Programming Manual

Allen-Bradley

Logix5000 Controllers Major, Minor, and I/O Faults

1756 ControlLogix, 1756 GuardLogix, 1769 CompactLogix, 1769 Compact GuardLogix, 1789 SoftLogix, 5069 CompactLogix, Studio 5000 Logix Emulate

Allen-Bradley • Rockwell Software

Rockwell Automation

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.

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).

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

Changes	Page
Updated the Major Fault Code table that lists the type and code displayed on the **Major Faults** tab and in the MajorFaultRecord attribute.	Major Fault Codes on page 27
Updated the table that explains how to use ladder logic to monitor information about common minor faults.	Identifying Minor Faults on page 33
Updated the Minor Fault Code table that lists the type and code displayed on the **Minor Faults** tab and in the MinorFaultRecord attribute.	Minor Fault Codes on page 35

This manual shows how to monitor and handle major and minor controller faults. It also provides lists of major, minor, and I/O fault codes that you can use to troubleshoot your system.

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 products from Rockwell Automation.

Resource	Description
Logix5000 Controllers General Instructions Reference Manual , publication 1756-RM003	Provides programmers with details about each available instruction for a Logix5000 controller.
Product Certifications website, http://www.ab.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 Allen-Bradley distributor or Rockwell Automation sales representative.

Legal notices

Warranty

This product is warranted in accordance with the product license. The product's performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product's implementation may vary among users.

This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at any time without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

Environmental Compliance

Rockwell Automation maintains current product environmental information on its website at
http://www.rockwellautomation.com/rockwellautomation/about-us/sustainability-ethics/product-environmental-compliance.page

Contact Rockwell

Customer Support Telephone — 1.440.646.3434

Online Support — http://www.rockwellautomation.com/support/

Major Faults

This chapter explains major fault codes and how to work with them in the Logix Designer application.

Major Fault State

If a fault condition occurs that would prevent an instruction from running, the instruction is aborted and a major fault is reported. A major fault halts logic execution and the controller switches to faulted mode (the OK LED flashes red).

Depending on your application, you may not want all major faults to shut down your system. If you do not want all major faults to shut down your system, create a fault routine to clear the fault and let your application continue to run. See Create a routine for the controller fault handler on page 17 and Clear a major fault during prescan on page 23.

The process of resuming execution after a fault is cleared is known as fault recovery.

Important: Do not use fault routines to continually clear all faults on the controller. Program the fault routine to be selective in the types and number of faults cleaned. It is also a good idea to log the fault occurrence so you can analyze it later.

Important: When an instruction generates an error due to a fault (for example, a COP with an indirect addressing programming error), the instruction is skipped and does not run. This occurs with all instructions.

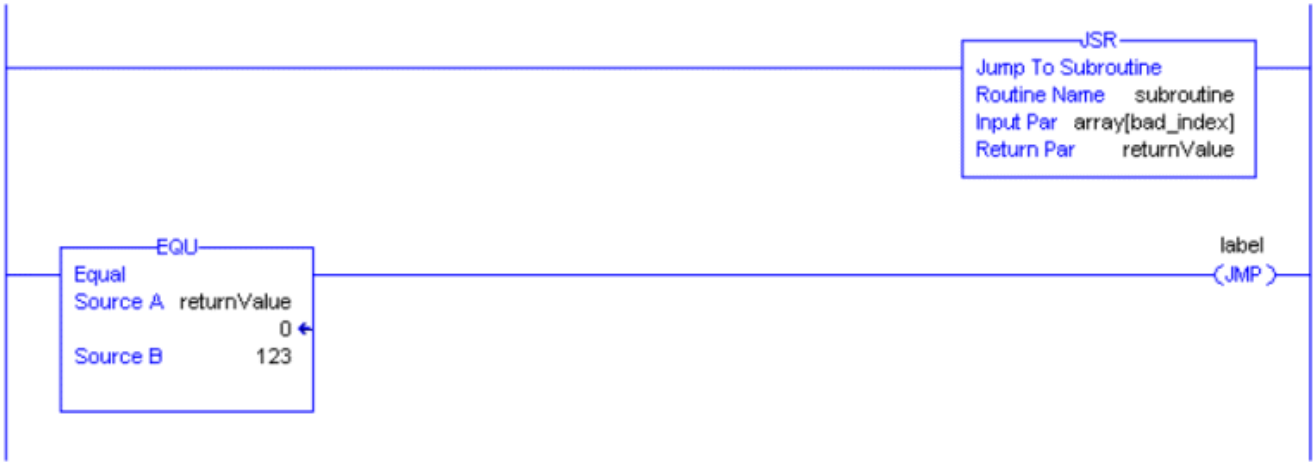
Example: In a system that uses recipe numbers as indirect addresses, an incorrectly typed number could produce a major fault. To keep the entire system from shutting down in the event of this fault, you can program a fault routine to clear type 4, code 20, major faults.

Recover from a major fault

If the fault is cleared, the faulted instruction does not continue to run.

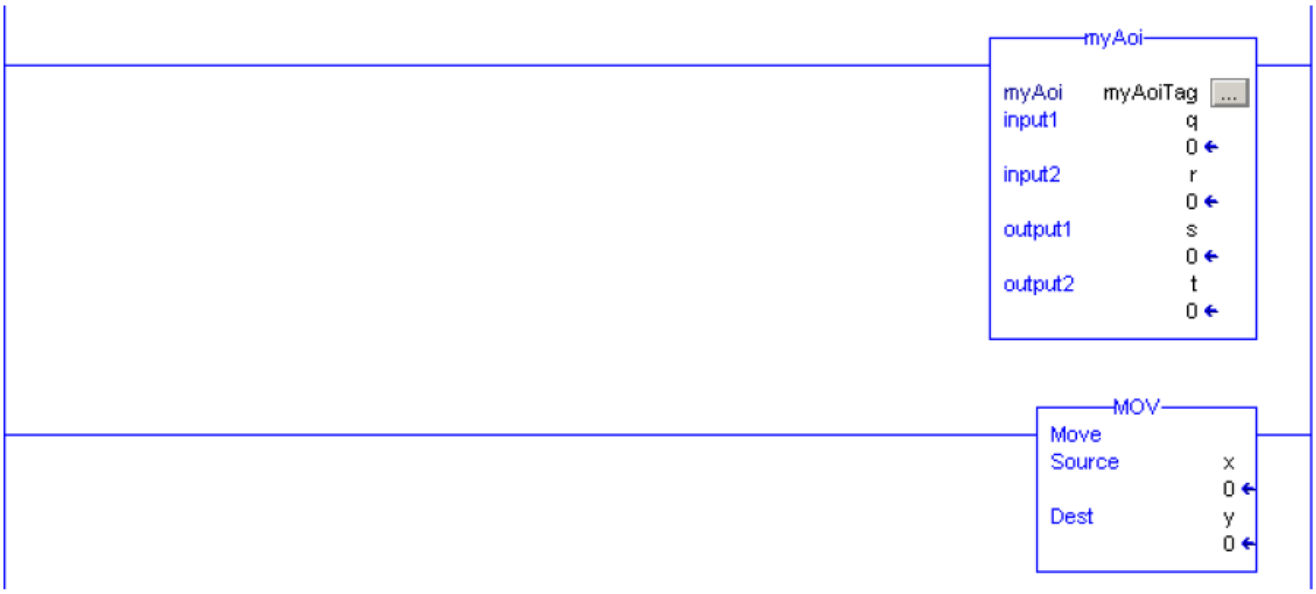
Example 1

In the following example, a JSR instruction passes an input parameter containing an indirect address that is out of bounds. If the fault is cleared, the JSR instruction is aborted (the subroutine does not run) and execution resumes with the EQU.



Example 2

In the following example, the logic inside an Add-On Instruction generates a fault. While the logic of an Add-On Instruction may look like a subroutine, it is not–the Add-On Instruction is an instruction. When a fault occurs inside an Add-On Instruction, the remainder of the Add-On Instruction is aborted. If the fault is cleared, execution resumes with the MOV.



Important points regarding Add-On Instructions

Keep the following considerations in mind when using Add-On Instructions and major faults.

- The Add-On Instruction stops running at the instruction that caused the fault. This means that the remainder of the scan mode routine does not run.

- If the fault is cleared, execution resumes at the instruction following the top-level Add-On Instruction invocation. For example, assume the Add-On Instruction *myAoi* in Example 2 invokes a nested Add-On Instruction *myNested*, which invokes another nested Add-On Instruction inner. Furthermore, assume that an instruction inside of inner causes a fault. If the fault is cleared, execution resumes with the MOV (the remainder of inner does not execute; the remainder of *myNested* does not execute; and the remainder of *myAoi* does not execute.)
- During prescan:
  - The Logic routine runs (in prescan mode).
  - The Prescan routine runs (in normal scan mode).
- During postscan:
  - The Logic routine runs (in postscan mode).
  - The Postscan routine runs (in normal scan mode).

If a fault occurs while processing the Logic routine, the Add-On Instruction aborts (the remainder of the Logic routine does not run and the pre-scan and post-scan routines do not run). If the fault is cleared, execution resumes at the instruction following the top-level Add-On Instruction invocation.

Fault handling during prescan and postscan

The behavior of each instruction varies depending on the mode in which it runs–true, false, prescan, or postscan. For details about what a specific instruction does in each mode, see the Logix5000 Controllers General Instructions Reference Manual , publication number 1756-RM003 .

- Prescan provides a system-defined initialization of the user program when the controller is switched from program mode to run mode.
- Postscan provides a system-defined re-initialization of the logic invoked from an SFC action, when the action is shut down (if SFCs are configured for Automatic Reset).

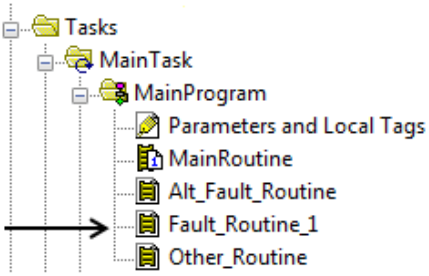
If an array index is out of range during prescan, the controller could generate a major fault. There are a number of ways this could happen: the controller loses power, encounters a major fault, or the project is saved while online. Because the user program, during prescan and postscan, cannot assign values to tags, the only way to correct these issues is to manually initialize the index variables using the Logix Designer application or to write a fault handler to ignore the array faults during prescan. To reduce the need for manual intervention, the Logix Designer application includes an internal fault handler. This handler is only used during prescan and only clears array faults (type 4, fault codes of 20 of 83).

