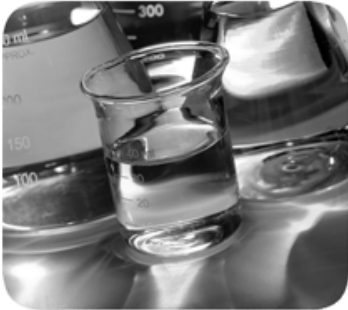


Logix5000 Controllers Ladder Diagram

Catalog Numbers 1756 ControlLogix, 1769 CompactLogix, 1789 SoftLogix, 1794 FlexLogix, PowerFlex 700S with DriveLogix



Important user information

Read this document and the documents listed in the additional resources section about installation, configuration, and operation of this equipment before you install, configure, operate, or maintain this product. Users are required to familiarize themselves with installation and wiring instructions in addition to requirements of all applicable codes, laws, and standards.

Activities including installation, adjustments, putting into service, use, assembly, disassembly, and maintenance are required to be carried out by suitably trained personnel in accordance with applicable code of practice. If this equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

No patent liability is assumed by Rockwell Automation, Inc. with respect to use of information, circuits, equipment, or software described in this manual.

Reproduction of the contents of this manual, in whole or in part, without written permission of Rockwell Automation, Inc., is prohibited.

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.



WARNING: Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.



ATTENTION: Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence

Important:

Identifies information that is critical for successful application and understanding of the product.

Labels may also be on or inside the equipment to provide specific precautions.



SHOCK HAZARD: Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.



BURN HAZARD: Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.



ARC FLASH HAZARD: Labels may be on or inside the equipment, for example, a motor control center, to alert people to potential Arc Flash. Arc Flash will cause severe injury or death. Wear proper Personal Protective Equipment (PPE). Follow ALL Regulatory requirements for safe work practices and for Personal Protective Equipment (PPE).

Allen-Bradley, Rockwell Software, Rockwell Automation, and TechConnect are trademarks of Rockwell Automation, Inc.

Trademarks not belonging to Rockwell Automation are property of their respective companies.

Summary of changes

This manual contains new and updated information. The following table contains the changes made to this revision.

Change	Topic
Updated graphics for version 24 and Windows 7.	Throughout the manual.
Input and Output program parameters do not require buffering during logic execution.	Choose the required instructions on page 11

Preface	Studio 5000 environment.....7
	Additional resources.....7
	Chapter 1
Program ladder diagram	Introduction.....9
	Instruction.....9
	Branch.....9
	Rung condition..... 11
	Write ladder logic..... 11
	Choose the required instructions..... 11
	Arrange the input instructions..... 13
	Arrange the output instructions..... 14
	Choose a tag name for an operand..... 14
	Enter ladder logic..... 15
	Append an element to the cursor location..... 16
	Drag and drop an element..... 16
	Assign instruction operands..... 16
	Create and assign a new tag..... 16
	Choose a name or an existing tag..... 18
	Drag and drop a tag from the Tags window..... 18
	Assign an immediate (constant) value..... 18
	Enter a rung comment..... 19
	Language switching..... 19
	Verify the routine..... 20
Index	

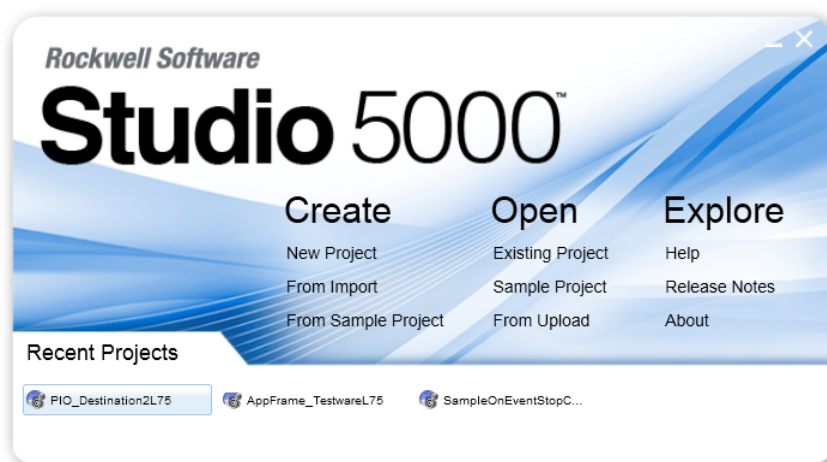
This manual shows how to program Logix5000 controllers with the relay ladder programming language. This manual is one of a set of related manuals that show common procedures for programming and operating Logix5000 controllers.

