Logix 5000 Controllers Information and Status

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

Rockwell Automation Publication 1756-PM015M-EN-P - March 2022
Supersedes Publication 1756-PM015L-EN-P - November 2020

Rockwell Automation

Programming Manual

Important User Information

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

Activities including installation, adjustments, putting into service, use, assembly, disassembly, and maintenance are required to be carried out by suitably trained personnel in accordance with applicable code of practice.

If this equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

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

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

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

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

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

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

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

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

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

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

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

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

Rockwell Automation recognizes that some of the terms that are currently used in our industry and in this publication are not in alignment with the movement toward inclusive language in technology. We are proactively collaborating with industry peers to find alternatives to such terms and making changes to our products and content. Please excuse the use of such terms in our content while we implement these changes.

This manual includes new and updated information. Use these reference tables to locate changed information.

Grammatical and editorial style changes are not included in this summary.

Global changes

This table contains a list of topics changed in this version, the reason for the change, and a link to the topic that contains the changed information.

Topic Name	Reason
Studio 5000 environment	Updated Studio 5000 Logix Designer branding image.
Component tracking	Updated the supported versions for component tracking to version 30 or later.
Controller Log Events	Added four missing log entries: Vendor Certificate Status, Port state modified, Constant Tag attribute set, and Constant Tag attribute clear.

In this manual

This manual describes how Logix 5000 controllers use connections with other devices. This manual also describes status keywords and how to get controller information, such as memory resources. This manual is one of a set of related manuals that show common procedures for programming and operating Logix 5000 controllers.

For a complete list of common procedures manuals, refer to the Logix 5000 Controllers Common Procedures Programming Manual, publication 1756-PM001.

The term Logix 5000 controller refers to any controller based on the Logix 5000 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 Logix 5000™ controllers for discrete, process, batch, motion, safety, and drive-based solutions.

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

Additional Resources

These documents contain additional information concerning related Rockwell Automation products.

Resource	Description
Industrial Automation Wiring and Grounding Guidelines, publication, 1770-4.1	Provides general guidelines for installing a Rockwell Automation industrial system.
Product Certifications website, http://www.ab.com	Provides declarations of conformity, certificates, and other certification details.

View or download publications at http://www.rockwellautomation.com/literature. To order paper copies of technical documentation, contact a local Rockwell Automation distributor or sales representative.

Rockwell Automation recognizes that some of the terms that are currently used in our industry and in this publication are not in alignment with the movement toward inclusive language in technology. We are proactively collaborating with industry peers to find alternatives to such terms and making changes to our products and content. Please excuse the use of such terms in our content while we implement these changes.

Legal Notices

Rockwell Automation publishes legal notices, such as privacy policies, license agreements, trademark disclosures, and other terms and conditions on the Legal Notices page of the Rockwell Automation website.

End User License Agreement (EULA)

You can view the Rockwell Automation End-User License Agreement ("EULA") by opening the License.rtf file located in your product's install folder on your hard drive.

Open Source Licenses

The software included in this product contains copyrighted software that is licensed under one or more open source licenses. Copies of those licenses are included with the software. Corresponding Source code for open source packages included in this product are located at their respective web site(s).

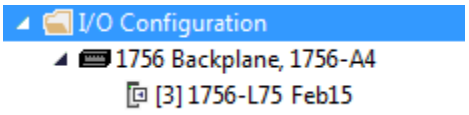
Alternately, obtain complete Corresponding Source code by contacting Rockwell Automation via the Contact form on the Rockwell Automation website:
http://www.rockwellautomation.com/global/about-us/contact/contact.page
Please include "Open Source" as part of the request text.

A full list of all open source software used in this product and their corresponding licenses can be found in the OPENSOURCE folder. The default installed location of these licenses is `C:\Program Files (x86)\Common Files\Rockwell\Help\<Product Name>\Release Notes\OPENSOURCE\index.htm`.

Connections

Introduction

A Logix 5000 controller uses connections for most, but not all, of its communication with other devices.

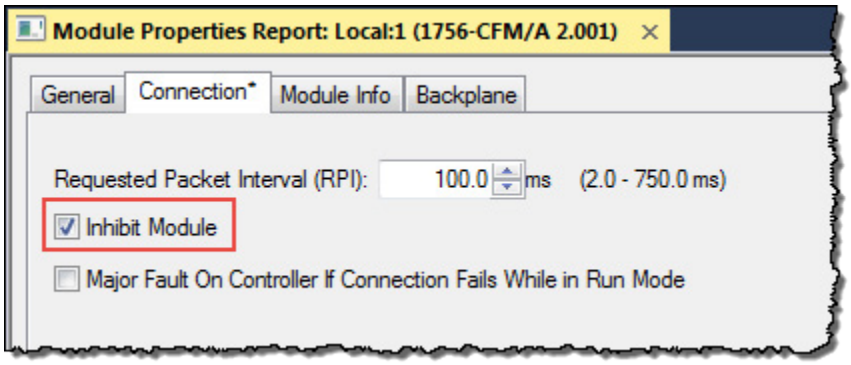
Term	Definition
Connection	A communication link between two devices, such as between a controller and an I/O module, PanelView terminal, or another controller. Connections are allocations of resources that provide more reliable communication between devices than unconnected messages. The number of connections that a single controller can have is limited. Configuring the controller to communicate with other devices in the system indirectly determines the number of connections the controller uses. These communication types use these connections: • I/O modules • Produced and consumed tags • Program parameters • Certain types of Message (MSG) instructions (not all types use a connection)
Requested packet interval (RPI)	The RPI specifies the period at which data updates over a connection. For example, an input module sends data to a controller at the RPI assigned to the module. • Typically, an RPI is configured in milliseconds (ms). The range is 0.2 ms (200 microseconds)...750 ms. • If a ControlNet network connects the devices, the RPI reserves a slot in the stream of data flowing across the ControlNet network. The timing of this slot may not coincide with the exact value of the RPI, but the control system guarantees that the data transfers at least as often as the RPI.
Path	The path describes the route that a connection takes to get to the destination. Typically, the path for a connection is defined when a device is added to the I/O Configuration folder of the controller. 

Inhibit a Connection

In some situations, such as when initially commissioning a system, it is useful to disable portions of a control system and enable them as they are connected to the control system. Inhibiting individual modules or groups of modules prevents the controller from trying to communicate with the modules.

ATTENTION: Inhibiting a module breaks the connection to the module and prevents communication of I/O data.

An installed I/O module defaults to not inhibited. To inhibit a module, change an individual module's properties.



To	Then
Communicate with the module	Do not inhibit the module. Clear the **Inhibit Module** checkbox.
Prevent communication with the module	Inhibit the module. Select the **Inhibit Module** checkbox.

When a communication bridge module is inhibited, the controller shuts down the connections to the bridge module and to all the modules that depend on that bridge module. Inhibit a communication bridge module to disable an entire branch of the I/O network.

When the module is inhibited, the Controller Organizer displays a yellow attention symbol over the module.

If	And you	And	Then
Offline	---------->	-------->	The inhibit status is stored in the project. When the project downloads to the controller, the module still is inhibited.
Online	Inhibit a module while connected to the module	-------->	The connection to the module is closed. The module's outputs go to the last configured Program mode.
	Inhibit a module but a connection to the module was not established (perhaps due to an error condition or fault)	-------->	The module is inhibited. The module status information changes to indicate that the module is inhibited and not faulted.
	Uninhibit a module (clear the checkbox)	No fault occurs	A connection is made to the module and the module is dynamically reconfigured (if the controller is the owner-controller) with the configuration created for that module. If the controller is configured for listen-only, it cannot reconfigure the module.
		Fault occurs	A connection is not made to the module. The module status information changes to indicate the fault condition.

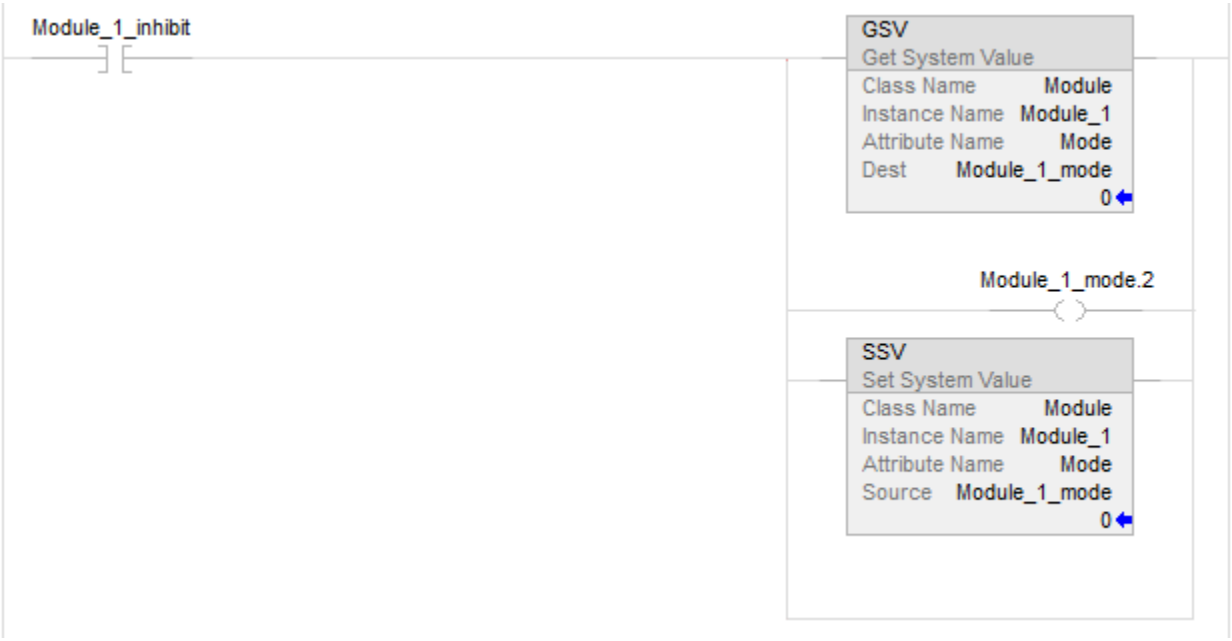
To inhibit or uninhibit a connection

1. Use a Get System Value (GSV) instruction to read the Mode attribute for the module.
2. To inhibit the module, set bit 2. To uninhibit the module, clear bit 2.
3. Use a Set System Value (SSV) instruction to write the Mode attribute back to the module.

Example: Inhibit a connection

If Module_1_Inhibit = 1, then inhibit the operation of the I/O module named Module_1.

- The GSV instruction sets Module_1_Mode = value of the Mode attribute for the module.
- The OTE instruction sets bit 2 of Module_1_Mode = 1. This means inhibit the connection.
- The SSV instruction sets the Mode attribute for the module = Module_1_Mode.



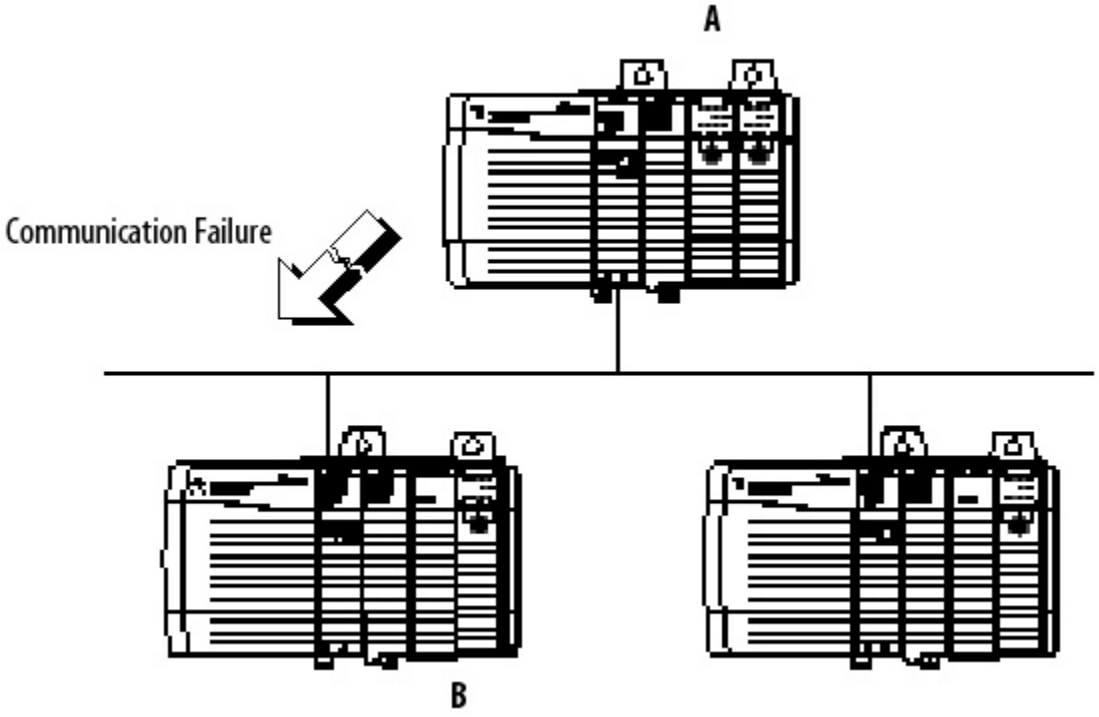
Manage a connection failure

If the controller loses communication with a module, data from that device does not update. When this occurs, the logic acts on the data in ways that may or may not be correct. Program the controller to manage faults safely and efficiently.

ATTENTION: Outputs respond to the last, non-faulted state of the controlling inputs. To avoid potential injury and damage to machinery, make sure this does not create an unsafe operation. Configure critical I/O modules to generate a controller major fault when they lose their connections to the controller, or monitor the status of I/O modules.

Example: Loss of communication

Controller B requires data from controller A. If communication fails between the controllers, controller B continues to act on the last data that it received from controller A.



If communication with a device in the I/O configuration of the controller does not occur for 100 ms, the communication times out. If this occurs, choose between these options.

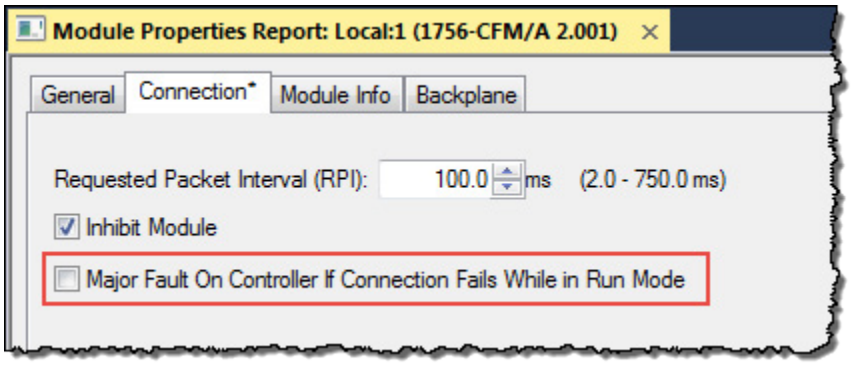
If the controller should	Then
Fault (major fault)	*Configure a major fault to occur*
Continue operating	*Monitor the health of a module*

See also

Monitor the health of a module on page 13

Configure a major fault to occur

To force a module to generate a major fault in the controller if it loses connection with the controller, configure a major fault to occur. This interrupts the execution of logic and runs the Controller Fault Handler. If the Controller Fault Handler does not clear the fault, the controller shuts down.



When selecting **Major Fault On Controller…Run Mode**, the controller:

- Must be connected to the module during the Program transition to Run mode. During the Program to Run mode transition, there can be a 20-second delay.