Tip: Array faults are not ignored during postscan because the user program controls index tag values when an action is shut down.

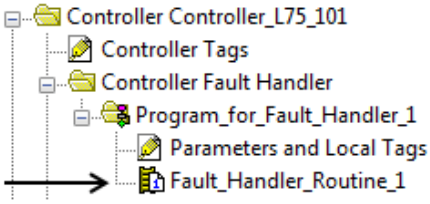
Placement of fault routines

You use a fault routine to program logic to take specific action after a fault, such as clearing the fault and continuing to run. Fault routines can be configured specific to a program, controller, or to the Power-Up Handler.

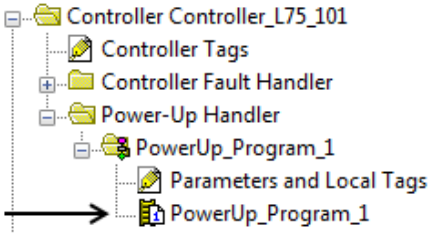
Program Fault Routine



Controller Fault Routine



Power-up Fault Handler Routine



Choose where to place the fault routine

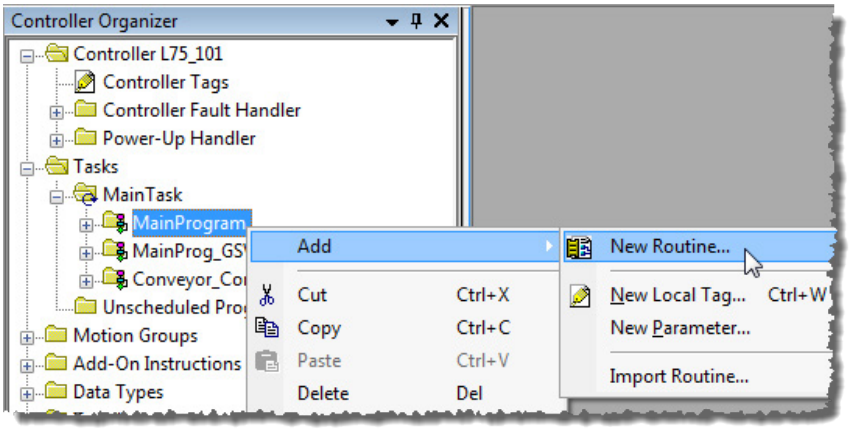
Where you place the routine depends on the type of fault that you want to handle. Use this table to determine where in the project your fault routine should be configured.

If you want to clear the fault when		See this section
Condition	**Fault Type**	
The execution of an instruction faults	4	Creating a Fault Routine for a Program on page 15
Communication with an I/O module fails	3	Creating a Routine for the Controller Fault Handler on page 17
Watchdog timer for a task expires	6	
A motion axis faults	11	
The controller powers up in Run or Remote Run mode	1	Creating a Routine for the Power-Up Handler on page 19

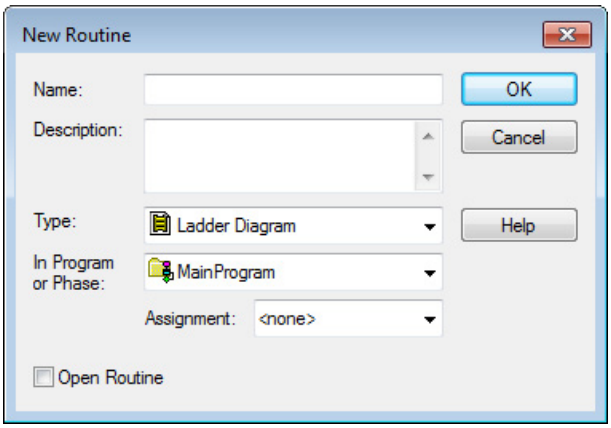
Create a fault routine for a program

Complete the following steps to create a fault routine.

1. Open the project in the Logix Designer application.
2. In the Controller Organizer, right-click **MainProgram** and click **Add>New Routine**.



3. On the **New Routine** dialog box, in the **Name** field, type the name of the routine.



4. (optional) In the **Description** field, type a description of the routine.
5. In the **Type** field, use the default setting, **Ladder Diagram**.
6. In the **In Program or Phase** field, use the default setting, **MainProgram**.

   Tip: If you are creating a fault routine for the Power-Up Handler or Controller Fault Handler, you can specify the program name of either program for the In Program or Phase option.

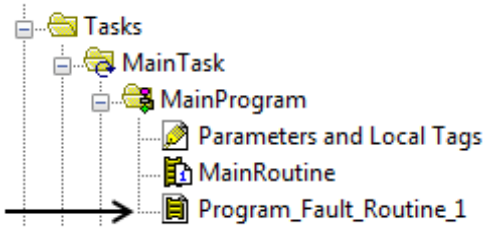
7. In the **Assignment** field, select **Fault**.
8. (optional) Select the **Open Routine** check box to immediately open the ladder logic program.
9. Click **OK**.

Change a fault routine assignment of a program

Complete the following steps to change the routine that is assigned as the fault routine.

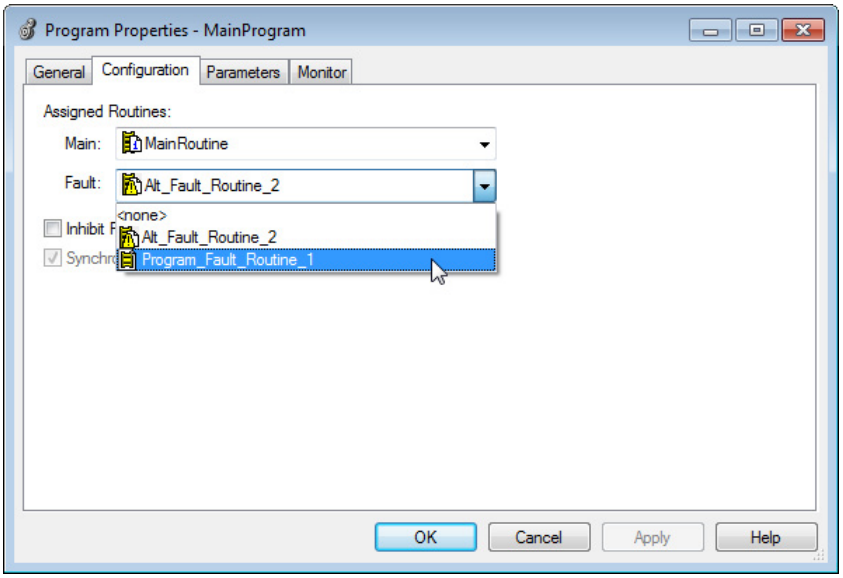
1. In the Controller Organizer, expand the **MainTask**.

   If a fault routine has already been created, it is included in the main program.



2. Right-click **MainProgram** and click **Properties**.
3. On the **Program Properties - MainProgram** dialog box, click the **Configuration** tab.

4. In the **Fault** field, select the routine you want to be the program's fault routine.



5. Click **OK**.

   The program you specified in step 4 is now indicated as the fault routine in the main program.

Create a routine for the controller fault handler

This section provides the steps for creating a fault routine to operate as the controller fault handler. Program tags are automatically created during this process.

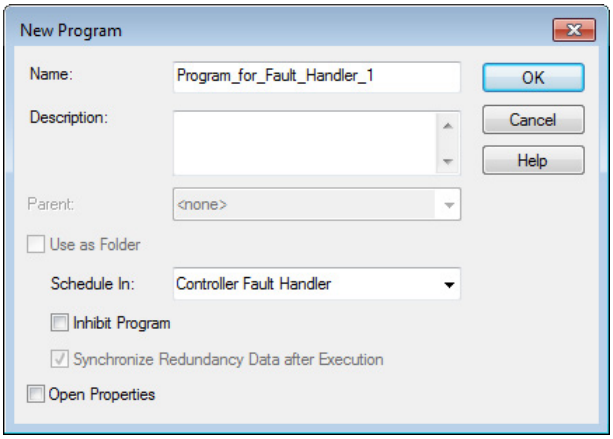
Important: When you program the fault handler, remember that any instruction that is skipped as part of the fault-handling program does not run when the main tasks and associated programs run.

For example, if your fault handler skips a JSR instruction that is causing a major fault, then that JSR instruction, including all of the programming within the subroutine, does not run.

When an instruction generates an error due to a fault (for example, a COP with an indirect addressing programming error), the instruction is skipped and does not run. This occurs with all instructions.

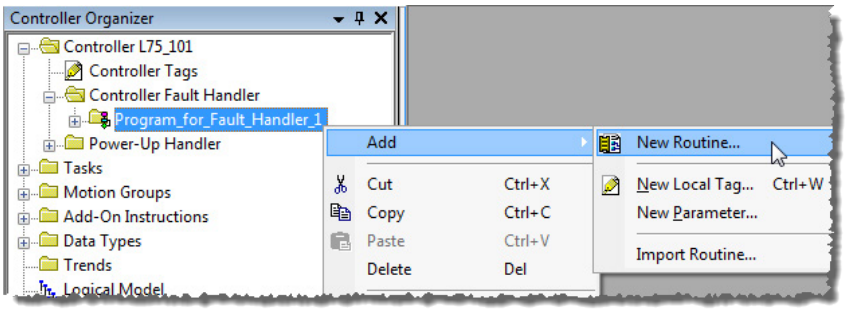
1. In the Controller Organizer, right-click **Controller Fault Handler** and click **New Program**.

2. On the **New Program** dialog box, in the **Name** field, type a program name. Verify that the **Schedule in** field is set to **Controller Fault Handler**.

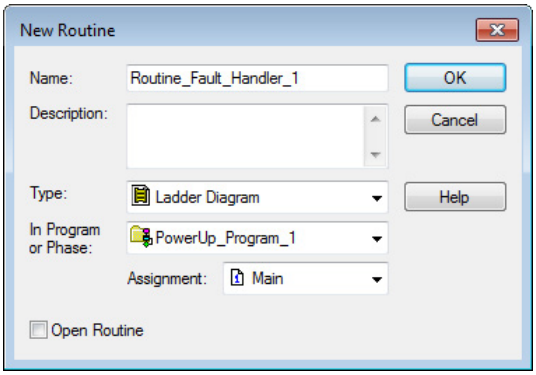


3. Click **OK**.

4. In the Controller Organizer, right-click the program you created in step 2 and click **Add>New Routine**.



5. On the **New Routine** dialog box, in the **Name** field, type a name for the routine.

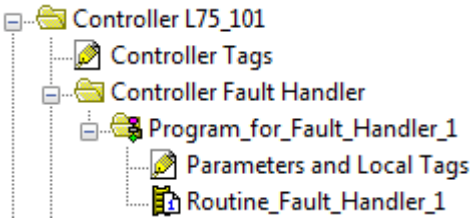


6. In the **Assignment** field, use the default setting, **Main**.

Tip: Even though you can choose **Fault** in the **Assignment** field, assigning the routine as a fault routine within the Controller Fault Handler is not necessary.

