Programming Manual

Allen-Bradley

Logix 5000 Produced and Consumed Tags

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. This table contains the changes made to this revision.

Summary of Changes	Topics
Updated information on controller models that support produced and consumed tags.	Controllers and networks that support produced/consumed tags on page 11

This manual details how, with a Logix5000 controller, to produce and consume system-shared tags and produce a large array. This manual is one of a set of related manuals that show common procedures for programming and operating Logix5000™ controllers.

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

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

Studio 5000 environment

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

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

Additional Resources

These documents contain additional information concerning related Rockwell Automation products.

Resource	Description
Replacement Guidelines: Logix5000 Controllers Reference Manual, publication 1756-RM100	Provides guidelines for migrating projects from ControlLogix 5560/5570 controllers to ControlLogix 5580 controllers, and from CompactLogix 5370 controllers to CompactLogix 5380 controllers.
Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1	Provides general guidelines for installing a Rockwell Automation industrial system.
Product Certifications webpage, available at http://ab.rockwellautomation.com	Provides declarations of conformity, certificates, and other certification details.

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

Legal notices

Copyright notice

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.

Trademark Notices

Allen-Bradley, ControlBus, ControlFLASH, Compact GuardLogix, Compact I/O, ControlLogix, CompactLogix, DCM, DH+, Data Highway Plus, DriveLogix, DPI, DriveTools, Explorer, FactoryTalk, FactoryTalk Administration Console, FactoryTalk Alarms and Events, FactoryTalk Batch, FactoryTalk Directory, FactoryTalk Security, FactoryTalk Services Platform, FactoryTalk View, FactoryTalk View SE, FLEX Ex, FlexLogix, FLEX I/O, Guard I/O, High Performance Drive, Integrated Architecture, Kinetix, Logix5000, Logix5550, MicroLogix, DeviceNet, EtherNet/IP, PLC-2, PLC-3, PLC-5, PanelBuilder, PowerFlex, PhaseManager, POINT I/O, PowerFlex, Rockwell Automation, RSBizWare, Rockwell Software, RSEmulate, Historian, RSFieldbus, RSLinx, RSLogix, RSNetWorx for DeviceNet, RSNetWorx for EtherNet/IP, RSMACC, RSView, RSView32, Rockwell Software Studio 5000 Automation Engineering & Design Environment, Studio 5000 View Designer, SCANport, SLC, SoftLogix,

SMC Flex, Studio 5000, Ultra 100, Ultra 200, VersaView, WINtelligent, XM, SequenceManager are trademarks of Rockwell Automation, Inc.

Any Rockwell Automation logo, software or hardware product not mentioned herein is also a trademark, registered or otherwise, of Rockwell Automation, Inc.

Other Trademarks

CmFAS Assistant, CmDongle, CodeMeter, CodeMeter Control Center, and WIBU are trademarks of WIBU-SYSTEMS AG in the United States and/or other countries. Microsoft is a registered trademark of Microsoft Corporation in the United States and/or other countries. ControlNet is a trademark of ControlNet International. DeviceNet is a trademark of the Open DeviceNet Vendors Association (ODVA). Ethernet/IP is a trademark of ControlNet International under license by ODVA.

All other trademarks are the property of their respective holders and are hereby acknowledged.