For a complete list of common procedures manuals, refer to the [Logix5000 Controllers Common Procedures Programming Manual](#), publication [1756-PM001](#).

The term Logix5000 controller refers to any controller that is based on the Logix5000 operating system.

Studio 5000 environment

The Studio 5000 Automation Engineering & Design Environment™ combines engineering and design elements into a common environment. The first element is the Studio 5000 Logix Designer™ application. The Logix Designer application is the rebranding of RSLogix™ 5000 software and will continue to be the product to program Logix5000™ controllers for discrete, process, batch, motion, safety, and drive-based solutions.



The Studio 5000® environment is the foundation for the future of Rockwell Automation® engineering design tools and capabilities. The Studio 5000 environment is the one place for design engineers to develop all elements of their control system.

Additional resources

These documents contain additional information concerning related Rockwell Automation products.

Resource	Description
Logix5000 Controllers Program Parameters Programming Manual , publication 1756-PM021	Describes how to use program parameters when programming Logix5000 controllers.

Resource	Description
Logix5000 Controllers General Instructions Reference Manual , publication 1756-RM003	Describes the available instructions for a Logix5000 controller.
Logix5000 Controllers Process and Drives Instructions Reference Manual , publication 1756-RM006	Describes how to program a Logix5000 controller for process or drives applications.
Logix5000 Controllers Motion Instruction Set Reference Manual , publication motion-RM002	Describes how to program a Logix5000 controller for motion applications.
Product Certifications website, http://ab.rockwellautomation.com	Provides declarations of conformity, certificates, and other certification details.

You can view or download publications at <http://www.rockwellautomation.com/literature>. To order paper copies of technical documentation, contact your local Rockwell Automation distributor or sales representative.

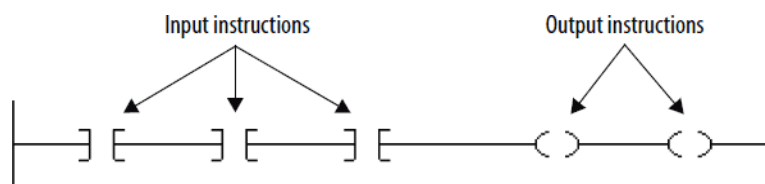
Program ladder diagram

Introduction

Instruction

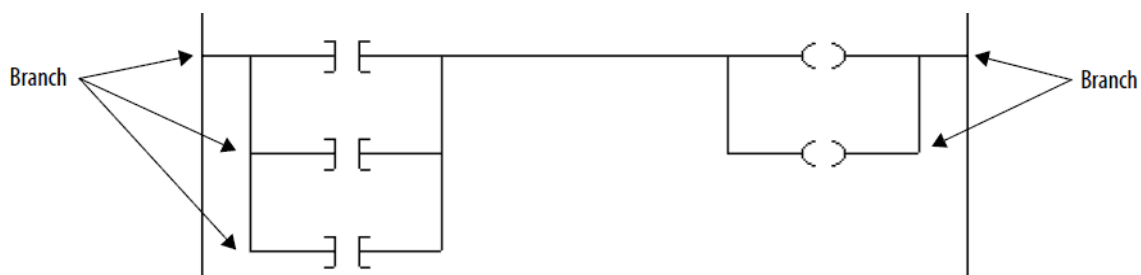
You organize ladder diagram as rungs on a ladder and put instructions on each rung. There are two basic types of instructions:

- **Input instruction:** An instruction that checks, compares, or examines specific conditions in your machine or process.
- **Output instruction:** An instruction that takes some action, such as turn on a device, turn off a device, copy data, or calculate a value.