  During this delay, the controller makes one attempt to connect to a module. If **Major Fault On Controller…Run Mode** is selected and no connection occurs during the 20-second delay, a fault occurs because at least one required connection is not established before going to Run mode. This is a 3/23 type fault code. This fault can occur in large systems with networked I/O.
- Registers a fault if the connection is dropped while in Run mode. A required I/O module connection failed, creating a 3/16 type fault.

For fault codes, see the *Logix 5000 Controllers Major and Minor Faults Programming Manual*, publication no. *1756-PM014*.

See also

Logix 5000 Controllers Major and Minor Faults Programming Manual, publication no. 1756-PM014

Monitor the health of a module

Monitor the module status if the module is not configured to generate a major fault. If a module loses its connection to the controller, outputs go to their configured faulted state. The controller and other I/O modules continue to operate based on old data from the module.

If communication with a module times out, the controller produces these warnings.

- The I/O status indicator on the front of the controller flashes green.
- A Warning (⚠) icon shows over the I/O configuration folder and over the device that has timed out.
- A module fault code is produced, which is accessible through:
  - Module Properties window for the module.
  - GSV instruction.

To monitor the health of connections, use a Get System Value (GSV) instruction to monitor the Module object for either the controller or a specific module.

To	Get this attribute	Data Type	Description

To	Get this attribute	Data Type	Description
Determine if communication has timed out with any device	LEDStatus	INT For efficiency, use a DINT as the destination data type.	Current state of the I/O status indicator on the front of the controller. No need to enter an instance name with this attribute. This attribute applies to the entire collection of modules.

This table describes the meaning of the I/O status indicator on the front of the controller.

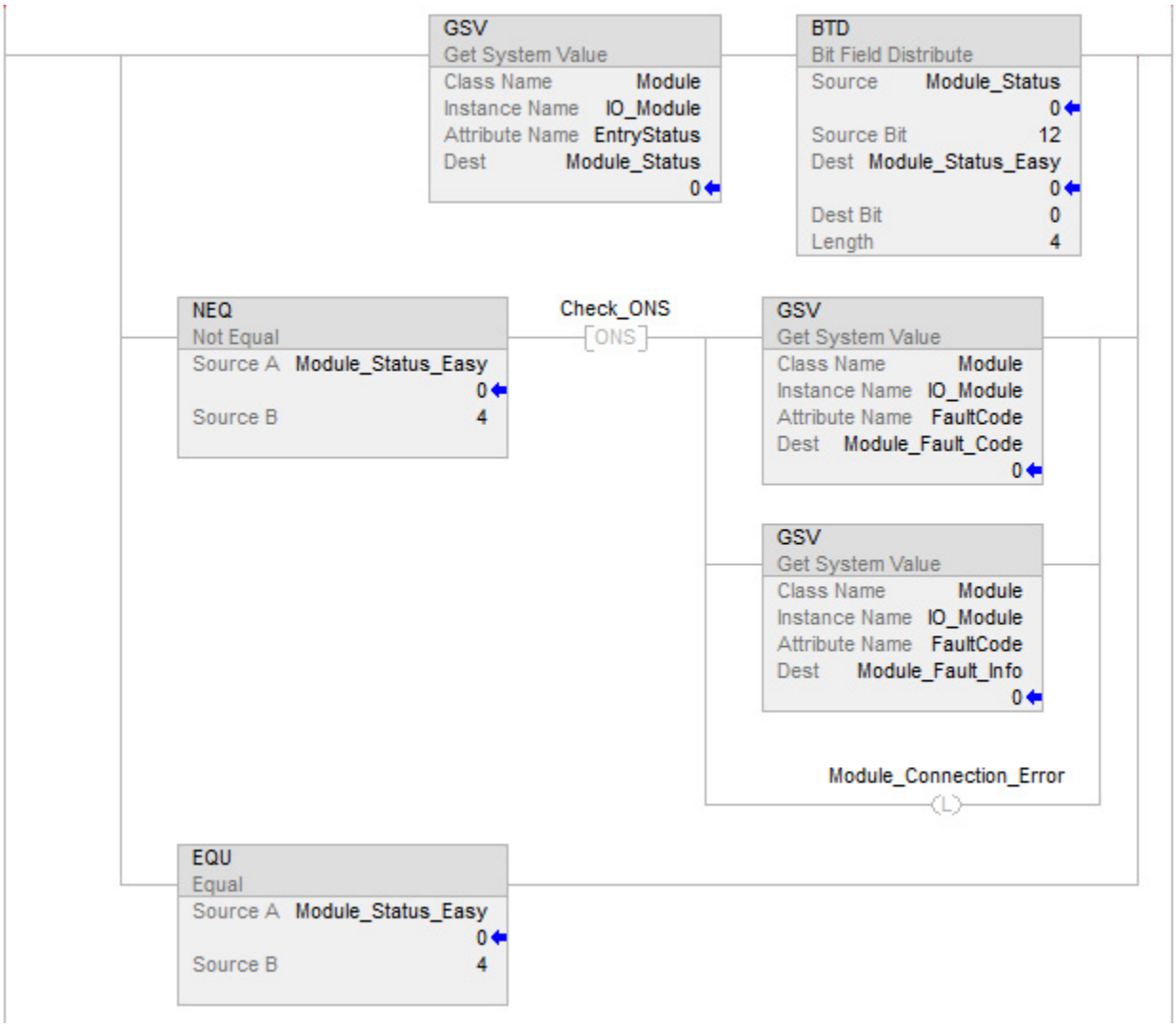
Value	Meaning
0	Status Indicator off. No Module objects are configured for the controller (there are no modules in the I/O Configuration section of the controller organizer).
1	Flashing red. None of the Module objects are Running.
2	Flashing green. At least one Module object is not Running.
3	Solid green. All the Module objects are Running.

To	Get this attribute	Data Type	Description
Determine if communication has timed out with a specific device	FaultCode	INT For efficiency, use a DINT as the destination data type.	A number that identifies a module fault, if one occurs. In the **Instance Name**, choose the device with a connection to monitor. Make sure to assign a name to the device in the I/O Configuration folder of the project.

If Module_Status is any value other than 4, the controller is not communicating with the module. See the example.

Example:

This rung checks the status of an I/O connection. The controller checks the entry status of the connection; any value other than 4 indicates that the connection is not functioning correctly. When the controller detects an error, the error code and information is trapped, and the controller tries to re-establish the connection.


GSV
Get System Value
Class Name Module
Instance Name IO_Module
Attribute Name EntryStatus
Dest Module_Status
0
BTD
Bit Field Distribute
Source Module_Status
0
Source Bit 12
Dest Module_Status_Easy
0
Dest Bit 0
Length 4
NEQ
Not Equal
Source A Module_Status_Easy
0
Source B 4
Check_ONS
ONS
GSV
Get System Value
Class Name Module
Instance Name IO_Module
Attribute Name FaultCode
Dest Module_Fault_Code
0
GSV
Get System Value
Class Name Module
Instance Name IO_Module
Attribute Name FaultCode
Dest Module_Fault_Info
0
Module_Connection_Error
L
EQU
Equal
Source A Module_Status_Easy
0
Source B 4

Determine controller memory information

Introduction

Depending on the type of controller, the memory of the controller is divided into several areas.

This controller	Stores this	In this memory
ControlLogix 5570 GuardLogix 5570S CompactLogix 5370 Compact GuardLogix 5370S ControlLogix 1756-L5x, L6x GuardLogix 1756-L6xS	I/O tags	I/O memory
	Produced / Consumed tags	
	Communication via Message (MSG) instructions	
	Communication with workstations	
	Communication with polled (OPC/DDE) tags that use RSLinx software[1]	
	Tags other than I/O, produced, or consumed tags	Data and logic memory[2]
	Logic routines	
	Communication with polled (OPC/DDE) tags that use RSLinx software[1]	
CompactLogix 1769-L2x, L3x SoftLogix5800 CompactLogix 5380 CompactGuardLogix 5380 CompactLogix 5480 ControlLogix 5580 GuardLogixlLogix 5580	These controllers do not divide their memory. They store all elements in one common memory area.	

(1) To communicate with polled tags, the controller uses both I/O data and logic memory.

(2) 1756-L55M16 controllers have an additional memory section for logic.

View data usage

In the Logix Designer application, the **Capacity** tab (formerly the **Memory** tab) on the **Controller Properties** dialog box shows data usage in the controller. The data displayed on the **Capacity** tab depends on the controller.

Tip: Refer to the *Memory information table* to see whether a controller divides its memory or has one common memory area.

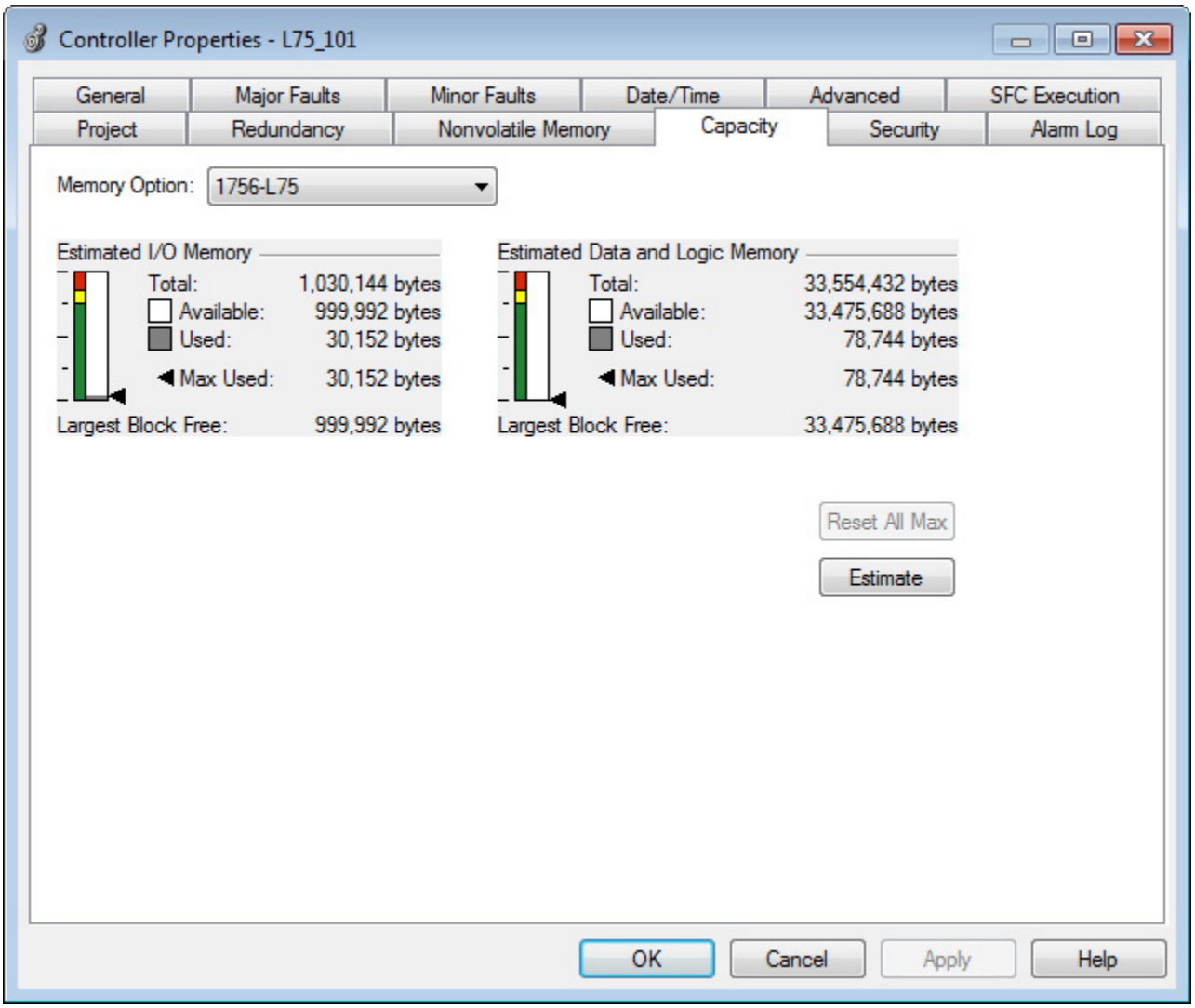
See also

Memory information table on page 17

Capacity tab for controllers that divide memory

The **Capacity** tab for controllers that divide memory shows I/O memory and Data and Logic memory separately. For each area of controller memory, the tab shows the consumed and available memory in bytes.

The tab provides an estimation tool to estimate the amount of controller memory a project requires for I/O, Data, and Logic. It also includes a **Max Used** field for each type of memory to show peak memory usage as communication occurs.



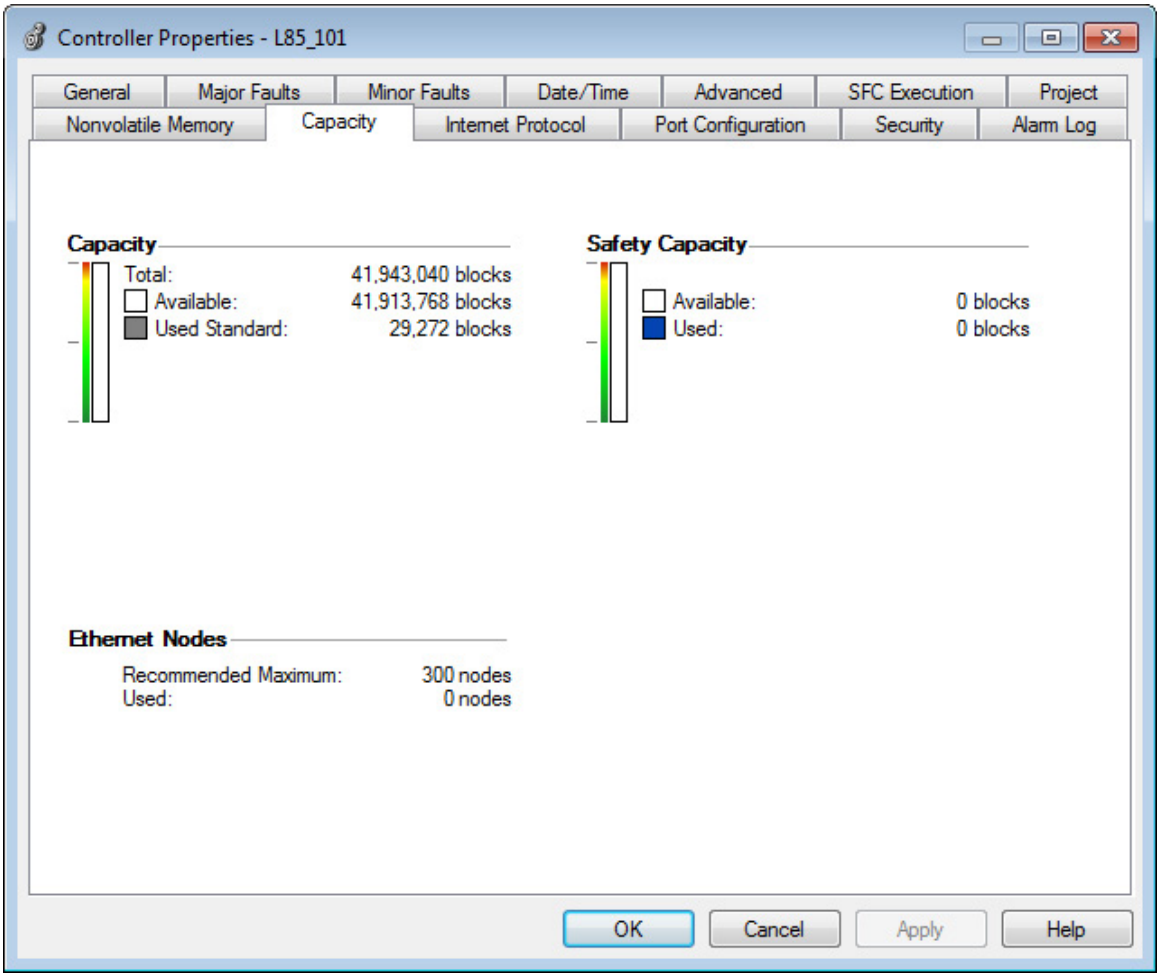
Capacity tab for controllers with common memory area

The **Capacity** tab for controllers that store elements in one common area differs slightly from the **Capacity** tab for other controllers.

