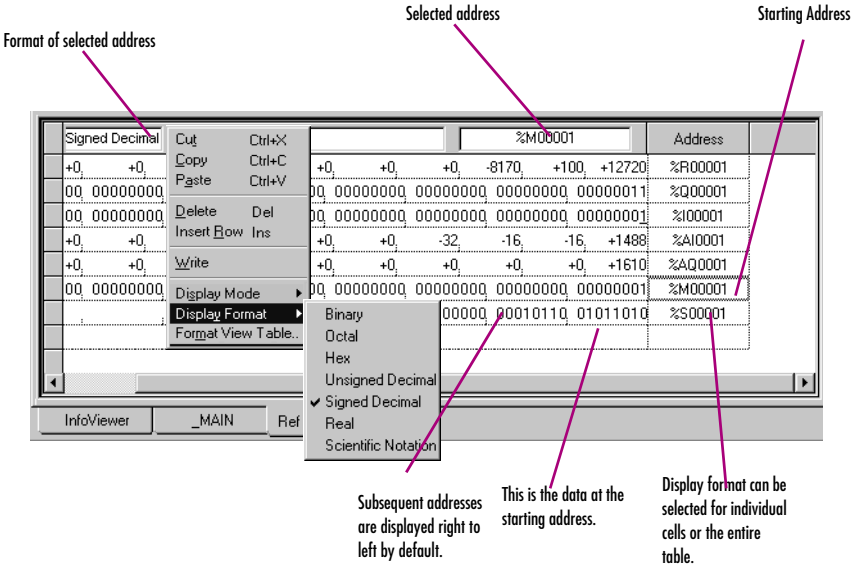
Want to know more? In the online help index, look up "fault" (in lower case), then select "Working with the Fault Tables".

Reference View Tables

Reference View Tables (RVTs) are tables in which you can monitor reference data when you are online to a GE Fanuc PLC. If you are in online programmer mode, you can also use RVTs to change the values of reference data. In the Project tab of the Navigator, the Reference View Tables folder contains a Default Tables folder. You can add user-defined tables to the Reference View Tables folder. A target can have zero or more user-defined RVTs.

You can configure the default appearance of your RVTs in the Options tab of the Navigator. For more information, see "Options", on page 23.

Data values at sequential addresses are displayed from right to left, by default, starting at the reference address specified in the Address column. Both default and user-defined RVTs display rows of 8 cells for discrete memory (each cell corresponding to 8 bits) and rows of 10 cells for register memory (each cell corresponding to one 16-bit register). The amount of data displayed in the columns depends on the data display format.

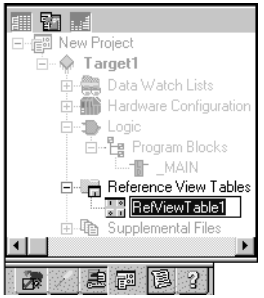


To create a user-defined Reference View Table

- In the Project tab of the Navigator, right-click the Reference View Tables folder and choose **New**.

  A new Reference View Table with a default name is added to the folder.

To work with a user-defined Reference View Table



Navigator: Project Tab

Reference View Table node

1. In the Project tab of the Navigator, expand the Reference View Tables folder and double-click the table you want to view.

   The Reference View Table appears in the main Machine Edition window.
2. Add reference addresses to the table as required.

   Note: You cannot add reference addresses to a default RVT.
3. Format the table entries as desired.

> **Want to know more?** In the online help index, look up "RVT", then select "Working with user-defined Reference View Tables" or "Working with default Reference View Tables".

Reports

Reports provide summaries and tables of information about your project. Most reports are displayed in the InfoViewer (see page 24). The Reports tab of the Feedback Zone contains a list of all reports generated since the last Machine Edition project was opened. The following are types of reports and logic printouts available in Logic Developer - PLC:

Address Use report	Online Force Report	IL block report *
Application Structure reports	Variables reports	LD block report *
EGD reports		Local Logic block report *
Hardware Configuration report		Motion block report *
Modbus Address report	CAM profile report	Structured Text Blocks *

An asterisk (*) indicates a logic printout.

To generate reports

- In the Project tab of the Navigator, right-click a node and choose **Report** to generate a report on that node.

  A report is automatically generated and displayed in the InfoViewer.

  Note: To generate an Online Force report for a target, you must be online to the target PLC. Right-click the target, point to **Online Commands**, and then choose **Online Force Report**.

To redisplay a previously generated report

1. In the Feedback Zone, double-click the Reports tab.

   A list of previously generated reports appears in the Feedback Zone.

2. In the list, click the report you want to view.

   The report displays in the InfoViewer.

   Many items in a report contain hyperlinks. Click a hyperlinked item to jump to that item in the project. For example, if a variable's name appears hyperlinked in a report, clicking it selects that variable in the Variables tab of the Navigator.

   Large reports are often separated into several pages. To view a different page of the report, scroll to the bottom of the report in the InfoViewer and click the number of the page you want to view.

To print a report displayed in the InfoViewer

1. Generate the report you want to print, or redisplay a previously generated report.
2. When the report is displayed, right-click the InfoViewer window and choose **Print**.

To print LD Blocks

1. In the Project tab of the Navigator, expand the Logic node.
2. Right-click the Program Blocks node and choose **Print LD Blocks**.

   The Print dialog box appears.

3. Select the blocks to print.
   - To print all of the target's LD blocks, select the **All** option.
   - To print only some of the target's LD blocks, select the **Selection** option, then select the check box in front of each block you want to print.
4. Select options as required and click **OK**.

To print ST blocks

1. In the Project tab of the Navigator, expand the Logic node.
2. Right-click the Program Blocks node and choose **Report of ST Blocks**.

   The Structured Text Blocks logic printout is automatically generated and displayed in the InfoViewer.

Want to know more? In the online help index, look up "Reports" and "printing LD logic".

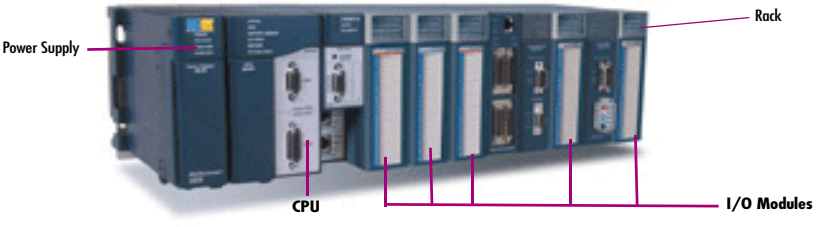
4 Hardware Configuration

Logic Developer - PLC supports several GE Fanuc PLC families and various GE Fanuc remote I/O interfaces (see page 72) with a variety of CPUs, racks, and modules for each. In order to operate, GE Fanuc PLC hardware must be configured with Logic Developer - PLC or some other GE Fanuc tool. The HWC component of Logic Developer - PLC provides a way to configure your target equipment. This chapter details specifics on configuring PLC hardware for your operational needs.

The first step in configuring PLC hardware is to select the PLC you want to configure (see page 30). When creating a new project, you can use a project template containing a default Hardware Configuration, or you can create an empty project and configure it manually.

PACSYSTEMS RX3I

The PACSystems RX3i, pictured below, is the newest family of GE Fanuc PLCs. The PACSystems RX3i is an upward compatible upgrade for Series 90-30 PLCs, offering more memory, faster clock speed, and an enhanced backplane bus.



The PACSystems RX3i consists of a main rack and up to seven expansion racks. Six types of Series 90-30 expansion racks are supported; they have IC693CHSnnn catalog numbers and are colored black, in keeping with the Series 90-30 color scheme. Two expansion racks have IC694CHSnnn catalog numbers. These are identical to the IC693CHSnnn expansion racks of the same numbers except that they are colored blue, in keeping with the PACSystems RX3i color scheme.

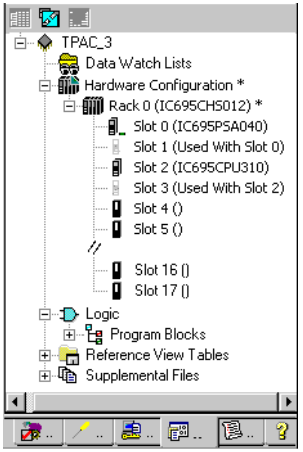
The main rack and all expansion racks support most Series 90-30 modules; these have IC693NNNnnn catalog numbers and are colored black, in keeping with the Series 90-30 color scheme. Almost all of the IC693NNNnnn modules supported by PACSystems RX3i also come as IC694NNNnnn modules that have identical functionality. The only difference is that the IC694NNNnnn modules are colored blue, in keeping with the PACSystems RX3i color scheme. For example, the IC693DSM314 and IC694DSM314 are functionally identical and both can be used interchangeably in a Series 90-30 rack system or a PACSystems RX3i rack system. The only hardware difference is their color.

When you configure a PACSystems RX3i rack system, you can select IC693NNNnnn or IC694NNNnnn racks or modules. When you configure a Series 90-30 rack system, however, you can select only IC693NNNnnn racks or modules. Even if you are physically using IC694NNNnnn racks or modules, selecting IC693NNNnnn racks or modules in Logic Developer - PLC is adequate.

The PACSystems RX3i default main rack, IC695CHS012, has 13 slots: Slots 0 through 12. The alternate main rack, IC695CHS016, contains 17 slots: Slots 0 through 16.

On either main rack, slot 0 is reserved for a power supply or the CPU, but the power supply or CPU does not have to be in slot 0. The last slot is reserved for the Serial Bus Transmitter module (IC695LRE001). The slot just before the last slot can contain any single-width module native to PACSystems RX3i (IC695...) except the Serial Bus Transmitter. Any other slot can contain any single-width module native to PACSystems RX3i (IC695...) except the Serial Bus Transmitter, and if the next slot is empty, they can contain any supported double-width module.

The IC695PSA040 power supply and the CPU are two-slot modules. The IC695PSD040 power supply is a one-slot module. PACSystems RX3i supports many Series 90-30 modules: each one occupies a single slot.



Navigator: Project Tab displaying the HWC node of a PACSystems RX3i

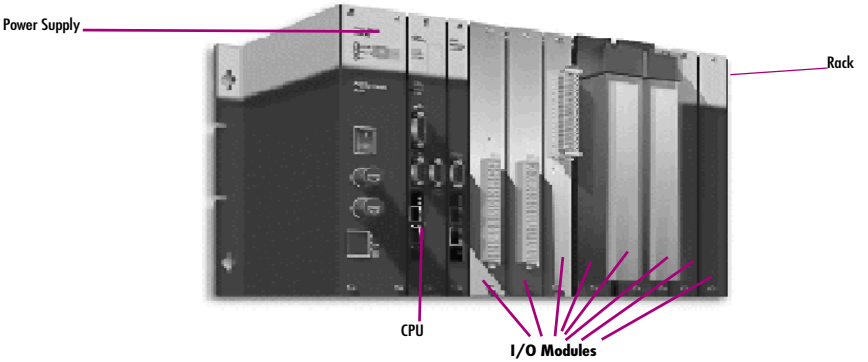
Configuring PACSystems RX3i Hardware

When you create a target with a PACSystems RX3i rack system, the default HWC consists of the main rack, with a power supply in slots 0 and 1, and a CPU in slots 2 and 3. You can replace the default power supply. You can replace the CPU with itself to update it to the latest catalog version, that is, the latest set of properties supported for the CPU in Logic Developer - PLC. You can move the power supply and the CPU to any empty slot in the target with an adjacent empty slot; however, the second adjacent slot cannot be the last slot. You can add up to seven expansion racks, and on each of these, you can add IC693NNNnnn modules and the IC694NNNnnn that are supported by Series 90-30.

Note: On PACSystems RX3i targets, only the main rack is added by default. On Series 90-70 and Series 90-30 PLCs, seven expansion racks are added by default to the HWC. You do not need to add them.

PACSYSTEMS RX7I

The PACSystems RX7i is pictured below.



The PACSystems RX7i consists of a main rack and up to seven expansion racks. The PACSystems RX7i supports a few types of main racks. It supports five types of Series 90-70 expansion racks. All main racks and expansion racks support a few non-intelligent Series 90-70 fan kits.

The PACSystems RX7i main racks have 10 or 18 single-width VME slots. Slot 0 supports one of the power supply modules. Slots 1 and 2 are used to house a double-width CPU. PACSystems RX7i supports a few CPU models. Each PACSystems RX7i CPU has an embedded Ethernet adapter. In addition to the Ethernet adapter, up to three IC698ETM001 Ethernet modules can be installed in a PACSystems RX7i main rack. The Ethernet modules occupy only one slot. All of the IC698... modules are supported only on the main rack.

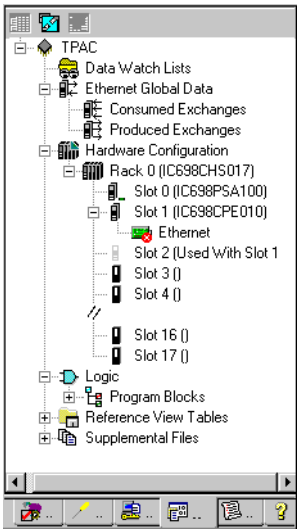
PACSystems RX7i supports most Series 90-70 modules. Most Series 90-70 modules are double-width, that is, they occupy two slots on the main rack, but they occupy only one slot on a Series 90-70 expansion rack. You can plug double-width modules into any two adjacent slots of the main rack except slots 0, 1, or 2. Because slot 17 is a double-width slot, any module in slot 17 occupies only slot 17. Slot 17 can also accept a single-width module.

PACSystems RX7i supports the following Series 90-70 bus expansion modules:

- IC697BEM711
- IC687BEM713
- IC697BEM713

PACSystems RX7i also supports Series 90-70 Genius Bus Controllers, which support numerous Genius devices.

Configuring PACSystems RX7i Hardware



Navigator: Project Tab displaying the HWC node of a PACSystems RX7i

When you create a target with a PACSystems RX7i rack system, the default Hardware Configuration consists of the main rack, with a power supply in slot 0 and a single CPU in slots 1 and 2. You can replace the default power supply and CPU, and add single-width or double-width modules. You can add up to seven expansion racks, and on each of these, you can add Series 90-70 modules.

Note: On PACSystems RX7i targets, only one rack is added by default. On Series 90-70 and Series 90-30 PLCs, seven expansion racks are added by default to the HWC. You do not need to add them.

CONFIGURING PLC HARDWARE

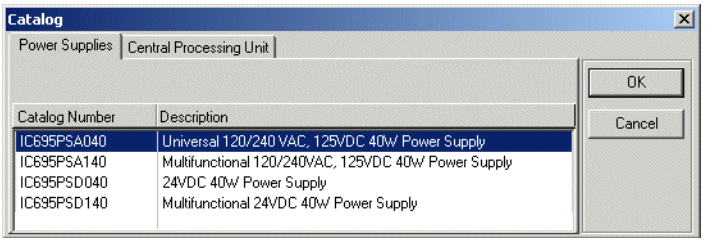
The following configuration procedures focus on the PACSystems RX3i. Procedures for the other PLC families supported by Logic Developer - PLC are nearly identical.

The default power supply for the PACSystems RX3i is the IC695PSA040.

To replace a power supply

1. In the Project tab of the Navigator, right-click the power supply slot and choose **Replace Module**.

   A list of available power supplies appears.

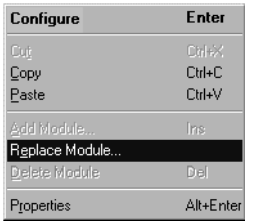


   Note: For non-PACSystems GE Fanuc PLC families, there is no Slot 0. Right-click the PWR slot instead.

2. Select the power supply you have installed in your rack and click **OK**.

   The default CPU specified in the project template for a PACSystems RX3i is the IC695CPU310.

To replace a CPU



HWC right-click menu

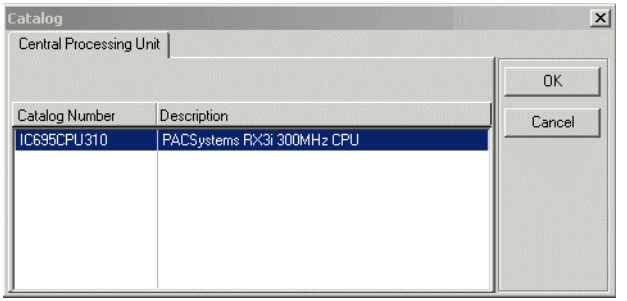
1. In the Project tab of the Navigator, expand the Hardware Configuration.

   All racks are revealed.

2. Expand the main rack.
3. Right-click the CPU slot and choose **Replace Module**.

   Note: PACSystems supports only one CPU; it can be replaced with itself to update it to the latest catalog version, that is, the latest set of properties supported for the CPU in Logic Developer - PLC.

The Module Catalog dialog box appears.



4. From the list, select a CPU.
5. Click **OK**.

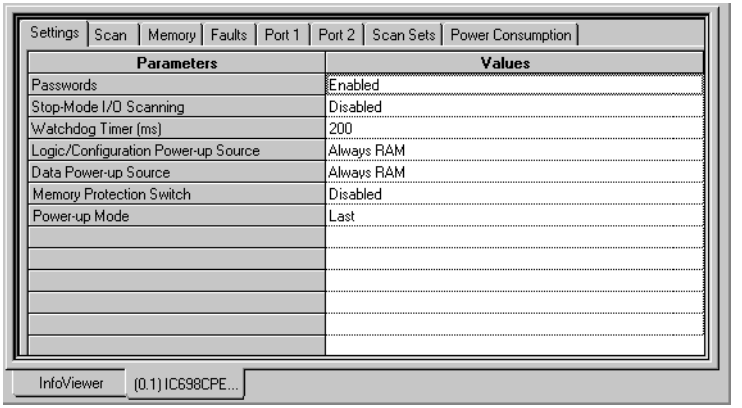
   A dialog box appears asking if you want to retain the settings from the existing CPU.
6. Click **Yes** or **No**.

   The target is configured with the selected CPU.

To configure a CPU

1. In the Project tab of the Navigator, right-click a slot containing a CPU and choose **Configure**.

   The Parameter editor displays all configurable settings for the CPU.



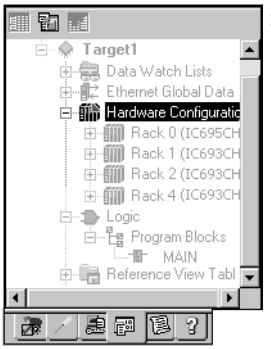
2. Modify the settings as required. For information on any parameter of a PACSystems CPU, select the parameter.

   Help topics for PACSystems CPU parameters appear in the Companion.

Tip: To open the Companion, press SHIFT+F11 or click on the Tools toolbar (see page 17).

To add an expansion rack (PACSystems only)

Note: On Series 90-70 and Series 90-30 PLCs, seven expansion racks are added by default to the HWC. You do not need to add them.

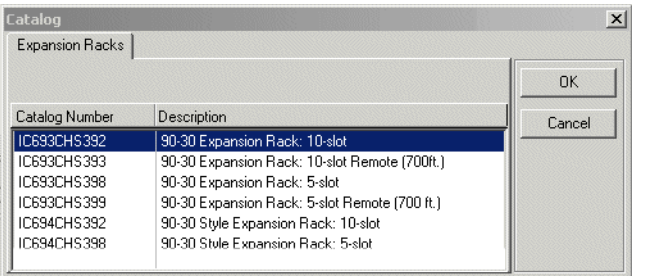


Navigator: Project tab

PACSystems HWC configuration

1. In the Project tab of the Navigator, right-click the Hardware Configuration and choose **Add Rack**.

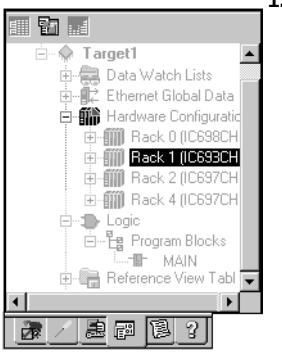
   The Module Catalog dialog box appears listing available rack types.



2. Select a rack and click **OK**.

The Series 90-30 expansion rack is added to the PACSystems RX3i rack system. It is assigned an available number in the range of 1 through 7, excluding numbers already assigned to other expansion racks.

To replace a rack

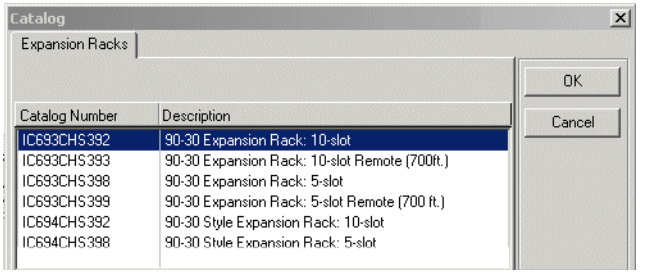


Navigator: Project tab

HWC configuration

1. In the Project tab of the Navigator, right-click a rack and choose **Replace Rack**.

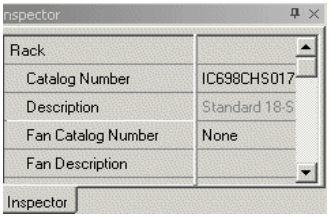
   The Module Catalog dialog box appears listing available rack types.



2. Select a rack and click **OK**.

To add a fan kit to a rack (PACSystems RX7i only)

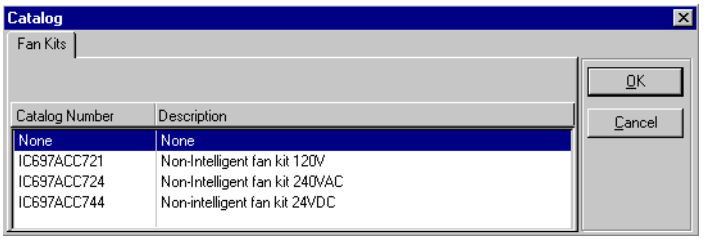
1. In the Project tab of the Navigator, expand the Hardware Configuration.
2. Right-click a rack and choose **Properties**.
   The Inspector displays the rack's properties.