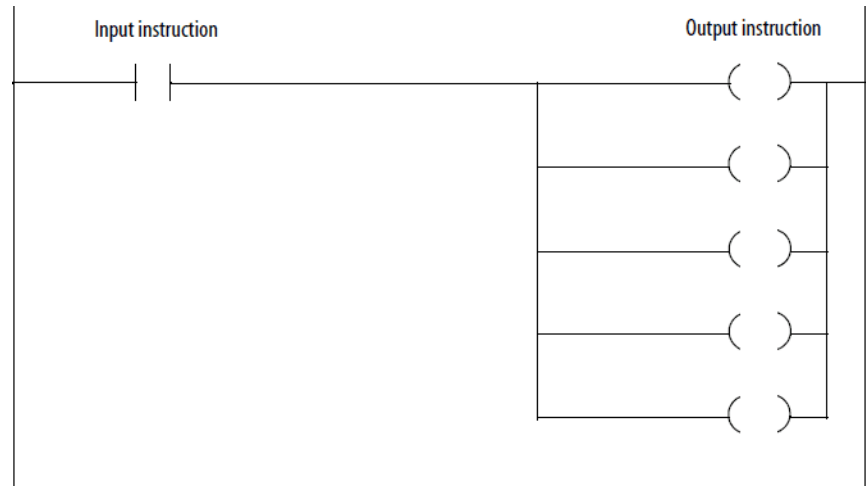


Branch

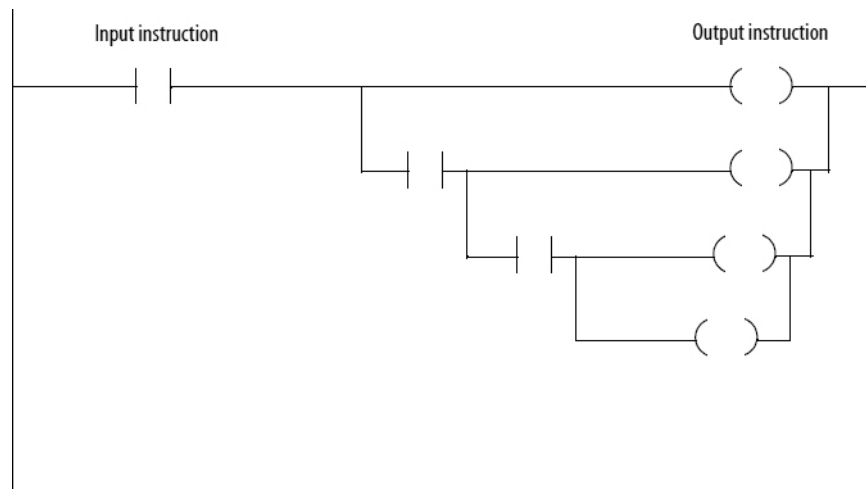
A branch is two or more instructions in parallel.



There is no limit to the number of parallel branch levels that you can enter. This example shows a parallel branch with five levels. The main rung is the first branch level, followed by four additional branches.



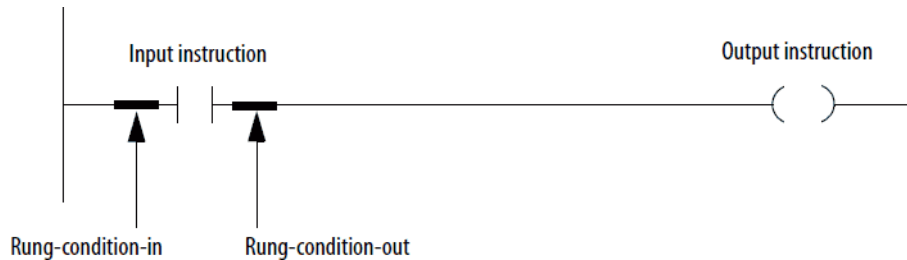
You can nest branches to as many as 6 levels. This example shows a nested branch. The bottom output instruction is on a nested branch that is three levels deep.



Large rungs with complex, nested branches result in having to scroll through the ladder editor and may end up spanning multiple pages when you print the logic. To make it easier to maintain, divide the logic into multiple smaller rungs.

Rung condition

The controller evaluates ladder instructions based on the rung condition preceding the instruction (rung-condition-in).



Only create instructions that affect the rung-condition-in of subsequent instructions on the rung.

- If the rung-condition-in to an input instruction is true, the controller evaluates the instruction and sets the rung-condition-out to match the results of the evaluation.
 - If the instruction evaluates to true, the rung-condition-out is true.
 - If the instruction evaluates to false, the rung-condition-out is false.
- An output instruction does not change the rung-condition-out.
 - If the rung-condition-in to an output instruction is true, the rung-condition-out is set to true.
 - If the rung-condition-in to an output instruction is false, the rung-condition-out is set to false.