Instead of describing consumed memory in bytes, the **Capacity** tab shows logical program blocks which represent units of executable code. The **Capacity** tab shows project size in blocks compared to the specified product capacity in blocks to determine if the project can be downloaded to the controller. If the size of the project exceeds the controller capacity available, a message appears that states the overage size in blocks. When the project size exceeds the memory available, the project does not download to the controller.

Block units are only used for application source code sizes. Data structures and messages are still described in bytes.

IMPORTANT Although block sizes are not physically equivalent to the bytes measured in controllers with divided memory controllers, in general, applications that fit in a divided memory controller should fit in the same controller with one common memory area.



Estimate memory information offline for controllers with divided memory

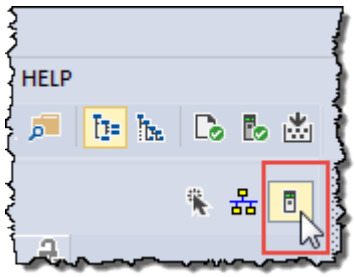
For controllers with divided memory, the applications in the project have a close correlation between the size of the source code and the physical memory in the controller. To estimate how much controller memory a project requires, use the **Estimate** button on the **Capacity** tab. It estimates the number of bytes of:

- Free (unused) memory.
- Used memory.
- Largest free contiguous block of memory.

Tip: This section only applies to controllers with divided memory. Refer to the *Memory information table* for a list of controllers with divided memory. Refer to *Capacity tab controllers with common memory area* for information about the other controllers.

To estimate memory information offline for controllers with divided memory

1. On the **Online** toolbar, select the **Controller Properties** icon.



2. On the **Controller Properties** dialog box, select the **Capacity** tab.
3. In the Estimated Data and Logic Memory area, view the memory information since the last estimate.
4. Select **Estimate** to re-estimate the amount of controller memory.
5. Select **OK**.

See also

Memory information table on page 17

Capacity tab for controllers with common memory area on page 18

View run-time memory information for controllers with divided memory

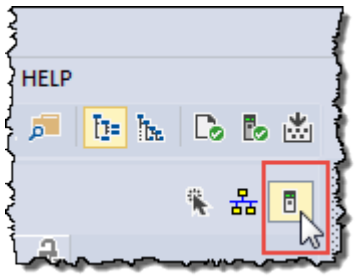
When online with a controller with divided memory, the **Capacity** tab shows the actual memory usage of the controller. While the controller is running, it uses additional memory for communication. The amount of memory the controller needs varies depending on the state of the communication.

The **Capacity** tab of the controller includes a **Max Used** entry for each type of memory. The **Max Used** values show the peak memory usage as communication occurs.

Tip: This section only applies to controllers with divided memory. Refer to the *Memory information table* for a list of controllers with divided memory. Refer to *Capacity tab controllers with common memory area* for information about the other controllers.

To view run-time memory information for controllers with divided memory

1. On the **Online** toolbar, select the **Controller Properties** icon.



2. On the **Controller Properties** dialog box, select the **Capacity** tab.
3. Select **Reset All Max** to reset values.
4. Select **OK**.

See also

Memory information table on page 17

Capacity tab for controllers with common memory area on page 18

View the maximum and used Ethernet nodes for a controller

Some controllers have limits on the number of Ethernet nodes they support in the I/O Configuration tree. When a device is added directly to the Ethernet I/O configuration, it is counted toward the node limitation of the controller.

The **Capacity** tab of the **Controller Properties** dialog box shows the current number of used Ethernet nodes in the I/O Configuration tree and the maximum number of Ethernet nodes supported by the controller.

When the controller is online, the controller enforces the node count, and the node count exceeds the limit:

- An error dialog box appears.
- The I/O module cannot be added to the I/O Configuration tree.

When the controller is offline, the controller enforces the node count, and the node count exceeds the limit when the project is verified or the controller attempts to go online:

- An error message is generated on the **Error** tab of the **Output window**.
- The project cannot be downloaded to the controller.

Tip: The 1756-L85E controllers do not enforce the node count, meaning the system does not prevent connecting additional nodes, and it may be possible to download the project. The system provides a warning when the recommended node limit has been exceeded.

To view the maximum and used Ethernet nodes for a controller:

1. On the **Online** toolbar, select the **Controller Properties** icon.

2. On the **Controller Properties** dialog box, select the **Capacity** tab.
3. In the **Ethernet Node** group, view the information about the Ethernet nodes.
   - **Maximum/Recommended Maximum** - Number of Ethernet nodes supported by the controller, or the recommended maximum number of Ethernet nodes for the controller.
   - **Used -** Current number of Ethernet nodes in the I/O Configuration tree

     If the controller is offline, appears next to the **Used** parameter when the node count exceeds the limit.
     For controllers that do not limit the number of Ethernet nodes, appears next to the **Used** parameter when the node count exceeds the recommended limit and this message appears:
     The recommended node limit has been exceeded. It may be possible to connect additional nodes depending on the node type and system architecture.

Write logic to get memory information

Use a Message (MSG) instruction to get memory information from the controller.

Tip: This procedure applies to controllers with divided memory. The Logix Designer application determines capacity information for controllers that store elements in one common area.

Get memory information from the controller

To get memory information from the controller, execute a Message (MSG) instruction. This table lists configuration information for the instruction.

On this tab	For this item	Type or select	Which means	
Configuration	Message Type	CIP Generic	Execute a Control and Information Protocol command.	
	Service Type	Custom	Create a CIP Generic message that is not available in the list.	
	Service Code	3	Read specific information about the controller (GetAttributeList service).	
	Class	72	Get information from the user memory object.	
	Instance	1	This object contains one instance.	
	Attribute	0	Null value.	
	Source Element	*source_array* of type SINT[12]		
		In this element	**Enter**	**Which means**
		source_array[0]	5	Get 5 attributes.
		source_array[1]	0	Null value.
		source_array[2]	1	Get free memory.
		source_array[3]	0	Null value.
		source_array[4]	2	Get total memory.
		source_array[5]	0	Null value.
		source_array[6]	5	Get largest contiguous block of additional free logic memory.
		source_array[7]	0	Null value.
		source_array[8]	6	Get largest contiguous block of free I/O memory.
		source_array[9]	0	Null value.
		source_array[10]	7	Get largest contiguous block of free data and logic memory.
		source_array[11]	0	Null value.

On this tab	For this item	Type or select	Which means
	Source Length	12	Write 12 bytes (12 SINTs).
	Destination	*INT_array* of type INT[29]	
Communication	Path	1, *slot_number_of_controller*	

Choose the memory information

This table lists the information that the MSG instruction returns to INT_array (destination tag of the MSG).

Tip: This procedure applies to controllers with divided memory. The Logix Designer application determines capacity information for controllers that store elements in one common area.

IMPORTANT The controller returns the values in 32-bit words. To see a value in bytes, multiply it by 4. If a controller does not divide its memory, the values show up as I/O memory. For the 1756-L55M16 controller, the MSG instruction returns two values for each logic memory category. To determine the free or total logic memory of a 1756-L55M16 controller, add both values for the category.

To find this information	Then copy these array elements	Description
Amount of free I/O memory (32-bit words)	*INT_array*[3]	Lower 16 bits of the 32 bit value
	INT_array[4]	Upper 16 bits of the 32 bit value
Amount of free data and logic memory (32-bit words)	*INT_array*[5]	Lower 16 bits of the 32 bit value
	INT_array[6]	Upper 16 bits of the 32 bit value
1756-L55M16 controllers only—amount of additional free logic memory (32-bit words)	*INT_array*[7]	Lower 16 bits of the 32 bit value
	INT_array[8]	Upper 16 bits of the 32 bit value
Total size of I/O memory (32-bit words)	*INT_array*[11]	Lower 16 bits of the 32 bit value
	INT_array[12]	Upper 16 bits of the 32 bit value
Total size of data and logic memory (32-bit words)	*INT_array*[13]	Lower 16 bits of the 32 bit value
	INT_array[14]	Upper 16 bits of the 32 bit value
1756-L55M16 controllers only—additional logic memory (32-bit words)	*INT_array*[15]	Lower 16 bits of the 32 bit value
	INT_array[16]	Upper 16 bits of the 32 bit value
1756-L55M16 controllers only—largest contiguous block of additional free logic memory (32-bit words)	*INT_array*[19]	Lower 16 bits of the 32 bit value
	INT_array[20]	Upper 16 bits of the 32 bit value
Largest contiguous block of free I/O memory (32-bit words)	*INT_array*[23]	Lower 16 bits of the 32 bit value
	INT_array[24]	Upper 16 bits of the 32 bit value
Largest contiguous block of free data and logic memory (32-bit words)	*INT_array*[27]	Lower 16 bits of the 32 bit value
	INT_array[28]	Upper 16 bits of the 32 bit value

Convert INTs to a DINT

The MSG instruction returns each memory value as two separate INTs.

- The first INT represents the lower 16 bits of the value.
- The second INT represents the upper 16 bits of the value.

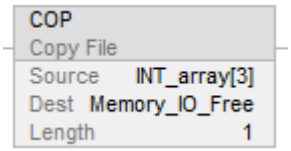
To convert the separate INTs into one usable value, use a Copy (COP) instruction.

In this operand	Specify	Which means
Source	First INT of the 2 element pair (lower 16 bits)	Start with the lower 16 bits.
Destination	DINT tag in which to store the 32-bit value	Copy the value to the DINT tag.
Length	1	Copy 1 times the number of bytes in the Destination data type. In this case, the instruction copies 4 bytes (32 bits), which combines the lower and upper 16 bits into one 32-bit value.

In this example, the COP instruction produces the 32-bit value that represents the amount of free I/O memory, in 32-bit words.

Example: Convert INTs to a DINT.

- Elements 3 of INT_array is the lower 16 bits of the amount of free I/O memory. Element 4 is the upper 16 bits.
- Memory_IO_Free is a DINT tag (32 bits) in which to store the value for the amount of free I/O memory.
- To copy all 32 bits, specify a Length of 1. This tells the instruction to copy 1 times the size of the Destination (32 bits). This copies both element 3 (16 bits) and element 4 (16 bits) and places the 32-bit result in Memory_IO_Free.



Controller logging

Introduction

Use the controller logging feature to detect and log changes made to Logix5000 controllers without adding any auditing software. With controller logging, the controllers:

- Detect changes and create logs entries containing information about the changes.
- Store the log entries to removable media for later review.
- Provide programmatic access to log entry counters to provide change detection information remotely.

Note these logging considerations:

- The 1769-L3x and 1769-L4x CompactLogix Controllers do not support storing log entries to removable media, and the audit value is not populated.
- The Audit Value is not supported in versions 19 and earlier.

Controller log

A controller log is a record of interactions that occurred in the controller due to physical conditions such as keyswitches, changes to removable media, fault conditions, and programming changes configured in the application.

- For the Compact GuardLogix 5370, CompactLogix 5370, ControlLogix 5570, and GuardLogix 5570 controllers, up to 100 log entries are buffered in the controller memory. The controller can save these buffered entries to removable media using a message instruction. Additionally, the controller can be configured to write buffered entries to the removable media after accumulating 80 entries.
- For the Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, and GuardLogix 5580 controllers, up to 500 log entries are buffered in the controller memory. The controller can be configured to write buffered entries to the removable media after accumulating 400 entries.

In the example table below, record numbers 4 through 6 are related to Change Detection.

Record Number	Time	Entry Description	User Name	Workstation Name	FactoryTalk ID	Extended Information	Change Detection Audit Value
1	12-Feb 03:39:34	Project download	John Doe	Laptop	FT\JDoe	Project L71	16#FD60_CB89_029F_3500
2	12-Feb 04:05:12	Forces Enabled	Jones	USMAYLT	FT\Jones		
3	12-Feb 04:22:03	Online edits modified controller program	John Doe	Laptop	FT\JDoe		

Record Number	Time	Entry Description	User Name	Workstation Name	FactoryTalk ID	Extended Information	Change Detection Audit Value
4	12-Feb 04:42:12	Change Log entry added			FT\JDoe		16#FD60_CB89_029F_3521
5	12-Feb 04:50:43	Change detection mask modified		None	None	Old mask 16#FFFF_FFFF_FFFF_FFFF, New mask 16#FFFF_FFFF_FFFC_FFFF	16#FD60_CB89_029F_3566
6	12-Feb 04:58:29	Change Log entry added		None	None		16#FD60_CB89_029F_35BF

Controller log header

When the controller creates a log file on the removable media, it includes some header information. The header information includes these items.

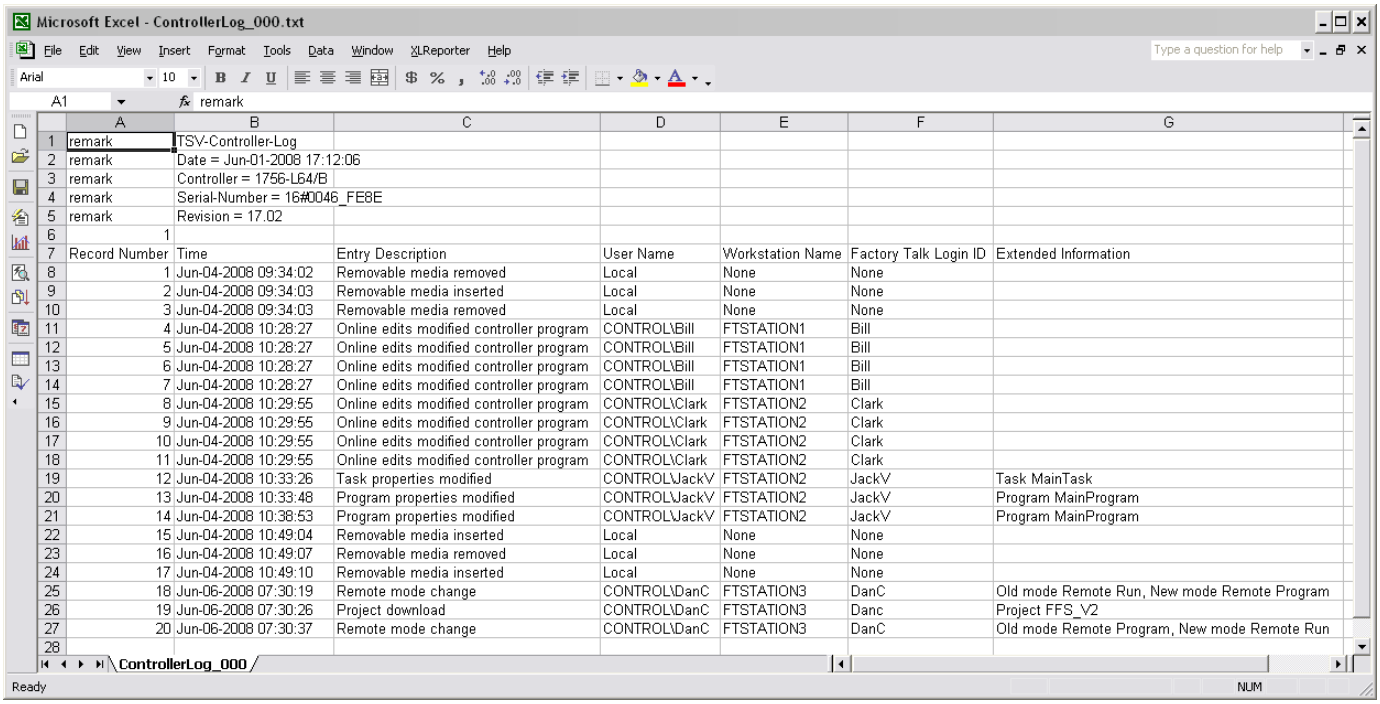
- Date the log file was created.
- Controller model number.
- Controller serial number.
- Version of firmware running on the controller.