3. In the Inspector, click the Fan Catalog Number property.
   The ••• button appears to its right.

4. Click the ••• button.

   The Fan Kits catalog appears.



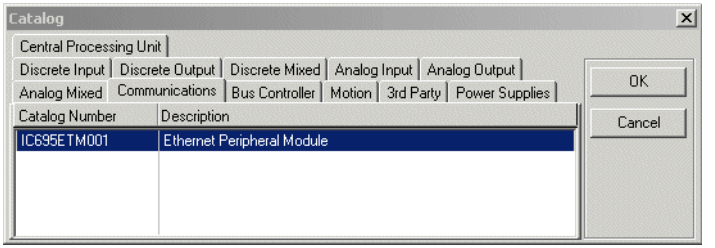
5. Select the fan kit you want to add and click **OK**.

   The fan kit is added to the rack.

To add an Ethernet module (PACSystems only)

1. In the Project tab of the Navigator, expand the Hardware Configuration and then Rack 0 (the main rack).
2. Double-click the slot you want to add an Ethernet module to, or right-click it and choose **Add Module**.

The Catalog appears.



3. In the Communications tab of the Catalog, select the Ethernet Module and click **OK**.

   The Ethernet module is added to the slot.

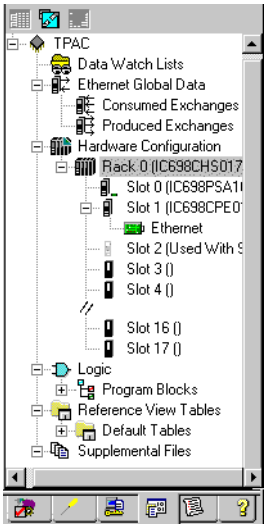
4. Double-click the Ethernet module.

   The Parameter editor appears.

5. Configure the Ethernet module's parameters as needed.

> Want to know more? In the online help index, look up "configuring PACSystems RX3i rack system".

To configure the Ethernet daughterboard (PACSystems RX7i only)



Navigator: Project tab PACSystems Ethernet Adapter

1. In the Project tab of the Navigator, expand the Hardware Configuration, then Rack 0 (the main rack), and then Slot 1.
2. Double-click Ethernet.

   The Parameter editor appears.

3. Configure the Ethernet daughterboard's parameters as needed.

To move a module

1. In the Project tab of the Navigator, expand the Hardware Configuration and then the rack that contains the module.
2. Do one of the following:
   - **To move the module from one target to another:** press SHIFT while dragging and dropping the module onto an appropriate empty slot.
     - or -
   - **To move the module within a target:** drag and drop the module onto an appropriate empty slot without pressing any keys.

   You cannot move modules between different types of PLC targets, for example from a Series 90-30 to a PACSystems RX7i.

   The module is removed from the original slot and inserted into the empty slot.

   Notes: (PACSystems)

   - You can drag a double-width module to a main rack slot only if both the slot and the slot after it are empty. Exceptions: The last slot on a PACSystems RX7i can contain a double-width module or one single-width module. The last slot on a PACSystems RX3i can contain only an IC695LRE001.
   - Only PACSystems RX3i supports moving a power supply or CPU module.

To copy a module

1. In the Project tab of the Navigator, expand the Hardware Configuration and then the rack that contains the module.
2. Do one of the following:
   - To copy the module from one target to another: drag and drop the module onto an appropriate empty slot without pressing any keys.
     - or -
   - To copy the module within a target: press CTRL while dragging and dropping onto an appropriate empty slot.

   You cannot copy modules between different types of PLC targets, for example from a Series 90-30 to a PACSystems RX7i.

   A copy of the module is inserted into the empty slot.

   Notes: (PACSystems)

   - You can drag and drop a double-width module to a main rack slot only if both the slot and the slot after it are empty. Exceptions: The last slot on a PACSystems

RX7i can contain a double-width module or one single-width module. The last slot on a PACSystems RX3i can contain only an IC695LRE001.

- Only PACSystems RX3i supports copying a power supply.

I/O VARIABLES

An I/O variable is a variable mapped to a terminal in the Hardware Configuration of a PACSystems. A terminal, for example, can be one of the following:

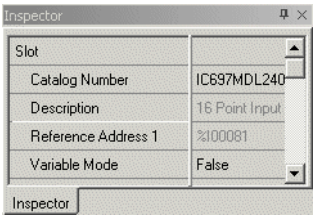
- A physical I/O discrete or analog point on a PACSystems module or on a Genius device
- A discrete or analog status returned from a PACSystems module
- Genius global data

Memory required to support I/O variables counts against your user memory. When you configure the PACSystems CPU (see page 55), select the Memory tab and set the I/O Discrete (# of Bits) and I/O Non-Discrete (# of Words) parameters to configure the space available for I/O variables.

To enable I/O variables for a module

1. In the Project tab of the Navigator, expand the target that contains the module.
2. Expand the Hardware Configuration and expand the rack that contains the module.
3. Right-click the module and choose **Properties**.

   The Inspector displays the module's properties.



4. In the Inspector, set the Variable Mode property to **True**.

   If the Variable Mode property is read-only and set to False, the module doesn't support I/O variables.

5. In the dialog box that appears, click **Yes**.

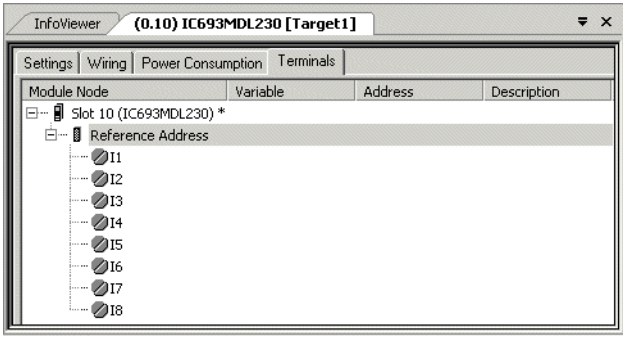
   If you enable I/O variables for a Genius bus controller, I/O variables are enabled for all the Genius devices on the Genius bus.

Note: When you enable I/O variables for a module, the Hardware Configuration and logic become coupled. This means that they must be downloaded, uploaded, or cleared together. You cannot change or add I/O variables in Run Mode Store.

To map a variable to a terminal in the Terminals tab of a module or Genius device

1. In the Project tab of the Navigator, expand the target that contains the module or Genius device.
2. Expand the Hardware Configuration and expand the rack that contains the module or device.
3. Double-click the module or expand the Genius Bus Controller (GBC) and double-click the Genius device.

   The Parameter editor displays the Terminals tab of the module or device.



4. In the Terminals tab, right-click a terminal node that has no I/O variable mapped to it and choose **Map Variable**.

   The Variables smart list appears.
5. In the smart list, do one of the following:

   - Enter a name that is not used elsewhere in the PACSystems target. A new I/O variable by that name is created and mapped to the terminal. Its Ref Address property is set to an I/O variable address. If the terminal is discrete, the new I/O variable's data type is set to BOOL. If the terminal is analog, the data type is set to INT, but you can change it to another 16-bit data type: UINT or WORD.

     - or -
   - Enter the name of an existing CPU-mapped variable, symbolic variable, or alias variable. (There are some mapping limitations.) The variable becomes an I/O variable mapped to the terminal and its Ref Address property is changed accordingly.

> **Want to know more?** In the online help index, look up "I/O Variables (PACSystems)", "working with I/O variables (PACSystems)", and "Coupled Hardware Configuration and Logic".

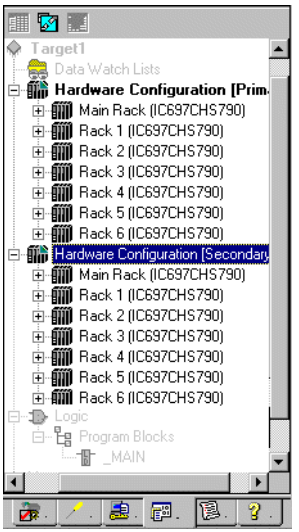
REDUNDANCY SYSTEMS

Redundancy systems apply only to Series 90-70 PLCs and PACSystems RX7i. In redundancy systems, two units are set up and configured identically. If one unit fails or is taken offline, the other unit assumes responsibility without interrupting operation of the entire system.

There are three types of redundancy systems:

- Basic CPU Redundancy
- Genius Redundancy
- CPU Redundancy Over Genius

Basic CPU Redundancy



Navigation: Project Tab
HWC Node
CPU Redundancy
(Series 90-70 and PACSystems RX7i only)

To implement basic CPU redundancy, a single Logic Developer - PLC target is associated with two physical PLCs, a Primary and a Secondary. Both PLCs share the same logic, but each has its own Hardware Configuration (HWC): Primary or Secondary. The selected HWC is the HWC that you can go online with, download to, upload from, and so on.

Three types of basic CPU redundancy are available. These redundancy systems can be used in combination with Genius redundancy schemes.

- **Single Bus with Preferred Master (Series 90-70 CPUs CGR772 and CGR935 only):** uses a single Genius bus with one or more bus controllers in each PLC. The primary unit is always chosen as the active unit when the units initially synchronize.
- **Single Bus with Floating Master (PACSystems RX7i and Series 90-70):** uses a single Genius bus with one or more bus controllers in each PLC. No switchover occurs on initial synchronization to make the primary unit the active unit.
- **Dual Bus with Floating Master (PACSystems RX7i and Series 90-70):** uses dual busses with one or more bus controllers in each PLC. No switchover occurs on initial synchronization. Bus Switching Modules (BSMs) are required in accordance with configuration of a dual bus network. This option provides redundancy of both the CPU and the Genius bus.

Genius Redundancy

A Genius redundancy system contains duplicate components that are configured to keep the Genius system operating properly even if one of the duplicate components fails or is taken out of service. Genius redundancy systems can be used in combination with PACSystems RX7i and Series 90-70 CPU redundancy systems.

You can configure five types of Genius redundancy systems:

- Genius dual bus redundancy (paired GBC internal)
- Genius dual bus redundancy (paired GBC external)
- Genius dual GBC redundancy (paired GBC internal)
- Genius dual GBC redundancy (paired GBC external)
- Genius dual bus & dual GBC redundancy

CPU Redundancy Over Genius

A CPU Redundancy Over Genius system contains duplicate components that are configured to keep the system operating properly if one of the duplicate components fails or is taken out of service.

Five types of CPU Redundancy Over Genius systems can be built upon the various types of redundancy Genius systems combined with the basic CPU Redundancy schemes.

- **CPU redundancy (GHS) using Genius dual GBC redundancy (paired GBC external):** Single bus with preferred master (Series 90-70 only)
- **CPU redundancy (HSB/GDB) using Genius dual GBC redundancy (paired GBC external):** Single bus with floating master
- **CPU redundancy (HSB/GDB) using Genius dual bus redundancy (paired GBC external):** Single bus with floating master
- **CPU redundancy (HSB/GDB) using Genius dual bus and dual GBC redundancy:** Dual bus with floating master
- **CPU redundancy (HSB/GDB) using a mixed Genius redundancy scheme**

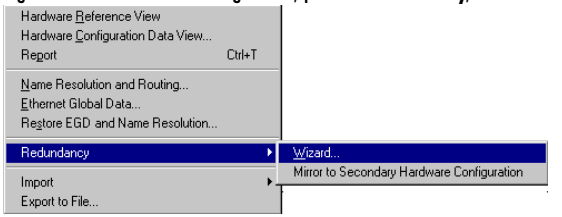
Want to know more? In the online help index, look up "Redundancy: an Overview".

Configuring Redundancy Systems

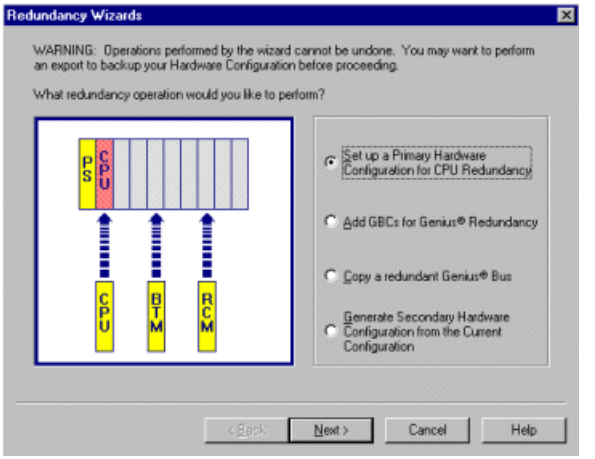
The procedures below are the general steps to follow for configuring redundancy systems. For the procedures on configuring specific redundancy systems, see the online help.

To set up the primary Hardware Configuration for CPU redundancy (PACSystems RX7i and Series 90-70)

1. In the Project tab of the Navigator, expand the target for which you want to set up CPU redundancy.
2. Right-click Hardware Configuration, point to **Redundancy**, and choose **Wizard**.

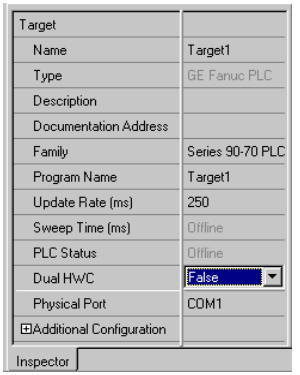


The Redundancy Wizards dialog box appears, with the **Set up a Primary Hardware Configuration for CPU Redundancy** option selected by default.



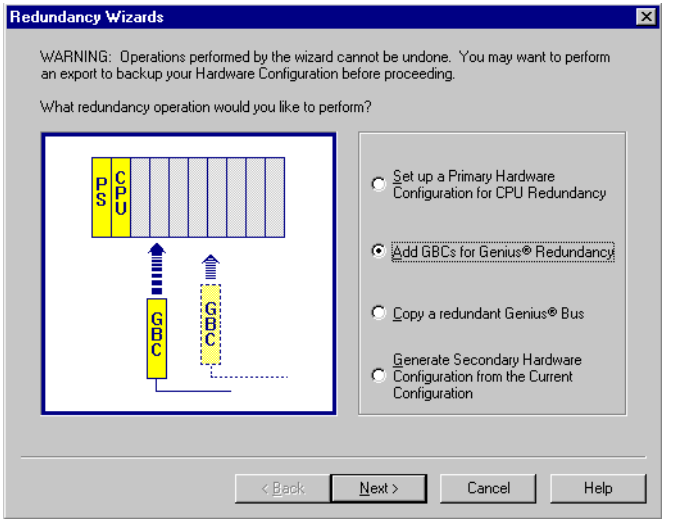
3. Click **Next** and follow the wizard to the end. The wizard adds a redundancy CPU and other redundancy modules to the configuration.

   The target property, **Dual HWC**, is now available in the Inspector and is set to False by default.



To add Genius Bus Controllers (GBC) to your system (PACSystems RX7i and Series 90-70)

1. In the Project tab of the Navigator, expand the target for which you want to set up Genius redundancy.
2. Right-click Hardware Configuration, point to **Redundancy**, and choose **Wizard**.
   The Redundancy Wizards dialog box appears.
3. Select **Add GBCs for Genius Redundancy**.

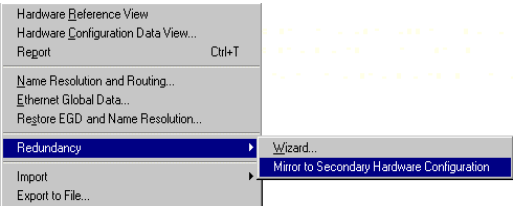


4. Click **Next** and follow the wizard to the end. The wizard enables you to select a Genius redundancy system and the location of the GBC modules. You can run this wizard multiple times to configure additional pairs of redundancy busses in the same system.

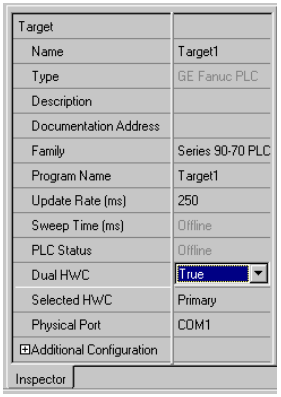
   Note: When you add a Genius Bus Controller (GBC) to a PACSystems RX7i or Series 90-70 rack, a new Genius bus network is automatically created and associated with that slot and GBC module. Up to 31 Genius I/O devices (blocks) can be connected to a GBC through its Genius bus.

To configure the secondary Hardware Configuration (PACSystems RX7i and Series 90-70)

1. In the Project tab of the Navigator, select the Hardware Configuration.
2. Right-click Hardware Configuration, point to **Redundancy**, and choose **Mirror to Secondary Hardware Configuration**.



A secondary rack system that is a copy of the primary rack system is generated. The target now displays two Hardware Configurations (HWCs), one labeled [Primary] and the other, [Secondary]. The Primary HWC is bolded because it is currently selected. The target property **Dual HWC** is now set to True. To select which controller to interact with, set the Selected HWC property, just below the Dual HWC property.



Note: You can mirror as many times as necessary to synchronize the two HWCs after modifying the primary HWC. Each time you mirror the primary HWC, the secondary HWC is updated to reflect those changes.

Want to know more? In the online help index, look up "Redundancy: an Overview".

PACSYSTEMS RX3I AND SERIES 90-30: DSM324I AND MOTION MATE DSM314 MOTION MODULES

The Series 90-30 PLC family supports various I/O modules (discrete input, discrete output, discrete mixed, analog input, analog output, and analog mixed), communication modules, intelligent modules, bus controllers, and motion modules. You configure Series 90-30 hardware as described in the section "Configuring PLC Hardware", on page 54.

PACSystems RX3i and Series 90-30 support some Motion modules that no other GE Fanuc PLC family supports: the DSM324i and the Motion Mate DSM314 (see page 133). Both are high performance, easy-to-use multi-axis motion control modules. Compatible with PLC logic solving and communications functions, the DSM314 supports the following servo types:

- Digital - GE Fanuc digital servo amplifiers and motors.
- Analog - GE Fanuc SL Series analog servos and third-party servos.

The DSM324i supports only the Digital servo types.

Both the DSM324i and the Motion Mate DSM314 modules have four axes that can be individually configured in Standard or Follower mode.

In order to perform motion programming in Logic Developer - PLC, you must program a DSM324i or a Motion Mate DSM314 module (see page 133).

To add a DSM324i or Motion Mate DSM314 module

1. With a rack of the Hardware Configuration node expanded, right-click the empty slot you want to add a DSM324i or a Motion Mate DSM314 module to and choose **Add Module**.

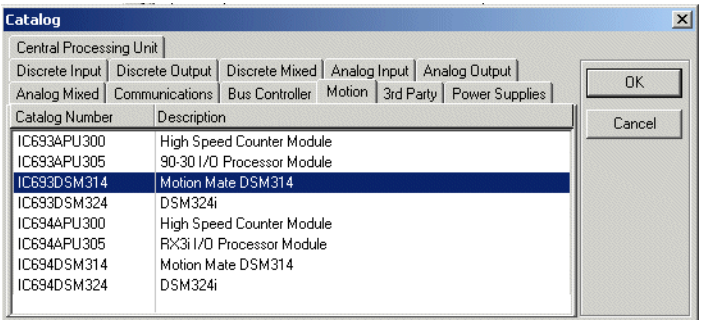
   The Catalog dialog box appears.

2. In the Catalog dialog box, click the Motion tab.

   A selection of motion modules appears in the Catalog dialog box.

3. From the list, select DSM324i or Motion Mate DSM314.

The following picture displays the selections available for a PACSystems RX3i.



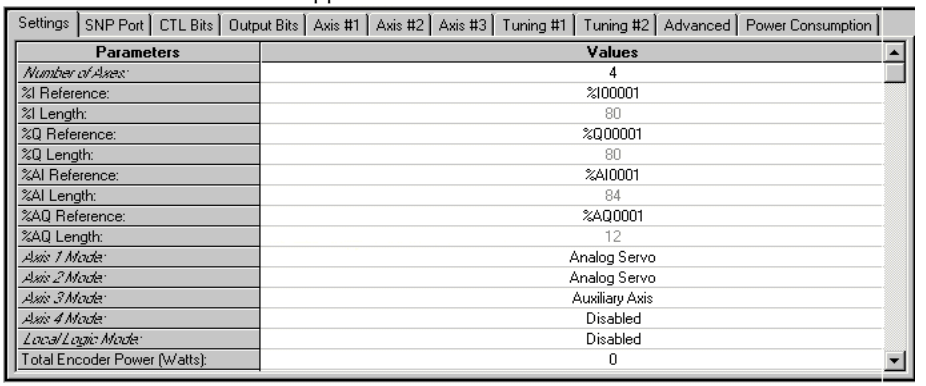
4. Click **OK**.

   As per the example illustration, an IC693DSM314 is added to the Hardware Configuration of your project.

To configure a DSM324i or a Motion Mate DSM314

1. In the Project tab of the Navigator, double-click the slot containing a DSM324i or Motion Mate DSM314.

   The Parameter editor appears.

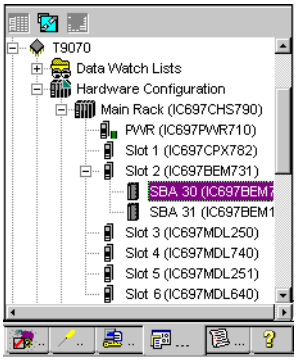


2. Configure the DSM324i or Motion Mate DSM314 by using the Parameter editor..

> **Want to know more?** In the online help index, look up "IC693DSM314: Motion Mate DSM314" and "IC693DSM324: DSM324i".

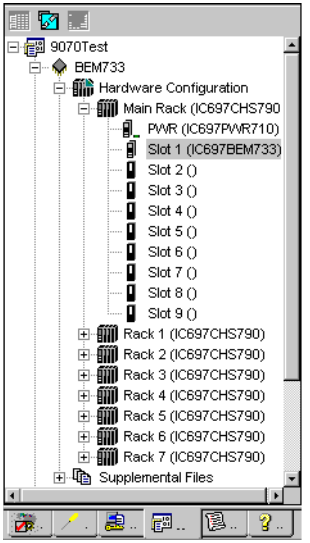
REMOTE I/O

Series 90-70 Genius Remote I/O Scanner



Navigator: Series 90-70 Remote I/O Scanner as a Genius device under a GBC

A Remote I/O Scanner (IC697BEM733) is a Genius device that is part of a Genius Bus system. As such, it is added as a Genius device to the Genius Bus Controller (GBC) and it is represented in Logic Developer - PLC by an SBA node under the GBC in a Series 90-70 target, as illustrated to the left. It is also represented by a separate target in the Machine Edition project, as illustrated below.