7. Click **OK**.

   The fault routine is created in the Controller Fault Handler program. If you want to edit the fault routine, double-click it to open it.



Create a routine for the power-up handler

The Power-Up Handler is an optional task that executes when the controller powers up in Run or Remote Run modes.

To	Do this
Prevent the controller from returning to Run or Remote mode	Leave the routine for the Power-Up Handler empty. When power is restored, a major fault (type 1, code 1) occurs and the controller enters the faulted state.
Direct the controller to take specific actions, then resume normal operation when power is restored	In the Power-Up Handler fault routine, complete these steps. 1. Clear the major fault (type 1, code 1). 2. Run the appropriate logic for the specific actions required.

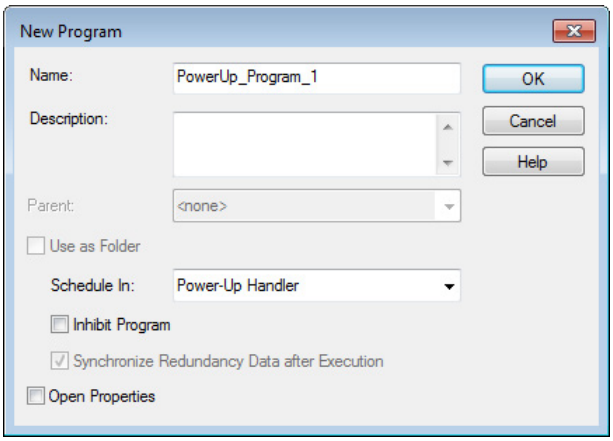
Important: Do not use fault routines to continually clear all faults on the controller. Program the fault routine to be selective in the types and number of faults cleared.

Important: When an instruction generates an error due to a fault (for example, a COP with an indirect addressing programming error), the instruction is skipped and does not run. This occurs with all instructions.

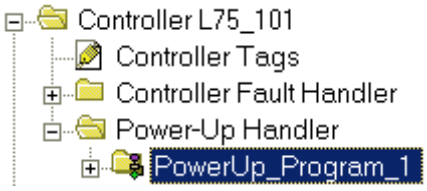
Complete the following steps to create a fault routine for the Power-Up Handler in the Logix Designer application.

1. In the Controller Organizer, right-click **Power-Up Handler** and click **New Program**.

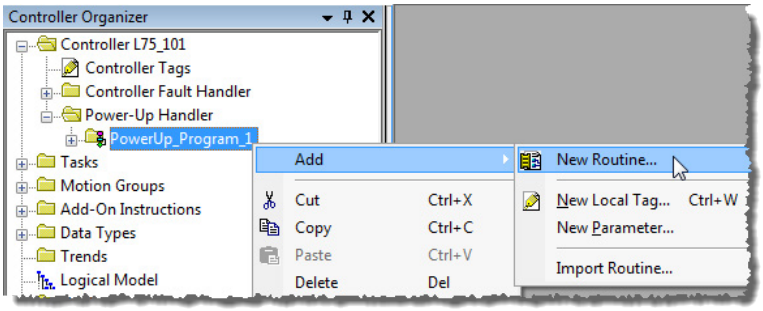
2. On the **New Program** dialog box, in the **Name** field, type a program name.



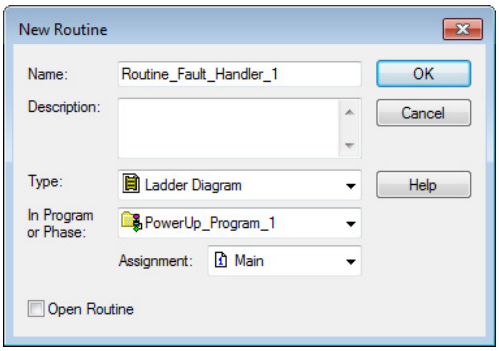
3. Click **OK**. The program is added to the Power-Up Handler.



4. Right-click the program you created in step 2 and click **Add>New Routine.**



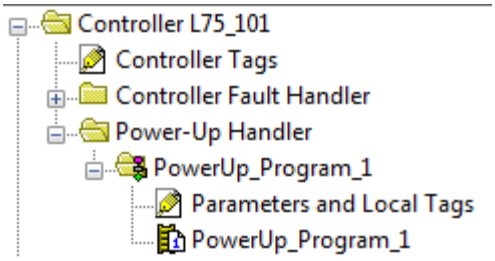
5. On the **New Routine** dialog box, in the **Name** field, type the name of the routine.



6. In the **Assignment** field, keep the default setting, **Main**.

   Tip: Even though you can choose **Fault** in the **Assignment** field, assigning the routine as a fault routine within the Power-Up Handler is not necessary.

7. Click **OK**. The fault routine is added to the Power-Up Handler. If you want to edit the new routine, double-click it to open it.



Programmatically clearing a major fault

To clear a major fault that occurs during the execution of your project, see the following sections to programmatically clear a major fault.

- Create a data type to store fault information on page 21
- Write a routine to clear the fault on page 23

Important: Do not use fault routines to continually clear all faults on the controller. Program the fault routine to be selective in the types and number of faults cleared.

Important: When an instruction generates an error due to a fault (for example, a COP with an indirect addressing programming error), the instruction is skipped and does not run. This occurs with all instructions.

Create a data type to store fault information

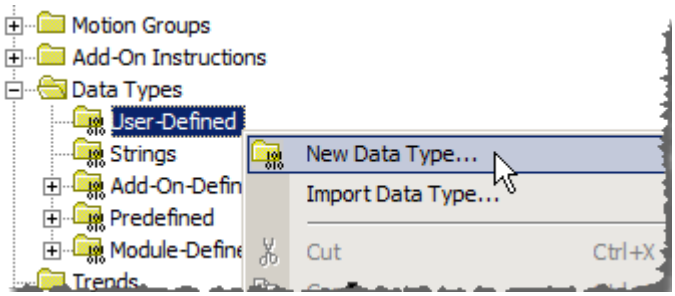
Logix5000 controllers store system information in objects. Unlike PLC-5 or SLC 500 controllers, there is no status file.

- To access system information, use a Get System Value (GSV) or Set System Value (SSV) instruction.
- For status information about a program, access the Program object.
- For fault information, access the MajorFaultRecord attribute of the Program object.

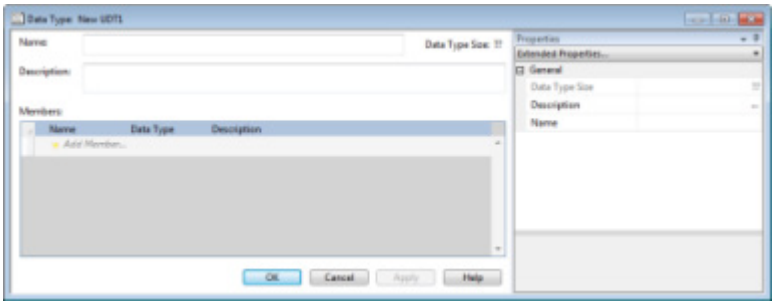
Attribute	Data Type	Instruction	Description
MajorFaultRecord	DINT[11]	GSV SSV	Records major faults for this program. Specifies the program name to determine which Program object you want, or specifies THIS to access the Program object for the program that contains the GSV or SSV instruction.

To simplify access to the MajorFaultRecord attribute, complete these steps to create a user-defined data type.

1. In the Controller Organizer, right-click **User-Defined** and click **New Data Type**.



2. On the **New Data Type** window, enter the data type information as shown in the following table.

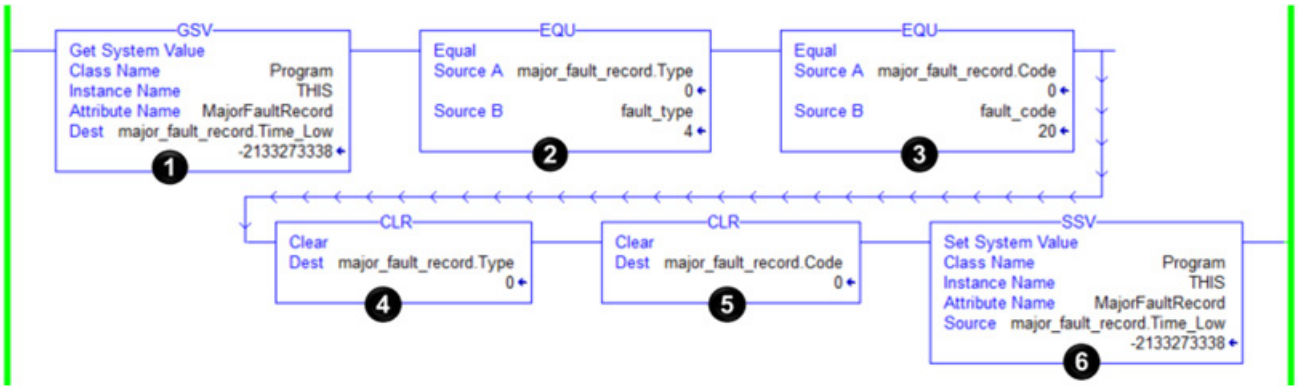


Data Type: FAULTRECORD			
Name	FAULTRECORD		
Description	Stores the MajorFaultRecord attribute or MinorFaultRecord attribute of the Program object.		
Members			
Name	**Data Type**	**Style**	**Description**
Time_Low	DINT	Decimal	Lower 32 bits of the fault timestamp value
Time_High	DINT	Decimal	Upper 32 bits of the fault timestamp value
Type	INT	Decimal	Fault type (program, I/O, and so forth)
Code	INT	Decimal	Unique code for the fault
Info	DINT[8]	Hex	Fault specific information

3. Click **OK**.

Write a routine to clear the fault

The following example shows a fault routine to clear a major fault.