Controller log entry

Each entry in the log can include this information.

- Record Number
- Time of Occurrence (UTC - 24 hour clock)
- Entry Description
- Windows User Name
- Workstation Name
- FactoryTalk User ID (if available)
- Extended Information
- Change Detection Audit Value

This example shows controller log entries in a spreadsheet. Controller log files are stored in plain text files in TSV format, so no special tools are required to read them. Open them in any text editor, or in a spreadsheet application such as Microsoft Excel.



Microsoft Excel - ControllerLog_000.txt

File Edit View Insert Format Tools Data Window XLReporter Help

Arial 10

A1 remark

	A	B	C	D	E	F	G
1	remark	TSV-Controller-Log					
2	remark	Date = Jun-01-2008 17:12:06					
3	remark	Controller = 1756-L64/B					
4	remark	Serial-Number = 16#0046_FE8E					
5	remark	Revision = 17.02					
6	1						
7	Record Number	Time	Entry Description	User Name	Workstation Name	Factory Talk Login ID	Extended Information
8	1	Jun-04-2008 09:34:02	Removable media removed	Local	None	None	
9	2	Jun-04-2008 09:34:03	Removable media inserted	Local	None	None	
10	3	Jun-04-2008 09:34:03	Removable media removed	Local	None	None	
11	4	Jun-04-2008 10:28:27	Online edits modified controller program	CONTROL\Bill	FTSTATION1	Bill	
12	5	Jun-04-2008 10:28:27	Online edits modified controller program	CONTROL\Bill	FTSTATION1	Bill	
13	6	Jun-04-2008 10:28:27	Online edits modified controller program	CONTROL\Bill	FTSTATION1	Bill	
14	7	Jun-04-2008 10:28:27	Online edits modified controller program	CONTROL\Bill	FTSTATION1	Bill	
15	8	Jun-04-2008 10:29:55	Online edits modified controller program	CONTROL\Clark	FTSTATION2	Clark	
16	9	Jun-04-2008 10:29:55	Online edits modified controller program	CONTROL\Clark	FTSTATION2	Clark	
17	10	Jun-04-2008 10:29:55	Online edits modified controller program	CONTROL\Clark	FTSTATION2	Clark	
18	11	Jun-04-2008 10:29:55	Online edits modified controller program	CONTROL\Clark	FTSTATION2	Clark	
19	12	Jun-04-2008 10:33:26	Task properties modified	CONTROL\JackV	FTSTATION2	JackV	Task MainTask
20	13	Jun-04-2008 10:33:48	Program properties modified	CONTROL\JackV	FTSTATION2	JackV	Program MainProgram
21	14	Jun-04-2008 10:38:53	Program properties modified	CONTROL\JackV	FTSTATION2	JackV	Program MainProgram
22	15	Jun-04-2008 10:49:04	Removable media inserted	Local	None	None	
23	16	Jun-04-2008 10:49:07	Removable media removed	Local	None	None	
24	17	Jun-04-2008 10:49:10	Removable media inserted	Local	None	None	
25	18	Jun-06-2008 07:30:19	Remote mode change	CONTROL\DanC	FTSTATION3	DanC	Old mode Remote Run, New mode Remote Program
26	19	Jun-06-2008 07:30:26	Project download	CONTROL\DanC	FTSTATION3	Danc	Project FFS_V2
27	20	Jun-06-2008 07:30:37	Remote mode change	CONTROL\DanC	FTSTATION3	DanC	Old mode Remote Program, New mode Remote Run
28							

ControllerLog_000

Ready NUM

Entries captured in the controller log

These entries are detected and logged by the controller. These events are described in more detail in *Controller Log Events*.

Tip: In Logix Designer versions 30 and later, audit log entries also appear in the controller log. Audit log entries describe events such as changes to component properties, edits in a routine, and changes to tag values. The audit log is used for auditing by the FactoryTalk® AssetCentre.

- Project downloaded
- Project loaded from removable media
- Project stored to removable media
- Online edits modified controller program
- Partial import online completed or Transaction committed
- I/O forces enabled, disabled, removed, or modified
- SFC forces enabled, disabled, removed, or modified
- Firmware update
- Constant tag data changed
- Multiple constant tag data changed
- Change to constant tag configuration reset
- Mode change
- Major fault, major fault cleared
- Program properties modified
- Task properties modified
- Controller timeslice modified
- Removable media inserted or removed
- Safety signature created or deleted
- Safety locked or unlocked
- Custom entry: User-defined logic to create a log entry, with user-defined entry description and extended information
- Safety signature delete inhibited in Run mode

- Safety signature delete allowed in Run mode
- The Changes To Detect value has changed
- Log Collected Data Cleared
- Program Connection Modified
- USB cable connected and disconnected

See also

Controller log events on page 38

Controller log buffer

For the Compact GuardLogix 5370, CompactLogix 5370, ControlLogix 5570, and GuardLogix 5570 controllers, the controller keeps up to 100 log entries buffered in its internal memory.

For the Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, and GuardLogix 5580 controllers, the controller keeps up to 500 log entries buffered in its internal memory.

Configure the controller to write the buffered entries to the removable media when its internal buffer is 80% full, or command the controller to write the buffered entries to the removable media with a message instruction. This procedure is detailed below. Once a log entry is written to the removable media, it is removed from the buffer.

If the removable media is not present, is full, or if the controller is not configured to automatically write buffered entries to the removable media, and the internal buffer becomes full, entries are saved in the buffer in a circular fashion. As new entries are stored, the oldest entries are discarded.

Controller log files and the removable media

When written to the removable media, controller logs are stored in plain text files in the Tab Separated Value (TSV) format. Each time the controller writes entries to the CompactFlash card, the entries are appended to the text file until the file reaches 1 MB in size. At this point, the controller creates a new text file.

The controller does not attempt to write log entries to a full removable media card. If the removable media becomes full, the system behaves as if the removable media is not present.

Controller log files are stored in plain text files in the TSV format, so no special tools are required to read them. Open them in any text editor, or in a spreadsheet application such as Microsoft Excel.

Writing the controller log to the SD card

The controller log can be written to the Secure Digital (SD) card either automatically or on demand.

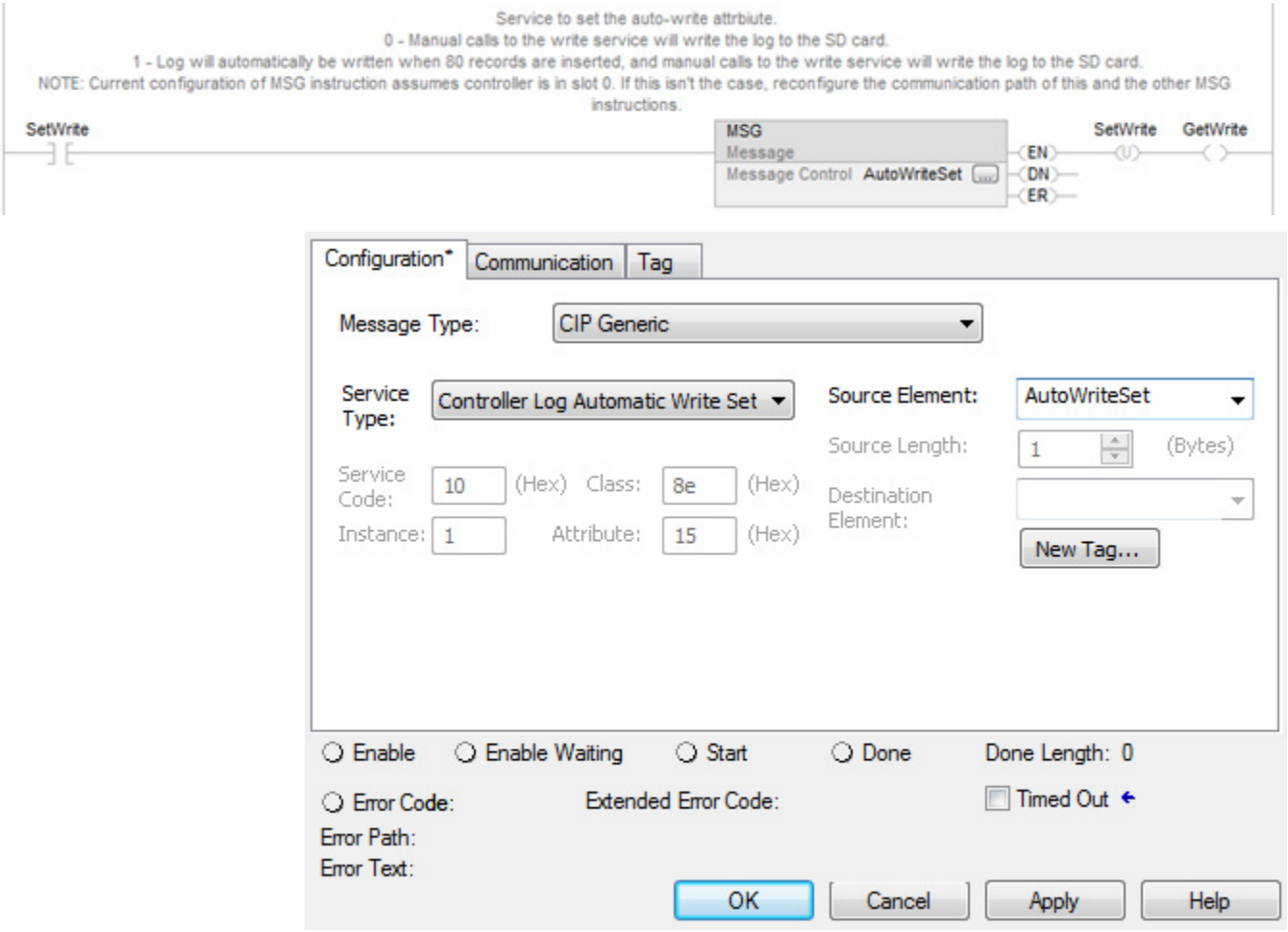
Tip: Some Logix 5000 controllers support additional types of removable media that can be used to write the controller log entries. Refer to the Logix 5000 controller documentation for information regarding the type of removable media the controller supports.

Automatic save

Configure the controller to automatically write buffered log entries to the SD card when the controller's internal log entry buffer becomes 80% full. Alternately, configure the controller to write the buffered entries before a firmware update. To write buffered entries before a firmware update, send a message instruction to the controller using a CIP Generic message type and a service type of Controller Log Automatic Write Set. Sending a value of 0 turns off automatic writes, and sending a value of 1 turns on automatic writes. By default, entries are not automatically written.

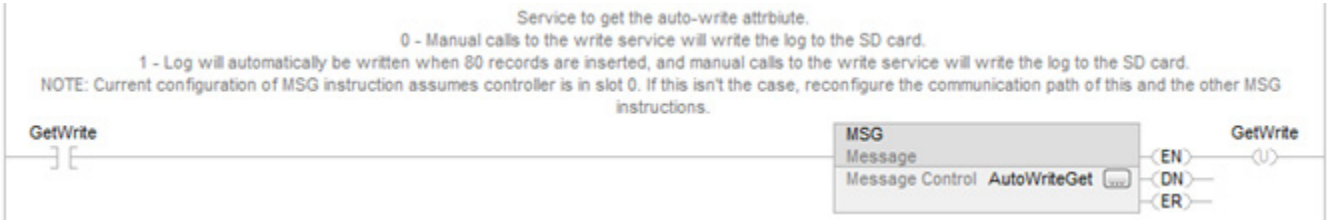
A rung of logic that performs this configuration and the configuration dialog box of the message instruction are shown in these examples.

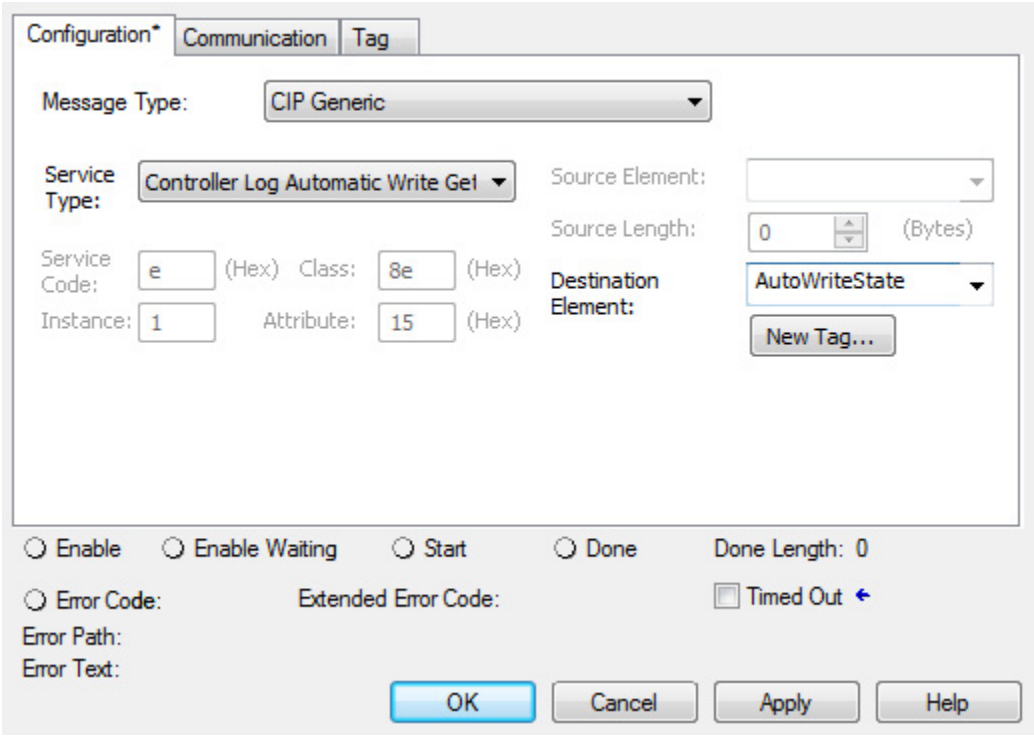
Example: Automatic Save Set ladder instruction and configuration dialog box



Retrieve the current state of the automatic write setting by using a CIP Generic message type and a service type of Controller Log Automatic Write Get. Use this rung of ladder logic and message instruction configuration to get this value.