Write ladder logic

Writing ladder logic requires that you choose the input and output instructions, and choose the tag names for operands.

Choose the required instructions

1. Identify the conditions to check and separate them from the action to take for the rung.
2. Choose the appropriate input instruction for each condition and the appropriate output instruction for each action.



For more information on specific instructions, see:

- [Logix5000 Controllers General Instructions Reference Manual](#), publication [1756-RM003](#)
- [Logix5000 Controllers Process and Drives Instructions Reference Manual](#), publication [1756-RM006](#)
- [Logix5000 Controllers Motion Instruction Set Reference Manual](#), publication [1756-RM007](#)

Tip: I/O module data updates asynchronously to the execution of logic. If you reference an input multiple times in your logic, the input could change state between separate references. If you need the input to have the same state for each reference, buffer the input value and reference that buffer tag.

You can also use Input and Output program parameters which automatically buffer the data during logic execution. See the [Logix5000 Controllers Program Parameters Programming Manual](#), publication [1756-PM021](#).

The examples in this chapter use two simple instructions to help you learn how to write ladder diagram logic. The rules that you learn for these instructions apply to all other instructions.

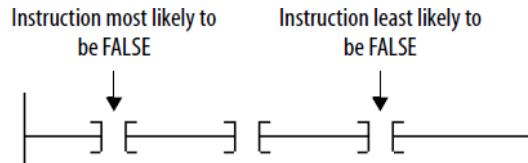
Symbol	Name	Mnemonic	Description						
	Examine If Closed	XIC	An input instruction that looks at one bit of data.						
			<table border="1"> <thead> <tr> <th>If the bit is</th> <th>Then the instruction (rung-condition-out) is</th> </tr> </thead> <tbody> <tr> <td>On (1)</td> <td>True</td> </tr> <tr> <td>Off (0)</td> <td>False</td> </tr> </tbody> </table>	If the bit is	Then the instruction (rung-condition-out) is	On (1)	True	Off (0)	False
If the bit is	Then the instruction (rung-condition-out) is								
On (1)	True								
Off (0)	False								
	Output Energize	OTE	An output instruction that controls one bit of data.						
			<table border="1"> <thead> <tr> <th>If the instructions to the left (rung-condition-in) are</th> <th>Then the instruction turns the bit</th> </tr> </thead> <tbody> <tr> <td>True</td> <td>On (1)</td> </tr> <tr> <td>False</td> <td>Off (0)</td> </tr> </tbody> </table>	If the instructions to the left (rung-condition-in) are	Then the instruction turns the bit	True	On (1)	False	Off (0)
If the instructions to the left (rung-condition-in) are	Then the instruction turns the bit								
True	On (1)								
False	Off (0)								

Arrange the input instructions

Determine how to arrange the input instructions on the rung, as shown below.

To check multiple input conditions when:	Arrange the input instructions:
<ul style="list-style-type: none"> All conditions must be met in order to take action. For example, If condition_1 AND condition_2 AND condition_3... 	In series:
<ul style="list-style-type: none"> Any one of several conditions must be met in order to take action. For example, If condition_1 OR condition_2 OR condition_3... 	In parallel:
<ul style="list-style-type: none"> There is a combination of the above. For example: If condition_1 AND condition_2... OR If condition_3 AND condition_2... 	In combination:


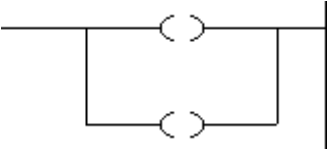

Tip: The controller executes all instructions on a rung regardless of their rung-condition-in. For optimal performance of a series of instructions, sequence the instructions from most likely to be false on the left to least likely to be false on the right.



When the controller finds a false instruction, it executes the remaining instructions in the series with their rung-condition-in set to false. Typically, an instruction executes faster when its rung-condition-in (rung) is false rather than true.

Arrange the output instructions

Place at least one output instruction to the right of the input instructions. You can enter multiple output instructions on a rung of logic:

Option	Example
Place the output instructions in sequence on the rung (serial).	
Place the output instructions in branches (parallel).	
Place the output instructions between input instructions. The last instruction on the rung must be an output instruction.	