Navigator: Series 90-70 Remote I/O Scanner as a target and as a module in Slot 1 of the main rack

A Remote I/O Scanner resides in Slot 1 of the Series 90-70 main rack. I/O modules can be inserted into the other slots of the main rack and on all the slots of the expansion racks. As such, the Remote I/O Scanner is represented by a target in the Project tab of the Navigator, and is also represented by a module in slot 1 of the main rack of that target.

To add a Remote I/O Scanner

1. In the Project tab of the Navigator, expand the Hardware Configuration and expand the main rack.
2. Right-click the slot containing the Genius Bus Controller (GBC) and choose **Add Genius Device**.

   The Choose SBA dialog box appears.
3. Select a Serial Bus Address (SBA) for the Genius device and click **OK**.

   The Catalog appears.
4. Select the device you want to add and click **OK**.

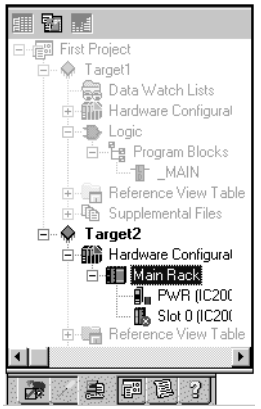
   The device is added to the bus configuration with the SBA you selected.
5. When prompted to add a new target for the Remote I/O Scanner, click **Yes**.

   A new GE Fanuc Remote I/O target appears in your project. This target represents the Remote I/O Scanner. In the Hardware Configuration of the new target, the Remote I/O Scanner also appears in Slot 1 of the main rack.
6. Configure the communication with the Remote I/O Scanner target.
7. Add and configure I/O modules for the rack system of the Remote I/O Scanner target.
8. Configure the Remote I/O Scanner module's parameters in Slot 1 of the main rack of the Remote I/O Scanner target.
9. Set up the Remote I/O Scanner Genius device in the same SBA node.

   The I/O Map of both Remote I/O Scanner configurations must match (at the Remote I/O Scanner target level and at the Remote I/O Scanner Genius Bus device level). Now whenever you update the I/O map on the Remote I/O Scanner, the I/O map is automatically updated to the GBC.
10. Download the Hardware Configuration of the new GE Fanuc Series 90-70 Remote I/O target.
11. Download the Hardware Configuration of the original GE Fanuc PLC target that contains the GBC.

> **Want to know more?** In the online help index, look up "IC697BEM733: Remote I/O Scanner".

VersaMax Remote I/O



Navigator: VersaMax Remote I/O

VersaMax remote I/O consists of a Network Interface Unit (NIU) (Ethernet, Genius or Profibus), and one or more I/O modules. You can use Logic Developer - PLC to configure this hardware. Once configured, the remote I/O can be controlled by a PLC or PC controller. The differentiating factor between a VersaMax remote I/O and a PLC is that a remote I/O is simply an input/output device with a communication interface. Unlike a PLC, a remote I/O has no CPU.

When working with the VersaMax Remote I/O Hardware Configuration, Logic Developer - PLC enables you to add racks, configure the power supply, and configure modules.

Remote I/O targets are generally added to a project when you create a project from a template. You can also add a GE Fanuc Remote I/O target to an existing project.

To create a project containing a GE Fanuc Remote I/O target from a template

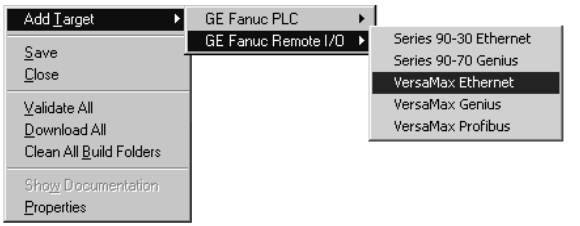
1. From the **File** menu, choose **New Project**.

   The New Project dialog box appears.
2. From the Project Template list, choose the GE Fanuc Remote I/O you want to add to your project.
3. Enter a descriptive Project Name.
4. Click **OK**.

   A new Remote I/O project is started.

To add a GE Fanuc Remote I/O target to an existing project

1. In the Project tab of the Navigator, right-click the Project node.
2. Point to **Add Target**, then to **GE Fanuc Remote I/O,** and choose the remote I/O you want to add.

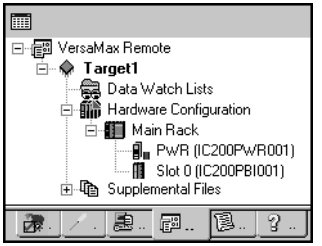


Note: The **Series 90-70 Genius** option is a Series 90-70 Remote I/O Scanner (see page 70).

To replace the power supply in your remote I/O configuration

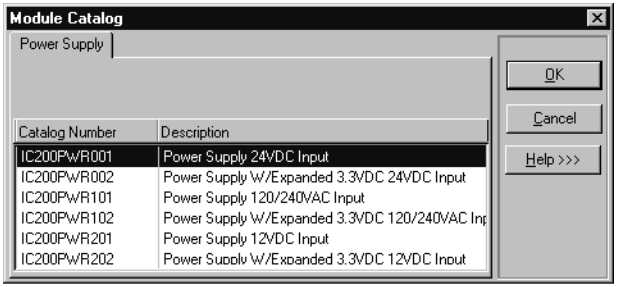
1. In the Project tab of the Navigator, expand the Hardware Configuration of the remote I/O.

   The Navigator displays the following:



2. Right-click the PWR slot and choose **Replace Module**.

   The Module Catalogue dialog box appears:

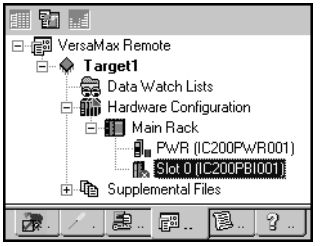


3. From the list, select the power supply that you want to configure for your system.
4. Click **OK**.

To add a new Carrier/Base to your VersaMax Remote I/O

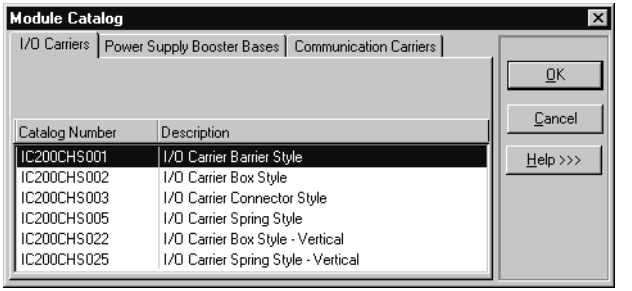
1. In the Project tab of the Navigator, select a remote I/O target.
2. Expand the Hardware Configuration node and the main rack.

   The Navigator displays the following:



3. Right-click the Slot 0 node and choose **Add Carrier/Base**.

   The Module Catalog dialog box appears.



4. Select the carrier/base that you want to add to the remote I/O target.

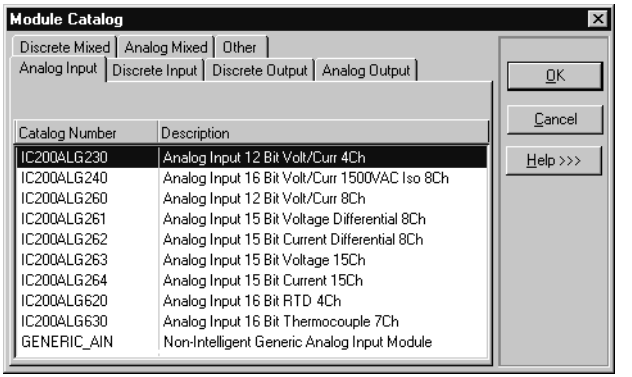
   Note: You can add a maximum of eight carrier modules to each VersaMax rack.

5. Click **OK**.

To add a module to a carrier/base

1. In the Project tab of the Navigator, expand the Remote I/O target you want to add a base to.
2. Double-click an empty carrier/base.

   The Module Catalog appears.



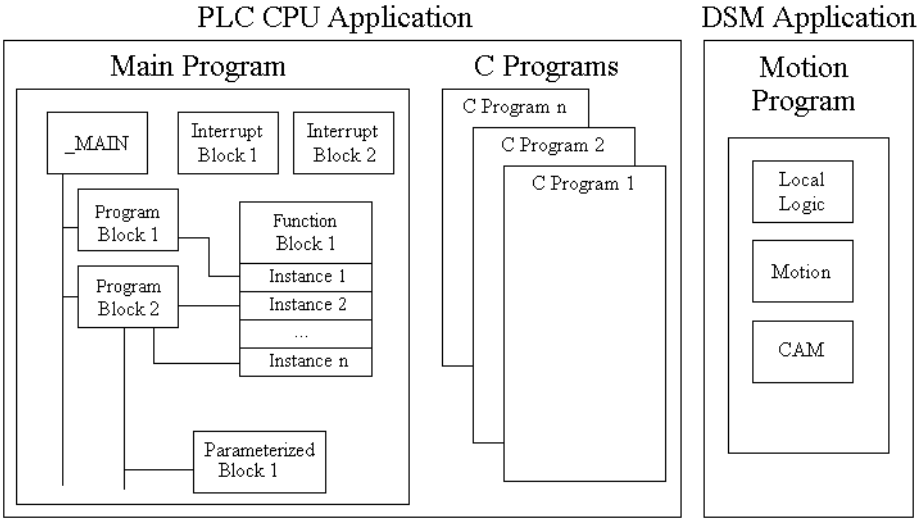
3. From the Module Catalog, select the module that you want to install.
4. Click **OK**.

> **Want to know more?** In the online help index, look up "Hardware Configuration".

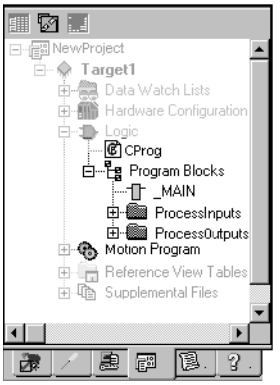
5 Logic Programs and Blocks

Program Types

For most GE Fanuc PLCs, all the logic that is downloaded to a PLC constitutes one program. However, there are cases when distinctions need to be made. Therefore, we can speak of a main program, a C program, and a Motion program.



Each C program is a named section of executable code. The main program and the Motion program contain blocks; that is, named sections of executable code that can be written in various languages.



Navigator: Project tab

Programs

The Project tab of the Navigator contains all of the programs:

- The main program and its blocks reside in the Program Blocks folder. You can create user-defined subfolders to further organize these blocks.
- C programs reside in the Logic folder.
- The Motion program resides in the Motion Program folder.

Main Program

Most GE Fanuc PLCs support only a main program. The main program consists of a block of logic named _MAIN and optionally, all of the following kinds of blocks of logic, found under the Program Blocks folder in the Logic folder of a GE Fanuc PLC target:

- Called blocks developed in various languages:
  - Ladder Diagram (LD) Blocks
  - Structured Text (ST) Blocks
  - Function Block Diagram (FBD) Blocks
  - Instruction List (IL) Blocks
  - C Blocks
- Called parameterized blocks (LD, ST, FBD, and/or C).
- Interrupt blocks (LD, ST, FBD, C, and/or IL) scheduled to execute at certain time intervals or when specific memory registers reach certain values.
- User-defined function blocks (UDFBs) in LD, ST, and FBD, of which independent instances with their own data structures and local memory can be used in logic.

Want to know more? In the online help index, look up "UDFB" then choose "User-Defined Function Blocks (UDFBs) (PACSystems)".

- Specialty function blocks, like HART utilities. Like UDFBs, you can create independent instances of specialty functions blocks with their own instance data (a structure variable). Unlike UDFBs, you cannot edit the logic of specialty function blocks because they are read-only and provided to you by GE Fanuc.

Want to know more? In the online help index, look up "Specialty Function Blocks (PACSystems)".

- All GE Fanuc PLCs support LD blocks, but only certain PLC families or models support one or more of the other kinds of blocks.

The _MAIN block can be in various languages, depending on the PLC family or model.

You download the main program to the PLC as logic; that is, after you initiate a download, a dialog box presents you with three Download to PLC options and you select the Logic option. If the Logic option is not present, select the Program option.

C Programs

Series 90-70 CPUs firmware version 6.00 and later support either the main program, or various C programs, or a combination of a main program and C programs. You download these various programs to the PLC together as logic; that is, after you initiate a download, a dialog box presents you with three Download to RAM options and you select the Logic option.

If there is no _MAIN block, then there is no main program. In this case, you would have only one or more C programs and possibly various standalone interrupt blocks (LD or C). These standalone interrupt blocks do not constitute a main program if there is no _MAIN block.

A C program can be named _MAIN only if the target contains no LD blocks.

A C program is not the same as a C block. For more information, see “C Programs”, on page 125.

Motion Program

In addition to the main program, Series 90-30 CPUs firmware version 10.00 and later, as well as PACSystems RX3i CPUs firmware version 2.80 and later, support a Motion program for use on a DSM324i or Motion Mate DSM314 module. The Motion program consists of:

- CAM profiles
- CAM blocks
- Local Logic block
- Motion blocks

You download the Motion program to the PLC as part of its Hardware Configuration (HWC); that is, after you initiate a download, a dialog box presents you with three Download to RAM options and you select the Hardware Configuration option. If you want to download the main program at the same time, you also select the Logic option. If neither option is present, select the Program

option to download both the Hardware Configuration and the Logic. For more information about the Motion Program, see page 133.

Number of Blocks in the Main Program

CPU Type	Number of Blocks	Maximum Number of Blocks
PACSystems	511 subroutine blocks plus one _MAIN block	512
Series 90-70	255 subroutine blocks plus one _MAIN block	256
Series 90-30	64 subroutine blocks plus one _MAIN block	65
VersaMax PLC and VersaMax Micro	64 subroutine blocks plus one _MAIN block	65
VersaMax Nano	8 subroutine blocks plus one _MAIN block	9
Series 90 Micro	1 _MAIN block	1

Scheduling Programs

On GE Fanuc PLCs that support only one program (the main program), the program as a whole cannot be scheduled. Every scan, logic execution begins with the _MAIN block. You can schedule individual interrupt blocks of logic but not the _MAIN block.

When you execute both a main program and a Motion program on a Series 90-30 PLC with a CPU firmware version 10.00 and later, or on a PACSystems RX3i with a firmware version of 2.80 or later, you cannot schedule the programs because they are executed concurrently in two different locations.

- In the CPU, every scan, logic execution begins with the _MAIN block of the main program.
- In the DSM324i or Motion Mate DSM314 module, the Motion program executes independently of CPU scan times.
- You can schedule individual interrupt blocks of logic in the main program (except the _MAIN block).

Series 90-70 CPUs firmware version 6.00 and later support scheduling programs. On targets that contain such CPUs, you can schedule any program, even the main program. That is, logic execution of any scan does not have to begin with the

_MAIN block of the main program: you can schedule any C program to begin the logic execution part of the scan. Within the main program, you can further schedule individual interrupt blocks of logic (except the _MAIN block). If you have no main program, that is, if your logic has no _MAIN block, you can still have stand-alone interrupt blocks that you can schedule individually.

To schedule the execution of programs on a Series 90-70

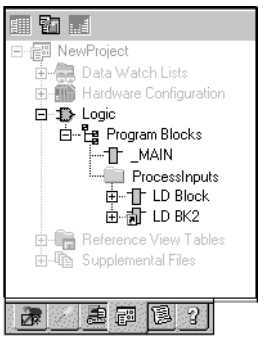
1. In the Project tab of the Navigator, right-click a C program or the _MAIN LD block, and choose **Properties**.

   The Inspector displays the program properties. The main program properties (Program Name, Scheduling, and Stack Size) are shown together with the _MAIN block's individual block properties.
2. In the Inspector, expand the **Scheduling** property.
3. Select a Schedule Mode.
4. Select the other properties used in the schedule mode you selected.
5. Click **OK**.

   Note: If you made any entries, they are validated when you click **OK**. You must correct any errors before you can click OK to close the dialog box.

> **Want to know more?** In the online help index, look up "Programs on GE Fanuc PLCs: an Overview".

To create a user-defined folder



1. In the Project tab of the Navigator, expand the target with the main program you want to organize and then expand the Logic folder.
2. Right-click the Program Blocks folder, point to **New**, and choose **Folder**.

   A new user-defined folder appears with a default name.
3. Optionally enter a new folder name, which must be unique among the folders directly under the parent folder.

> **Want to know more?** In the online help index, look up "organizing logic (main program of a PLC)".

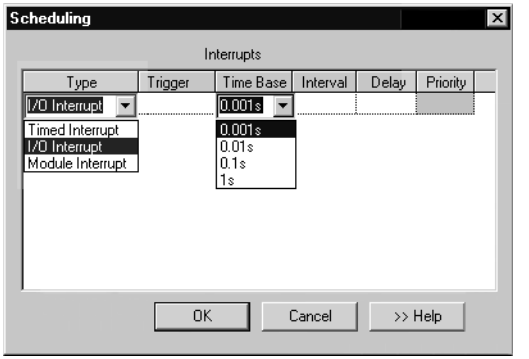
To schedule the execution of a block of logic

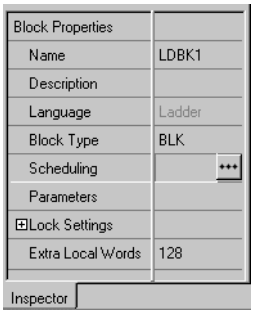
1. In the Project tab of the Navigator, right-click an existing LD, FBD, ST, C, or IL block, and choose **Properties**.

   The block's properties appear in the Inspector.

2. In the Inspector, click the ••• button in the Scheduling property.

   The Scheduling dialog box appears.





Inspector: LD block properties. .

3. Configure scheduling by entering values in each of the fields.

> **Want to know more?** In the online help index, look up "Scheduling execution of blocks of logic".

To control access to a block

1. Right-click an LD, ST, FBD, or IL block and choose **Properties**.
2. In the Inspector, expand the Lock Settings property.
3. In the Lock Type property, choose a setting from the list.
4. In the Password property, enter a password.

To search/replace in one block

1. Double-click an LD, ST, FBD, or IL block to open it.
2. From the **Search** menu, choose **Find** or **Replace**.

   A dialog box appears.
3. In the **Find what** or in the **Text to find** field, enter the text to find.
4. (Only if you want to replace text.) In the **Replace with** field or in the **New text** field, enter the text to replace the found text with.
5. Select or clear each search/replace option, as required.
6. Click **Find**, **Find next**, **Replace**, **Replace All**, **Close**, or **Cancel**, as required.

Note: Logic Developer - PLC enables you to search for some text in an entire target or portions thereof, which you can specify. Various options are available to narrow a search.

Want to know more? In the online help index, look up "Global" and select the "Conducting a Global Search" help topic.

Indirect References

The LD, ST, and FBD editors support indirect references. An indirect reference enables you to treat the value of a variable assigned to an instruction operand as a pointer to other data, rather than as actual data. Indirect references are sometimes referred to as relative pointers.

Indirect references can be used only as follows:

- With PACSystems CPUs and Series 90-70 CPUs.
- In the LD, ST, and FBD editors.
- With %R, %AI, %AQ, %P, %L, and %W memory areas.
- On instructions that support indirect references.

Notes for PACSystems

- The index for an indirect reference to %W is a 32-bit DWORD value.
- Indirect references cannot be used to address bits in 16-bit memory.
- Indirect references are not supported on symbolic variables or I/O variables.

Possible Uses

- To perform the same operation to many registers.
- To avoid repetitious logic within the application program.
- In loop situations where each register is incremented by a constant or by a value specified until a maximum is reached.

To assign an indirect reference

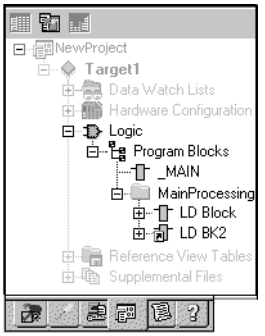
- **Where a variable operand is expected, type the @ symbol, followed by a valid reference address or variable name.**
  The LD, ST, or FBD editor converts a reference address to a variable name or the editor substitutes the name of a variable already mapped to the address.

Notes:

- The actual variable name does not contain the @ symbol. The @ symbol, when used at the start of a reference address or variable name, merely indicates that the address or variable must be treated as an indirect reference.
- The @ symbol can be thought of as an operator with the lowest precedence. In other words, MyArray[2,5] becomes a reference address that is used for the indirect reference. Another way to think of this is @(MyArray[2,5]) as opposed to (@MyArray)[2,5].

The following are valid indirect references:

- @R0001 (a variable name automatically created from an address)
- @MyArray[2, 5] (an array reference)
- @$MainSwitch (a universal variable)
- @MyTimer.PV (a structure element)



Navigator: Project tab

An LD Block

LD EDITOR

The Ladder Diagram (LD) editor is used to create programs with the Ladder Diagram programming language. LD logic graphically represents the programmed actions performed by a PLC as it executes.

The LD editor is cell-based with rungs constructed of horizontal sequences of instructions that are wired together. A given instruction and its operands can occupy one or more cells.

You can work with the LD editor while offline to edit a disk copy of a project, or while online to monitor the execution of the logic while you fine tune the project, for example by making word-for-word changes (see page 91).

You can customize the appearance and behavior of the LD editor by setting options (see page 23).

An LD block is a named section of LD Logic that is compiled and downloaded to the PLC represented by the associated target.

To customize the LD editor

1. In the Options tab of the Navigator, expand the Editors folder and then the Ladder folder.
2. Right-click a page (Confirmations, Editing, Font and Colors, or View), and choose **Properties**.