Example: Automatic Save Get ladder instruction and configuration dialog box



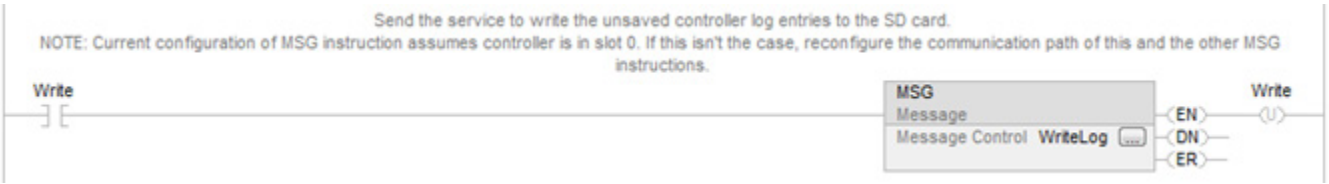


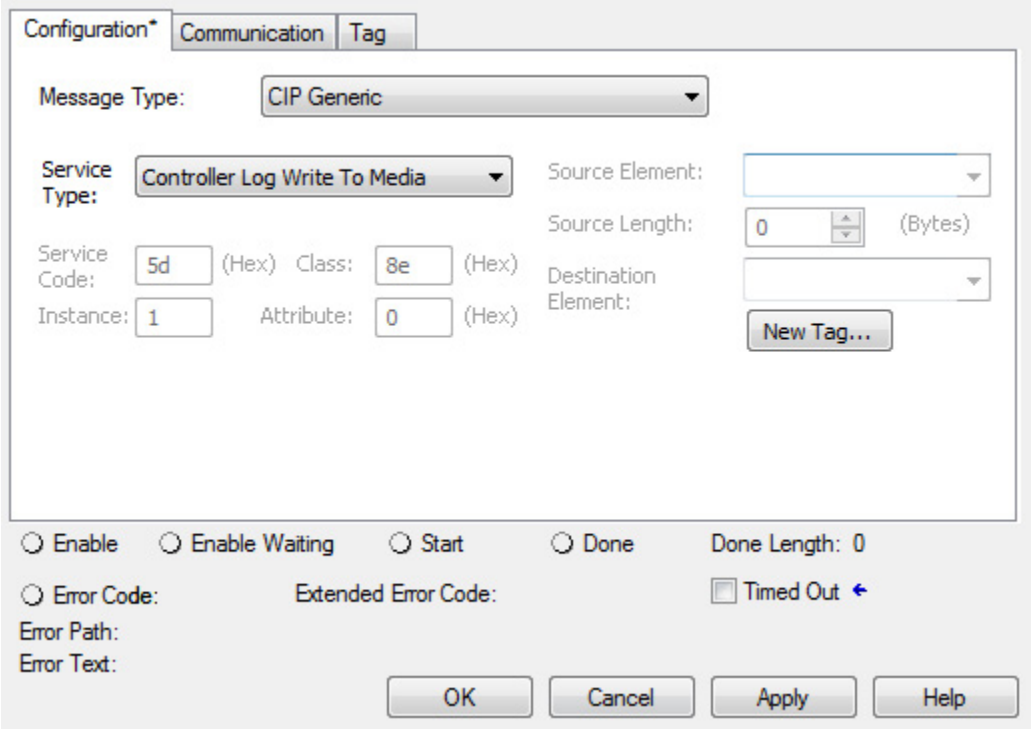
Save on demand

Command the controller to write buffered entries to the SD card by using a message instruction with a CIP Generic message type and a service type of Controller Log Write To Media.

A rung of ladder logic that sends this message and the configuration of the message instruction are shown in these examples.

Example: Write Buffered Entries to the SD Card ladder instruction and configuration dialog box





Controller logging counters

Three counters, listed in the table, provide real-time statistics about modifications to the controller.

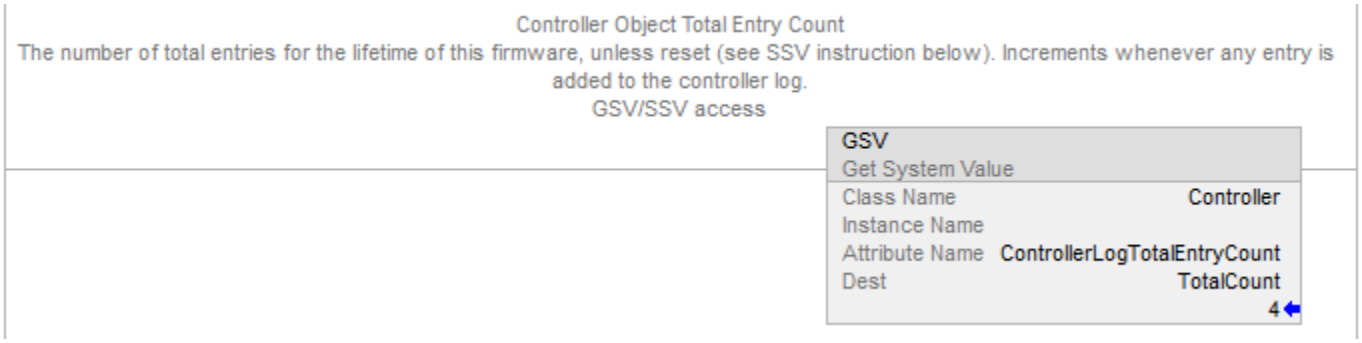
Counter Name	Description	Access
Total Entry Count	Number of entries added to the log since the last firmware update.	GSV/SSV
Unsaved Entry Count	Number of entries in controller RAM not yet written to the CompactFlash card.	GSV
Modify Execution Count	Count that specifically tracks modifications that can change behavior of a running controller. A subset of entries increment this count.	GSV/SSV

Total entry count

Total Entry Count is the number of controller entries that have been added since the last firmware update. This counter increments after any entry is added to the log, and it is written to the log in the Record Number field. Use a Set System Value (SSV) instruction to set it to a known value. This can be useful, for example, for monitoring system changes during a production run.

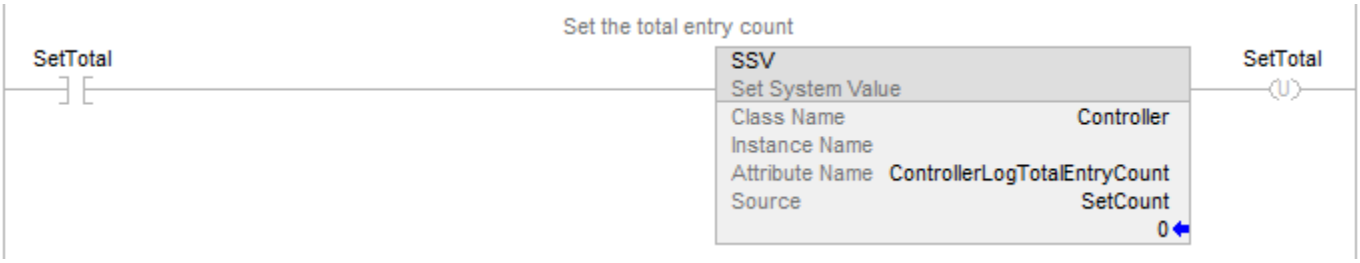
This rung of ladder logic shows how to retrieve the Total Entry Count by using a Get System Value (GSV) instruction.

Example: Total Entry Count by using a Get System Value (GSV) ladder instruction



This rung of ladder logic shows how to set the Total Entry Count to a known value (in this example, 0) by using an SSV instruction.

Example: Set the Total Entry Count to a Known Value ladder instruction



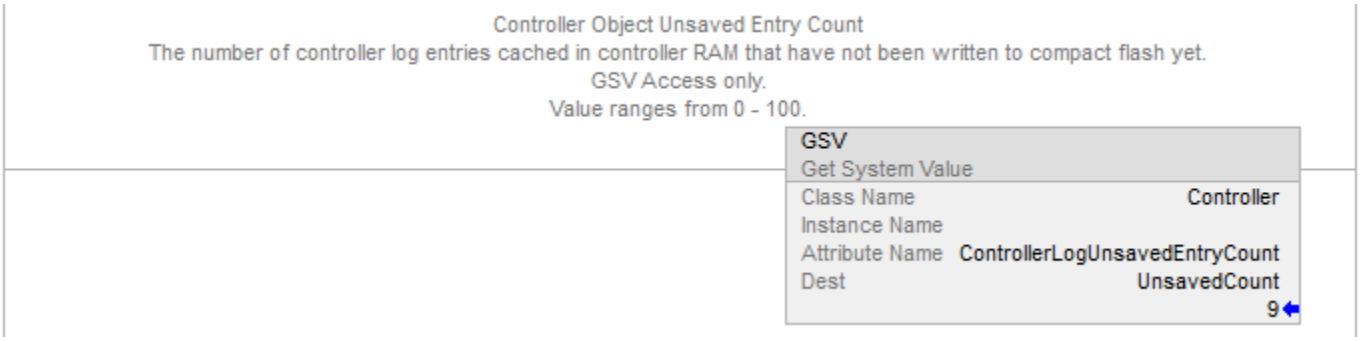
Unsaved entry count

The Unsaved Entry Count is the number of log entries that are in controller memory but have not yet been stored to the CompactFlash card.

This counter value is available via a GSV instruction, and can range from 0...100, the maximum number of entries that the controller can buffer.

This rung of ladder logic shows how to retrieve the Unsaved Entry Count by using a Get System Value (GSV) instruction.

Example: Retrieve the Unsaved Entry Count using a Get System Value (GSV) ladder instruction



Execution modification count

The Execution Modification Count tracks the number of changes that occur that can change the behavior of a running controller. Configure the counter to include or exclude force changes.

These events cause the Execution Modification Count to increment:

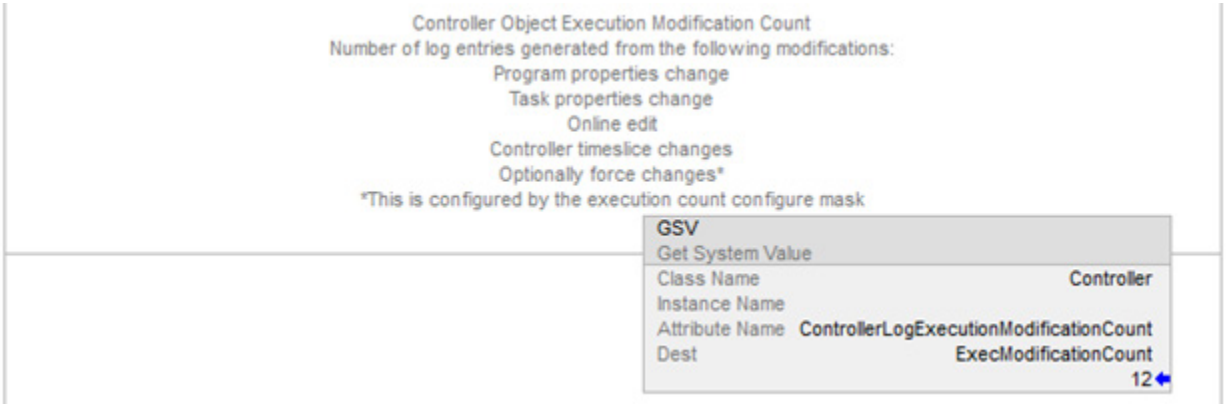
- Online edits tested or assembled
- Forces enabled or disabled (if so configured)
- Program properties modified

- Task properties modified
- Controller timeslice modified

Set this counter to a known value using an SSV instruction.

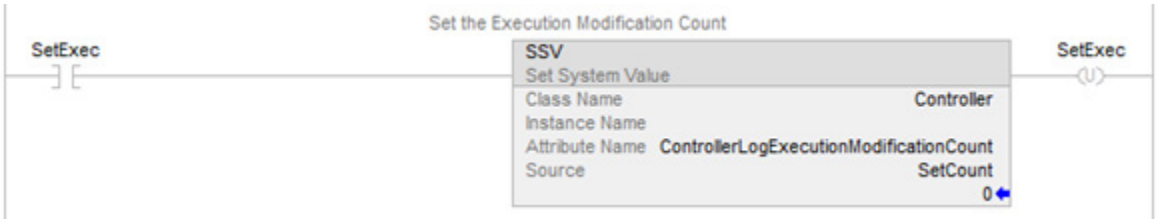
This example rung of ladder logic shows how to retrieve the Execution Modification Count by using a GSV instruction.

Example: Retrieve the Execution Modification Count by using a GSV ladder instruction



This rung of ladder logic shows how to set the Execution Modification Count to a known value.

Example: Ladder instruction to set the Execution Modification Count to a known value

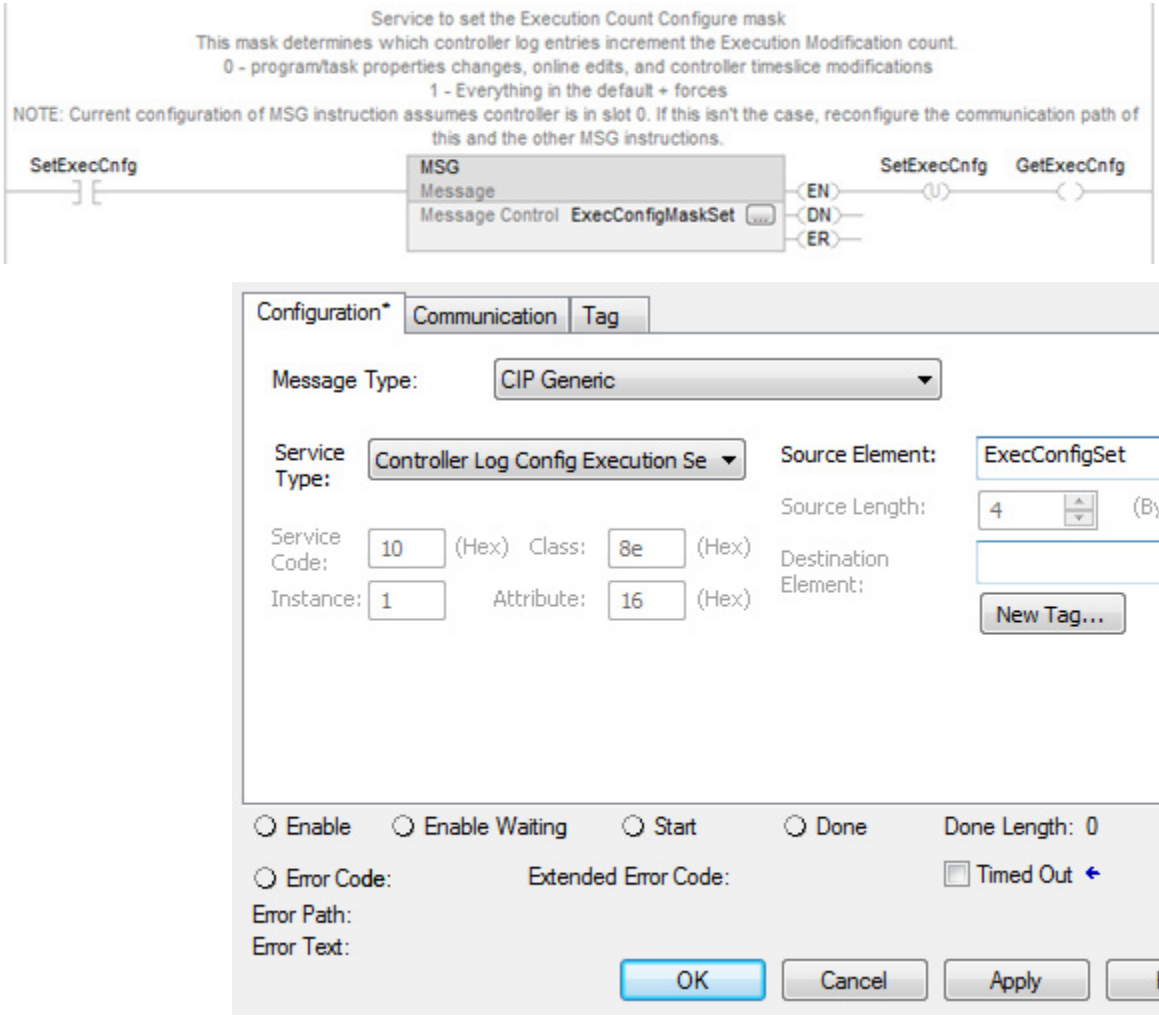


To configure whether the Execution Modification Count includes forces, use a message instruction of message type CIP Generic and a service type of Controller Log Config Execution Set.

If it is sent a value of 1, forces are included in the counter. If it is sent a value of 0, forces are not included.