Choose a tag name for an operand

Tag names follow these formats:

For a:	Specify:
Tag	<i>tag_name</i>
Bit number of a larger data type	<i>tag_name.bit_number</i>
Member of a structure	<i>tag_name.member_name</i>
Element of a one dimension array	<i>tag_name[x]</i>
Element of a two dimension array	<i>tag_name[x,y]</i>
Element of a three dimension array	<i>tag_name[x,y,z]</i>
Element of an array within a structure	<i>tag_name.member_name[x]</i>
Member of an element of an array	<i>tag_name[x,y,z].member_name</i>

where:

- *x* is the location of the element in the first dimension.
- *y* is the location of the element in the second dimension.
- *z* is the location of the element in the third dimension.

For a structure within a structure, add *.member_name*.

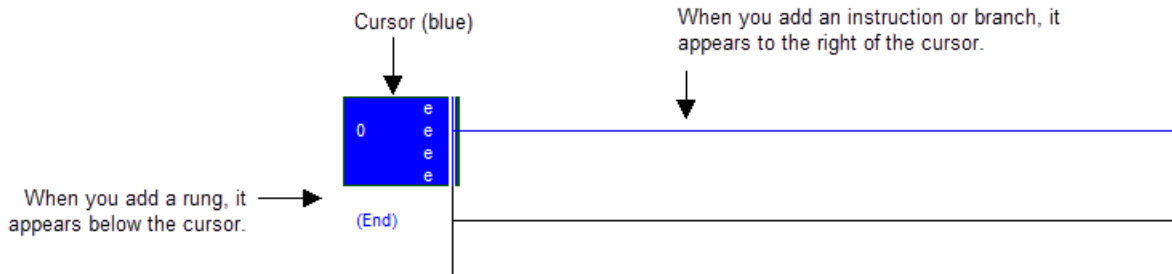
Example:

Choose a Tag Name for an Operand

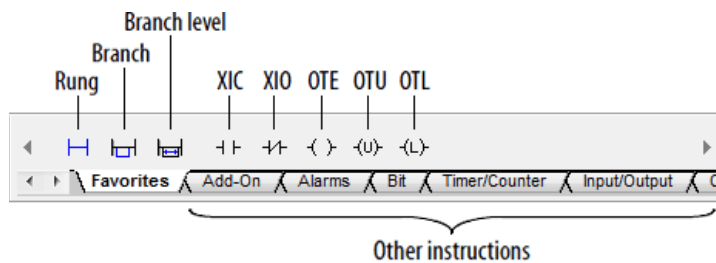
To Access:	The tag name looks like this:
machine_on tag	machine_on _____] [_____
bit number 1 of the one_shots tag	one_shots.1 _____] [_____
DN member (bit) of the running_seconds timer	running_seconds.DN _____] [_____
mix member of the north_tank tag	north_tank.mix _____] [_____
element 2 in the recipe array and element 1,1 in the tanks array	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p style="text-align: center;">COP</p> <p>Copy File</p> <p>Source recipe[2]</p> <p>Dest tanks[1,1]</p> <p>Length 1</p> </div>
element 2 in the preset array within the north_tank tag	<div style="border: 1px solid black; padding: 5px; width: fit-content;"> <p style="text-align: center;">CLR</p> <p>Clear</p> <p>Dest north_tank.preset[2]</p> <p style="text-align: right;">0</p> </div>
part_advance member of element 1 in the drill array	drill[1].part_advance _____] [_____

Enter ladder logic

A new routine contains a rung that is ready for instructions.



Use the **Language Element** toolbar to add a ladder diagram element.



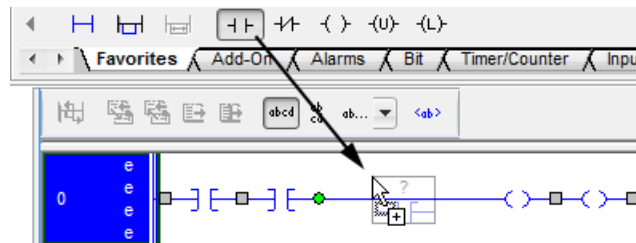
Append an element to the cursor location

Important: Use caution when copying and pasting components between different versions of the Logix Designer application. The application only supports pasting to the same version or newer version. Pasting to a prior version of the application is not supported. When pasting to a prior version, the paste action may succeed but the results may not be as intended.

1. Click to select the instruction, branch, or rung that is above or to the left of where you want to add an element.
2. On the **Language Element** toolbar, click the button for the element that you want to add.