   The configurable options appear as properties in the Inspector.
3. In the Inspector, adjust settings as required.

To create an LD block

1. In the Project tab of the Navigator, right-click Program Blocks, point to **New**, and then choose LD Block.

   A new LD block with a default name is created.
2. Rename the block as desired.

To open an LD block for editing

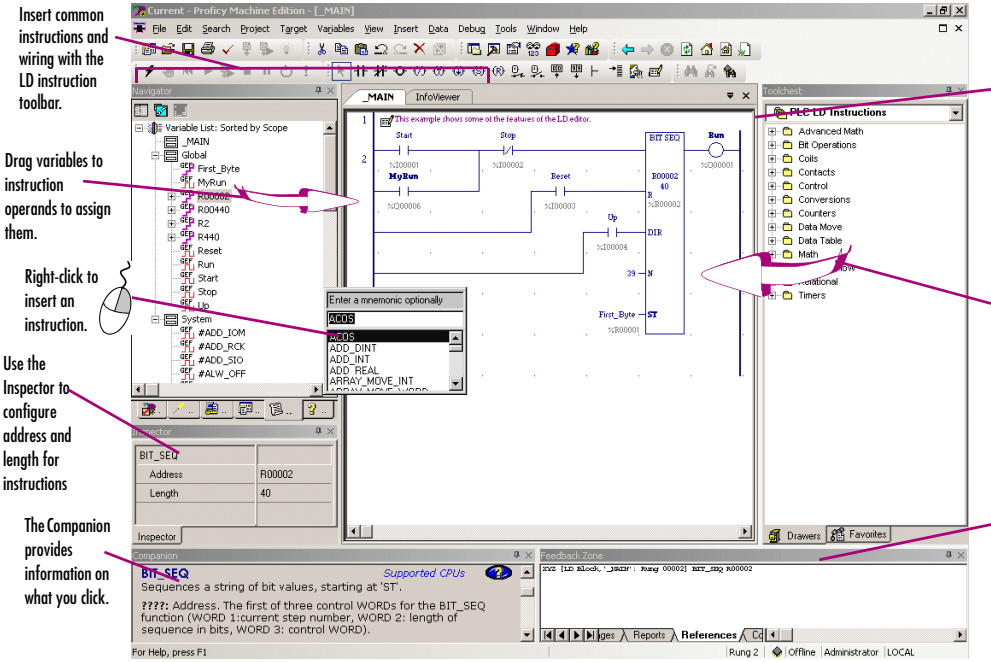
- In the Project tab of the Navigator, double-click an LD block.

  The block opens in the LD editor.

  Note: You can have multiple blocks open for editing. To navigate to another open LD block, click the tab that displays its name at the top of the editor window.

Working with the LD Editor Offline

When you are offline, there is no live communication between the LD editor and the target. Most logic development is done while offline. The following diagram illustrates some of the more common operations you can perform when you use the LD editor offline.



To insert an instruction

1. In the LD editor, right-click an empty cell and choose **Place Instruction**.

   A smart list appears listing all available instruction mnemonics.

2. Choose an instruction mnemonic from the list and press ENTER.

   Note: If you attempt to place an instruction that has inputs into the first column, it is pushed to the second column and a horizontal wire is inserted in the first column.

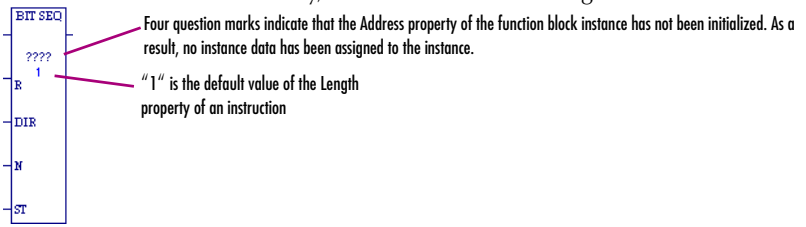
To assign instance data to a built-in function block instance and assign a length to an instruction

There are two methods.

First method

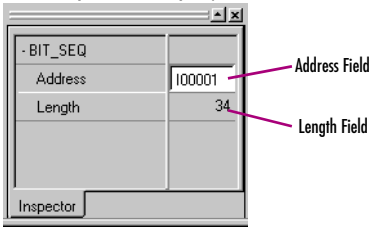
1. In the LD editor, insert a built-in function block or insert an instruction that requires a Length.

   For illustrative purposes, the following diagram displays a built-in function block instance that does double duty, because it also has a length.



2. Right-click the function block instance or instruction and choose **Properties**.

   The Inspector displays the instance's or instruction's properties.



3. In the Address property, enter a variable name or a reference address to specify the start of a memory block used for the instance data of the function block instance.

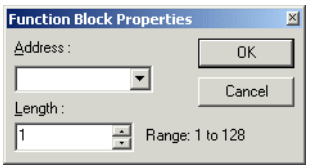
   Note: If you type a reference address, it is converted to a variable name automatically.

4. In the Length property, enter the number of bits or 16-bit registers on which the function block instance or instruction is to operate.

Second method

1. In the LD editor, insert a built-in function block or insert an instruction that requires a Length.
2. Double-click the function block instance or instruction or select it and press ENTER.

   When you double-click a function block instance, the Function Block Properties dialog box appears, as pictured below. When you double-click an instruction, the Function Properties dialog box appears, in which the Address property is unavailable.



3. In the Address property, enter a variable name or a reference address to specify the start of a memory block used for the instance data of the function block instance.

   Note: If you type a reference address, it is converted to a variable name automatically.
4. In the Length property, enter the number of bit or 16-bit registers on which the function block instance or instruction is to operate.

To assign variables to instruction operands

1. In the LD editor, double-click beside any operand of an instruction, or click there and press ENTER.

   A smart list appears prompting you to enter a variable name or reference address.
2. Type, or choose from the list, a variable name or reference address.

   If you enter a reference address, a variable name will be automatically substituted.

To check (validate) a single LD block

1. In the Project tab of the Navigator, expand the target that contains the block to check, and then expand the Logic folder.
2. Expand the Program Blocks folder and then, if the block resides in a user-defined folder, expand the user-defined folder.
3. Right-click the block and choose **Check Block**.

   Machine Edition checks the block for errors. Any messages or errors are displayed in the Build tab of the Feedback Zone.

Editing Logic as Text

You can enter LD logic into the LD editor by starting to type right into a cell. A smart list appears and you can continue to type your instructions and operands. When you press ENTER, the logic you typed appears in customary graphic form in the LD editor. For example, in the first cell of a new rung, if you type "AD MyDint1 MyDint2 Result12;AD MyDint3 MyDint4 Result34", the following logic appears in the LD editor:

A horizontal wire was inserted in the first cell to make room for the ADD_DINT instruction's operands and the operands were inserted in their proper places. In the keyboard entry, "AD" stood for "ADD_DINT", because ADD_DINT is the first available instruction that begins with "AD" in the smart list. A semicolon separates the ADD_DINT instructions. The required horizontal wires are supplied automatically.

> **Want to know more?** In the online help index, look up "Keyboard Method (to insert LD instructions)".

A powerful extension of typing logic into the LD editor is the ability to write LD logic in any text editor and copy and paste it into the LD editor, or to copy and paste logic from the LD editor to a text editor. You can copy an entire block of LD logic from the Navigator to any text editor, or copy sections of LD logic as text from the LD editor to any text editor. You can then edit the logic in the text editor and copy the edited logic back into the LD editor.

When LD logic is copied as text, a verbose format is used to make it easier to read, but when you edit the logic as text you can use the same shorthand used in the keyboard method of entering LD logic in the LD editor.

To copy an entire LD block as text

1. In the Project tab of the Navigator, expand the target and expand the Logic folder.
2. Expand the Program Blocks folder and, if the LD block resides in a user-defined folder, expand the user-defined folder.
3. Optionally right-click the LD block and choose **Check Block**. This validates the LD block. If the logic contains errors, error messages appear in the Feedback Zone. If the logic contains errors, fix them.
4. Right-click the LD block and choose **Copy**.

The content of the LD block is copied to the Windows Clipboard.

5. Paste the content of the Windows Clipboard into a text editor.

   The LD logic is pasted as text.

To copy a section of LD logic as text

1. Optionally right-click the LD block and choose **Check Block**. This validates the LD block. If the logic contains errors, error messages appear in the Feedback Zone. If the logic contains errors, fix them.
2. In the LD editor, select the cells that contain the logic you want to copy as text.
3. Right-click the selected logic and choose **Copy**.

   The LD logic is copied to the Windows Clipboard.
4. Paste the content of the Windows Clipboard into a text editor.

   The LD logic is pasted as text.

To copy text into the LD editor

1. In the text editor, select and copy the text representing LD logic to the Windows clipboard.
2. In the LD editor, do one of the following:

   - Select the cells that you want to overwrite.

   - or -

   - Click the cell that will be at the upper left corner of the LD logic you want to paste.
3. Right-click and choose **Paste**.

   Existing cells in the LD editor are overwritten with the content of the Windows clipboard.

To move or duplicate LD logic

1. In the LD editor, select a range of logic.
2. To move it:

   - Click the selection and drag it to a new location.

   - or -

   To duplicate it:

   - Press CTRL while clicking the selection, and drag the selection to where you want to place the duplicate.

   When you release the mouse button, the selection is respectively moved to the new location or a duplicate copy of the selected logic is placed in the new area.

Want to know more? In the online help index, look up "Copying LD Logic as Text: overview".

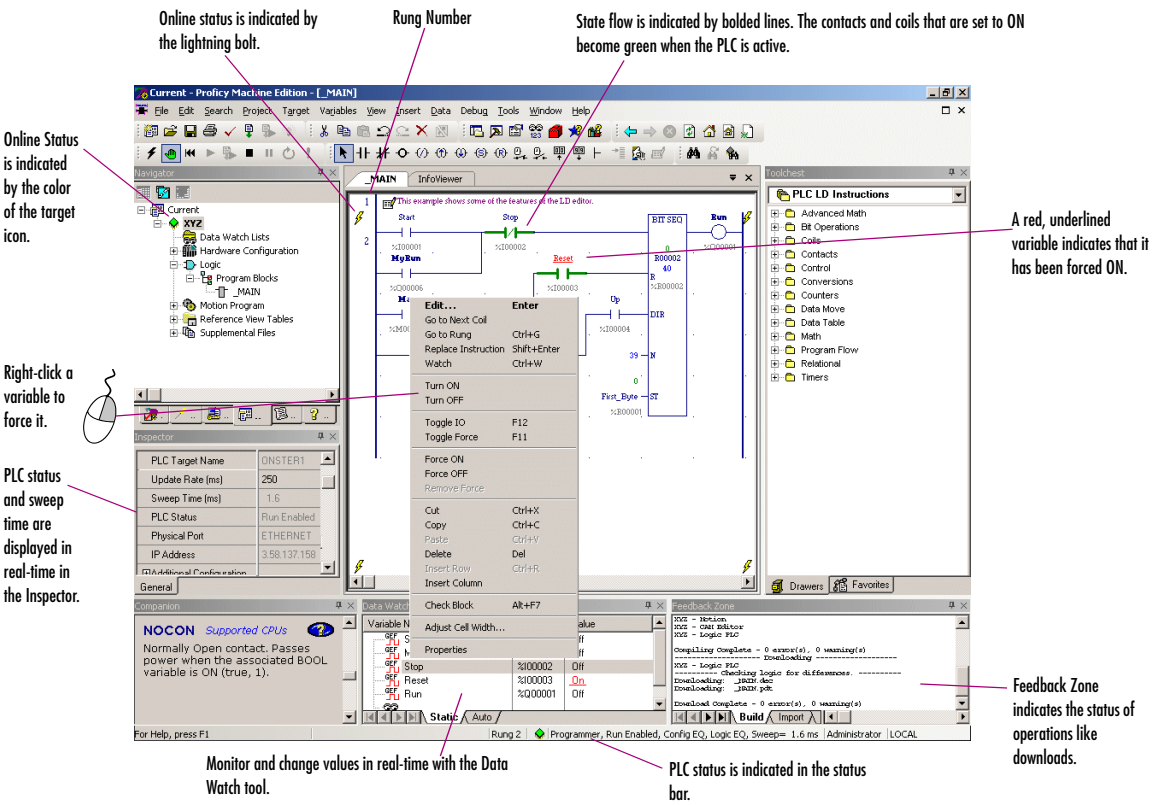
Working with the LD Editor Online

When online in monitor mode, you can monitor the PLC but cannot change anything in the PLC or edit your logic. The LD editor animates LD logic to reflect program execution in the PLC. Data values change typically every 250 milliseconds while coils and contacts indicate power flow. The number of milliseconds is set in the Update Rate target property. (See “To configure a PLC target”, on page 31.)

When you are online in programmer mode, you can edit your logic, make changes in the PLC, and monitor the PLC. Many GE Fanuc targets also support Run Mode Store; that is, they support the downloading of logic to a running target.

See “Offline, Online: monitor mode, programmer mode”, on page 39

.



Three methods are at your disposal to make changes in your LD logic and download the changes to an online running PLC:

- **Test Edit** (PACSystems only). See page 90.
- **Word-for-word changes:** see page 91.
- **Go not equal, keep working, and download changes:** see page 92.

You can also affect BOOL variables during execution. See page 92.

Test Edit (PACSystems only)

Test Edit enables you to perform transacted online programming. That is, it enables you to modify an LD block in the LD editor while seeing both the original and modified logic, test the new logic's execution on the PACSystems, and then either keep the modified logic or roll back to the original logic.

To begin editing in Test Edit

1. In the LD editor, start modifying the logic.
   The Logic Change Options dialog box appears.
2. Select **Enter Test Edit Mode** and click **OK**.
   The LD editor is placed in Online Edit mode, with some visual cues to remind you that a Test Edit session is in progress. Because Test Edit is an extension of the Run Mode Store (RMS), only changes supported for an RMS are supported in Test Edit mode. At any time, you can test the modified logic.

To begin testing the modified logic

1. Click anywhere inside the LD editor.
2. From the Debug menu, choose **Begin Test**.
   The modified logic is downloaded to the PACSystems, which then executes the modified logic. The original block of logic is retained in PACSystems memory as an inactive block.
   You now have three possible courses of action:
   - Cancel the test and continue editing the logic while in Test Edit mode.
   - Cancel Test Edit mode and restore the original logic in the PACSystems.
   - Accept the changes you tested and commit them to the PACSystems.

To cancel the test and continue editing the logic while in Test Edit mode

1. Click anywhere inside the LD editor.

2. From the Debug menu, choose **Cancel Test**.

   The modified logic block is deactivated in the PACSystems. The original block is activated and executed. The modified logic block retains its changes in the LD editor and you can continue editing it and test it again later.

To cancel Test Edit mode and restore the original logic in the PACSystems

1. Click anywhere inside the LD editor.
2. From the Debug menu, choose **Cancel Edit**.

   If you made no changes to logic, the Test Edit session ends immediately. If you made any changes to logic, the Cancel Test Edit dialog box appears.
3. In the Cancel Test Edit dialog box, select one of the following options and click **OK**.

   - **Restore original logic in editor and maintain equality:** The PACSystems deactivates and deletes the modified block of logic, and it activates and executes the original block. The LD editor discards the modified logic from the block, retaining only the block's original logic. Your Test Edit session ends. You are online in programmer mode and logic equal.
   - **Keep modified logic in editor and lose equality:** The PACSystems deactivates and deletes the modified block of logic, and it activates and executes the original block. The LD editor retains the modified block of logic and no longer displays the original logic where it was different. Your Test Edit session ends. You are online in programmer mode and logic not equal.

To accept the changes you tested and commit them to the PACSystems

1. Click anywhere inside the LD editor.
2. From the Debug menu, choose **Accept Edit**.

   The modified logic is committed to the PACSystems. The original logic is completely deleted from the PACSystems; it no longer exists as an inactive block. The Test Edit session is ended. You are online in programmer mode and logic equal.

Word-for-Word Changes

A word-for-word is a small change in logic made while online that fits in the same amount of memory as the original logic. For example, changing the type of contact or coil or changing an operand is usually a word-for-word change.

Word-for-word changes can be completed online to PACSystems, Series 90-30, Series 90-70, Series 90 Micro, and VersaMax modular PLCs. (VersaMax Nano / Micro PLCs do not support word-for-word changes.)

To make word-for-word changes

1. While online to a target PLC, make a change to LD logic that does not change the logic size.
2. Do one of the following:
   - If the Logic Change Options dialog box appears, select Word-for-word change and click **OK**.
     - or -
   - If the Word for Word Change dialog box appears, click **Yes**.

   The change is downloaded to the PLC.

Go not equal, keep working, and download changes

If you make changes in logic that do not qualify for Test Edit or a word-for-word change, or if you choose not to enter Test Edit mode or make a word-for-word change, logic becomes not equal.

To regain equality, you can download your changes to the PLC (see page 41) or upload the logic from the PLC (see page 42). You can generally download to the PLC whether the target is running or not. All PLC families support this Run Mode Store capability, but not every PLC in every family.

Affecting BOOL variables

To turn on/off or force a variable

- In the LD editor, right-click a BOOL variable anywhere in LD logic and choose **Force ON**, **Force OFF**, **Toggle Force**, **Turn ON**, **Turn OFF**, or **Toggle IO**.

  Note: Forcing a variable ON or OFF overrides any actions the application may take during runtime. That is, if a variable is forced OFF (0), but LD logic is trying to set it to ON (1), it remains set to OFF.

> **Want to know more?** In the online help index, look up "Test Edit: an Overview" and "working online with LD editor: overview".

LD Instructions

The following is a list of all LD instructions available in Logic Developer - PLC. Companion help indicates which PLCs support the instructions.

Want to know more? In the online help index, look up "LD Instructions".

Advanced Math

ACOS	COS	LN	SQRT_DINT	TAN
ASIN	EXP	LOG	SQRT_INT	
ATAN	EXPT	SIN	SQRT_REAL	

Bit Operations

AND_DWORD*	BIT_SET_DWORD*	NOT_WORD	SHIFTL_DWORD*
AND_WORD	BIT_SET_WORD	OR_DWORD*	SHIFTL_WORD
BIT_CLR_DWORD*	BIT_TEST_DWORD*	OR_WORD	SHIFTR_DWORD*
BIT_CLR_WORD	BIT_TEST_WORD	ROL_DWORD*	SHIFTR_WORD
BIT_POS_DWORD*	MASK_COMP_DWORD	ROL_WORD	XOR_DWORD*
BIT_POS_WORD	MASK_COMP_WORD	ROR_DWORD*	XOR_WORD
BIT_SEQ	NOT_DWORD*	ROR_WORD	

Coils

COIL	NEGCOIL	PTCOIL*P
CONTCOIL	NTCOIL*P	RESETCOIL
NCCOIL	POSCOIL	SETCOIL

Contacts

CONTCON	LOALR*	NOCON	POSCON*
FAULT*	NCCON	NOFLT*	PTCON*P
HIALR*	NEGCON*	NTCON*P	

An asterisk followed by the letter P (*P) indicates the instructions exclusive to PACSystems PLCs. An asterisk followed by "70" (*70) indicates the instructions exclusive to Series 90-70 PLCs. A simple asterisk (*) indicates the instructions shared by PACSystems and Series 90-70 PLCs and exclusive to them.