This rung of ladder logic shows how to send the message instruction, followed by the configuration dialog box of the message instruction.

Example: Controller Log Config Execution Set ladder instruction and configuration dialog box

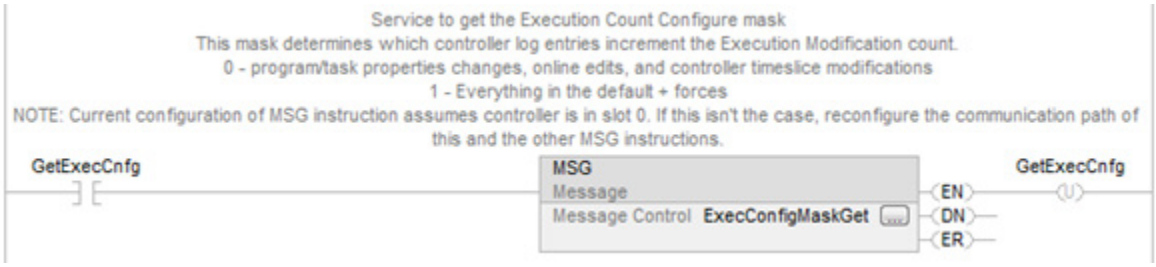


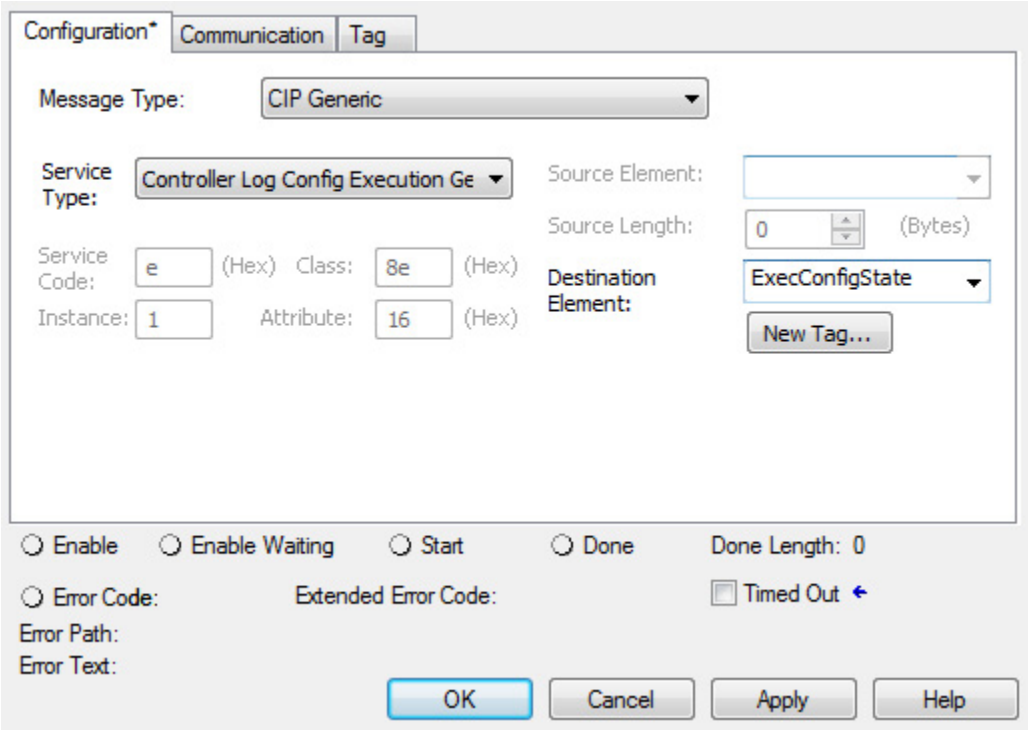
The Source Element should be of data type DINT.

Use a message instruction to retrieve the current value of this configuration. This message uses a message type of CIP Generic and a service type of Controller Log Config Execution Get.

This rung of ladder logic shows how to send the message instruction, followed by the configuration dialog box of the message instruction.

Example: Controller Log Config Execution Get ladder instruction and configuration dialog box



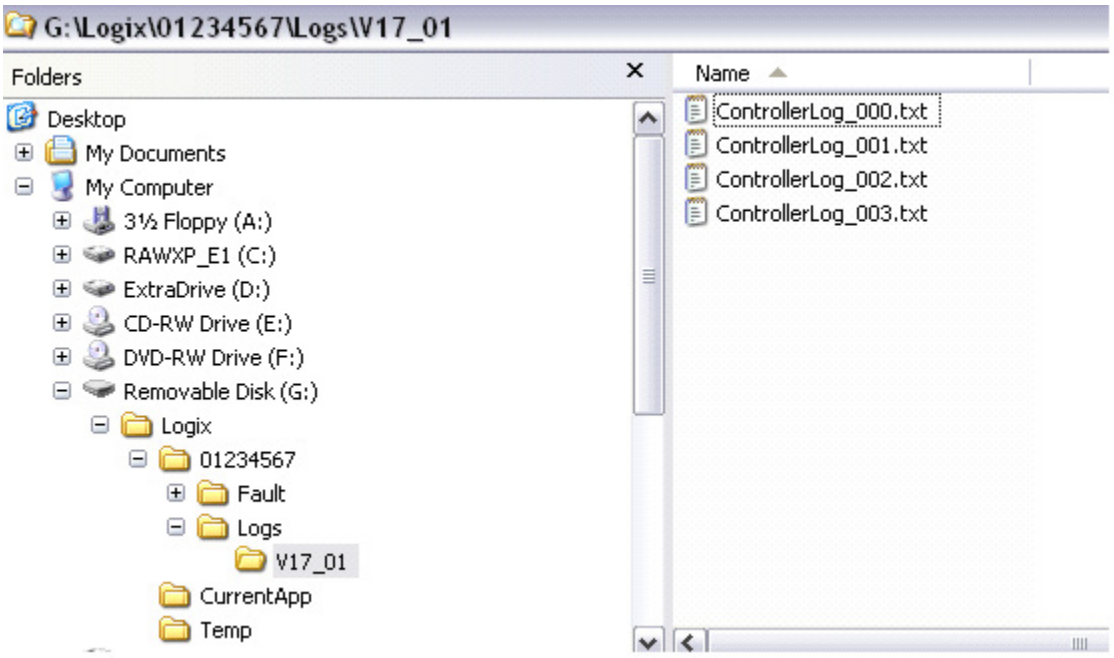


The Destination tag should be of type DINT.

Log file storage

When a log file is written to the CompactFlash card, it is stored at \Logix\XXXXXXXX\Logs\VYY_ZZ, where XXXXXXXX is the eight-digit serial number of the controller and YY_ZZ is the revision number of the firmware (major_minor revision).

Log file location



The file is named ControllerLog_yyy.txt, where yyy is a sequential number from 000...999. The controller adds to the log file until it reaches a size greater

than 1 MB. At that point, the next write of the controller log causes a new file to be created with the next sequence number.

When there are 1000 files larger than 1 MB, no more logs are created. However, the controller searches for the file name with the smallest possible sequence number that it can create or write to. For example, if a user deletes files 001...100 but leaves the rest, the controller starts creating logs again starting at a sequence number of 001. If there are already 1000 log files and a user deletes log entries out of file 005, the controller writes the next log entries to that file. The controller starts at 000 and looks for the first file that does not exist or is less than 1 MB in size.

Each time the controller opens a log file for writing, it creates a back-up file that is a copy of the log file before the write. This file is called Backup.txt. The controller overwrites the backup every time it opens a log file is opened for writing.

Rockwell Automation recommends periodically clearing space on the card for new log files. The controller does not delete any files off the card to create more space for new log files.

Log file format

This table lists the information that is contained in the controller log file.

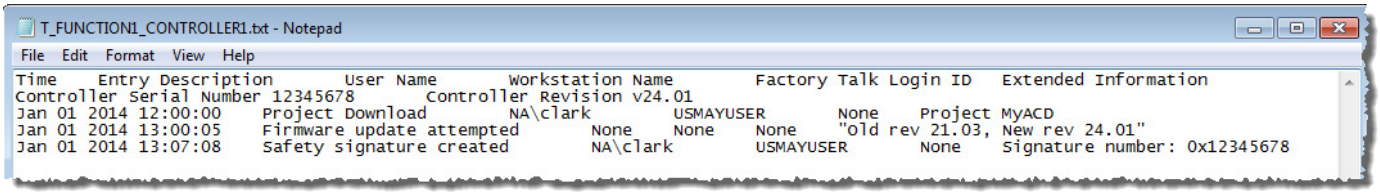
Content	Description	Format
Time	Controller's GMT time	MMM-DD-YY HH:MM:SS 24-hour time
Entry	Entry Description (defined in Entry List section)	
User Name	User's login ID	Windows domain name with display name if available
Workstation Name	User's computer name	Computer Name
FactoryTalk ID	User's FactoryTalk login ID	Alphanumeric characters
Extended Information	Entry specific information (defined in Entry List section)	
Change Detection Audit Value[1]	Changes that the controller detected in the Audit value	

(1) Version 20 or later. See *Change detection*.

The log file is formatted in UTF-16, and has a file extension of .txt. Double-clicking the file opens it in Notepad on most systems. However, since it is formatted as Tab Separated Values (TSV), it also opens correctly in a spreadsheet application, like Microsoft Excel.

This screen is an example of the log file in Notepad.

Tip: For version 20 and later, a column for the Audit Value changes is included in the log file.



This screen is an example of the log file in Excel.

	A	B	C	D	E	F	G	H	I	J
1	Time	Entry Description	User Name	Workstation Nam	Factory Talk Log	Extended Informa	Controller Serial Nu	Controller Revision v24.01		
2	Jan 01 2014 12:00:00	Project Download	NA\clark	USMAYUSER	None	Project MyACD				
3	Jan 01 2014 13:00:05	Firmware update att	None	None	None	Old rev 21.03, New rev 24.01				
4	Jan 01 2014 13:07:08	Safety signature cre	NA\clark	USMAYUSER	None	Signature number: 0x12345678				
5										

See also

Change detection on page 50

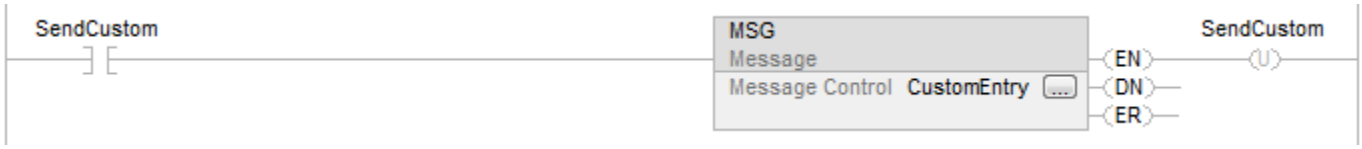
Create Custom Log Entries

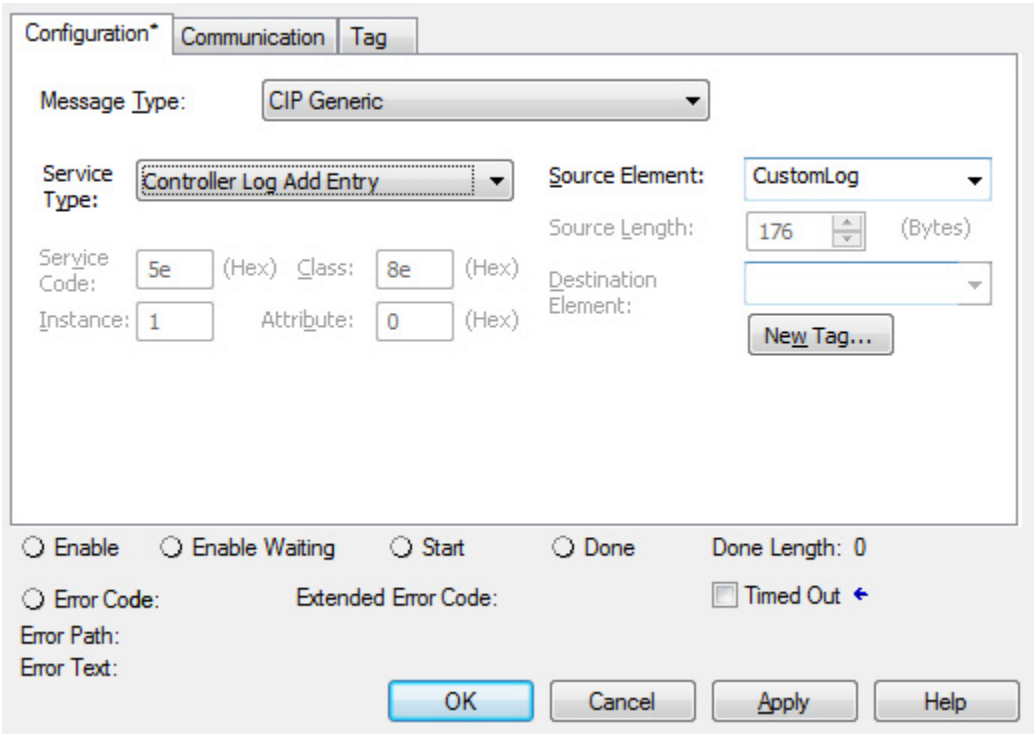
Use a message instruction to add custom entries to the controller log. The message instruction uses a CIP Generic message type and a service type of Controller Log Add Entry.

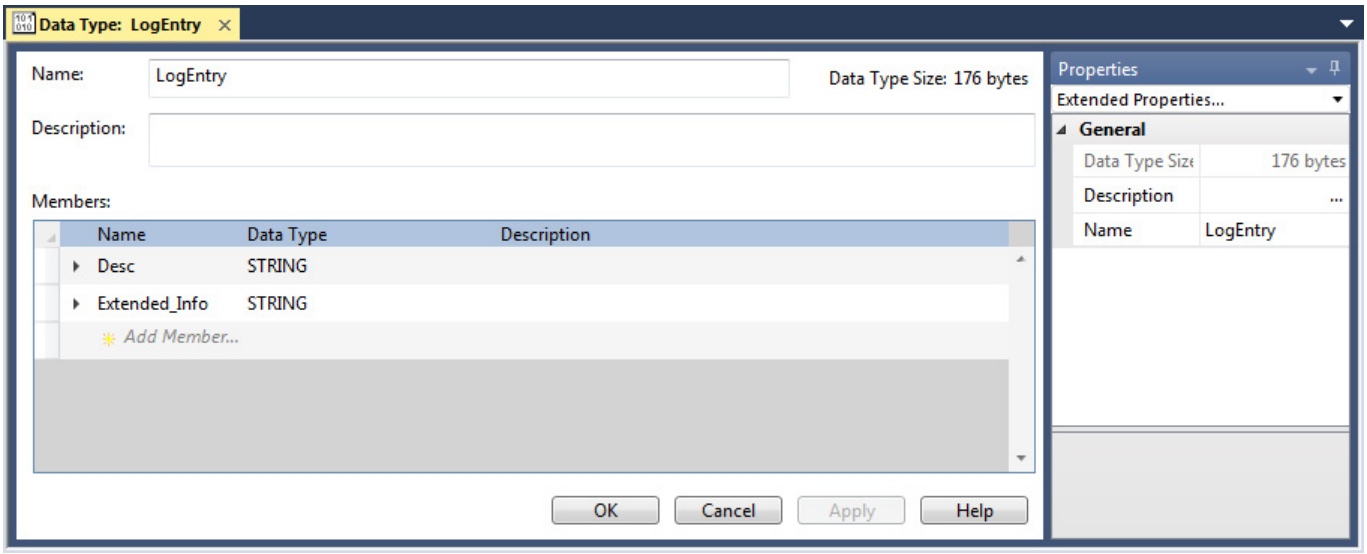
The source element of this message should be a tag of a user-defined data type. The user-defined data type should contain two string members. The controller writes the first string to the log entry's Description field. The controller writes the second string to the log entry's Extended Information field.