Drag and drop an element

Drag the button for the element directly to the desired location. A green dot shows a valid placement location (drop point).

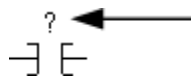


Assign instruction operands

After you add an instruction to a ladder rung, you assign tags to the instruction operands. You can create a new tag, use an existing tag, or assign a constant value.

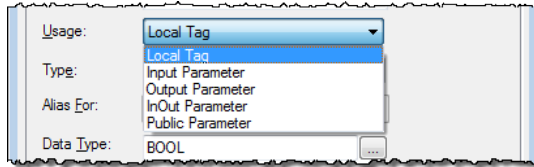
Create and assign a new tag

1. Click the operand area of the instruction.

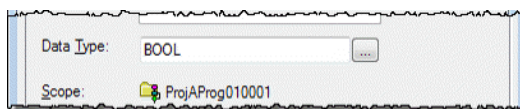


2. Type a name for the tag and press the **Enter** key.
3. Right-click the tag name and then click **New "tag_name"**.

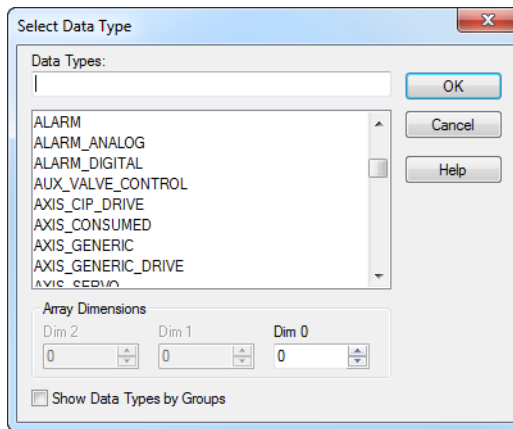
- In the **New Parameter or Tag** dialog box, in the **Usage** box, choose the usage.



- In the **New Parameter or Tag** dialog box, in the **Data Type** box, click the  button.

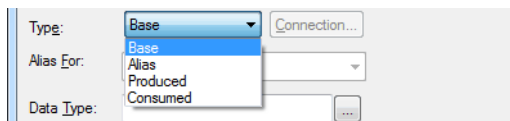


- In the **Select Data Type** dialog box, choose the data type for the tag.



If you want to define the tag as an array, in the **Array Dimensions** boxes, enter the number of elements in each dimension.


- Click **OK**.
- In the **New Parameter or Tag** dialog box, choose the scope for the tag.



- Click **OK**.


Choose a name or an existing tag



1. Double-click the operand area, and then click .

The Tag Browser window appears.

2. Select the name or tag:

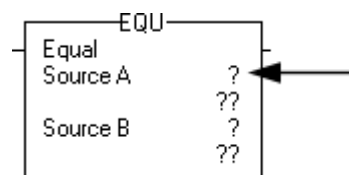
To select a:	Do this:
Label, routine name, or similar type of name	Click the name.
Tag	Double-click the tag name.
Bit number	<ol style="list-style-type: none"> A. Click the tag name. B. To the right of the tag name, click . C. Click the required bit.

3. Press the **Enter** key or click a different spot on the ladder diagram to close the Tag Browser.

Drag and drop a tag from the Tags window

1. Find the tag in the **Controller Tags** or the **Program Parameters and Local Tags** window.
2. Double-click the tag to select it.
3. Click and drag the tag to its location on the instruction. A green dot appears to show you where you can drop the tag.

Assign an immediate (constant) value



1. Click the operand area of the instruction.

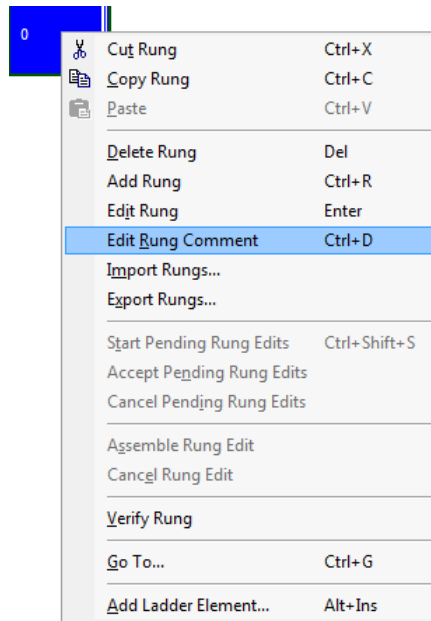
2. Type the value and press the **Enter** key.