Item	Reason	Description
1	Gets the fault type and code	The GSV instruction: • Accesses the MajorFaultRecord attribute of this program. This attribute store information about the fault. • Stores the fault information in the major_fault_record (of type FAULTRECORD) tag. When the tag is based on a structure, enter the first member of the tag.
2	Checks for a specific fault.	The first EQU instruction checks for a specific type of fault, such as program, I/O. In Source B, enter the value for the type of fault that you want to clear.
3		The second EQU instruction checks for a specific fault code. In Source B, enter the value for the code that you want to clear.
4	Sets the fault code and fault type to zero	The first CLR instruction sets the value of the fault type in the major_fault_record tag to zero.
5		Add the second CLR instruction sets the value of the fault code in major_fault_record tag to zero.
6	Clears the fault	The SSV instruction writes: • The new values to the MajorFaultRecord attribute of this program. • The values contained in the major_fault_record tag. Because the Type and Code member are set to zero, the fault clears and the controller resumes execution.

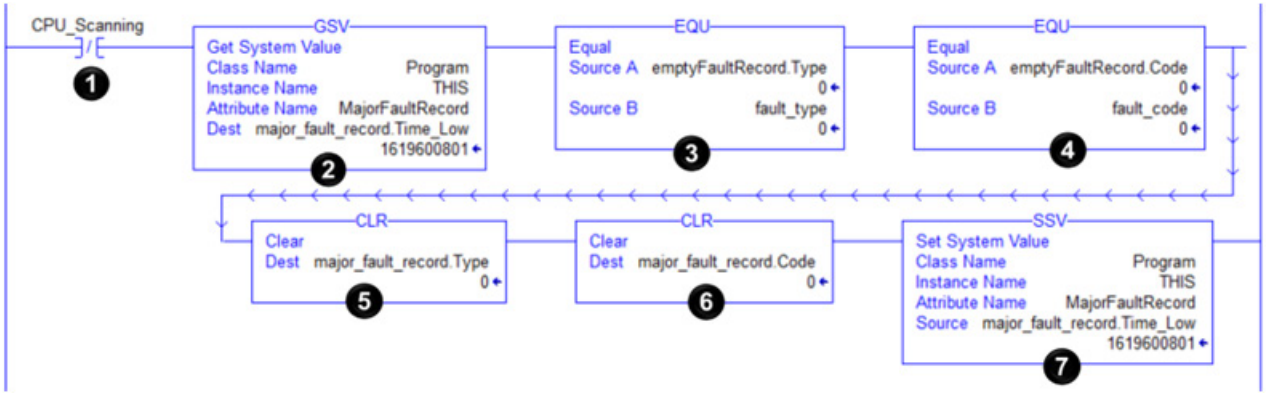
Clear a major fault during prescan

If the controller faults immediately after you switch it to the Run mode, examine the prescan operation for the fault. Depending on the revision of your controller, an array subscript that is beyond the range of the array (out of range) during prescan might cause a fault.

If controller is revision	Then
11.x or earlier	During prescan, an array subscript that is beyond the range of the array (out of range) produces a major fault.
12.x	See the release notes for the firmware of your controller.
13.0 or later	During prescan, the controller automatically clears any faults due to an array subscript that is beyond the range of the array (out of range).

The following example shows a fault routine to clear a major fault that occurs during prescan.

Important: It is good programming practice to check for a specific fault before clearing that fault.



Item	Reason	Description
1	Identifies when the controller is in prescan.	The fault routine of this program uses the status of this bit to determine if the fault occurred during prescan or normal scan of the logic. • During prescan, this bit is off. During prescan, the controller resets all bits that are referenced by OTE instructions. • When the controller begins to run the logic, the CPU_scanning bit is always on.
2	Gets the fault type and code	The GSV instruction does the following: • Accesses the MajorFaultRecord attribute of this program. This attribute stores information about the fault. • Stores the fault information in the major_fault_record (of type FAULTRECORD) tag. When you enter a tag that is based on a structure, enter the first member of the tag.
3	Checks for a specific fault	The first EQU instruction checks for a fault of type 4, which means that an instruction in this program caused the fault.
4		The second EQU instruction checks for a fault of code 20, which means that either an array subscript is too large, or a POS or LEN value of a CONTROL structure is invalid.
5		The first CLR instruction sets the value of the fault type in the major_fault_record tag to zero.
6		The second CLR instruction sets the value of the fault type in the major_fault_record tag to zero
7	Clears the fault	The SSV instruction does the following: • Writes the new values to the MajorFaultRecord attribute of this program. • Writes the values contained in the major_fault_record tag. Because the Type and Code member are set to zero, the fault clears and the logix starts running again.

Test a fault routine

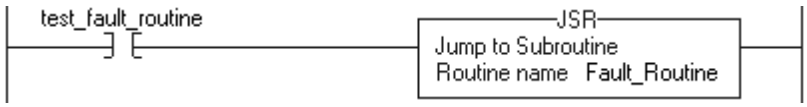
Use a JSR instruction to test the fault routine of a program without creating an error (simulate a fault).

1. Create a BOOL tag that is used to initiate the fault.

2. In the main routine or a subroutine of the program, enter the following rung, where:

   - test_fault_routine is the tag used to initiate the fault.
   - Fault_Routine is the fault routine of the program.

   When test_fault_routine is on, a major fault occurs and the controller executes Fault_Routine.



Create a user-defined major fault

To suspend (shut down) the controller based on conditions in the application, create a user-defined major fault. With a user-defined major fault:

- The fault type = 4.
- Define a value for the fault code. Choose a value between 990 and 999. These codes are reserved for user-defined faults.
- The controller handles the fault the same as other major faults:
  - The controller changes to the Program mode and stops running the logic.
  - Outputs are set to their configured state or value for faulted mode.

Example: When Tag_1.0 = 1, produce a major fault and generate a fault code of 999.

See the following procedures to create a user-defined major fault.

- Create a fault routine for the program. on page 25
- Configure the program to use the fault routine. on page 26
- Jump to the fault routine. on page 26

Create a fault routine for the program

If a program already contains a fault routine, see Jump to the fault routine on page 26.

If a program does not contain a fault routine, complete these steps to add one.

1. In the Controller Organizer, right-click the program and click **Add>New Routine.**

2. On the **New Routine** dialog box, in the **Name** field, type a name for the fault routine.
3. In the **Type** field, use the default setting, **Ladder Diagram**.
4. In the **In Program or Phase** field, select the program or phase where the routine will reside.
5. In the **Assignment** field, select **Fault**.
6. (optional) Select the **Open Routine** check box, to immediately open the ladder logic program.
7. Click **OK**.

Configure the program to use the fault routine

1. In the Controller Organizer, right-click the program and click **Properties**.
2. On the **Properties** dialog box, click the **Configuration** tab.
3. In the **Fault** field, select the fault routine.
4. Click **OK**.

Jump to the fault routine

In the main routine of the program, enter the following rung, where:

- Fault_Routine is the name of the fault routine for the program.
- 999 is the value for the fault code.

Create a user-defined major fault

When Tag_1.0 = 1, execution jumps to name_of_fault_routine. A major fault occurs and the controller enters the faulted mode. Outputs go to the faulted state. The **Controller Properties** dialog box, **Major Faults** tab, displays the code 999.

Major fault codes

The type and code correspond to the type and code displayed in the following locations.

- **Controller Properties** dialog box, **Major Faults** tab
- Program object, MajorFaultRecord attribute

Type	Code	Cause	Recovery Method
1	1	The controller powered on in Run mode.	Run the power-loss handler.
1	16	I/O communication configuration fault detected. (CompactLogix 1768-L4x controllers only.)	Reconfigure the number of communication modules on the 1768 bus side of the controller: • 1768-L43 has a maximum of two modules. • 1768-L45 has a maximum of four modules. • Up to four Sercos modules • Up to two NetLinx communication modules

1	40	If the controller uses a battery, the battery does not contain enough charge to save the user program on power-down.
		If the controller uses an ESM (Energy Storage Module), the ESM does not contain enough charge to save the user program on power-down.
			• For controllers that use a battery, replace the battery.
			• For controllers that use an ESM (Energy Storage Module):
			  • Allow the ESM to fully charge before powering down the controller.
			  • Replace the ESM if the ESM is removable, or replace the controller if the ESM is not removable.
			• If the problem persists, contact Rockwell Automation support.

1	60	For a controller with **no** memory card installed, the controller: • Detected a non-recoverable fault. • Cleared the project from memory.	1. Clear the fault. 2. Download the project. 3. Change to Remote Run or Run mode. Follow these steps if the fault persists. 1. Before you cycle power to the controller, record the state of the OK and RS232 status indicators. 2. Contact Rockwell Automation support.
1	61	For a controller with a memory card installed, the controller: • Detected a non-recoverable fault. • Wrote diagnostic information to the memory card. • Cleared the project from memory.	1. Clear the fault. 2. Download the project. 3. Change to Remote Run or Run mode. If the fault persists, contact Rockwell Automation support.
1	62	For a controller with a Secure Digital (SD) card installed, the controller: • Detected a nonrecoverable fault. • Wrote diagnostic information to the memory card. When in this state, the controller does not open any connections or allow a transition to Run mode.	1. Clear the fault. 2. Download the project. 3. Change to Remote Run or Run mode. If the fault persists, contact Rockwell Automation support.

Type	Code	Cause	Recovery Method
3	16	A required I/O module connection failed.	Check the: • I/O module is in the chassis. • Electronic keying requirements. • Controller Properties **Major Fault** tab and the Module Properties **Connection** tab for more information about the fault.
3	20	Possible fault with the chassis.	Not recoverable. Replace the chassis.
	21		
3	23	At least one required connection was not established before going to Run mode.	Wait for the controller I/O light to turn green before changing to Run mode.
4	16	Unknown instruction encountered.	Remove the unknown instruction. This probably happened due to a program conversion process.
4	20	Array subscript too big, control structure .POS or .LEN is invalid.	Adjust the value to be within the valid range. Do not exceed the array size or go beyond dimensions defined.
4	21	Control structure .LEN or .POS < 0.	Adjust the value so it is > 0.
4	31	The parameters of the JSR instruction do not match those of the associated SBR or RET instruction.	Pass the appropriate number of parameters. If you pass too many parameters, the extra ones are ignored without any error.
4	34	A timer instruction has a negative preset or accumulated value.	Fix the program so that it cannot load a negative value into timer preset or accumulated value.
4	42	JMP to a label that did not exist or was deleted.	Correct the JMP target or add the missing label.
4	82	A sequential function chart (SFC) called a subroutine and the subroutine tried to jump back to the calling SFC. This occurs when the SFC uses either a JSR or FOR instruction to call the subroutine.	Remove the jump back to the calling SFC.
4	83	The data tested was not inside the required limits. This occurs with array subscripts used with Boolean arrays and bit level addressing.	Adjust the value to be within the valid range. Do not exceed the array size or go beyond the dimensions defined.
4	84	Stack overflow.	Reduce the subroutine nesting levels or the number of parameters passed.
4	89	In an SFR instruction, the target routine does not contain the target step.	Correct the SFR target or add the missing step.
4	90	A safety instruction occurs outside a safety task.	Place the safety instruction inside the safety task.
4	91	Equipment Phase instruction is being called from outside an Equipment Phase program.	Only use the instruction in an Equipment Phase program.
4	94	Nesting limits exceeded.	Restructure the project to reduce the subroutine nesting levels.
4	990	User-defined major faults. See Creating a User-Defined Major Fault on page 25 for more information.	
4	991		
4	992		
4	993		
4	994		
4	995		
4	996		
4	997		
4	998		