This rung of ladder logic shows how to send the message instruction, followed by the **Configuration** dialog box of the message instruction and the definition of the user-defined data type used for the source element.

Example: Send the Message ladder instruction, Controller Log Add Entry configuration dialog box, and Data Type dialog box.







Sample Ladder Logic File

The Logix Designer application contains a controller logging sample ladder file. If the sample files were installed during the installation, open the sample files from the **Help** menu. On the **Help** menu, select **Vendor Sample Projects**.

Controller Log Events

This table describes the events that the controller stores in the controller log.

Tip: In Logix Designer versions 30 and later, audit log entries also appear in the controller log. Audit log entries describe events such as changes to component properties, edits in a routine, and changes to tag values. The audit log is used for auditing by the FactoryTalk® AssetCentre.

Entry	Information Logged
Project download[1]	• Time Stamp • Entry Description: Project download • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Project
Load from removable media	• Time Stamp • Entry Description: Project load • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Project
Load from removable media auto-initiated	• Time Stamp • Entry Description: Project auto load • UserName: Local • Workstation Name: None • FactoryTalk Login Id: None • Extended Information: Project
Store to removable media	• Time Stamp • Entry Description: Project store • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Project

Online edits tested or assembled	• Time Stamp
	• Entry Description: Online edits modified controller program
	• UserName
	• Workstation Name
	• FactoryTalk Login Id
	• Extended Information: None
	• Edits logged are:
	  • Test Program Edits
	  • UnTest Program Edits
	  • Assemble Program Edits
	  • Accept Program Edits
	  • Accept Pending Rung Edits

Partial import online completed[2]	• Time Stamp • Entry Description: Partial import online modified controller • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
I/O forces enabled	• Time Stamp • Entry Description: I/O forces enabled • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
I/O forces disabled	• Time Stamp • Entry Description: I/O forces disabled • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None

Entry	Information Logged
I/O forces removed	• Time Stamp • Entry Description: I/O forces removed • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
I/O forces modified	• Time Stamp • Entry Description: I/O force value changed • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Tag
SFC forces enabled	• Time Stamp • Entry Description: SFC forces enabled • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
SFC forces disabled	• Time Stamp • Entry Description: SFC forces disabled • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
SFC forces removed	• Time Stamp • Entry Description: SFC forces removed • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
SFC forces modified	• Time Stamp • Entry Description: SFC element force value changed • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Routine
Firmware update from workstation	• Time Stamp • Entry Description: Firmware update attempted • UserName: None • Workstation: None • FactoryTalk Login Id: None • Extended Information: Old revision <major.<minor>, New revision <major.<minor>, where the major and minor revision numbers are each two digits.
Firmware update from removable media	• Time Stamp • Entry Description: Firmware update from removable media attempted • UserName: Local • Workstation: None • FactoryTalk Login Id: None • Extended Information: Old revision <major.<minor>, New revision <major.<minor>, where the major and minor revision numbers are each two digits.

Entry	Information Logged

Mode changed through Logix Designer	• Mode change started Time Stamp
	• Entry Description: Remote mode change
	• UserName
	• Workstation Name
	• FactoryTalk Login Id
	• Extended Information: Old mode <mode>, New mode <mode>
	• Possible modes:
	  • Run
	  • Remote Run
	  • Test
	  • Program
	  • Remote Program

Mode changed through keyswitch	• Time Stamp
	• Entry Description: Keyswitch mode change
	• UserName: Local
	• Workstation Name: None
	• FactoryTalk Login Id: None
	• Extended Information: Old mode <mode>, New mode <mode>
	• Possible modes:
	  • Run
	  • Remote Run
	  • Test
	  • Program
	  • Remote Program

Major fault	• Time Stamp • Entry Description: A major fault occurred • UserName: None • Workstation Name: None • FactoryTalk Login Id: None • Extended Information: Fault type <type number>, Fault code <code number>
Major faults cleared	• Time Stamp • Entry Description: All major faults cleared • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
Major faults cleared through key switch	• Time Stamp • Entry Description: All major faults cleared • UserName: Local • Workstation Name: None • FactoryTalk Login Id: None • Extended Information: None

Program properties modified	• Time Stamp
	• Entry Description: Program properties modified
	• UserName
	• Workstation Name
	• FactoryTalk Login Id
	• Extended Information: Program
	• Property changes logged:
	  • Inhibit checkbox
	  • Main routine changed
	  • Fault routine changed

Entry	Information Logged

Task properties modified	• Time Stamp
	• Entry Description: Task properties modified
	• UserName
	• Workstation Name
	• FactoryTalk Login Id
	• Extended Information: Task
	• Task property changes logged:
	  • Type changed
	  • Inhibit checkbox
	  • Disable Automatic Output Processing to Reduce Task Overhead checkbox
	  • Priority value
	  • Period Value
	  • Execute if no Event occurs within X ms checkbox
	  • Trigger changed
	  • Trigger Tag changed
	  • Schedule changed/Service operation

Controller time slice modified	• Time Stamp
	• Entry Description: Controller timeslice modified
	• UserName
	• Workstation Name
	• FactoryTalk Login Id
	• Extended Information
	• Changes logged:
	  • System Overhead Time Slice
	  • During unused System Overhead Time Slice radio buttons

Removable media removed	• Time Stamp • Entry Description: Removable media removed • UserName: Local • Workstation Name: None • FactoryTalk Login Id: None • Extended Information: None
Removable media inserted	• Time Stamp • Entry Description: Removable media inserted • UserName: Local • Workstation Name: None • FactoryTalk Login Id: None • Extended Information: None

Safety signature create	• Time Stamp
	• Entry Description: Safety signature create
	• UserName
	• Workstation Name
	• FactoryTalk Login Id
	• Extended Information:
	  • Eight-character signature number on Compact GuardLogix 5370 and GuardLogix 5570 controllers: 0xYYYYYYYY (hex format)
	  • 64-character signature number on Compact GuardLogix 5380 and GuardLogix 5580 controllers: 0xYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY (hex format)

Entry	Information Logged

Safety signature delete	• Time Stamp
	• Entry Description: Safety signature delete
	• UserName
	• Workstation Name
	• FactoryTalk Login Id
	• Extended Information:
	  • Eight-character signature number on Compact GuardLogix 5370 and GuardLogix 5570 controllers: 0xYYYYYYYY (hex format)
	  • 64-character signature number on Compact GuardLogix 5380 and GuardLogix 5580 controllers: 0xYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYYY (hex format)

Safety lock	• Time Stamp • Entry Description: Safety lock • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
Safety unlocked	• Time Stamp • Entry Description: Safety unlock • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
Custom entry	• Time Stamp • Entry Description: <User supplied string> maximum 40 characters • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: <User supplied info>, maximum 82 characters
Constant tag data changed	• Time Stamp • Entry Description: Constant tag data changed • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Tag: <tag name><old Value> to <new value>
Multiple constant tag data changed	• Time Stamp • Entry Description: Multiple constant tag data changed • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Tag: <tag name>
Change to constant tag configuration reset	• Time Stamp • Entry Description: Constant tag configuration reset • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Tag: <tag name>

Entry	Information Logged
Safety signature delete inhibited in Run mode	• Time Stamp • Entry Description: Safety signature delete inhibited in Run mode • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
Safety signature delete allowed in Run mode	• Time Stamp • Entry Description: Safety signature delete allowed in Run mode • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: None
Audit Value Mask Modified	• Time Stamp • Entry Description: Change detection mask modified • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Old mask: 0xFFFF_FFFF_FFFF_FFFF New mask: 0xFFFF_FFFF_FFFF_DFFF
Log Collected Data Cleared	• Time Stamp • Entry Description: Log Collected Data Cleared • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Log: <log name>
Program Connection Modified	• Time Stamp • Entry Description: Program connection modified • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Program, <connection type>
USB Cable Connected[3]	• Time Stamp • Entry Description: USB connected • UserName: None • Workstation Name: None • FactoryTalk Login Id: None • Extended Information: None
USB Cable Disconnected[3]	• Time Stamp • Entry Description: USB disconnected • UserName: None • Workstation Name: None • FactoryTalk Login Id: None • Extended Information: None
Vendor Certificate Status	• Time Stamp • Entry Description: Vendor Certificate Status • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Status string

Entry	Information Logged

Port state modified	• Time Stamp
	• Entry Description: I/O forces enabled
	• UserName
	• Workstation Name
	• FactoryTalk Login Id
	• Extended Information:
	  • PortId
	  • PhyId
	  • NewState

Constant Tag attribute set	• Time Stamp • Entry Description: Constant Tag attribute set • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Tag name
Constant Tag attribute clear	• Time Stamp • Entry Description: Constant Tag attribute clear • UserName • Workstation Name • FactoryTalk Login Id • Extended Information: Tag name

(1) In versions 20 and later, the Change Detection Audit Value column is included in the controller log. This column is used to record the Audit Value for Change Detection.

(2) In versions 20 and later, this value in the controller log equates to Transaction committed in the Change Detection mask.

(3) This feature only applies for the Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, and GuardLogix 5580 controllers.

Change detection

Introduction

Use the controller change detection feature as an additional means of detecting changes made to Logix 5000 controllers.

- A unique audit value is generated when a project is downloaded to the controller.
- When a change is detected in the controller, a new audit value is generated.
- ChangesToDetect mask allows programmatic configuration of the events to monitor for changes.
- Controller change detection is integrated into the Logix Designer application.

IMPORTANT The change detection feature is not supported in version 19 or earlier.

Keep these considerations in mind when using change detection:

- Change detection is not available on the Studio 5000® Logix Emulate™ Controller and the SoftLogix5860 Controller, and the audit value is not populated.
- The 1769-L3x and 1769-L4x CompactLogix controllers do not support storing log entries to removable media.
- Change detection is integrated in FactoryTalk AssetCentre version 4.1 and later. FactoryTalk AssetCentre can be configured to detect changes in the controller and read the controller's Controller Log.
- Change detection is not integrated with RSMACC utilities.

Controller change detection

Two controller attributes support the Change Detection feature in version 20 and later.

Attribute Name	Description	Access
AuditValue	A unique value that is generated when a project is downloaded to the controller or loaded from removable storage. When a change is detected this value is updated. To specify which changes are monitored, use the ChangesToDetect attribute.	GSV
ChangesToDetect	Used to specify which changes are monitored. When a monitored change occurs, the Audit Value is updated.	GSV/SSV

ChangesToDetect

The ChangesToDetect mask is a 64-bit value. Each bit of the ChangesToDetect mask corresponds to a particular event that could cause the Audit Value to change. See *ChangeToDecect* format.

IMPORTANT Change detection is unavailable on the Studio 5000 Logix Emulate Controller and the SoftLogix5860 Controller.

If a bit is set to a value of 1, the corresponding event is monitored for changes and when a change for that event occurs the AuditValue updates. If a bit is set to a value of 2, the corresponding event is not monitored and does not affect the Audit Value. The default Audit Value of: 0xFFFFFFFFFFFFFFFF indicates that all events are monitored.

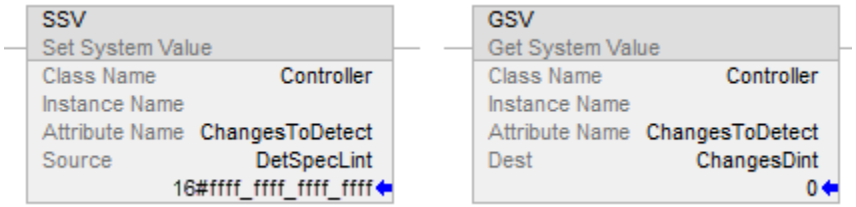
Tip: The Audit Value updates when the controller is online.

Some events always update the audit value when they occur. For example, the audit value changes when a project is downloaded to the controller, or when ChangesToDetect is reconfigured. These types of events are not included in the ChangesToDetect mask.

Configure the ChangesToDetect mask programmatically using SSV, GSV and MSG instructions.

Tip: Use the **Security** tab in the **Controller Properties** dialog box to configure Change Detection on version 20 or later of the Logix Designer application.

Use the Set System Value (SSV) instruction to write to the ChangesTo Detect attribute and the Get System Value (GSV) instruction to read the ChangesTo Detect attribute programmatically.



Tip: Rockwell recommends using the DINT[2] data type to avoid limitations when working with LINT data types in Rockwell Automation controllers.

Use the **Message Configuration** dialog box to read or write to the ChangesToDetect attribute through CIP Generic Messages. For complete information on how to access data using an MSG instruction, refer to the *<LOGIX> Controllers Message Programming Manual*, publication *1756-PM012*.

Use these settings to configure the ChangesToDetect attribute using the **Message Configuration** dialog box.

If you want to:	In this property	Type or select
Set controller events monitored for changes	Message Type	CIP Generic
	Service Type	Changes to Detect Set
	Source	tag_name of type DINT[2] or LINT 1 This tag represents a bit mask of the changes monitored for the controller.
	Destination	Leave blank
Get controller events monitored for changes	Message Type	CIP Generic

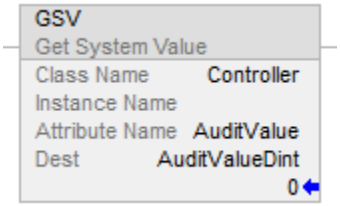
If you want to:	In this property	Type or select
	Service Type	Changes to Detect Get
	Source	Leave blank
	Destination	tag_name of type DINT[2] or LINT 1 This tag represents a bit mask of the changes monitored for the controller.

See also

Logix 5000 Controllers Messages Programming Manual, publication no. 1756-PM012

AuditValue

Use the Get System Value (GSV) instruction to read the AuditValue attribute programmatically.



Use the **Message Configuration** dialog box to read the AuditValue attribute through CIP Generic Messages. For complete information on how to access data using an MSG instruction, refer to the *Logix 5000 Controllers Messages Programming Manual*, publication no. *1756-PM0012*.

Use these settings to read the Audit Value attribute using the Message Configuration.

To get AuditValue in this property	Type or select
Message Type	CIP Generic
Service Type	Audit Value Get
Source	Leave blank
Destination	tag_name of type DINT[2] or LINT 1 This tag contains the AuditValue for the controller.