Control

DO_IO	EXIT_FOR*	PID_IND	SUS_IO*	SWITCH_POS*[P]
DRUM	FOR_LOOP*	PID_ISA	SUSP_IO_INTR	
END_FOR*	MASK_IO_INTR	SER	SVC_REQ	

Conversions

BCD4_TO_INT	DINT_TO_BCD8*	INT_TO_REAL	REAL_TO_WORD	UINT_TO_REAL*
BCD4_TO_REAL	DINT_TO_INT*	INT_TO_UINT*	TRUNC_DINT	WORD_TO_REAL
BCD4_TO_UINT*	DINT_TO_REAL	RAD_TO_DEG	TRUNC_INT	
BCD8_TO_DINT*	DINT_TO_UINT*	REAL_TO_DINT	UINT_TO_BCD4*	
BCD8_TO_REAL*	INT_TO_BCD4	REAL_TO_INT	UINT_TO_DINT*	
DEG_TO_RAD	INT_TO_DINT*	REAL_TO_UINT*	UINT_TO_INT*	

Counters

DNCTR	UPCTR

Data Move

BLK_CLR_WORD	BUS_TS_BYTE*[P]	DATA_INIT_UINT*	SWAP_WORD*
BLKMOV_DINT*	BUS_TS_DWORD*[P]	DATA_INIT_WORD*	VME_CFG_READ*[70]
BLKMOV_DWORD*	BUS_WRT_BYTE*[P]	MOVE_BOOL	VME_CFG_WRITE*[70]
BLKMOV_INT	BUS_WRT_DWORD*[P]	MOVE_DINT*	VME_RD_BYTE*[70]
BLKMOV_REAL	BUS_WRT_WORD*[P]	MOVE_DWORD*	VME_RD_WORD*[70]
BLKMOV_UINT*	COMM_REQ	MOVE_INT	VME_RMW_BYTE*[70]
BLKMOV_WORD	DATA_INIT_ASCII*	MOVE_REAL	VME_RMW_WORD*[70]
BUS_RD_BYTE*[P]	DATA_INIT_COMM*	MOVE_UINT*	VME_TS_BYTE*[70]
BUS_RD_DWORD*[P]	DATA_INIT_DINT*	MOVE_WORD	VME_TS_WORD*[70]
BUS_RD_WORD*[P]	DATA_INIT_DLAN	SHFR_BIT	VME_WRT_BYTE*[70]
BUS_RMW_BYTE*[P]	DATA_INIT_DWORD*	SHFR_DWORD*	VME_WRT_WORD*[70]
BUS_RMW_DWORD*[P]	DATA_INIT_INT*	SHFR_WORD	
BUS_RMW_WORD*[P]	DATA_INIT_REAL*	SWAP_DWORD*	

Data Table

ARRAY_MOVE_BOOL	FIFO_WRT_WORD*	SEARCH_GE_UINT*	SEARCH_NE_DINT
ARRAY_MOVE_BYTE	LIFO_RD_DINT*	SEARCH_GE_WORD	SEARCH_NE_DWORD*
ARRAY_MOVE_DINT	LIFO_RD_DWORD*	SEARCH_GT_BYTE	SEARCH_NE_INT
ARRAY_MOVE_DWORD*	LIFO_RD_INT*	SEARCH_GT_DINT	SEARCH_NE_UINT*
ARRAY_MOVE_INT	LIFO_RD_UINT*	SEARCH_GT_DWORD*	SEARCH_NE_WORD
ARRAY_MOVE_UINT*	LIFO_RD_WORD*	SEARCH_GT_INT	SORT_INT*
ARRAY_MOVE_WORD	LIFO_WRT_DINT*	SEARCH_GT_UINT*	SORT_DINT*
ARRAY_RANGE_DINT*	LIFO_WRT_DWORD*	SEARCH_GT_WORD	SORT_WORD*
ARRAY_RANGE_DWORD*	LIFO_WRT_INT*	SEARCH_LE_BYTE	TBL_RD_DINT*
ARRAY_RANGE_INT*	LIFO_WRT_UINT*	SEARCH_LE_DINT	TBL_RD_DWORD*
ARRAY_RANGE_UINT*	LIFO_WRT_WORD*	SEARCH_LE_DWORD*	TBL_RD_INT*
ARRAY_RANGE_WORD*	SEARCH_EQ_BYTE	SEARCH_LE_INT	TBL_RD_UINT*
FIFO_RD_DINT*	SEARCH_EQ_DINT	SEARCH_LE_UINT*	TBL_RD_WORD*
FIFO_RD_DWORD*	SEARCH_EQ_DWORD*	SEARCH_LE_WORD	TBL_WRT_DINT*
FIFO_RD_INT*	SEARCH_EQ_INT	SEARCH_LT_BYTE	TBL_WRT_DWORD*
FIFO_RD_UINT*	SEARCH_EQ_UINT*	SEARCH_LT_DINT	TBL_WRT_INT*
FIFO_RD_WORD*	SEARCH_EQ_WORD	SEARCH_LT_DWORD*	TBL_WRT_UINT*
FIFO_WRT_DINT*	SEARCH_GE_BYTE	SEARCH_LT_INT	TBL_WRT_WORD
FIFO_WRT_DWORD*	SEARCH_GE_DINT	SEARCH_LT_UINT*	
FIFO_WRT_INT*	SEARCH_GE_DWORD*	SEARCH_LT_WORD	
FIFO_WRT_UINT*	SEARCH_GE_INT	SEARCH_NE_BYTE	

Math

ABS_DINT*	ADD_UINT*	MOD_DINT	MUL_REAL	SUB_DINT
ABS_INT*	DIV_DINT	MOD_INT	MUL_UINT*	SUB_INT
ABS_REAL*	DIV_INT	MOD_UINT*	SCALE_DINT*[P]	SUB_REAL
ADD_DINT	DIV_MIXED*	MUL_DINT	SCALE_INT	SUB_UINT*
ADD_INT	DIV_REAL	MUL_INT	SCALE_UINIT*[P]	
ADD_REAL	DIV_UINT*	MUL_MIXED*	SCALE_WORD	

Program Flow

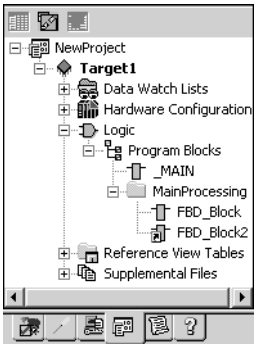
CALL	ENDMCR	JUMP	LABELN	V_WIRE
COMMENT	ENDMCRN	JUMPN	MCR	
END	H_WIRE	LABEL	MCRN	

Relational

CMP_DINT*	EQ_UINT*	GT_REAL	LT_INT	RANGE_DINT
CMP_INT*	GE_DINT	GT_UINT*	LT_REAL	RANGE_DWORD*
CMP_REAL*	GE_INT	LE_DINT	LT_UINT*	RANGE_INT
CMP_UINT*	GE_REAL	LE_INT	NE_DINT	RANGE_UINT*
EQ_DINT	GE_UINT*	LE_REAL	NE_INT	RANGE_WORD
EQ_INT	GT_DINT	LE_UINT*	NE_REAL	
EQ_REAL	GT_INT	LT_DINT	NE_UINT*	

Timer Function Blocks

OFDT_HUNDS	ONDTR_HUNDS	TMR_HUNDS	TOF
OFDT_SEC*	ONDTR_SEC*	TMR_SEC*	TON
OFDT_TENTHS	ONDTR_TENTHS	TMR_TENTHS	TP
OFDT_THOUS	ONDTR_THOUS	TMR_THOUS	



Navigator: Project tab

An FBD Block

FBD EDITOR

The Function Block Diagram (FBD) editor is used to create logic with the Function Block Diagram programming language. FBD is a process language. FBD logic graphically represents the programmed actions performed by a PLC as it executes.

The FBD editor is free form; that is, instructions and text boxes can be placed anywhere on the FBD editor where there is empty space. Sequences of instructions can be wired together horizontally and vertically.

You can work with the FBD editor while offline to edit a disk copy of a project, or you can edit an FBD block of logic online, but this causes the logic to become not equal until you download the FBD block.

You can customize the appearance and behavior of the FBD editor by setting options (see page 23).

An FBD block is a named section of FBD Logic that is compiled and downloaded to the PLC represented by the associated target.

To customize the FBD editor

1. In the Options tab of the Navigator, expand the Editors folder, then expand the Function Block Diagram folder.
2. Right-click the Colors and Preferences page and choose **Properties**.

   The configurable settings appear as properties in the Inspector.
3. In the Inspector, adjust settings as required.

To create an FBD block

1. In the Project tab of the Navigator, right-click the Program Blocks folder, point to **New**, and then choose **FBD Block**.

   A new FBD block with a default name is created.
2. Rename the block as desired.

To open an FBD block for editing

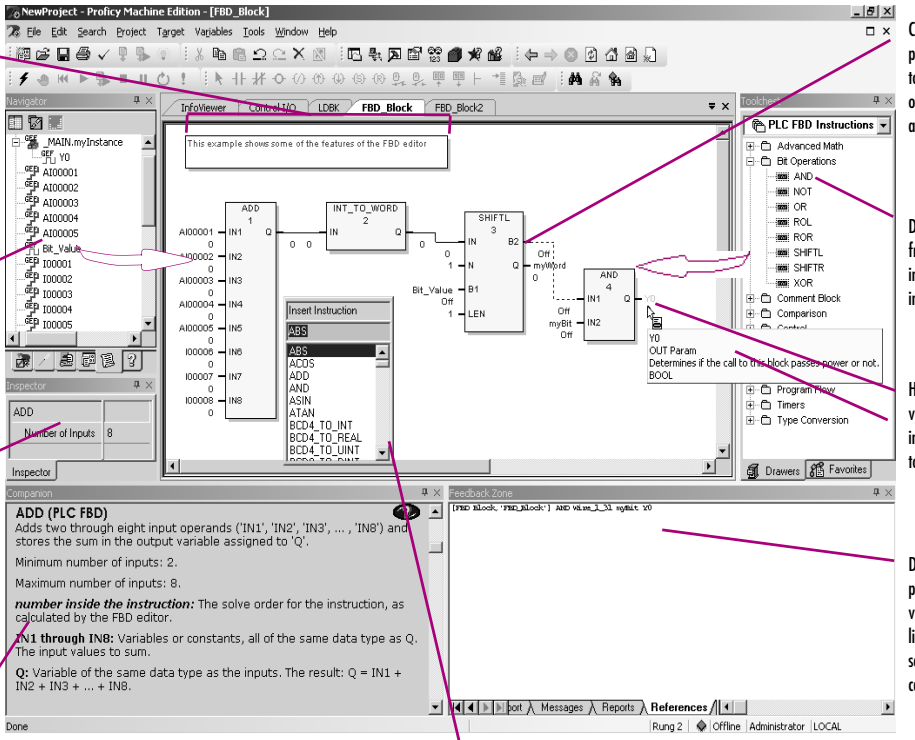
- In the Project tab of the Navigator, double-click an FBD block.

  The block opens in the FBD editor.

  Note: You can have multiple blocks open for editing. To navigate to another open FBD block, click the tab that displays its name at the top of the editor window.

Working with the FBD Editor Offline

While in offline mode, there is no live communication between the FBD editor and the target. Most logic development is done while offline. The following diagram illustrates some of the more common operations you can perform using the FBD editor offline.



To insert an instruction

1. In the FBD editor, right-click an empty cell and choose **Insert Instruction**.

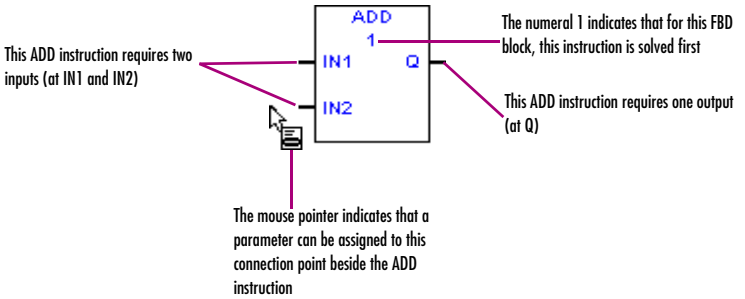
   A smart list appears listing all available instruction mnemonics.
2. Choose an instruction mnemonic from the list and press ENTER.

To assign a parameter beside an instruction

There are two methods to assign a parameter to an instruction.

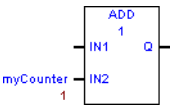
First method (from the FBD editor)

1. In the FBD editor, hover the mouse pointer immediately before an input connection point or immediately beyond an output connection point of an FBD instruction, so that the mouse pointer appears as .



2. Double-click the connection point.
3. In the smart list that appears, enter or choose an existing variable, constant, or expression to assign to the instruction parameter.

   In the FBD editor, the instruction may display as follows:



Note: Inside the rectangle of the ADD instruction, the "1" indicates the solve order of the instruction. Under the variable *myCounter*, the "1" is the initial value of *myCounter*.

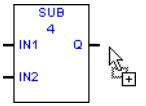
Second method (outside the FBD editor)

1. In the Variables tab of the Navigator, or the Data Watch tool, select a variable.

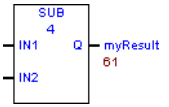
2. **Drag the selected variable to the connection point of an FBD instruction.**

3. **When the mouse pointer appears as , release the left mouse button to assign the variable to a parameter of the FBD instruction.**

   For example, the following diagram shows that you can assign a parameter to the FBD SUBTRACT instruction at the output connection point named Q.



   After you have assigned the parameter, the instruction may appear as follows:



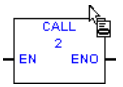
   Note: In the above diagram, "61" is the initial value of myResult. "4" inside the rectangle of the FBD instruction indicates its solve order.

> **Want to know more?** From the Help menu, select Search. Look up "solve order". Select "FBD Editor: an Overview".

To assign a parameter above an FBD instruction or function block instance

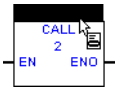
The FBD Call instruction and every function block instance (an instance of a counter, PID, timer, HART utility, or user-defined function block) require a parameter to be assigned immediately above the instruction or instance.

1. In the FBD editor, hover with the mouse pointer immediately above an FBD Call or function block instance so that the mouse pointer appears as .



2. When the mouse pointer appears as , click the left mouse button.

   The instruction or function block instance appears in the FBD editor with the parameter selected, as shown below.



3. Without moving the mouse pointer, click again.

   A smart list appears. If this is an FBD Call instruction, the smart list contains a list of all callable blocks of logic in your target. If this is an FBD function block instance, the smart list contains a list of variables that you can choose from to assign instance data to the function block instance. Normally you would choose or create instance data that is not used by any other function block instance, but you can assign the same instance data to multiple function block instances if you want.

4. In the smart list, enter or choose an existing block of logic, or enter a new variable, or enter or choose an existing variable.

   The block appears as the destination of the FBD Call or the variable is assigned as instance data to the function block instance.

To check (validate) a single FBD block

1. In the Project tab of the Navigator, expand the target that contains the block to check, and then expand the Logic folder.
2. Expand the Program Blocks folder and then, if the block resides in a user-defined folder, expand the user-defined folder.
3. Right-click the block and choose **Check Block**.

   Machine Edition checks the block for errors. Any messages or errors are displayed in the Build tab of the Feedback Zone.

To change the number of inputs for FBD instructions (ADD, AND, MUL, OR, SUB, XOR)

1. In the FBD editor, select an ADD, AND, MUL, OR, SUB, or XOR instruction.
2. In the Inspector, select the Number of Inputs property, and then enter or choose from the list the number of inputs you need.
3. If required, draw a wire or assign a variable or constant to the input and output parameters.

> **Want to know more?** In the online help index, look up "FBD Block". Click the Related Topics button and choose "FBD PLC logic: an Overview".

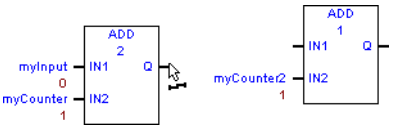
To draw a wire

Notes:

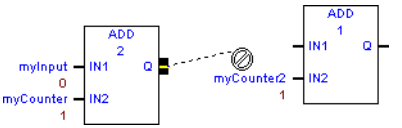
- At least two instructions must exist in the FBD editor.
- You cannot draw a wire to or from a Text box.

1. In the FBD editor, hover the mouse pointer above a connection point of an FBD instruction or function block instance.

   When the mouse pointer appears as , you can start a wire. See the diagram below.

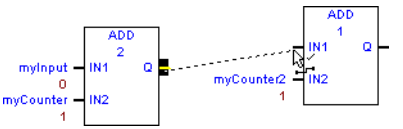


2. Hold down the left mouse button, and then drag the mouse pointer to a connection point of another FBD instruction



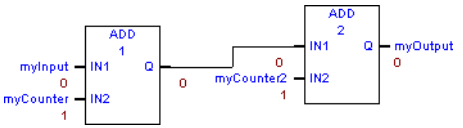
3. Continue to hold down the left mouse button and drag the mouse pointer to a connection point of another FBD instruction, until the mouse pointer appears as .

   This indicates that the current location is a valid end point for the wire. See the diagram below.



4. Release the left mouse button.

The newly drawn wire appears as a solid (analog) or a dashed (discrete) line in the FBD editor. See the diagram below.



Note: In the above diagram:

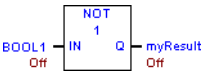
- The numbers "1" and "2" inside the rectangles have been reversed, indicating a change in the solve order.
- The zeroes under the wire are the value that is travelling over the wire. It's the same value at both ends.

To negate an FBD parameter

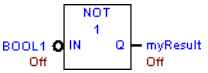
(For discrete parameters only.)

1. In the FBD editor, assign a BOOL parameter to a connection point of an FBD instruction that requires a BOOL variable.

   The diagram may appear as shown below.



2. Right-click a BOOL variable assigned to a connection point, for example, in the diagram above, *BOOL1*.
3. In the list box that appears, choose **Negate**. The diagram now appears as shown below.



Note: Negating a BOOL parameter while online causes logic to be not equal; turning a BOOL parameter ON/OFF while online does not affect logic equality.

To negate an FBD wire

(For discrete FBD wires only.)

1. If required, draw the FBD wire from a connection point that requires a BOOL variable, to a connection point that also requires a BOOL variable.

   If the wire is discrete, it appears as a dotted line .....!.....'.....'.

2. Right-click the wire, and then choose **Negate**.

   The wire now appears as ○.....!.....'.....'.

To move or duplicate FBD logic

1. In the FBD editor, select an FBD instruction or Text box.
2. To move it:

   - Click the selection and drag it to a new valid location. When you move an instruction, the parameters and wires assigned to it move along with it.

   - or -

   To duplicate it:

   Press CTRL while clicking the selection and drag the selection to where you want to place the duplicate. When you release the mouse button, a duplicate copy of the selected instruction is placed in the new area. When you duplicate an instruction, its parameters are also duplicated; wires are not duplicated.

Want to know more? From the Help menu, select Search. Paste "editing FBD blocks" (including the quotation marks) and then choose "Editing FBD blocks (PLC)".

To zoom in or zoom out an FBD block

Note: To zoom in an FBD block display in the FBD editor makes the FBD appear larger; to zoom out an FBD block display makes the FBD appear smaller.

1. Click anywhere in the FBD editor.
2. Do one of the following:

   - In the Inspector, expand the Diagram Settings group of properties, and then set the zoom property.

     - or -

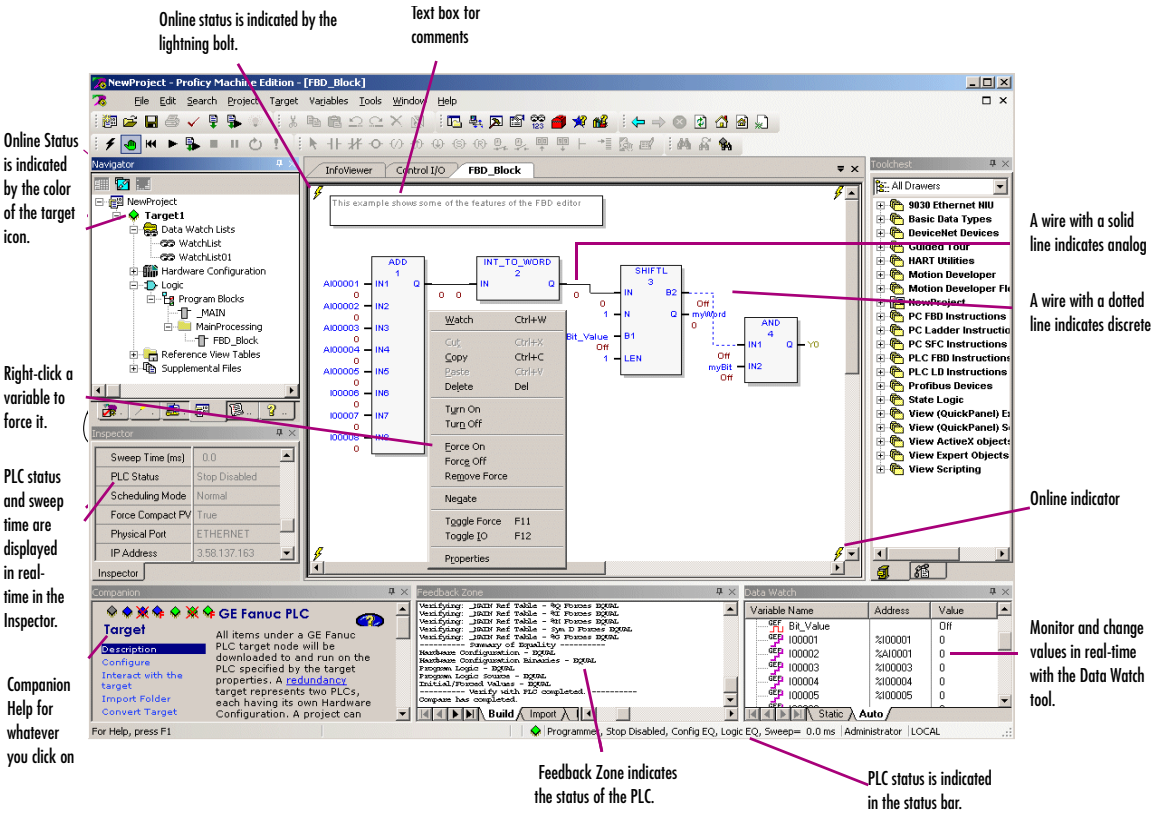
   - To zoom out, press the "-" key. To zoom in, press the "+" key.

Working with the FBD Editor Online

When online in monitor mode, you can monitor the PLC but cannot change anything in the PLC or edit your logic.

When you are online in programmer mode, you can edit your logic, make changes in the PLC, and monitor the PLC. GE Fanuc PACSystems targets also support Run Mode Store; that is, they support the downloading of logic to a running target.

See "Offline, Online: monitor mode, programmer mode", on page 39.



To turn on/off or force a variable

- In the FBD editor, right-click a BOOL variable anywhere in FBD logic and choose **Force ON**, **Force OFF**, **Toggle Force**, **Turn ON**, **Turn OFF**, or **Toggle IO**.

Note: Forcing a variable ON or OFF overrides any actions the logic may take during runtime. That is, if a variable is forced OFF (0), but FBD logic is trying to turn it ON (1), it stays OFF.

Want to know more? From the Help menu, select Search. Enter "Working with Variables and FBD Logic - Online (PLC)" (including the quotation marks).

FBD Instructions, Functions, and Function Blocks

(PACSystems only.) The following is a list of all FBD instructions, functions, and function blocks available in Logic Developer - PLC.

Want to know more? In the online help index, look up "Instructions (PLC FBD)".