Type	Code	Cause	Recovery Method
4	999		
6	1	Task watchdog expired. User task has not completed in the specified period. A program error caused an infinite loop, or the program is too complex to run as quickly as specified, or a higher priority task is keeping this task from finishing.	Take one or more of these steps: • Increase the task watchdog. • Shorten the execution time. • Make the priority of this task higher. • Simplify higher priority tasks. • Move some code to another controller.
7	40	Store to nonvolatile memory failed.	1. Try again to store the project to nonvolatile memory. 2. If the project fails to store to nonvolatile memory, replace the memory card. If you are using a 1756-L7*x* controller, verify that the SD card is unlocked.
7	41	Load from nonvolatile memory failed due to controller type mismatch.	Change to a controller of the correct type or download the project and store it on the memory card.
7	42	Load from nonvolatile memory failed because the firmware revision of the project in nonvolatile memory does not match the firmware revision of the controller.	Update the controller firmware to the same revision level as the project that is in nonvolatile memory.
7	43	Load from nonvolatile memory failed due to bad checksum.	Contact Rockwell Automation support.
7	44	Failed to restore processor memory.	Contact Rockwell Automation support.

7	50	The log file certificate cannot be verified. When the controller starts up it attempts to verify the log file key/certificate combination. Depending on the verification, the controller takes one of the following actions:
		• If the controller verifies the existing log file certificate, the controller continues with existing log directory.
		• If the existing certificate cannot be verified, the controller logs a major fault and attempts to create a new certificate.
		  • If the controller successfully creates a new certificate, it creates a backup log subdirectory, moves the existing files to that directory, and continues logging and signing with the new verification key and log file certificate.
		  • If the controller cannot create a new certificate, the controller writes log entries to the existing log directory, but does not update signature files in that directory.
			Clear the fault and power cycle the controller. If the problem persists, contact Rockwell Automation support.

8	1	Attempted to place controller in Run mode with keyswitch during download.	Wait for the download to complete and clear the fault.
11	1	Actual position has exceeded positive overtravel limit.	Move axis in negative direction until position is within overtravel limit and then execute Motion Axis Fault Reset.
11	2	Actual position has exceeded negative overtravel limit.	Move axis in positive direction until position is within overtravel limit and then run Motion Axis Fault Reset.
11	3	Actual position has exceeded position error tolerance.	Move the position within tolerance, then run Motion Axis Fault Reset.
11	4	Encoder channel A, B, or Z connection is broken.	Reconnect the encoder channel, then run Motion Axis Fault Reset.
11	5	Encoder noise event detected or the encoder signals are not in quadrature.	Fix encoder cabling, then run Motion Axis Fault Reset.

Type	Code	Cause	Recovery Method
11	6	Drive Fault input was activated.	Clear Drive Fault then run Motion Axis Fault Reset.
11	7	Synchronous connection incurred a failure.	Try these steps: • First, run Motion Axis Fault Reset. • If that does not work, pull the servo module out and plug it back in. • If this does not work, replace the servo module.
11	8	Servo module has detected a serious hardware fault.	Replace the module.
11	9	Asynchronous Connection has incurred a failure.	Try these steps: • First, run Motion Axis Fault Reset. • If that does not work, pull the servo module out and plug it back in. • If this does not work, replace the servo module.
11	10	Motor fault has occurred.	See the DriveFaults axis tag for more information.
11	11	Motor thermal fault has occurred.	See the DriveFaults axis tag for more information.
11	12	Drive thermal fault has occurred.	See the DriveFaults axis tag for more information.
11	13	SERCOS ring fault has occurred.	Verify the integrity of the SERCOS fiber-optic ring network and the devices on it.
11	14	Drive enable input fault has occurred.	Re-enable the drive enable input and clear the fault.
11	15	Drive phase loss fault has occurred.	Restore full power connection to the drive and clear the fault.
11	16	Drive guard fault has occurred.	See the GuardFaults axis tag for more information.
11	32	The motion task has experienced an overlap.	The group's course update rate is too high to maintain correct operation. Clear the group fault tag, raise the group's update rate, and then clear the major fault.
11	33	Coordinated System Time reference has been lost.	

12	32	A disqualified secondary controller cycled power and no partner chassis or controller was found upon powerup.	Verify the following items.
			• A partner chassis is connected.
			• Power is applied to both redundant chassis.
			• Partner controllers have the same:
			  • Catalog number
			  • Slot number
			  • Firmware revision

12	33	An unpartnered controller has been identified in the new primary chassis after a switchover.	Either: • Remove the unpartnered controller and troubleshoot the cause of the switch over. • Add a partner controller to the secondary chassis, troubleshoot the cause of the switch over, and synchronize the system.
12	34	Just after a switchover occurs, the keyswitch positions of the primary and secondary controllers are mismatched. The old primary controller is in Program mode and the new primary controller is in Run mode.	Either: • Change the keyswitches from Run to Program to Run mode twice to clear the fault. • Use the Logix Designer application to go online with the controllers. Then, clear the faults and change both the controller's mode to Run.
14	1	Safety Task watchdog expired. User task has not completed in a specified period. A program error caused an infinite loop, the program is too complex to run as quickly as specified, a higher priority task is keeping this task from finishing, or the safety partner has been removed.	Clear the fault. If a safety task signature exists, safety memory is re-initialized and the safety task begins executing. If a safety task signature does not exist, you must re-download the program to allow the safety task to run. Reinsert the safety partner, if it was removed.
14	2	An error exists in a routine of the safety task.	Correct the error in the user-program logic.
14	3	Safety partner is missing.	Install a compatible safety partner.

Type	Code	Cause	Recovery Method
14	4	Safety partner is unavailable.	Install a compatible safety partner.
14	5	Safety partner hardware is incompatible.	Install a compatible safety partner.
14	6	Safety partner firmware is incompatible.	Update the safety partner so that the firmware major and minor revision matches the primary controller.
14	7	Safety task is inoperable. This fault occurs when the safety logic is invalid. For example, a mismatch in logix exists between the primary controller and safety partner, a watchdog timeout occurred, or memory is corrupt.	Clear the fault. If a safety task signature exists, safety memory is re-initialized via the safety task signature and the safety task begins executing. If a safety task signature does not exist, you must download the program again to allow the safety task to run.
14	8	Coordinated system time (CST) not found.	Clear the fault. Configure a device to be the CST master.
14	9	Safety partner nonrecoverable controller fault.	Clear the fault and download the program. If the fault persists, replace the safety partner.
17	34	Controller internal temperature has exceeded operating limit.	Measures should be taken to reduce the ambient temperature of the module. Follow the recommended limits for the ambient (inlet) temperature and apply the required clearance around the chassis.
18	1	The CIP Motion drive has not initialized correctly.	To determine corrective action, see the CIP Initialization attribute for details about the type of fault that occurred. For more information about the CIP Initialization attribute, see the Integrated Motion on the Ethernet/IP Network Configuration and Startup User Manual , publication MOTION-UM003 .
18	2	The CIP Motion drive has not initialized correctly. This fault is indicated when a manufacturer-specific initialization fault has occurred.	To determine the corrective action, see the CIP Initialization Fault - Mfg attributes for details about the fault that occurred. For more information about the CIP Initialization Fault - Mfg attribute, see the Integrated Motion on the Ethernet/IP Network Configuration and Startup User Manual , publication MOTION-UM003 .
18	3	The Physical Axis Fault bit is set, indicating a fault on the physical axis.	To determine corrective action, see the CIP Axis Fault attributes for details about the fault that occurred. For more information about the CIP Axis Fault attributes, see the Integrated Motion on the Ethernet/IP Network Configuration and Startup User Manual , publication MOTION-UM003 .
18	4	The Physical Axis Fault bit is set, indicating a fault on the physical axis. This fault is indicated when a manufacturer-specific axis fault has occurred.	To determine corrective action, see the CIP Axis Fault - Mfg attributes for details about the fault that occurred. For more information about the CIP Axis Fault -Mfg attributes, see the Integrated Motion on the Ethernet/IP Network Configuration and Startup User Manual , publication MOTION-UM003 .
18	5	A motion fault occurred.	To determine corrective action, see the Motion Fault attribute and Motion Fault bits for details about the fault that occurred. For more information about the Motion Fault attribute and Motion Fault bits, see the Integrated Motion on the Ethernet/IP Network Configuration and Startup User Manual , publication MOTION-UM003 .
18	6	A CIP Motion Drive fault has occurred. Usually the fault affects all the axes associated with the module and all of the associated axes are shut down.	Reconfigure the faulted motion module to correct the fault. For more information about the Module Fault attribute and Module Fault bits, see the Integrated Motion on the Ethernet/IP Network Configuration and Startup User Manual , publication MOTION-UM003 .
18	7	A motion group fault has occurred. Usually the fault affects all of the axes associated with a motion group.	Reconfigure the entire motion subsystem to correct the fault. See the Group Fault attribute if details about the fault are needed.

Type	Code	Cause	Recovery Method
18	8	A fault has occurred during the configuration of a CIP Motion Drive. Typically, this fault occurs after an attempt to update an axis configuration attribute of a CIP Motion Drive was unsuccessful.	To determine the corrective action, see the Configuration Fault in the Attribute Error Code and Attribute Error ID attributes associated with the motion or 1756-EN*x*T module.
18	9	An Absolute Position Recovery (APR) fault has occurred and the absolute position of the axis cannot be recovered.	To determine the corrective action, see the APR Fault attributes to determine the cause of the fault. For more information about the APR Fault attributes, see the Integrated Motion on the Ethernet/IP Network Configuration and Startup User Manual , publication MOTION-UM003 .
18	10	An Absolute Position Recovery (APR) fault has occurred and the absolute position of the axis cannot be recovered. This fault is indicated when a manufacturer-specific APR fault has occurred.	To determine the corrective action, see the APR Fault - Mfg attributes to determine the cause of the fault. For more information about the APR Fault - Mfg attributes, see the Integrated Motion on the Ethernet/IP Network Configuration and Startup User Manual , publication MOTION-UM003 .
18	128	A fault specific to the Guard Motion safety function has occurred. This fault is applicable only when a drive with Guard Safety functionality is used.	To determine the corrective action, see the Guard Motion attributes and Guard Status bits to determine the cause of the fault. For more information about the Guard Motion attributes and Guard Status bits, see the Integrated Motion on the Ethernet/IP Network Configuration and Startup User Manual , publication MOTION-UM003 .
20	1	A required license is missing or expired during the transition to run or test mode.	Insert a CmCard containing all licenses required by the project in the controller.