See also

Logix 5000 Controllers Messages Programming Manual, publication no. 1756-PM012, publication no. 1756-PM012

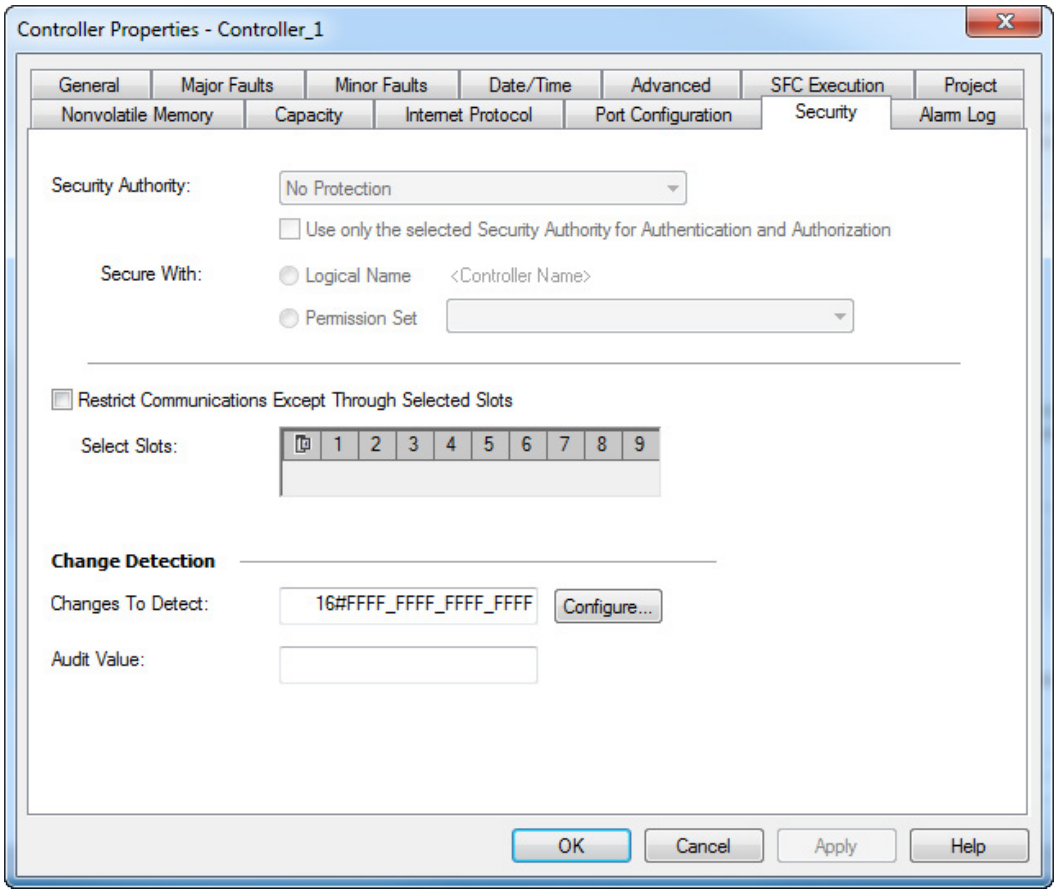
ChangesToDetect format

Bit number	Event description
0	Project stored to removable media
1	Online edits modified controller program

Bit number	Event description
2	Transaction committed. Indicates that a batch of one or more changes has been applied to the controller. For example, this might happen when a routine is imported online or when a module configuration is changed online.
3	SFC forces enabled
4	SFC forces disabled
5	SFC forces removed
6	SFC element force value changed
7	I/O forces enabled
8	I/O forces disabled
9	I/O forces removed
10	I/O forces modified
11	Firmware update attempted
12	Firmware update from removable media attempted
13	Remote mode change
14	Keyswitch mode change
15	A major fault occurred
16	All major faults cleared
17	All major faults cleared through keyswitch
18	Task properties modified
19	Program properties modified
20	Controller time slice modified
21	Removable media removed
22	Removable media inserted
23	Safety signature created
24	Safety signature deleted
25	Safety lock
26	Safety unlock
27	Constant Tag value changed
28	Multiple constant Tag values changed
29	Constant Tag attribute cleared
30	Constant Tag attribute set
31	Custom Log Entry Added
32	Correlation Affected. Indicates a change occurred in the controller that affects the synchronization between the controller and the project file that was downloaded to it. **Tip**: Correlation Affected can include detection of a component that has been created, deleted, or modified, or logic that has been modified. Including the Correlation Affected event updates the audit value for these types of changes, even if other similar events (such as **Online edits modified controller program** or **Task properties modified**) are masked.
33	Safety signature delete inhibited in Run mode
34	Safety signature delete allowed in Run mode
35	Data log or Alarm log cleared
36	Program connection modified

Change detection in the Logix Designer application

Use the **Security** tab in the **Controller Properties** dialog box (available in versions 20 and later of the application) to configure Change Detection.



To edit the **Changes To Detect** field, type a new value. To select the events to monitor for changes from a list, select **Configure** to open the **Configure Changes to Detect** dialog box.

For additional information on configuring the settings on the **Security** tab, refer to the *<LOGIX> Security Programming Manual*, publication *1756-PM016*, or the online help for the Logix Designer application.

Tip: To configure Change Detection programmatically use SSV, GSV, or MSG instructions. See *Controller change detection*.

See also

Controller Change Detection on page 47

Logix 5000 Security Programming Manual, publication no. 1756-PM016

Component tracking

Use component tracking to determine whether tracked components have changed. Like change detection, component tracking indicates changes to a project, but at the component level. The Logix Designer application creates a unique signature for each tracked component and compiles those signatures into one overall tracked value to indicate their current state. Component tracking applies to:

- Routines

- Add-On Instructions
- I/O Modules
- Constant tags

Tip: Component tracking is supported only on CompactLogix 5370, ControlLogix 5570, Compact GuardLogix 5370, and GuardLogix 5570 controllers in versions 30 or later of the Logix Designer application.

Tracked components appear in the **Tracked Components** dialog box, which is accessible on the **Controller Properties** dialog box - **Security** tab. The tracked value changes when a tracked component updates in a way that affects the execution of a routine or an Add-On Instruction. Updating a component attribute that does not affect execution--for example, descriptions and non-constant tag values--does not change the tracked value.

Tip: When an attribute is changed and then changed back to its previous value, the tracked value does not change. This behavior differs from Change Detection, which registers a change when the change is reversed.

Tip: The Tracked State value and the list of tracked components are static as long as the dialog box is open and will not reflect changes from other workstations. If a user on another workstation makes an update, if changes are not showing, close and re-open the Tracked Components dialog box to see the changes.

Recommended limits for tracked components

This table lists the recommended limits on the number of tracked components. If the limits are exceeded, a noticeable impact on performance may occur in the Logix Designer application.

Tip: To optimize performance, configure component tracking so that the tracked state value is calculated on demand rather than at regular intervals.

Tracked component	Recommended limit
Routines	100
Add-On Instructions	100
I/O Modules	50
Tags	300

FactoryTalk Security permissions that control component tracking

These FactoryTalk Security permissions control a user's ability to change tracking status for a component:

- Routine: Modify Properties
- Add-On Instruction: Modify
- Module: Modify Properties
- Tag: Modify Properties

Track the state of components and constant tags

Designate project components for tracking to monitor the current state of a component. Use these instructions to track the state of a routine, an Add-On Instruction, an I/O module, or a constant tag.

To track the state of a component:

1. In the **Controller Organizer**, highlight the component to track.
2. Right-click and select **Include in tracking group**.
3. To stop tracking a component, right-click and select **Include in tracking group** again.

Tip: When using the tracked state value to verify that code in a tracked routine is being executed, make sure to track all the routines that call the tracked routine, up to and including the main routine. Tracking all components that call the tracked routine helps to prevent a user from modifying the project so that the tracked routine is no longer called.

To track the state of constant tags

1. In the **Tag editor** or the **Data Monitor**, highlight the constant tag to track.
2. Right-click and select **Include in tracking group**.
3. To stop tracking a constant tag, right-click and select **Include in tracking group** again.

Tip: When a base tag or an alias tag is tracked, the base tag and all alias tags are tracked.

View tracked components

After selecting project components for tracking, check project component status in the **Tracked Components** dialog box, which is accessible on the **Controller Properties dialog box - Security** tab.

To view tracked components:

1. On the **Online** toolbar, select the **Controller Properties** icon.
2. On the **Controller Properties** dialog box, select the **Security** tab.
3. In the **Change Detection** section, select **View components** to open the **Tracked components** dialog box.

Retrieve the tracked state value using a Message instruction

Use a Message (MSG) instruction to retrieve the tracked state value. When the controller receives the MSG instruction, it calculates the tracked state value and stores it in a destination tag.

Tip: When using the tracked state value to verify that code in a tracked routine is being executed, make sure to track all the routines that call the tracked routine, up to and including the main routine. Tracking all components that call the tracked routine helps to prevent a user from modifying the project so that the tracked routine is no longer called.

To retrieve the tracked state value using an MSG instruction:

1. Create a MESSAGE tag.
2. Insert an MSG instruction in the ladder logic or Structured Text routine.

3. In the instruction, select **Browse** to open the **Message Configuration** dialog box.
4. On the **Configuration** tab, configure the following settings:
   - For **Message Type**, select **CIP Generic**.
   - For **Service Type**, select **Tracked State Value Get**.
   - For **Source Element**, select the source tag element. In this release of Logix Designer, the value of the SINT datatype must be 1.
   - For **Destination Element**, select the tag where the controller sends the tracked state value. For best results, create a destination tag that is an array of 32 SINTs.
5. On the **Communication** tab, configure the communication path.
6. Select **Apply**.

Access status information

Introduction

The controller supports status keywords inserted in logic for monitoring specific events.

- The status keywords are not case sensitive.
- Because the status flags can change quickly, the Logix Designer application does not display the status of the flags. For example, even when a status flag is set, an instruction that references that flag is not highlighted.
- A tag alias cannot be defined to a keyword.

The table lists the available key words.

To determine if	Use
The stored value cannot fit into the destination because it is either: • Greater than the maximum value for the destination. • Less than the minimum value for the destination. **Important:** Each time S:V goes from cleared to set, it generates a minor fault (type 4, code 4).	S:V
The instruction's destination value is 0.	S:Z
The instruction's destination value is negative.	S:N
An arithmetic operation causes a carry or borrow that tries to use bits that are outside the data type. For example: • Adding 3 + 9 causes a carry of 1. • Subtracting 25 - 18 causes a borrow of 10.	S:C
This is the first, normal scan of the routines in the current program.	S:FS
At least one minor fault has been generated. • The controller sets this bit when a minor fault occurs due to program execution. • The controller does not set this bit for minor faults that are not related to program execution, such as battery low.	S:MINOR

Status of S:FS when the project has an SFC

The state of S:FS depends on the status of the SFC.

- When using S:FS in an action of a sequential function chart (SFC), S:FS is set (on) for one scan each time the step goes active. S:FS = *step_name*.FS.
- If the SFC calls a routine, S:FS is set (on) for one scan each time the step that calls the routine goes active. S:FS = *step_name*.FS.

  If the SFC does not call a routine, S:FS is set (on) for the first scan of the task.

Example: SFC Calls a Ladder Diagram

Suppose several steps in an SFC call the same ladder diagram routine. And suppose the ladder diagram uses S:FS. Each time one of those steps goes active, S:FS turns on for one scan of the ladder diagram.

Example: Several Tasks but No SFC

Suppose two tasks use ladder diagrams. When the first task runs for the first time, S:FS turns on for one scan. After that, S:FS stays off for that task. When the other task runs for the first time, S:FS turns on for one scan in that task. S:FS stays off in the first task that ran.

Get and set system data

The controller stores system data in objects. There is no status file, as in the PLC-5 controller. Use the GSV/SSV instructions to get and set controller system data that is stored in objects.

- The GSV instruction retrieves the specified information and places it in the destination.
- The SSV instruction sets the specified attribute with data from the source.

ATTENTION: Use the SSV instruction carefully. Making changes to objects can cause unexpected controller operation or injury to personnel.

To create a GSV/SSV instruction, use Logix Designer online help to find the required information, and create the instruction in the Logix Designer application. This summary describes the steps.

- Open the online help in Logix Designer application to find the object and data type to use in the GSV/SSV instruction.
- In the Logix Designer application, create a tag for the value of the attribute.
- In the Ladder Logic editor, create the GSV/SSV instruction.

To get or set system data

1. Open the project in the Logix Designer application.
2. From the **Help** menu, choose **Contents**.
3. On the Help window, select the **Index** tab.
4. In the **Type in the keyword to find** field, type **GSV/SSV objects** and select **Display**.
5. In the GSV/SSV Objects help window, select the object.

To get or set	Select this object
Customize instructions for commonly-used logic	AddOnInstructionDefinition
Different Axis objects depending on motion application	Axis (Several types)
System-overhead time slice	Controller
Physical hardware of a controller	ControllerDevice
Groups one or more axes to generate coordinated motion	CoordinateSystem
Coordinated system time for the devices in one chassis	CST
DF1 communication driver for the serial port	DF1

To get or set	Select this object
Fault history for a controller	FaultLog
Attributes of a message instruction	Message
Sttus, faults, and mode of a module	Module
Group of axes	MotionGroup
Fault information or scan time for a program	Program
Instance number of a routine	Routine
Different objects for safety	Safety
Configuration of the serial port	SerialPort
Properties or elapsed time of a task	Task
Precision time management for motion control	TimeSynchronize
Wall clock time of a controller	WallClockTime

6. In the list of attributes for the object, identify the attribute to access.

   For attribute information, see the *Logix 5000 Controllers General Instruction Reference Manual*, publication no. *1756-PM003*.

7. In Logix Designer application, create a tag for the value of the attribute.

If the data type of the attribute is	Then
One element (for example, DINT)	Create a tag for the attribute.
More than one element (for example, DINT[7])	• Create a user-defined data type that matches the organization of data that is used by the attribute. • Create a tag for the attribute and use the data type identified for the object.

8. In the Ladder Logic routine, enter the appropriate instruction.

To	Enter this instruction
Get the value of an attribute	GSV
Set the value of an attribute	SSV

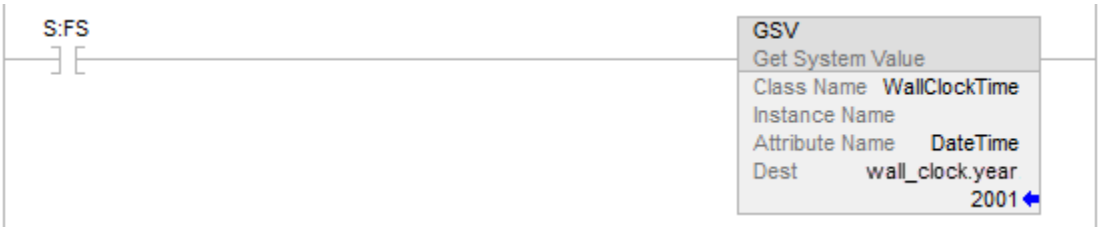
9. Assign the required operands to the instruction.

For this operand	Select
Class name	Name of the object.
Instance name	Name of the specific object (for example, name of the required I/O module, task, message). • Not all objects require this entry. • To specify the current task, program, or routine, select THIS.
Attribute name	Name of the attribute.
Dest (GSV)	Tag that stores the retrieved value. If the tag is a user-defined data type or an array, select the first member or element.
Source (SSV)	Tag that stores the value to be set. If the tag is a user-defined data type or an array, select the first member or element.

The following example gets the current date and time.

Example: Get a System Value

At the first scan, the following rung of logic gets the DateTime attribute of the WallClockTime object and stores it in the wall_clock tag, which is based on a user-defined data type.



For more GSV/SSV information, see the *Logix 5000 Controllers General Instructions Reference Manual*, publication no. *1756-RM003*.

See also

Logix 5000 Controllers General Instructions Reference Manual, publication no. 1756-RM003

Index

T

Rockwell Automation support

Use these resources to access support information.

Technical Support Center	Find help with how-to videos, FAQs, chat, user forums, and product notification updates.	rok.auto/support
Knowledgebase	Access Knowledgebase articles.	rok.auto/knowledgebase
Local Technical Support Phone Numbers	Locate the telephone number for your country.	rok.auto/phonesupport
Literature Library	Find installation instructions, manuals, brochures, and technical data publications.	rok.auto/literature
Product Compatibility and Download Center (PCDC)	Get help determining how products interact, check features and capabilities, and find associated firmware.	rok.auto/pcdc

Documentation feedback

Your comments help us serve your documentation needs better. If you have any suggestions on how to improve our content, complete the form at rok.auto/docfeedback.

Waste Electrical and Electronic Equipment (WEEE)

At the end of life, this equipment should be collected separately from any unsorted municipal waste.

Rockwell Automation maintains current product environmental information on its website at rok.auto/pec.

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

Connect with us.

rockwellautomation.com — expanding **human possibility**™

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-PM015M-EN-P - March 2022
Supersedes Publication 1756-PM015L-EN-P - November 2020