Advanced Math

ABS	ASIN	COS	EXPT	LOG	TAN
ACOS	ATAN	EXP	LN	SIN	

Bit Operations

AND	OR	ROR	SHIFTR
NOT	ROL	SHIFTL	XOR

Comparison

CMP	GE	LE	NE
EQ	GT	LT	RANGE

Control

DO_IO	PID_IND	SUS_IO	SVC_REQ
MASK_IO_INTR	PID_ISA	SUSP_IO_INTR	

Counters

DNCTR	UPCTR

Data Move

BUS_RD	BUS_RMW_WORD	BUS_WRT
BUS_RMW_BYTE	BUS_TS_BYTE	COMM_REQ
BUS_RMW_DWORD	BUS_TS_WORD	MOV

Math

ADD	MOD	NEG	SUB
DIV	MUL	SCALE	

Program Flow

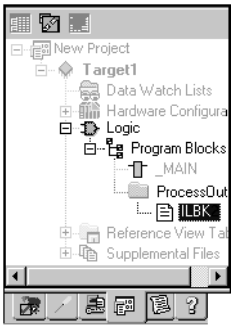
CALL

Timer Function Blocks

OFDT_HUNDS	ONDTR_HUNDS	TMR_HUNDS	TOF
OFDT_SEC	ONDTR_SEC	TMR_SEC	TON
OFDT_TENTHS	ONDTR_TENTHS	TMR_TENTHS	TP
OFDT_THOUS	ONDTR_THOUS	TMR_THOUS	

Type Conversion

BCD4_TO_INT	DINT_TO_BCD8	INT_TO_BCD4	REAL_TO_DINT	UINT_TO_DINT
BCD4_TO_REAL	DINT_TO_DWORD	INT_TO_DINT	REAL_TO_INT	UINT_TO_INT
BCD4_TO_UINT	DINT_TO_INT	INT_TO_REAL	REAL_TO_UINT	UINT_TO_REAL
BCD8_TO_DINT	DINT_TO_REAL	INT_TO_UINT	TRUNC_DINT	UINT_TO_WORD
BCD8_TO_REAL	DINT_TO_UINT	INT_TO_WORD	TRUNC_INT	WORD_TO_INT
DEG_TO_RAD	DWORD_TO_DINT	RAD_TO_DEG	UINT_TO_BCD4	WORD_TO_UINT



Navigator: Project tab

An IL Block

IL EDITOR

Instruction List (IL) is a programming language specified by the IEC 61131-3 standard. This text language is accumulator-based and much like the assembly languages used for programming microprocessors. The instructions executed by an IL block modify or use an accumulator that is located in PLC memory. Two types of accumulators are defined: one analog accumulator for numeric and bitwise operations and eight boolean accumulators for discrete logic to support eight levels of nested boolean expressions. The IL editor is free-form with an option to apply a standard formatting rule. The appearance and behavior of the IL editor is user-configurable.

Note: Only Series 90-30, VersaMax PLCs, and VersaMax Nano/Micro support IL logic.

To configure accumulators

1. In the Project tab of the Navigator, right-click the Program Blocks folder and choose **Properties**.

   The Inspector displays the Accumulator Address properties.
2. In the Boolean Start property, enter the reference address of the first of eight PLC memory locations to use for boolean accumulators.

   The ending address is calculated automatically. The memory area must be %T, %M, or %Q.
3. In the Analog Start property, enter the PLC memory locations to use for the analog accumulator.

   The ending address is calculated automatically. The memory area must be %R, %AI or %AQ.

To create an IL block

1. In the Project tab of the Navigator, right-click the Program Blocks folder, point to **New**, and choose **IL Block**.

   An empty IL block with the default name "ILBk*n*" is added to the folder, where *n* represents a unique number.
2. Rename the block as desired.

To open an IL block for editing

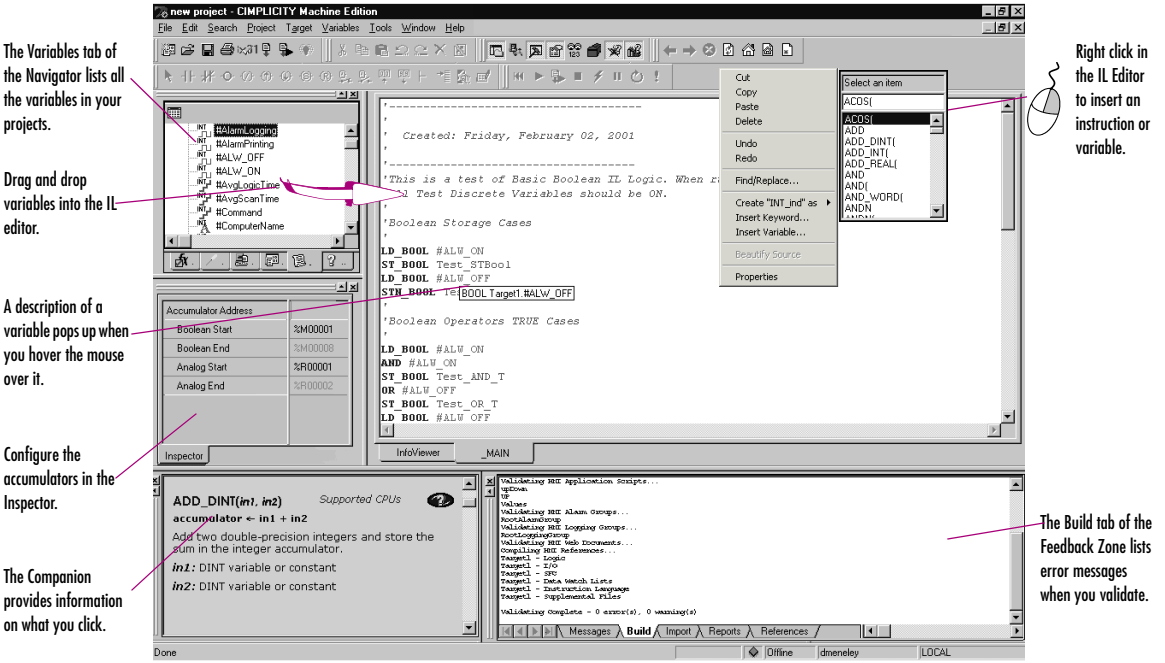
- In the Project tab of the Navigator, right-click an IL Block and choose **Open**.

  The block opens in the IL editor.

Note: You can have multiple blocks open for editing. To navigate to another open IL block, click the tab displaying its name at the top of the editor window.

Working with the IL Editor Offline

Most project development is carried out while offline from the target PLC. Editing while offline provides maximum flexibility and enables you to interact with the Machine Edition tools as shown in the following diagram.



To insert an instruction

1. In the IL editor, right-click and choose **Insert Keyword**.

   A smart list appears listing all available instruction mnemonics.
2. From the list, select an instruction and press ENTER.

   The instruction is inserted in your logic.

To assign operands to an instruction

1. In the IL editor, right-click and choose **Insert Variable**.

   A smart list appears showing all your defined variables.

2. Type, or choose from the list, a variable name or reference address and then press ENTER.

   The name appears in your logic.

   Note: If you entered a reference address or a new variable name, you must create a variable from it.

To create a variable from a reference address

- In the IL editor, right-click a reference address, point to **Create "name" as**, and then choose a data type.

  A variable is created and a default name is applied. For example, if the reference address is %R0032, the auto-created variable is named R00032.

To create a variable from a name

1. In the IL editor, right-click a name, point to **Create "name" as**, and then choose a data type.

   A variable is created with the name you right-clicked.

2. Map the variable to PLC memory. (See page 21.)

To move or duplicate IL logic

1. In the IL editor, select a range of logic.
2. To move it, click the selection and drag it to a new location. To duplicate it, press CTRL while clicking the selection, and drag the selection to where you want to place the duplicate.

   When you release the mouse button, the selection is respectively moved to the new location or a duplicate copy of the selected logic is placed in the new area.

To insert an inline comment

1. In the IL editor, click where you want to insert an inline comment.
2. Type an apostrophe (') followed by comment text.
3. Press ENTER to complete the comment.

To insert a block comment

1. In the IL editor, click where you want to insert a block comment.
2. Type (* followed by comment text.

   A block comment can contain any number of characters and can span multiple lines.

3. Type *) to complete the block comment.

To reformat IL logic

- In the IL editor, right-click and choose **Beautify Source**.

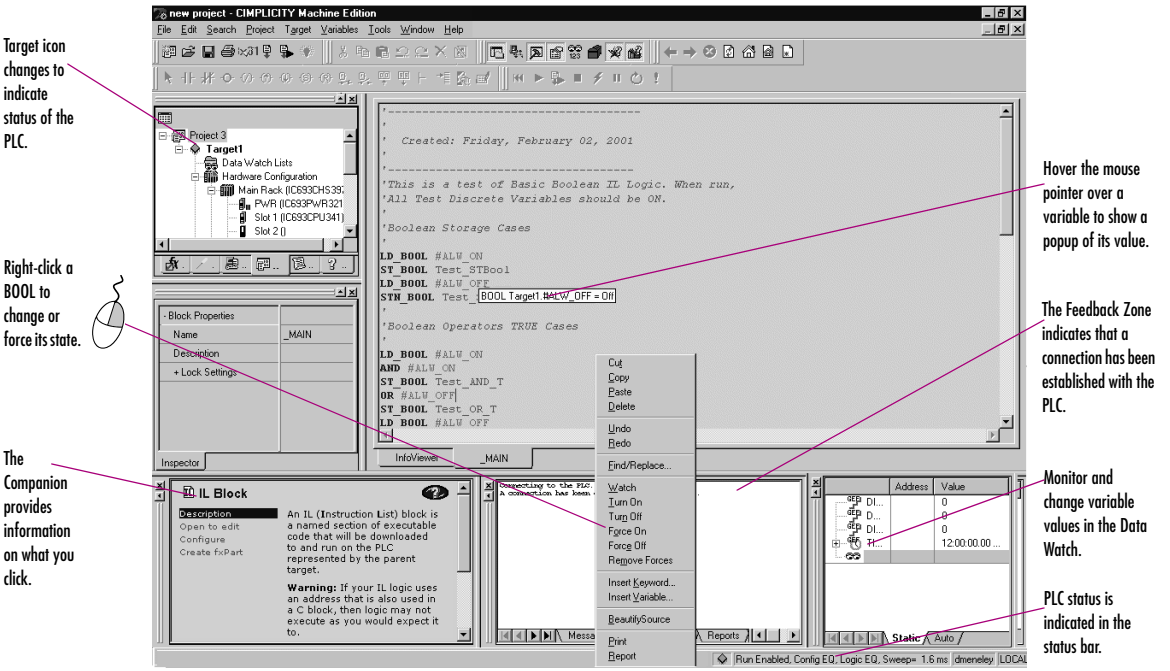
  The entire content of the IL editor is reformatted according to the default formatting (indentation) rules.

Working with the IL Editor Online

When online in monitor mode, you can monitor the PLC but cannot change anything in the PLC or edit your logic.

When you are online in programmer mode, you can monitor the PLC, edit your logic, and make changes in the PLC. Many GE Fanuc targets also support Run Mode Store; that is, they support the downloading of logic to a running target

See "Offline, Online: monitor mode, programmer mode", on page 39.



To monitor a data value

- Click anywhere in the IL editor and hover the mouse pointer over a variable.

  A tooltip appears, showing the variable's current value. This value, however, does not update automatically: you must move the mouse pointer away and back over the variable to update the value.

To change a BOOL variable's state

- In the IL editor, right-click the BOOL variable whose value you want to change and choose **Turn On**, **Turn Off**, or **Toggle IO**.

The state of the variable in the target PLC changes when the command is received. It remains in that state until acted on by the PLC's logic.

To force a BOOL variable's state

- In the IL editor, right-click a BOOL variable and choose **Force On**, **Force Off**, or **Toggle Force**.

The state of a forced variable remains unchanged, regardless of any actions by PLC logic.

To remove the force from a BOOL variable

- In the IL editor, right-click a BOOL variable and choose **Remove Forces**.

The state of the forced variable is controlled by PLC logic from now on.

Downloading Online Changes to a Target GE Fanuc PLC

You can change your IL logic while online, but you must download the changes to the PLC to maintain equality. Some GE Fanuc PLCs support the downloading of changed IL logic even when they are executing.

To make changes to IL logic and download them to a running target PLC (if the target PLC supports it)

1. While online to a target PLC, make a change to IL logic.
2. In the Project tab of the Navigator, right-click the target and choose **Download to PLC**.

You are prompted to confirm a Run Mode Store. That is, the altered IL block will be downloaded to the running PLC without stopping it.

Want to know more? In the online help index, look up "Downloading to running GE Fanuc PLCs".

IL Instructions

The following is a list of all IL instructions available in Logic Developer - PLC. Companion help indicates which PLCs support the instructions.

Basic Instructions

ADD	GT	LT	OR	ST_DINT	XOR
AND	LD_BOOL	MOD	ORN	ST_INT	XORN
ANDN	LD_ENO	MUL	PT	ST_REAL	
DIV	LD_INT	NE	R	ST_WORD	
EQ	LDN_BOOL	NOT	S	STN_BOOL	
GE	LE	NT	ST_BOOL	SUB	

Advanced Math

ACOS	COS	LN	SQRT_DINT	TAN
ASIN	EXP	LOG	SQRT_INT	
ATAN	EXPT	SIN	SQRT_REAL	

Bit Operations

AND_WORD	BIT_SEQ	MASK_COMP_DWORD	OR_WORD	SHIFTL_WORD
BIT_CLR_WORD	BIT_SET_WORD	MASK_COMP_WORD	ROL_WORD	SHIFTR_WORD
BIT_POS_WORD	BIT_TEST_WORD	NOT_WORD	ROR_WORD	XOR_WORD

Control

DO_IO	DRUM	PID_IND	PID_ISA	SER	SVC_REQ

Conversions

BCD4_TO_INT	DINT_TO_REAL	RAD_TO_DEG	REAL_TO_WORD	WORD_TO_REAL
BCD4_TO_REAL	INT_TO_BCD4	REAL_TO_DINT	TRUNC_DINT	
DEG_TO_RAD	INT_TO_REAL	REAL_TO_INT	TRUNC_INT	

Counters

DNCTR	UPCTR

An asterisk (*) indicates the instructions exclusive to VersaMax PLCs.

Data Move

BLK_CLR_WORD	BLKMOV_REAL	COMM_REQ	MOVE_BOOL	MOVE_REAL	SHFR_BIT
BLKMOV_INT	BLKMOV_WORD		MOVE_INT	MOVE_WORD	SHFR_WORD

Data Table

ARRAY_MOVE_BOOL	SEARCH_EQ_INT	SEARCH_GT_DINT	SEARCH_LT_BYTE	SEARCH_NE_WORD
ARRAY_MOVE_BYTE	SEARCH_EQ_WORD	SEARCH_GT_INT	SEARCH_LT_DINT	
ARRAY_MOVE_DINT	SEARCH_GE_BYTE	SEARCH_GT_WORD	SEARCH_LT_INT	
ARRAY_MOVE_INT	SEARCH_GE_DINT	SEARCH_LE_BYTE	SEARCH_LT_WORD	
ARRAY_MOVE_WORD	SEARCH_GE_INT	SEARCH_LE_DINT	SEARCH_NE_BYTE	
SEARCH_EQ_BYTE	SEARCH_GE_WORD	SEARCH_LE_INT	SEARCH_NE_DINT	
SEARCH_EQ_DINT	SEARCH_GT_BYTE	SEARCH_LE_WORD	SEARCH_NE_INT	

Math

ADD_DINT	DIV_DINT	MOD_DINT	MUL_DINT	SCALE_WORD*	SUB_REAL
ADD_INT	DIV_INT	MOD_INT	MUL_REAL	SUB_DINT	
ADD_REAL	DIV_REAL	MUL_INT	SCALE_INT*	SUB_INT	

Program Flow

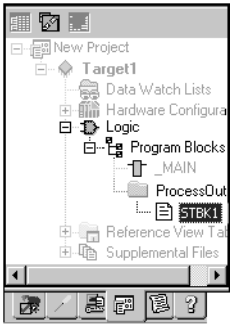
CAL	CALCN	END_MCRN	JMPC	MCRN	RETC
CALC	END	JMP	JMPCN	RET	RETCN

Relational

EQ_DINT	GE_DINT	GT_DINT	LE_DINT	LT_DINT	NE_DINT	RANGE_DINT
EQ_INT	GE_INT	GT_INT	LE_INT	LT_INT	NE_INT	RANGE_INT
EQ_REAL	GE_REAL	GT_REAL	LE_REAL	LT_REAL	NE_REAL	RANGE_WORD

Timer Function Blocks

OFDT_HUNDS	ONDTR_HUNDS	TMR_HUNDS
OFDT_TENTHS	ONDTR_TENTHS	TMR_TENTHS
OFDT_THOUS	ONDTR_THOUS	TMR_THOUS



Navigator: Project tab

An ST Block

ST EDITOR

Structured Text logic is a programming language specified by the IEC 61131-3 standard. The PLC Structured Text (ST) Editor is a free-form text editor for editing Structured Text logic in the Machine Edition environment. With the ST editor, you can work on a disk copy of a structured text block (offline) or monitor the execution of an ST block running in the PLC (online). You can edit an ST block online, but this causes the logic to become unequal until you download the ST block.

ST is a high-level language composed of various operators and functions. ST logic is supported by PACSystems PLCs.

To customize the ST Editor

1. In the Options tab of the Navigator, expand the Editors folder and then the Text folder.
2. Right-click a page (Colors or Preferences), and choose **Properties**.
   The configurable settings appear as properties in the Inspector.
3. In the Inspector, adjust the settings as required.

To create an ST block

1. In the Project tab of the Navigator, expand the Logic folder, right-click the Program Blocks folder or a user-defined folder, point to **New**, and then choose **ST block**.
   A new ST block with a default name is created under the Program Blocks folder or under a user-defined folder.
2. (Optional.) Rename the block.

To create a parameterized ST block

1. Create an ST block.
2. Do one of the following:
   - In the Inspector, set parameters for the block.
     - or -
   - Set the block's Block Type property to **Parameterized Block**, and then optionally schedule the parameterized block.

   Notes:
   - By assigning parameters to the block, its Block Type property is automatically changed to Parameterized Block.

- A parameterized block cannot have both interrupt scheduling and parameters.

Want to know more? In the online help index, look up "Parameterized Block (PLC)".

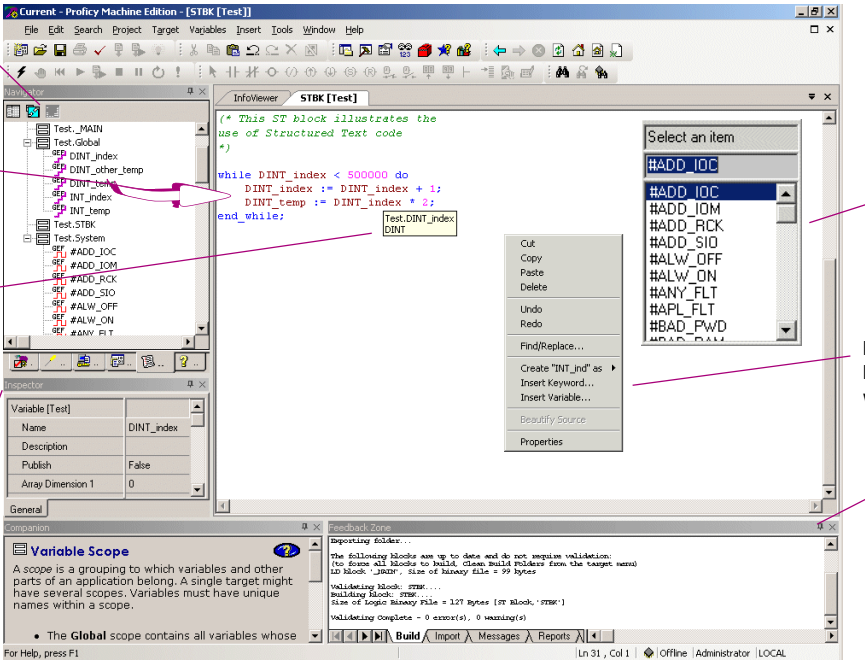
To open an ST block for editing

- In the Project tab of the Navigator, under the Logic folder, under the Program Blocks folder, or under a user-defined folder, double-click an ST block.

The ST block opens in the ST editor.

Working with the ST Editor Offline

The ST editor interacts with the Machine Edition tools to provide maximum flexibility when editing a program. The following diagram illustrates some of the operations you can perform.



To insert an ST variable or keyword

1. In the ST editor, right-click and choose **Insert Variable** or **Insert Keyword**.
   A smart list appears, prompting you to enter the name of an existing variable or keyword.
2. Type, or choose from the list, the item you want, and then press ENTER.
   The variable or keyword appears in the ST editor.
   Notes:
   - Parameters and constants must be manually entered.
   - When entering an indirect reference as an operand, prefix the variable name with @ after selecting it from the smart list, for example, @IntVar. Indirect referencing is available for all register references (%R, %P, %L, %W, %AI, and %AQ).

Want to know more? In the online help index, look up "Indirect References".

To create a variable from a name

- In the ST editor, right-click a name that was entered as an operand, point to **Create "[*name*]" as**, and then choose a data type from the list that appears.
  A new variable of the specified data type is created and substituted for the name at every occurrence in the ST block. The name of the new variable is identical with the original name.

To insert a line comment

1. In the ST editor, click where you want a line comment to begin.
2. Type an apostrophe (') or // followed by comment text.
3. Press ENTER to complete the line comment. The line comment ends when a new line starts.

To insert a block comment

1. In the ST editor, click where you want a block comment to begin.
2. Type **(*** followed by comment text. A block comment can contain any number of characters and can span multiple lines.
3. Type ***)** to complete the block comment.

To select a range of ST logic

1. In the ST editor, click at the beginning of the range.
2. Press and hold shift and then click at the end of the range. All logic within the range is selected.

   Tips:

   - You can also click and drag from the beginning of the range to the end.
   - After selecting a range of logic, you can click anywhere in it and drag it to another area in the ST editor.

To move or duplicate ST logic

1. In the ST editor, select a range of logic.
2. To move it, click the selection and drag it to the new location. To duplicate it, press CTRL while clicking the selection, and then drag the selection to where you want to place the duplicate.

   When you release the mouse button, the selection is respectively moved to the new location or a duplicate copy of the selected logic is placed in the new area.

To locate all occurrences of a variable

1. In the Feedback Zone, click the References tab.
2. Click a variable anywhere it appears in ST logic, in the Variables tab of the Navigator, or in the Data Watch tool.