Minor Faults

This chapter explains minor fault codes and how to work with them in the Logix Designer application.

Identify minor faults

The following table explains how to use ladder logic to monitor information about common minor faults.

To check for a	Do this
Task overlap	1. Enter a GSV instruction that gets the FaultLog object, MinorFaultBits attribute. 2. Monitor bit 6.
Load from nonvolatile memory	1. Enter a GSV instruction that gets the FaultLog object, MinorFaultBits attribute. 2. Monitor bit 7.
Serial port fault	1. Enter a GSV instruction that gets the FaultLog object, MinorFaultBits attribute. 2. Monitor bit 9.
Low battery, energy storage status or uninterruptable power supply (UPS) fault	1. Enter a GSV instruction that gets the FaultLog object, MinorFaultBits attribute. 2. Monitor bit 10.
Instruction-related fault	1. Create a user-defined data type that stores the fault information. Name the data type FaultRecord and assign the following members. **Name** / **Data Type** / **Style** TimeLow / DINT / Decimal TimeHigh / DINT / Decimal Type / INT / Decimal Code / INT / Decimal Info / DINT[8] / Hex 2. Create a tag that stores the values of the MinorFaultRecord attribute. 3. From the Data Type menu in step 1 of this instruction, choose the data type. 4. Monitor S:MINOR. 5. Use a GSV instruction to get the values of the MinorFaultRecord attribute if S:MINOR is on. 6. Reset S:MINOR if you want to detect a minor fault that is cause by another instruction. S:MINOR remains set until the end of the scan.

Minor fault examples

Use these examples to check for minor faults.

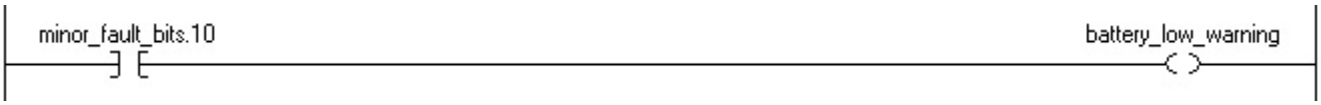
The following example checks for a low battery warning.

Example: **Check for a minor fault.**

Minor_fault_check times for 1 minute (60000 ms) and then automatically restarts itself.

Every minute, minor_fault_check.DN turns on for one scan. When this occurs, the GSV instruction gets the value of the FaultLog object, MinorFaultBits attribute, and stores it in the minor_fault_bits tag. Because the GSV instruction only runs once every minute, the scan time of most scans is reduced.

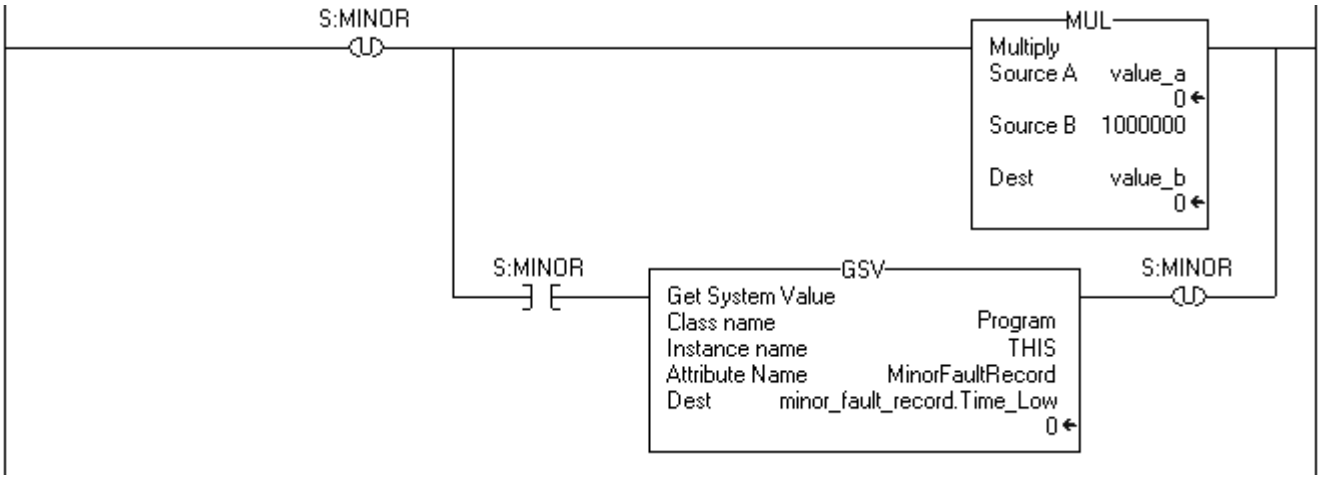
If minor_fault_bits.10 is on, depending on the controller, the battery is low or the ESM or UPS needs to be replaced or is missing.



The following example checks for a minor fault that is caused by a specific instruction.

Example: **Check for a minor fault that is caused by an instruction.**

- Multiply value_a by 1000000 and check for a minor fault, such as a math overflow.
- To make sure that a previous instruction did not produce the fault, the rung first clears S:MINOR.
- The rung then executes the multiply instruction.
- If the instruction produces a minor fault, the controller sets S:MINOR.
- If S:MINOR is set, the GSV instruction gets information about the fault and resets S:MINOR.



Minor fault codes

Minor faults are indicated in the following locations.

- **Controller Properties** dialog box, **Minor Faults** tab
- Program object, MinorFaultRecord attribute

The following tables identify the type, code, and suggested recovery method when applicable.

Type	Code	Cause	Recovery Method
1	15	• A 1769 power supply is connected directly to the controller's 1768 CompactBus, with an invalid configuration. • The 1768 power supply powering the controller has failed.	• Remove the power supply from the 1768 CompactBus and cycle power to the system. • Replace the power supply.
3	1	Bus off condition. The connections between the controller and the I/O modules are broken.	Complete these steps to identify the source of the BUS OFF fault: 1. The number of local expansion modules in the project matches the number of modules that are physically installed in the system. 2. All mounting bases are locked and I/O modules are securely installed on mounting bases. 3. All 1734 POINT I/O modules are configured to use the Autobaud rate. If these steps do not remedy the fault condition, contact Rockwell Automation support.
3	94	The current RPI update of an I/O module overlaps with its previous RPI update.	Set the RPI rate of the I/O modules to a higher numerical value. Rockwell Automation recommends that the CompactLogix 5370 L2 and CompactLogix 5370 L3 control systems do not run with Module RPI Overlap faults.
4	4	An arithmetic overflow occurred in an instruction.	Examine the arithmetic operations (order) or adjust the values.
4	5	In a GSV/SSV instruction, the specified instance was not found.	Check the instance name.
4	6	In a GSV/SSV instruction, either the: • Specified Class name is not supported. • Specified Attribute name is not valid.	Check the Class name and Attribute name.
4	7	The GSV/SSV destination tag was too small to hold the data.	Fix the destination so it has enough space.
4	30	Bad parameters passed through to the ASCII port.	Verify the ASCII configuration settings.
4	35	PID delta time ≤ 0.	Adjust the PID delta time so that it is > 0.
4	36	PID setpoint out of range.	Adjust the setpoint so that it is within range.
4	51	The LEN value of the string tag is greater than the DATA size of the string tag.	1. Check that no instruction is writing to the LEN member of the string tag. 2. In the LEN value, enter the number of characters that the string contains.
4	52	The output string is larger than the destination.	Create a new string data type that is large enough for the output string. Use the new string data type as the data type for the destination.

Type	Code	Cause	Recovery Method
4	53	The output number is beyond the limits of the destination data type.	Either: • Reduce the size of the ASCII value. • Use a larger data type for the destination.
4	56	The Start or Quantity value is invalid.	Check the following: 1. The Start value is between 1 and the DATA size of the Source. 1. The Start value plus the Quantity Value is less than or equal to the DATA size of the Source.
4	57	The AHL instruction failed to execute because the serial port is set to no handshaking.	Either: • Change the Control Line setting of the serial port. • Delete the AHL instruction.
6	2	Periodic task overlap. Periodic task has not completed before it is time to run again.	Make changes, such as simplifying programs, lengthening the period, or raising the relative priority.
6	3	Event task overlap. Event task does not complete before it is time to execute again.	Make changes such as simplifying programs, lengthening the period, raising the relative priority, or slowing the triggering event.
7	49	When the controller loads a project from nonvolatile memory, it logs this minor fault and sets the FaultLog object, MinorFaultBits attribute, bit 7.	Clear the fault.
9	0	Unknown error while servicing the serial port.	Contact Rockwell Automation Technical Support.
9	1	The CTS line is not correct for the current configuration.	Disconnect and reconnect the serial port cable to the controller. Make sure the cable is wired correctly.
9	2	Poll list error. A fault was detected with the DF1 master's poll list, such as specifying more stations than the size of the file, specifying more than 255 stations, trying to index past the end of the list, or polling the broadcast address (STN #255).	Check for the following errors in the poll list. • Total number of stations is greater than the space in the poll list tag. • Total number of stations is greater than 255. • Current station pointer is greater than the end of the poll list tag. • A station number greater than 254 was encountered.
9	3	The RS232 DFI Master Active Station tag is unspecified.	Specify a tag to be used for the Active Station Tag in the System Protocol tab of the Controller Properties.
9	5	DF1 slave poll timeout. The poll watchdog has timed out for slave. The master has not polled this controller in the specified amount of time.	Determine and correct delay for polling.
9	9	Modem contact was lost. DCD or DSR control lines are not being received in proper sequence or state.	Correct modem connection to the controller.
9	10	Data has been dropped or lost from the serial port.	Slow down the rate at which the initiator is sending data.
10	10	Battery not detected or needs to be replaced.	Install a new battery.
10	11	Safety partner battery not detected or needs to be replaced.	Install a new battery.
10	12	The Energy Storage Module (ESM) is not installed. If the controller is powered-down, the WallClockTime attribute and program are not maintained.	Install an ESM in the controller.
10	13	The installed ESM is not compatible with the controller.	Replace the installed ESM with one that is compatible with the controller.
10	14	The ESM needs to be replaced due to a hardware fault. It is not capable of maintaining the WallClockTime attribute or controller program at powerdown.	Replace the ESM.