Enter a rung comment

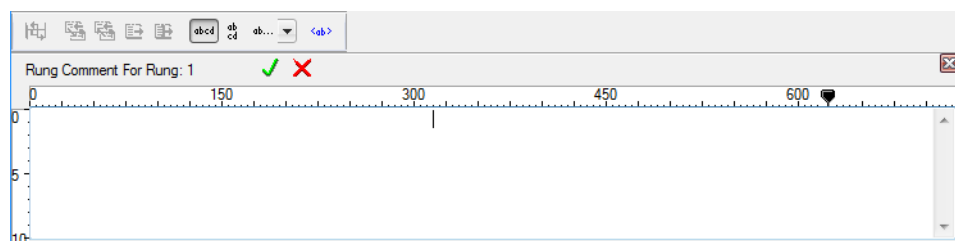
When entering a rung of ladder logic, you can add comments that explain the purpose of your rung.

To enter a rung comment, perform this procedure.

1. Right-click the rung number of your ladder logic and then click **Edit Rung Comment**.



The **Rung Comment** dialog box appears.



2. Type your rung comment, and then click the green check to save your changes or click the red X to discard your changes.

Language switching

With version 17 and later of the application, you have the option to display project documentation, such as tag descriptions and rung comments for any supported localized language. You can store project documentation for multiple languages in a single project file rather than in language-specific project files. You

define all the localized languages that the project supports and set the current, default, and optional custom localized language. The application uses the default language if the current language's content is blank for a particular component of the project. However, you can use a custom language to tailor documentation to a specific type of project file user.

Enter the localized descriptions in your project, either when programming in that language or by using the import/export utility to translate the documentation off-line and then import it back into the project. Once you enable language switching, you can dynamically switch between languages.


Project documentation that supports multiple languages within a project includes:

- Component descriptions in tags, routines, programs, user-defined data types, and Add-On Instructions.
- Equipment phases.
- Trends.
- Controllers.
- Alarm Messages (in ALARM_ANALOG and ALARM_DIGITAL configuration).
- Tasks.
- Property descriptions for modules in the Controller Organizer.
- Rung comments, SFC text boxes, and FBD text boxes.

For more information on enabling a project to support multiple translations of project documentation, see the online help.

Verify the routine

As you program your routine (s), it is a good idea to periodically verify your work.

1. In the **Standard** toolbar click .
2. Errors are listed in the **Output** window on the **Errors** tab at the bottom of the application.
 - a. To go to the first error or warning, press the **F4** key.
 - b. Correct the error according to the description in the **Errors** tab.
 - c. Repeat until you have corrected all errors.
3. To close the **Output** window, press the **Alt+I** keys.

B

branch
 ladder logic 9
buffer input 12

C

create
 tag 16

I

immediate value
 ladder logic 18

L

ladder logic
 arrange input instructions 13
 arrange output instructions 14
 assign immediate value 18
 branch 9
 develop 11
 rung condition 11

N

name
 tag name 14

R

routine
 verify 20
rung condition 11

T

tag
 assign 16
 choose name 14
 create 16
 enter 16
 ladder logic 14, 16

V

verify
 routine 20

Rockwell Automation support

Rockwell Automation provides technical information on the web to assist you in using its products. At <http://www.rockwellautomation.com/support> you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at <https://rockwellautomation.custhelp.com> for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit <http://www.rockwellautomation.com/services/online-phone>.

Installation assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

United States or Canada	1.440.646.3434
Outside United States or Canada	Use the Worldwide Locator available at http://www.rockwellautomation.com/locations , or contact your local Rockwell Automation representative.

New product satisfaction return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

United States	Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.
Outside United States	Please contact your local Rockwell Automation representative for the return procedure.

Documentation feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the feedback form, publication [RA-DU002](#).

Rockwell Otomasyon Ticaret A.Ş., Kar Plaza İş Merkezi E Blok Kat:6 34752 İçerenköy, İstanbul, Tel: +90 (216) 5698400

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Rockwell Automation Publication 1756-PM008E-EN-P - October 2014

Supersedes Publication 1756-PM008D-EN-P - November 2012

Copyright © 2014 Rockwell Automation, Inc. All rights reserved. Printed in the U.S.A.