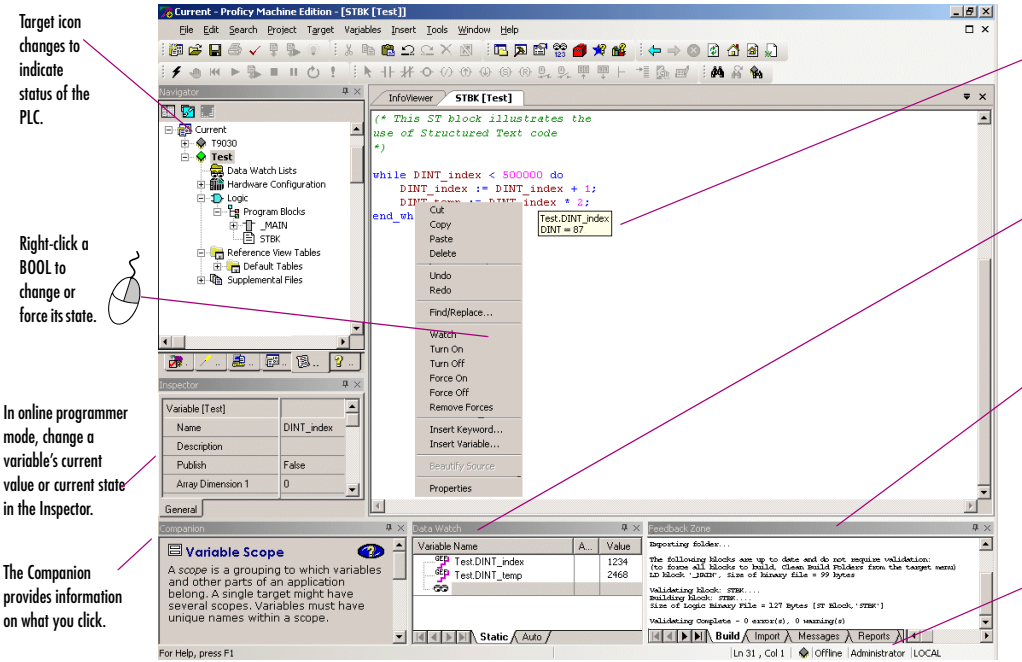
   Each occurrence of the variable in your project (for a given target) is listed in the **References** tab of the Feedback Zone.

Working with the ST Editor Online

When online in monitor mode, you can monitor the PLC but cannot change anything in the PLC or edit your logic.

When you are online in programmer mode, you can monitor the PLC, edit your logic, and make changes in the PLC. PACSystems also support Run Mode Store; that is, they support the downloading of logic to a running target.

See “Offline, Online: monitor mode, programmer mode”, on page 39.



To view a variable value

- In the ST editor, hover the mouse pointer over a variable to see its value. A tool tip displays the variable name, address, description, data type, and current value. If this is a BOOL variable, the tool tip also displays if it has been forced On (1) or Off (0).

To view the value of an ST parameterized block parameter

1. In the Project tab of the Navigator, expand the Program Blocks folder.
2. If the block that contains the call to the ST parameterized block whose parameter you want to monitor resides in a user-defined folder, expand the user-defined folder.
3. Right-click the block that contains the call and choose **Open**.
   The block appears in the appropriate editor.
4. In the editor, right-click the block call to the ST parameterized block and choose **Open Block**.
   The called ST parameterized block appears in the ST editor.
5. In the ST editor, hover the mouse pointer over the parameterized block parameter (variable) to see its value.
   Depending on the context of the parameter, the tool tip displays the following items:

   - If the text refers to a parameter instead of a variable, and the block was opened from a CALL instruction (in an LD, ST, or FBD block), then if possible, the parameter is associated with the variable used in the call the block was opened from. In this case, the tool tip displays the parameter name, address of the variable, description of the parameter, data type, and value of the variable.
   - If the text refers to a parameter instead of a variable, and the block was not opened from a CALL instruction (in an LD, ST, or FBD block), then the tool tip displays the parameter name, description of the parameter, and data type only.

   In either case, if this is a forced BOOL variable, the tool tip also displays if the variable has been forced On (1) or Off (0).

   Note: The tool tip displays a variable's current value only if logic is equal; the tool tip doesn't display the current value if logic is not equal.

To change a BOOL variable's state

- In the ST editor, right-click the BOOL variable whose value you want to change and choose **Turn On**, **Turn Off**, or **Toggle IO**.

  The state of the variable in the target PLC changes when the command is received.

To force a BOOL variable's state

- In the ST editor, right-click a BOOL variable and choose **Force On**, **Force Off**, or **Toggle Force**.

  The state of a forced variable remains unchanged, regardless of any actions by PLC logic.

To remove the force from a BOOL variable

- In the ST editor, right-click a BOOL variable and choose **Remove Forces**.

  The state of the unforced variable is controlled by PLC logic from now on.

ST Statements, Functions, and Function Blocks

The following is a list of all ST statements, functions, and function blocks available in Logic Developer - PLC. InfoViewer Help indicates which PLCs support the statements, functions, or function blocks.

> **Want to know more?** In the online help index, look up "ST PLC Logic: an Overview", and then click the "PLC" link at the top, right corner.

Statements

:= (Assignment)	Comment	Function block invocation		Repeat ... until
Block call	Exit	Function call		Return
Case	For ... do	If ... then ... elsif		While ... do

Advanced Math

ACOS	COS	LN	SQRT_DINT	TAN
ASIN	EXP	LOG	SQRT_INT	
ATAN	EXPT, **, or ^	SIN	SQRT_REAL	

Bitwise Operators

AND	NOT	OR	XOR

Control

DO_IO	MASK_IO_INTR	SUS_IO	SUSP_IO_INTR	SVC_REQ	SWITCH_POS

Conversions

ANGLES	INT to BCD4, DINT, REAL, UINT, or WORD
BCD4 to INT, REAL, or UINT	REAL to DINT, INT, or UINT
BCD8 to DINT or REAL	TRUNC_DINT, TRUNC_INT
DINT to BCD8, DWORD, INT, REAL, or UINT	UINT to BCD4, DINT, INT, REAL, or WORD
DWORD to DINT	WORD to INT or UINT

Data Move

COMM_REQ

Math

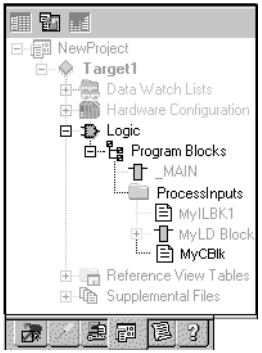
+ (addition)	- (subtraction)	- (negation)	* (multiplication)	/ (division)	mod (modulo)
ABS_DINT	ABS_INT	ABS_REAL	SCALE_DINT	SCALE_INT	SCALE_UINT

Relational Operators

= (Equal)	>= (Greater than or equal)	> (Greater than)	<= (Less than or equal)	< (Less than)	<>, != (Not equal)

Timer Function Blocks

TOF	TON	TP



Navigator: Project tab

A C Block

C BLOCKS

A C block is an independent section of executable code written in the C programming language that is downloaded to and executed on the target PLC. C blocks are created externally using GE Fanuc's C Programming Toolkit and then imported into a project. A C block compiled for PACSystems has a *.gefelf* extension. A C block compiled for Series 90-70 or Series 90-30 has a *.exe* extension. C Blocks can be called as a subroutine from another block (LD, ST, FBD, or IL) but cannot call another block.

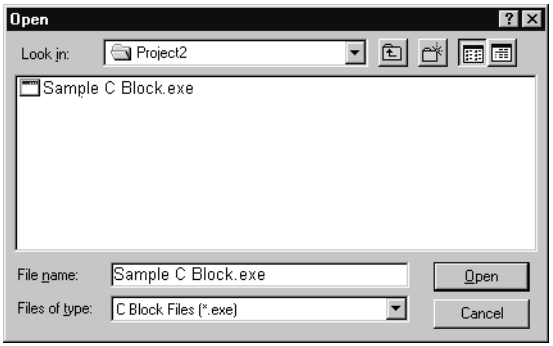
For more information on developing C Blocks, refer to *C Programmer's Toolkit for Series 90 PLCs* (GFK-0646) and *PACSystems RX7i C Toolkit User's Guide* (GFK-2259).

Working with C blocks

To import a C block

1. In the Project tab of the Navigator, right-click the Program Blocks folder and choose **Add C block**.

   The Open dialog box appears.



2. Browse to the .exe or .gefelf file you want to import and click **Open**.

   The selected file is added to the Program Blocks folder with the same name as the .exe or .gefelf file.

   Note: You can then move the C Block to a user-defined folder.

Want to know more? In the online help index, look up "C block" and "User-Defined Folder".

To set a C block's parameters

Note: This procedure applies only to C blocks that were written to require parameters for use on PACSystems or Series 90-70 targets.

1. In the Project tab of the Navigator, right-click the C block and choose **Properties**.

   The Inspector displays the block's properties.

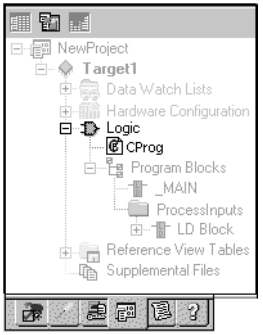
2. In the Inspector, select the Parameters property and click ....

   The Parameters dialog box appears.

3. Type a name and description for each required input and output parameter.

   For information on the required parameters, refer to the written documentation for the C block. The names you enter will display in the CALL instruction that calls the C block. The names and descriptions will display in a tooltip when you hover the mouse pointer over the CALL instruction.

Want to know more? In the online help index, enter "Parameterized" and select the "C Block" help topic.



Navigator: Project Tab

A C Program

C PROGRAMS

Note: C programs are supported only on Series 90-70 CPUs, firmware release 6.00 and later.

A C program is an independent section of executable code, written in C language, that is downloaded to and executed on the associated target PLC. To develop a C program, use GE Fanuc's C *Programmer's Toolkit for Series 90 PLCs User's Manual* (GFK-0646). You then import it into a project.

A C program has access to all the % reference tables of the PLC except for the _MAIN LD block's %P memory and the %L memory of any other LD block. A C program can also call any of the numerous PLC-embedded functions that are included in the C Programmer's Toolkit.

A C program cannot be called as a subroutine. Execution is controlled only through scheduling. A C program cannot call another block as a subroutine.

By setting the parameters of a C program, you enable it to access memory directly. When a C program begins to execute, it reads the data for all the parameters and makes a copy of the data. If the C program's execution is interrupted or time-sliced over multiple scans, the C program, when it resumes execution, uses the copy of the data that it made when it began to execute.

C programs can coexist with a main program and with State Logic on a Series 90-70 PLC. For such a target, the State Logic is compiled into a C program. This C program is named _MAIN if there is no LD logic; however, if State Logic coexists with LD logic, the State Logic is compiled into a C program named _STATE and one LD block must be named _MAIN.

Working with C Programs

Setting a C program's parameters

Note: You must provide a list of all the input and output parameters that the C program requires. The main program does not use parameters.

1. In the Project tab of the Navigator, right-click a C program and choose **Properties**.

   The Inspector displays the C program's properties.

2. In the Inspector, select the Parameters property and click ....

   The Parameters dialog box appears.

3. On the Input and Output tabs, enter up to 8 input and up to 8 output parameters. Each parameter has its own row on the tab. For each parameter, double-click the following cells and enter the required data:
   - **Name:** The parameter's name.
   - **Type:** The parameter's data type.
   - **Length:** The length of the input or output reference.
   - **Variable:** The first data item associated with the parameter.
   - **Description:** (Optional) The parameter's description.

Want to know more? In the online help index, choose "C Program".

6 Ethernet Global Data (EGD)

Ethernet Global Data is a mechanism that enables one CPU, referred to as a *producer*, to share a portion of its internal memory with one or more other CPUs, referred to as *consumers*, at a regularly scheduled periodic rate. Such a snapshot of internal memory, mediated by an Ethernet interface, is referred to as an exchange. An exchange is identified by a unique combination of three identifiers:

- The Producer ID (the producer's IP address)
- The Exchange ID (the exchange's identifier)
- The Adapter Name (the Ethernet interface identifier)

Exchanges vs. Pages

Some EGD tools group multiple produced exchanges into what is called a page, and some EGD tools can consume such pages. Logic Developer - PLC, however, does not consume or produce pages. It consumes and produces exchanges.

- When you add a consumed exchange in Logic Developer - PLC, if an EGD tool has published produced pages that contain multiple exchanges to the EGD Configuration Server, the produced exchanges are extracted from the pages and presented for your selection as exchanges.
- Logic Developer - PLC does not group multiple produced exchanges into a produced page. EGD tools that consume pages see the exchanges produced by Logic Developer - PLC as produced pages that contain only one exchange each.

Integration with the EGD Configuration Server

Logic Developer - PLC is integrated with the EGD Configuration Server. The EGD Configuration Server is a central repository of EGD configuration information used to facilitate the sharing of information between EGD tools in order to assist with the configuration of EGD.

After configuring a produced exchange on any EGD tool (such as the EGD component in Logic Developer - PLC), you can publish the configuration of the produced exchanges to the EGD Configuration Server. After this, Logic Developer - PLC, when connected to the server, can obtain the produced exchange configuration from the server and automatically create a consumed exchange that

matches the produced exchange exactly. At this point, the only thing left to do is to map the variables of the exchange to memory on the EGD consumer, or to replace some or all of the unmapped variables with mapped variables that already exist on the EGD consumer.

Using the EGD Configuration Server also provides you with extra EGD validation that is unavailable without the server.

Using the EGD Configuration Server also enables you to use EGD Signatures. When a PACSystems is configured to use signatures, then every scan, it compares the signature of every consumed exchange with the signature of its corresponding produced exchange to determine whether the configuration of the consumed exchange matches the configuration of the produced exchange; if not, the exchange is not consumed.

The EGD Configuration Server keeps track of which produced exchanges published on the server are consumed by EGD devices that are connected to the server. You can then obtain an Unconsumed Data report. You can also obtain reports that compare the produced or consumed exchange information on your computer with that on the server.

The EGD Configuration Server is not the authoritative source of the EGD configuration. It does not store past versions of EGD configurations. It contains only the latest snapshots of the EGD configurations published to it. The EGD tools are the authoritative source, while the server simply enables the tools to share information. It is possible to delete the server contents completely and restore them again by using the tools in a two-step process: publish the authoritative information from all the tools to the server, and then have each tool read the information it needs from the server and confirm to the server that it has received it.

Want to know more? In the online help index, look up "Ethernet Global Data in Logic Developer - PLC: an Overview" and "EGD Configuration Server: Integration with Logic Developer - PLC".

Integration with the EGD Management Tool (EMT)

Logic Developer - PLC is also integrated with the EGD Management Tool (EMT). You can open the EMT from within Logic Developer - PLC and vice-versa.

The EGD Management Tool (EMT) provides a high-level view of the EGD system published on the EGD Configuration Server, including a graphical display of individual EGD systems. The EMT provides validation of the EGD configuration.

The EMT enables the EGD configuration of devices by launching the appropriate EGD Configuration Tool from the graphical display.

Want to know more? Refer to the online help in the EGD Management Tool.

Logic Developer - PLC implementation of EGD: The EGD component

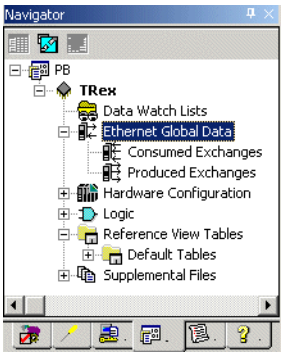
The first step in using Ethernet Global Data (EGD) in a GE Fanuc PLC or ENIU target is adding the EGD component. In a PLC target, you can do so even if the CPU presently configured in the target does not support EGD, as long as the target belongs to one of the PLC families that support EGD.

This flexibility enables you to start configuring EGD even if the hardware specifications of your system have not been determined.

To add the EGD component

- In the Project tab of the Navigator, right-click a GE Fanuc PLC target that belongs to one of the PLC or ENIU family types that support Ethernet Global Data (EGD), point to **Add Component**, and choose **Ethernet Global Data**.

  The Ethernet Global Data folder is added to the target. The folder contains an empty Consumed Exchanges folder and an empty Produced Exchanges folder.



Navigator: After the Ethernet Global Data component has been added to the target

Note: For Ethernet modules that support EGD uploads, the EGD component is automatically added when you upload the Hardware Configuration and EGD configuration. The EGD component is also automatically added to a target when you import a folder that contains an Ethernet Global Data (EGD) configuration or import a Hardware Configuration that contains an EGD configuration. In such cases, the Consumed Exchanges and Produced Exchanges folders are automatically populated.

To install the EGD Configuration Server

1. Insert the Machine Edition 5.50 supplemental CD into the computer on which you want to install the EGD Configuration Server.

   The computer must be connected to an Ethernet network. It may be local or remote.
2. On the Machine Edition 5.50 supplemental CD, navigate to the EGD Server and Tools folder.
3. Double-click the EgdCfgServer Setup.msi file.

The EGD Configuration Server is installed on the computer.

Note: The client library required for your copy of Logic Developer - PLC to communicate with the EGD Configuration Server is automatically installed as part of the installation of Logic Developer - PLC.

To install the EGD Management Tool (EMT) on your computer

1. Insert the Machine Edition 5.50 supplemental CD into your computer.
2. On the Machine Edition 5.50 supplemental CD, navigate to the EGD Server and Tools folder.
3. Double-click the EgdManagementTool Setup.msi file.
   The EGD Management Tool is installed on your computer.

To configure communications with the EGD Configuration Server

1. In the Options tab of the Navigator, expand the Machine Edition folder.
2. Right-click the EGD page and choose **Properties**.
   The Inspector displays the EGD options.
3. In the Inspector, configure the Host Name option. For help on this or any option, select the option and look up the help that automatically appears in the Companion. To open the Companion, press SHIFT+F11.
4. (Optional.) Configure the Local Server Cache Path and Timeout options.

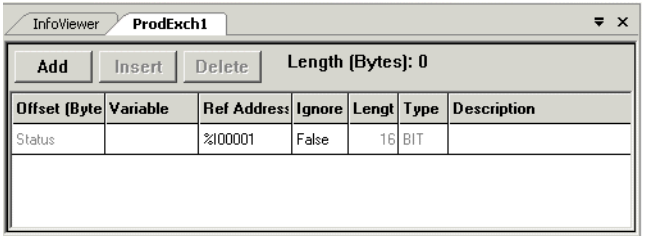
To configure a Logic Developer - PLC target to use the EGD Configuration Server

1. In the Project tab of the Navigator, expand the target.
2. Right-click the Ethernet Global Data folder and choose **Properties**.
   The Inspector displays the folder's properties.
3. Ensure that the Use Configuration Server property is set to True and configure the properties below it.

To add a new produced exchange and configure it

1. In the Project tab of the Navigator, expand the target's Ethernet Global Data folder.
2. Right-click the Produced Exchanges folder and choose **New**.
   A new produced exchange appears with a default name. The new produced exchange is invalid.
3. (Optional.) Enter a name that is more meaningful than the default name.
4. Right-click the produced exchange and choose **Properties**.
   The Inspector displays the exchange's properties.
5. In the Inspector, configure the Destination property and other properties as required.
   If the new produced exchange is now valid, its icon looks like this: .
6. In the Project tab of the Navigator, right-click the produced exchange and choose **Configure**.

The EGD Variable editor appears.



7. In the editor, use the buttons to add, insert, or delete rows.
8. For each row, define the Variable, Ref Address, Length, and/or Description parameters. When you have configured the target to use the EGD Configuration Server, you must specify a Variable for each row; you cannot use memory ranges that have no variables mapped to them. The Ignore parameter is available only for the Status. The Type parameter is read-only.

   For help on the EGD Variable editor, click inside the grey space at the top and press F1.

To publish a target's produced exchanges to the EGD Configuration Server, by using the Validate method

1. In the Project tab of the Navigator, right-click the target and choose **Set as active target**.

   If the option is unavailable, the target has already been set as the active target.
2. Right-click the target and choose **Validate**.

   Any errors found in the target's EGD configuration, Hardware Configuration, or logic are listed in the Build tab of the Feedback Zone.
3. Correct any errors found and repeat step 2 of this procedure.
4. When the Build tab of the Feedback Zone displays no errors, click the Messages tab of the Feedback Zone.

   If the message "EGD Produced Data published" appears, the target's produced exchanges have been updated (published) from your computer to the EGD Configuration Server.

To publish a target's produced exchanges to the EGD Configuration Server, by using the Bind and Build method

1. In the Project tab of the Navigator, right-click the target and choose **Set as active target**.

   If the option is unavailable, the target has already been set as the active target.
2. Expand the target.
3. Right-click the Ethernet Global Data folder and choose **Bind and Build**.

   Any errors found in the target's EGD configuration are listed in the Messages tab of the Feedback Zone. Some of these errors may pertain to consumed exchanges. If you want context-sensitive help on the errors, we recommend that you use the Validation method instead of the Bind and Build method.

4. If errors are found pertaining to produced exchanges, correct them and repeat step 3 of this procedure.

   The message "EGD Produced Data published" appears when the target's produced exchanges have been updated (published) from your computer to the EGD Configuration Server.

   If an error is found pertaining to a consumed exchange, you typically need to synchronize the consumed exchange.

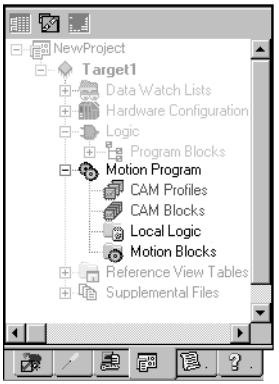
To synchronize a consumed exchange on your computer with the corresponding produced exchange published on the EGD Configuration Server:

1. If any of the following conditions is true, validate the target that contains the producer. See "Validating a Target", on page 39.)
   - You have uploaded the Hardware Configuration and EGD configuration from the producer PLC to your computer
   - You have converted the target that contains the producer
   - You have deleted the Ethernet Global Folder from the target that contains the producer and added it again.

   Validating the target that contains the producer updates the EGD Configuration Server with the produced exchange that this consumed exchange consumes.
2. In the Project tab of the Navigator, expand the target.
3. Expand the Ethernet Global Data folder and then expand the Consumed Exchanges folder.
4. If you have changed the Local Producer ID property of the producer that the consumed exchange consumes from, right-click the consumed exchange and choose **Properties**. Then, in the Inspector, set the Producer ID property on the consumed exchange to the same value as the new Local Producer ID.
5. In the Project tab of the Navigator, right-click the consumed exchange and choose **Synchronize to Server**.