Type	Code	Cause	Recovery Method
10	15	The ESM cannot store enough energy to maintain the WallClockTime attribute or the controller program at power-down.	Replace the ESM.
10	16	The uninterruptable power supply (UPS) is missing or not ready.	Either: • Install the UPS. • Check the UPS to make sure it is adequately charged to provide backup power in the event of power loss.
10	17	The UPS battery has failed and needs to be replaced.	Replace the battery in the UPS.
13	21	Wall Clock Time out of range.	Verify the Wall Clock Time is set to the correct date/time.
17	1…*n*	An internal controller diagnostic has failed.	Contact Rockwell Automation Technical Support with the fault type and fault code logged.
17	35	Controller internal temperature is approaching operating limit.	Measures should be taken to reduce the ambient temperature of the module. Follow the recommended limits for the ambient (inlet) temperature and apply the required clearance around the chassis.
20	1	A required license is missing or expired while the controller is in run or test mode.	Insert a CmCard containing all licenses required by the project in the controller.

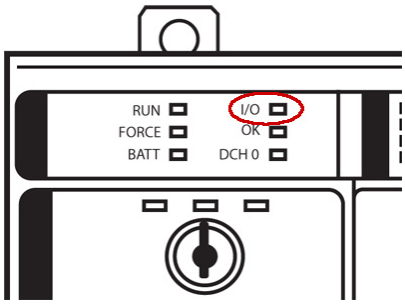
I/O Fault Codes

This chapter explains I/O fault codes and how to work with them in the Logix Designer application.

Indications of I/O faults

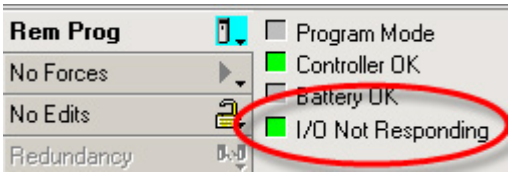
I/O faults are indicated in the following ways depending on the controller.

- The I/O indicator of the controller (shown in examples below) flashes green or red.

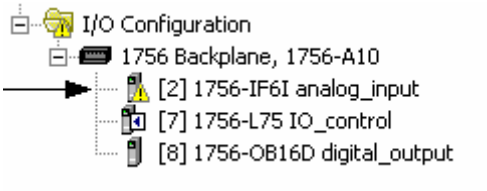


- I/O fault messages are indicated on the controller status display.

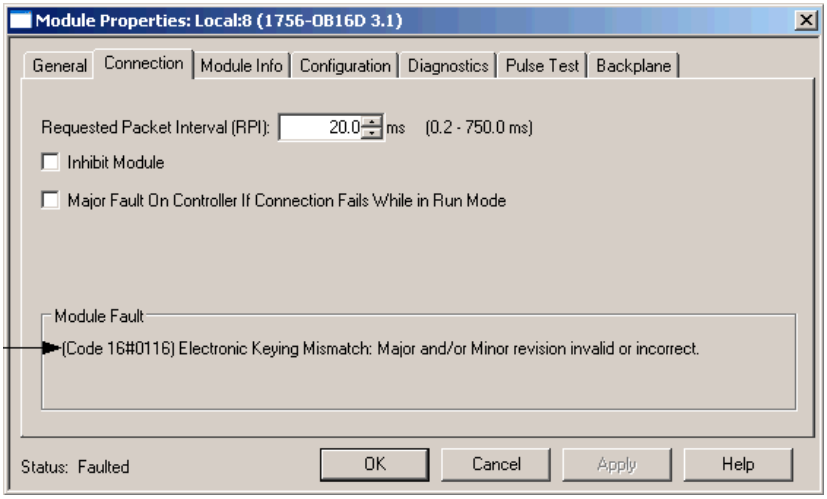
- The I/O status indicator and messages show in the controller status area of the Logix Designer application. Then indicator flashes green or red and the corresponding status message indicates an error.



- A yellow warning symbol appears on the module in the I/O Configuration tree of the Logix Designer application.



- A module fault code and description appear in the **Connection** tab of the **Module Properties** dialog box.



I/O Fault Codes

Depending where the fault code is indicated, the code format contains either the full Hexadecimal number (for example, 16#000A) or the last characters of the code (for example, #000A).

The following table lists common I/O fault codes and a corresponding description and recovery method when applicable. Each code is listed by the last characters of the full Hexadecimal number (that is, #*XXXX*).

Code	Interpretation
#0001	A connection to a module failed.
#0002	Possible causes include the following. • There are not enough connections available either for the controller or for the communication module being used to connect through. Check the connection use of the controller or communication module. If all of the connections are used, try to free some of the used connections or add another module to route the errant connection through. • The I/O memory limits of the controller are exceeded. Check the I/O memory available and make program or tag changes if needed. • The I/O module targeted does not have enough connections available. Check the number of controllers making a connection to this I/O module and verify that the number of connections is within the limits of the I/O module.

Code	Interpretation
#0005	The controller is attempting to make a connection to the module and has received an error. Possible causes include the following. • The configured address for the connection to the module is incorrect. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options were used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of Logix Designer application. For more information about electronic keying, see the user manual for the module you are using. If you are using a 1756-DHRIO module, verify that the Channel type selected in the software (DH+ or remote I/O network) matches the module's rotary switch settings.
#0006	Possible causes include the following. • The response buffer is too small to handle the response data. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passed the electronic keying test. This may result when Disable Keying or Compatible Keying options were used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#0007	A service request is unconnected, but should be connected.
#0008	The controller has requested a service which is not implemented on the target module.
#0009	The configuration for the module is invalid. The module configuration may have been changed in the Data Monitor or programmatically. Open the Connections tab of the Module Properties dialog box for the additional fault code. The additional fault code indicates the configuration parameter that is causing the fault. You may have to correct multiple parameters before this fault is cleared and the module is connected. Verify that the configuration is valid by using the module configuration software to validate your configuration. Consult the module documentation for a list of fault codes to determine the configuration parameter that is in error.
#000A	Possible causes include the following. • A connection is being created where the connection type is invalid. • An object attribute or tag value is invalid. If an object attribute or tag is invalid, export the Logix Designer file, then re-import it. Reschedule the ControlNet network after re-importing if applicable.
#000C	The controller is attempting to request a service from the module and has received an error. First, verify that the module is not faulted. For an I/O module, this may indicate that the module has one of the following conditions. • Limited communication is possible, but the module has a Major Fault. • A firmware update needs to be completed or is currently being completed. Refer to the Module Info tab to determine the exact cause.
#000D	An I/O map instance is created where the instance is already in use.
#000E	A MSG instruction is configured to change an attribute that cannot be changed.
#000F	A MSG instruction has been configured to delete a map object that cannot be deleted.
#0010	The state of the device prevents a service request from being handled.
#0011	The reply to a message has a data size that is too large for the destination. Change the destination to a tag that can handle the data size and type being returned.
#0013	The configuration for the module is invalid. Not enough configuration data was sent. Verify that the correct module is being targeted.

Code	Interpretation
#0014	A MSG instruction is configured to change an attribute that does not exist.
#0015	The configuration for the module is invalid. Too much configuration data was sent. Verify that the correct module is being targeted.
#0100	The connection being accessed is already in use. Possible causes include the following. • The controller is attempting to make a specific connection to a module and the module cannot support more than one of these connections. • The target of a connection recognizes that the owner is attempting to remake a connection that is already running.
#0103	Possible causes include the following. • The controller is requesting services not supported by the module. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passed the electronic keying test. This may result when Disable Keying or Compatible Keying options were used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#0106	An ownership conflict occurred for the connection. One of the following conditions exists. • The Connection Request to this module has been rejected due to an Ownership conflict with another Owner (for example, another Controller). This may occur with modules, such as output modules, that allow only a single Owner to configure and control its outputs. This fault may also occur if the module is configured as Listen Only and supports only one connection. • If the Owner is connected to the module with a Unicast connection over EtherNet/IP network, other connections to the module may fail because the Owner controls the one connection. If the Owner is connected to the module with a Multicast connection over EtherNet/IP network, Unicast connections to the module may fail because the Owner controls the one connection. Configure both the Owner and the Listen Only connection as Multicast.
#0107	A connection being accessed was not found.
#0108	An invalid connection type was used. One of the following conditions exists. • The controller is requesting a connection type not supported by the module. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. • The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using. • You have configured a consumed tag or module to use a Unicast connection over EtherNet/IP network, but the producer does not support Unicast connections.

Code	Interpretation
#0109	The connection size is inconsistent with the expected size. Possible causes include the following. • The controller is attempting to set up a connection with the module and cannot because the size of the connection is invalid. • The controller may be attempting to connect to a tag in a producing controller whose size does not match the tag in this controller. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. • The fault may occur even when the module passed the electronic keying test. This may result when Disable Keying or Compatible Keying options were used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using. If the module is a 1756 ControlNet module, verify that the chassis size is correct. For remote I/O adapters, verify that the rack size and rack density is correct.
#0110	The controller is attempting to set up a Listen Only connection with the module and cannot because the module has not been configured and connected to by an Owner (for example, another controller). This controller is not an Owner of this module because it is attempting to establish a Listen Only connection, which requires no module configuration. It cannot connect until an Owner configures and connects to the module first.
#0111	Possible causes include the following. • The Requested Packet Interval (RPI) specified is invalid for this module or for a module in the path to this module. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using. • For Listen Only connections: the RPI set by the owner of this module is slower than the one requested. Either increase the requested RPI or decrease the RPI the owner controller is using. See the Connection tab for valid RPI values.
#0114	The Product Code of the actual module hardware does not match the Product Code of the module created in the software. Electronic Keying failed for this module. You may have a mismatch between the module created in the software and the actual module hardware.
#0115	The Product Type of the actual module hardware does not match the Product Type of the module created in the software. Electronic Keying failed for this module. You may have a mismatch between the module created in the software and the actual module hardware.
#0116	The Major or Minor revisions of the module do not match the Major or Minor revisions of the module created in the software. Verify that you have specified the correct Major and Minor Revision if you have chosen Compatible Module or Exact Match keying Electronic Keying failed for this module. You may have a mismatch between the module created in the software and the actual module hardware.