Want to know more? In the online help index, look up "Developing EGD systems with the EGD Configuration Server: Possible work flow" and "Troubleshooting (PLC EGD)".

7 Motion Programming



Navigator: Project tab

Motion Program

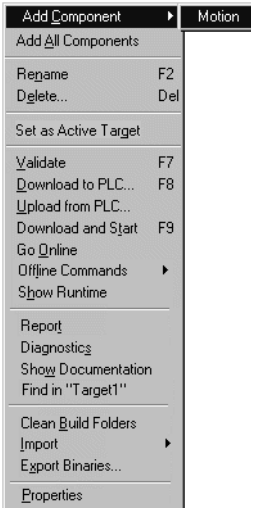
Logic Developer - PLC supports motion programming for the DSM324i and Motion Mate DSM314 motion control modules. High performance, easy-to-use, these multi-axis motion control modules are highly integrated with the PACSystems RX3i and the Series 90-30 PLC logic solving and communication functions.

Both the DSM324i and the DSM314 (see page 68) supports 10 motion blocks, 40 subroutines and a maximum total of 1000 motion program statements. Logic Developer - PLC, making motion programming possible, supports the following motion editors:

- PLC Motion Editor
- Local Logic Editor
- CAM Editor

This chapter outlines basic procedures that will get you started with Logic Developer - PLC to create motion programs with these three motion editors.

To add a Motion component to a target



Target right-click menu

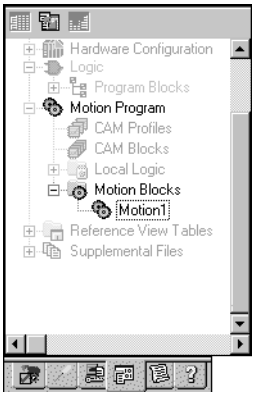
- In the Project tab of the Navigator, right-click the target, point to **Add Component**, and then choose **Motion**.

  A Motion Program node is added to your project. Included are empty Motion Blocks, Local Logic, CAM Profiles and CAM Blocks folders.

PLC MOTION EDITOR

Logic Developer - PLC includes a PLC Motion Editor, which enables you to create PLC Motion blocks for the DSM324i and the DSM314. This text-based editor is configurable as to its appearance and behavior. Comments and white space are not considered as PLC Motion block statements. The Motion block programming syntax is different from Local Logic syntax (see page 137).

To add a Motion Block



Navigator: Project tab

Motion Blocks

1. In the Project tab of the Navigator, expand the Motion Program.
2. Right-click Motion Blocks and choose **New**.

   A new empty PLC Motion block with a default name is added to your project.
3. Rename the block as desired.

To open a motion block for editing

1. In the Project tab of the Navigator, expand the Motion Program.
2. Expand Motion Blocks and double-click the Motion Block you want to open.

   The block opens for editing in the PLC Motion editor.

Working with the Motion Editor

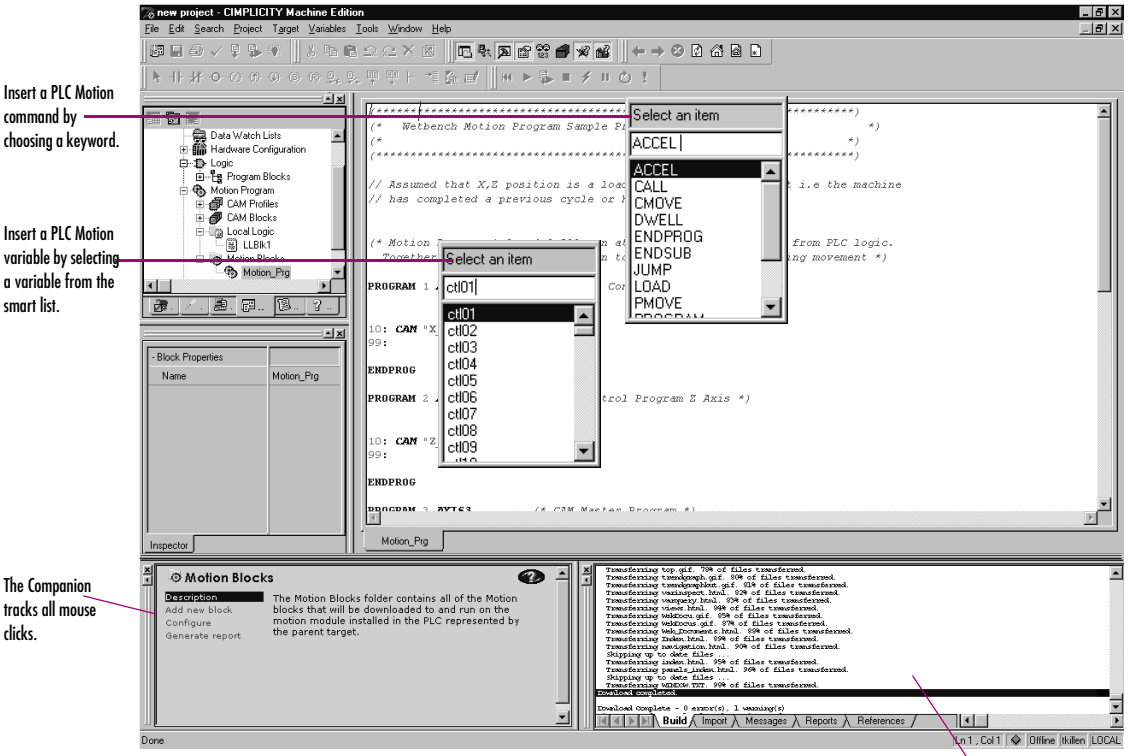
To insert a command

1. In the Motion editor, right-click and choose **Insert Keyword**.

   A smart list appears showing all available motion commands.

2. Select the appropriate command from the smart list and press ENTER.

   The command is placed in the motion editor.

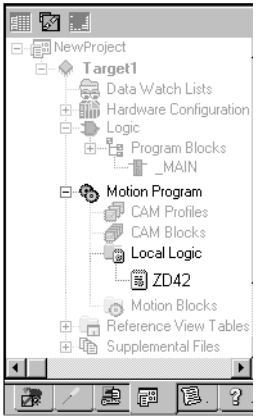


Want to know more? In the online help index, choose "PLC Motion Editor".

Motion Commands

ACCEL	The ACCEL statement sets the axis acceleration for subsequent moves and remains in effect in a given block unless changed. Note: If a move instruction is executed before ACCEL, the tag Acceleration is used.
BLOCK NUMBER	Block numbers may be used as the destination of JUMP commands. Block numbers must be unique and can be between 1 and 65535.
CAM	The CAM statement starts CAM motion and specifies exit conditions.
CAM-LOAD	CAM-LOAD loads a parameter register with the starting location for a CAM slave axis.
CAM-PHASE	CAM-PHASE sets the phase for CAM commands.
CALL	The CALL command executes another block as a subroutine
CMOVE	The CMOVE command programs a continuous move using the specified position and acceleration mode.
DWELL	DWELL causes motion to cease for a specified time period before processing the next command.
ENDPROG	The ENDPROG statement terminates a PLC Motion program definition.
ENDSUB	The ENDSUB statement terminates a PLC Motion subroutine definition.
JUMP	Jump to a block number or a sync block within the current program or subroutine. The jump may be unconditional or conditional based on the status of a CTL bit.
LOAD	Initializes or changes a parameter data register with a 32-bit twos-complement integer value.
PMOVE	The PMOVE command programs a positioning move using the specified position and accelerator mode.
PROGRAM	The PROGRAM statement is the first statement in a motion program. The program statement identifies the program number (valid range: 1 through 10) and the axis configuration. Program definitions cannot nest.
SUBROUTINE	The SUBROUTINE statement is the first statement in a motion subroutine. The subroutine statement identifies the subroutine number (valid range: 1 through 40) and the axis configuration.
SYNC BLOCK	A sync block is a special case of a block number. A sync block can be used only in multi-axis programs.
VELOC	Sets the process VELOCITY used by subsequent motion program move commands and remains in effect until changed by another VELOC statement.

LOCAL LOGIC



Navigator: Project tab

Local Logic

A Local Logic block runs synchronously with the Motion block, but is independent of the PLC's CPU scan. This enables the DSM324i or DSM314 to interact much more quickly with motion I/O signals on its faceplate connectors than would be possible if the logic for the signals were handled in the _MAIN program running on the PLC.

Local Logic language uses free-form, text-based circuits and contains basic mathematical and logical constructs. The Local Logic syntax enables you to assign a variety of logic tasks to your motion programs while working in conjunction with PLC Logic programs and motion blocks to yield a flexible programming environment. Because it uses straightforward, understandable syntax, it is easy to gain proficiency with this editor.

The Local Logic programming language supports assignments, conditional statements, arithmetic, logical, and relational statements. Local Logic provides the user access to motion controller data, parameters using a fixed set of variables, control, and status bits:

- Parameter data - accessible from Local Logic host PLC and motion blocks. The parameter data are similar to variables in a program.
- CTL bits - enable the Local Logic block or host PLC to signal the motion block to start an event.
- Motion block numbers - the current block number can be used within the Local Logic block or host PLC to make an action occur only during a specific motion programming section.

To create a Local Logic block

1. In the Project tab of the Navigator, expand the Motion Program.
2. Right-click Local Logic and choose **New**.

   A new Local Logic block with a default name is created.
3. Rename the block as required.

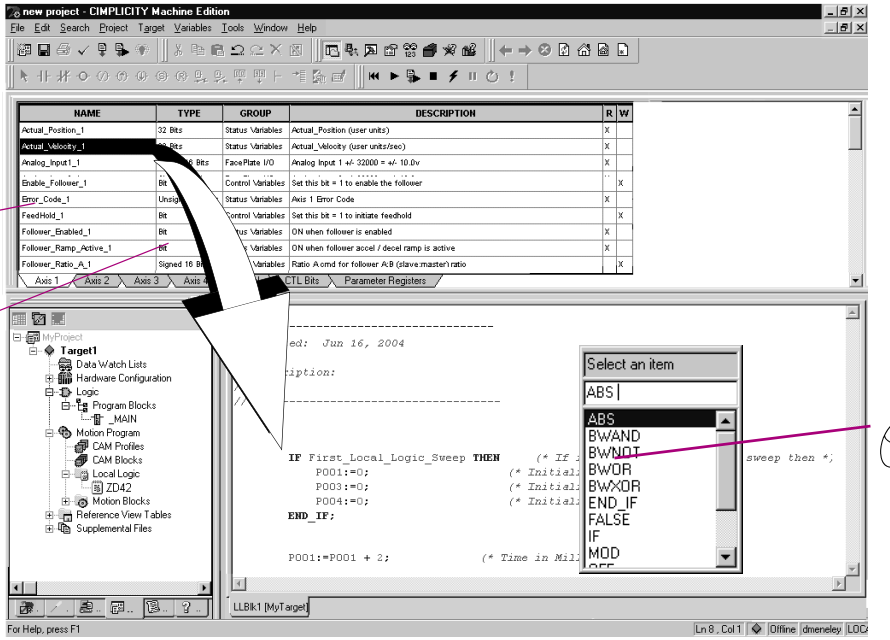
To open a Local Logic block for editing

- In the Project tab of the Navigator, expand the Motion Program and double-click the Local Logic block.

  The Local Logic block opens for editing in the Local Logic editor.

Working with the Local Logic Editor

The Local Logic editor has its own distinctive syntax for constructing Local Logic blocks.



Drag and drop variables from the Local Logic Variable Table (LLVW) into the Local Logic editor.

Right-click to choose commands from a list.

To insert a Local Logic command

1. In the Local Logic editor, right-click and choose **Insert Keyword**.

   A smart list displays all available Local Logic commands.

2. Select the desired command in the smart list and press ENTER

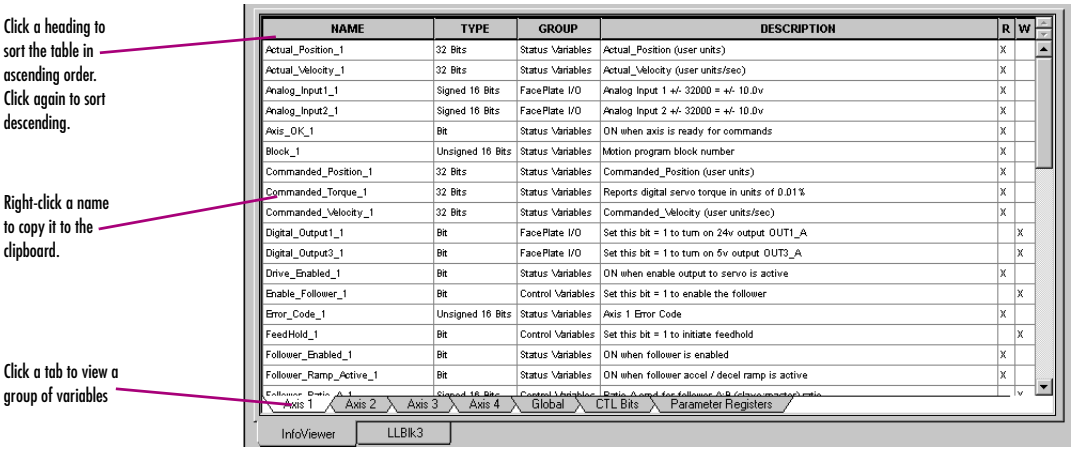
   The command is inserted.

   Note: You can drag variables from the Local Logic Variable Table (LLVT) to the Local Logic editor.

Local Logic Variables

Local Logic is designed to complement a PLC's logic and mathematical capabilities. Solving small Local Logic and mathematical sets requires a tight synchronization with the controlled motion.

Logic Developer - PLC includes a table containing Local Logic variables, the Local Logic Variable Table (LLVT), that you can drag into your Local Logic blocks. As illustrated in the diagram below, the LLVT has several tabs that organize the variables by category:



To view the LLVT

1. In the Project tab of the Navigator, expand the Motion Program.
2. Right-click Local Logic and choose **Local Logic Variable Table**.

   The "Which LLVT do you want?" help topic appears.
3. Select **Motion Mate DSM314** or **DSM324i**.

   The LLVT appears in the Infoviewer, displaying variables or data in each tab are:

Axis 1	Variables specific to axis 1
Axis 2	Variables specific to axis 2
Axis 3	Variables specific to axis 3
Axis 4	Variables specific to axis 4
Global	Global data such as module status code
CTL bits	DSM general Control/Status bits
Parameter Registers	DSM parameter data

The table has six columns:

Name	Contains the variable name that is to be used within a Local Logic block
Type	The data type for this variable. For example, 32-bit means that this variable is a 32-bit variable.
Group	The group this variable is placed in. For example, Faceplate I/O means that this variable refers to a point on the module faceplate.
Description	This column contains a textual description of the variable. When the user hovers the mouse pointer over the description, a tool tip displays the description.
R	This column indicates if the variable can be read by a Local Logic block.
W	This column indicates if the variable can be written by a Local Logic block.

To insert a Local Logic variable

1. In the Local Logic editor, right-click and choose **Insert Variable**.

   A smart list appears prompting you to choose a Local Logic variable name.

2. Select a variable in the list and press ENTER.

   The variable is inserted in your Local Logic.

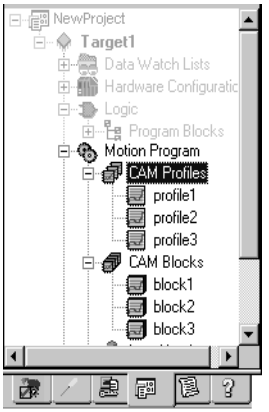
Local Logic Commands and Operators

Local Logic enables you to execute basic logic and mathematical functions on the DSM324i or Motion Mate DSM314 module. Commands use upper case characters only and are case sensitive.

ABS	TRUE	-(minus)	<= (less than or equal to)
BWAND	FALSE	/ (divide)	<> (not equal)
BWOR	IF	* (multiply)	
BWXOR	THEN	:=(assign)	
BWNOT	END_IF	>(greater than)	
ON	MOD	< (less than)	
OFF	+ plus	>= (greater than or equal to)	

Want to know more? In the online help index, choose "Local Logic: an Overview".

CAM EDITOR



Project Tab: Navigator

CAM profiles and CAM blocks

The CAM editor is an accessory for Logic Developer - PLC motion programming that provides a means to create, edit, and manage electronic CAM profiles. Each CAM profile is a user-defined curve that specifies the response of a slave servo to a master position index. CAM profiles are referenced by name in the parent motion program and grouped into CAM blocks. Each block is intended for download to a specific motion module via its PLC controller. The hardware components are specified in the Hardware Configuration (HWC) of the parent target.

You can reuse a CAM profile by including it in multiple CAM blocks. To include a CAM profile in a CAM block, see "To add an alias to a CAM block", on page 144.

To create a CAM block

1. In the Project tab of the Navigator, expand the Motion Program.
2. Right-click CAM blocks and choose **New**.

   A new CAM block with a default name is created.
3. Rename the block as desired.

To import CAM blocks

1. In the Project tab of the Navigator, expand the Motion Program.
2. Right-click CAM blocks and choose **Import from File**.

   The Open dialog box appears.
3. Browse to the CAM block you want to import (.csv or .txt file).
4. Click **Open**.

   The imported block appears in your project.

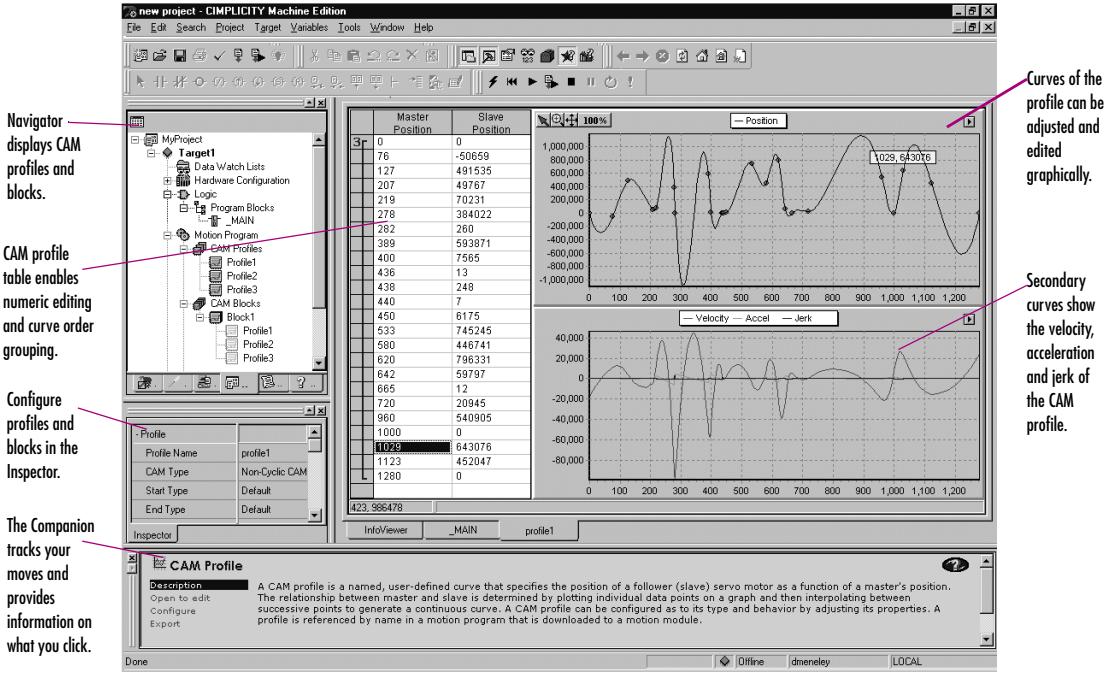
To open a CAM block for editing

1. In the Project tab of the Navigator, expand the Motion Program.
2. Expand CAM Blocks and double-click the CAM block you want to open.

   The block opens for editing in the PLC Motion editor.

Working with the CAM Editor

You can adjust the curves of your CAM profile to suit the specific needs of your project. With the CAM editor you create profiles by defining points on a master/slave position curve. Groups of adjoining points are allocated to sectors. Each sector is assigned a polynomial curve fit order (1,2,3) that specifies how the curve will be interpolated between points.



To create a CAM profile

1. In the Project tab of the Navigator, expand the Motion Program.
2. Right-click CAM Profiles and choose **New**.

   A new CAM profile with a default name is added to your project.

To configure a CAM profile

1. In the Project tab of the Navigator, expand the Motion Program.
2. Right-click CAM Profiles, right-click a CAM profile, and choose **Properties**.

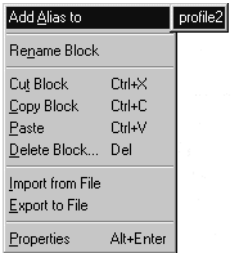
   The Inspector displays the properties of the CAM profile.
3. Adjust the properties of the CAM profile in the Inspector to specify its type and boundary conditions.

To edit a CAM profile

1. In the Project tab of the Navigator, expand the Motion Program.
2. Right-click CAM Profiles and double-click a CAM profile.

   A graphical representation of your profile appears in the profile editor and a numeric representation appears in the profile table.
3. Insert and move points in the profile editor or table.
4. Group points into sectors in the profile table and assign curve fit order to each sector.

To add an alias to a CAM block



Right-click method

1. In the Project tab of the Navigator, expand the Motion Program and expand CAM blocks.
2. Right-click a CAM block, point to **Add Alias to**, and then choose a profile.

Note: Aliases correspond to CAM profiles within CAM blocks. In order to create aliases for CAM blocks, you must have previously created CAM profiles.

Drag and drop method

1. In the Project tab of the Navigator, expand the Motion Program.
2. Expand CAM Profiles and expand CAM blocks.
3. Drag a CAM profile and drop it onto a CAM block.

The CAM profile is added to the CAM block.

Want to know more? In the online help index, choose "CAM".

Index

D

E

F

G

H

I

N

O

P

Q

R

T

U

V

W