Code	Interpretation
#0117	The connection is to an invalid port or port that is already in use. One of these conditions exists. • Another controller owns this module and has connected with a Communications Format different than the one chosen by this controller. Verify that the Communications Format chosen is identical to that chosen by the first owner controller of the module. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passed the electronic keying test. This may result when Disable Keying or Compatible Keying options were used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using. • The controller may be attempting to connect to a nonexistent tag in a producing controller.
#0118	An invalid configuration format was used. One of these conditions exists. • The configuration class specified does not match the class supported by the module. • The connection instance is not recognized by the module. • The path specified for the connection is inconsistent. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passed the electronic keying test. This may result when Disable Keying or Compatible Keying options were used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#0119	The controlling connection is not open. Where a Listen Only connection is requested, the controlling connection is not open.
#011A	The controller is attempting to set up a connection with the module and cannot because required resources are unavailable. If the module is a 1756 ControlNet module, up to five controllers can make Rack Optimization connections to the module. Verify that this number has not been exceeded. If the module is a 1794-ACN15, 1794-ACNR15, or 1797-ACNR15 adapter, only one controller can make a Rack Optimization connection to the module. Verify that this number has not been exceeded.
#0203	The owner or originator recognizes that the target device is on the network or backplane. However, I/O data and messages are not being responded to. In other words, the target can be reached, but its response is not as expected. For example, this fault may be indicated where multicast Ethernet packets are not returned. When this fault occurs, the controller usually attempts to continuously remove and remake the connection. If you use FLEX I/O modules, verify that you are using the correct terminal device.
#0204	The controller is attempting to make a connection; however, the target module is not responding. The device also appears to be missing from the backplane or network. To recover, take the following steps. • Verify that the module has not been removed and is still functioning and receiving power. • Verify that the correct slot number has been specified. • Verify that the module is properly connected to the network. If you are using FLEX I/O modules, verify that the correct terminal block is in use.

Code	Interpretation
#0205	Possible causes include the following. • The controller is attempting to set up a connection with the module and has received an error (a parameter is in error). • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#0206	Possible causes include the following. • The controller is attempting to set up a connection with the module and has received an error - the request size is too large. Verify that the path to this module is sufficiently close to the controller. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#0301	One of these conditions may exist. • The controller is attempting to set up a connection with the module and has received an error: a module in the path is out of memory. • The controller may be attempting to connect to a tag in a producing controller that is not marked as being produced. • The controller may be attempting to connect to a tag in a producing controller. That tag may not be configured to allow enough consumers. • The size or number of connections through this module needs to be reduced. • One of the network modules between the module and the controller may be out of memory. Check network configuration of the system. • The module may be out of memory. Check the system configuration and capabilities of the module. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#0302	The controller is attempting to set up a connection with the module and has received an error: a module in the path has exceeded its communication bandwidth capacity. Increase the Requested Packet Interval (RPI) and reconfigure your network with RSNetWorx software. Distribute the load on another bridge module.
#0303	The controller is attempting to set up a connection with the module and has received an error: a module in the path has exceeded its communication bandwidth capacity. Distribute the load on another bridge module.
#0304	The ControlNet module is not scheduled to send data. Use RSNetWorx for ControlNet software to schedule or reschedule the ControlNet network.
#0305	The ControlNet configuration in the controller does not match the configuration in the bridge module. This may occur because a ControlNet module was changed after the network was scheduled, or because a new control program has been loaded into the controller. Use RSNetWorx for ControlNet software to reschedule the connections.

Code	Interpretation
#0306	The ControlNet Configuration Master (CCM) cannot be found. The 1756-CNB module and PLC-5 ControlNet processor are the only devices capable of being a CCM and the CCM must be node 1. Verify that a 1756-CNB modules or PLC-5 ControlNet processor is at node 1 and is functioning properly. This fault may temporarily occur when the system is powered up and is being cleared when the CCM is located.
#0311	The controller is attempting to set up a connection with the module and has received an error. Verify that all modules in the I/O Configuration tree are the correct modules.
#0312	The controller is attempting to set up a connection with the module and has received an error: an invalid link address has been specified. A link address can be a slot number, a network address, or the remote I/O chassis number and starting group. Verify that the chosen slot number for this module is not greater than the size of the rack. Verify that the ControlNet node number is not greater than the maximum node number configured for the network in RSNetWorx for ControlNet software.
#0315	The segment type or route is invalid. Possible causes include the following. • The controller is attempting to set up a connection with the module and has received an error: the connection request is invalid. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#0317	The controller is attempting to set up a ControlNet connection with the module and has received an error. Use RSNetWorx for ControlNet software to schedule or reschedule the connection to this module.
#0318	The controller is attempting to set up a connection with the module and has received an error: the link address is invalid. Verify that the associated ControlNet module has the correct slot or node number selected.
#0319	The controller is attempting to set up a connection with the module and has received an error: the redundant module does not have the necessary resources to support the connection. Reduce the size or number of connections through this module or add another controller or ControlNet module to the system.
#0322	A connection point mismatch has occurred. Possible causes include the following. • A new connection requested does not match the existing connection. Check the controllers that are using the connection and verify that all the configurations are identical. • The connection requested is not a listener or a controlling connection type.
#031E	The controller is attempting to connect to a tag in a producing controller and has received an error. The controller is attempting to connect to a tag in a producing controller and that tag has already been used by too many consumers. Increase the maximum number of consumers on the tag.
#031F	No SC (servicing controller) connection object was found that corresponds to a symbol instance.
#0800	No interpretation available.
#0801	No interpretation available.
#0814	Invalid connection status information was found.
#FD01	I/O map object attributes were found to be invalid. A fault has occurred while clearing memory.
#FD02	No error code is supplied by an I/O module to describe an I/O fault.
#FD03	The controller is attempting to set up a connection with the module and has received an error: this module requires a particular set of connections and connection types, and one of those connection types is missing.
#FD04	The module requires a Coordinated System Time (CST) master in the chassis. Configure a module (typically a controller) in this chassis to be the CST master.

Code	Interpretation
#FD05	The module requires an assigned axis or group table. Assign a Group or Axis.
#FD06	The controller command to transition the SERCOS ring to a new phase returned an error from the module. Check for duplicate Drive Nodes.
#FD07	An attempt to configure the SERCOS ring failed. The baud rate for all devices must be the same and supported by the drives and the SERCOS module.

#FD08	Mainly two sets of faults may cause this fault, physical and interface faults.
	Possible sources of physical faults include the following conditions.
	• Broken ring
	• Loose connector
	• Fiber optics not clean
	• Electrical noise due to improper drive grounding
	• Too many nodes on the ring
	• Interface errors are encountered when you are configuring third party drives. Potential interface errors include the following.
	  • No SERCOS MST (Protocol Error).
	  • Missed AT (drive did not send data when expected).
	  • SERCOS timing error in phase 3.
	  • Error in drive data returned to SERCOS module.

#FD09	An attempt by the controller to configure the node for cyclic operation returned an error.
#FD0A	A bad response was received from a motion module.
#FD1F	An error occurred adding the safety network segment to a route.
#FD20	No safety task appears to be running.
#FD22	Verify the number of physical expansion I/O modules configured for the controller and then update the number of modules selected from the Expansion I/O list on the General page in the Controller Properties dialog.
#FD23	To verify the number of physical expansion I/O the controller supports, open the Controller Properties dialog and expand the Expansion I/O list on the General page. Configure the number of physical expansion I/O modules to match the selection in the Expansion I/O list.
#FE01	An invalid configuration format was encountered.
#FE02	The Requested Packet Interval (RPI) specified is invalid for this module. See the Connection tab for valid RPI values.
#FE03	The input connection point has not been set.
#FE04	The controller is attempting to set up a connection with the module and has received an error.

#FE05	Possible causes include the following.
	• The controller is attempting to set up a connection with the module and has received an error.
	• The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail.
	  The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option.
	  Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted.
	  Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application.
	  For more information about electronic keying, see the user manual for the module you are using.

#FE06	The input force pointer has not been set.
#FE07	The output connection point has not been set.
#FE08	The controller is attempting to set up a connection with the module and has received an error.

Code	Interpretation
#FE09	Possible causes include the following. • The controller is attempting to set up a connection with the module and has received an error. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#FE0A	A connection has been opened without the output force pointer being set.
#FE0B	Possible causes include the following. • The tag to be consumed on this module is invalid. Verify that the tag is marked as being produced. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#FE0C	The controller is attempting to set up a connection with the PLC-5 controller and has received an error. Verify that the instance number specified has been properly specified in the PLC-5 controller.
#FE0D	The symbol instance number was found to not be set.
#FE0E	The module is currently being updated.
#FE0F	Firmware supervisor has attempted to update an unsupported module.
#FE10	The firmware file to update the module cannot be found.
#FE11	The firmware file is corrupted.
#FE12	An error has occurred while updating the module.
#FE13	An active connection could not be made to the target module.
#FE14	The firmware file is currently being read.
#FE22	The target-to-originator netparams connection type is invalid.
#FE23	The target-to-originator netparams connection does not specify whether unicast is allowed.
#FF00	The controller is attempting to set up a connection with the module and has received an error. Verify that the physical module is the same module type (or is a compatible module) as created in the software. If the module is a 1756-DHRIO module in a remote chassis (connected via a ControlNet network), verify that the network has been scheduled with RSNetWorx software. Even after the network has been scheduled with RSNetWorx for ControlNet software, if you are online and if the 1756-DHRIO module is configured for DH+ network only, a #ff00 Module Fault (no connection instance) may occur. The module is properly communicating even though Faulted is displayed as its Status on the Module Properties dialog box. Disregard the error message and fault status and continue.
#FF01	The controller is attempting to set up a connection with the module and has received an error. Verify that the path to this module is a valid length.
#FF04	The controller's map instance attempted to access a connection while in an invalid state.
#FF08	The controller is attempting to set up a connection with the module and has received an error. Verify that the path to this module is a valid length.

Code	Interpretation
#FF0B	Possible causes include the following. • The configuration for the module is invalid. • The module in use (that is, the physical module) is different than the module specified in the I/O configuration tree and is therefore causing the connection or service to fail. The fault may occur even when the module passes the electronic keying test. This may result when Disable Keying or Compatible Keying options are used in the module configuration instead of the Exact Match option. Despite passing the electronic keying test, the module being connected to does not have the same features or settings as the module specified in the I/O configuration tree and does not support the connection or service being attempted. Check the module in use and verify that it exactly matches the module specified in the I/O configuration tree of the Logix Designer application. For more information about electronic keying, see the user manual for the module you are using.
#FF0E	The controller is attempting to set up a connection with the module and has received an error.

S

T

U

W

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-PM014J-EN-P - December 2016

Supersedes Publication 1756-PM014I-EN-P - July 2016