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 Automation

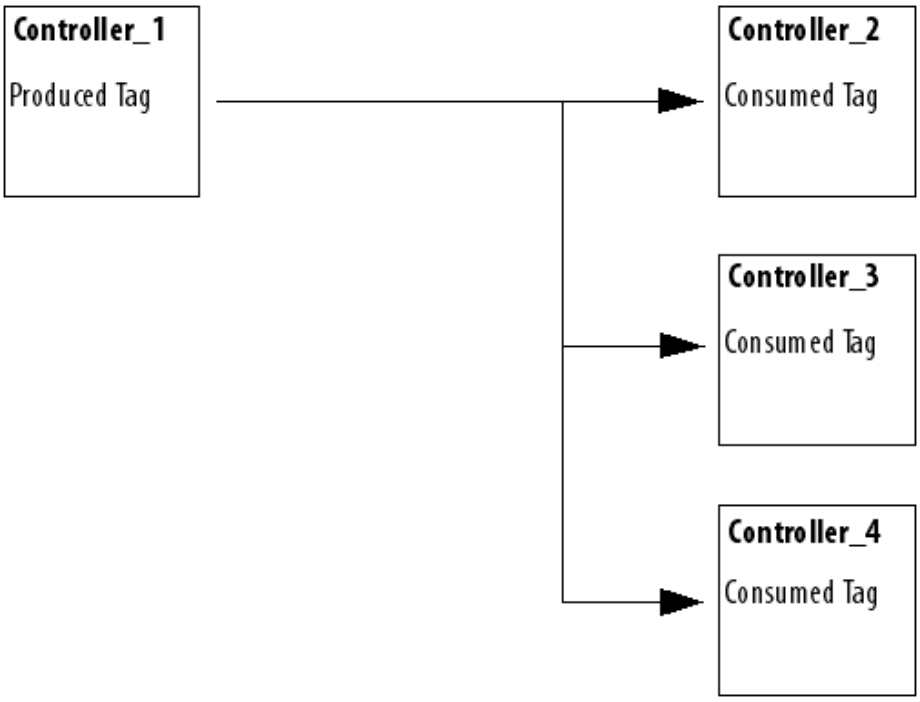
Customer Support Telephone — 1.440.646.3434

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

Produce and Consume a Tag

Introduction

A Logix5000 controller lets you produce (broadcast) and consume (receive) system-shared tags.



Term	Definition
Produced tag	A tag that a controller makes available for use by other controllers. Multiple controllers can simultaneously consume (receive) the data. A produced tag sends its data to one or more consumed tags (consumers) without using logic.
Consumed tag	A tag that receives the data of a produced tag. The data type of the consumed tag must match the data type (including any array dimensions) of the produced tag. The RPI of the consumed tag determines the period at which the data updates.

Controllers and networks that support produced/consumed tags

For two controllers to share produced or consumed tags, both controllers must be attached to the same network, such as a ControlNet or Ethernet/IP network. You cannot bridge produced and consumed tags over two networks. Check the documentation specific to your controller to determine which network connections it supports.

Connection requirements of a produced or consumed tag

Produced and consumed tags each require connections. As you increase the number of controllers that can consume a produced tag, you also reduce the number of connections the controller has available for other operations, like communication and I/O.

Important:	If a consumed-tag connection fails, all of the other tags being consumed from that remote controller stop receiving new data.

Each produced or consumed tag uses the following connections.

This Type of Tag	Uses This Many Connections
Produced tag	number_of_consumers + 1
Consumed tag	1

Example: Connection Requirements of a Produced or Consumed Tag

- A ControlLogix controller producing 4 tags for 1 controller uses 8 connections:
  - Each tag uses 2 connections (1 consumer + 1 = 2).
  - 2 connections per tag x 4 tags = 8 connections.
- Consuming 4 tags from a controller uses 4 connections (1 connection per tag x 4 tags = 4 connections).

Communication Paths to CompactLogix 5380, CompactLogix 5480, and ControlLogix 5580 controllers

If a controller running release v24 or earlier is to consume tag data from a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller, it must have a target device in its I/O configuration for the connection. However, releases v24 and earlier do not have profiles for CompactLogix 5380, CompactLogix 5480, and ControlLogix 5580 controllers, so a substitute must be used.

Rockwell recommends the following substitutes:

- If you want to consume tags through the embedded Ethernet port on a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller, represent the controller as a CompactLogix 5370 Controller, such as the 1769-L36ERM, in your consuming controller's I/O configuration. These controllers have a built-in Ethernet port and therefore can be addressed using an EtherNet/IP Address.
- If you want to consume tags through the backplane port on a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller, represent the controller as a ControlLogix 5570, such as the 1756-L75, in your consuming controller's I/O configuration. This process allows connection using the backplane.

Considerations when migrating projects that contain multicast produce tags

When you migrate projects from an earlier controller model to a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller, you need to verify that projects containing multicast produce tags are configured correctly.

In releases before v28, produce tags produce data at the requested packet interval (RPI) of the fastest requesting consumer. This behavior allows multiple consumers, with different RPI settings, to successfully connect to a producer. In releases v28 and later, the first consumer of a produce tag determines the RPI at which data is produced. All subsequent consumers must request the same RPI value as the first consumer or they fail to connect and display error code 0112.

The first consumer of a produce tag is the device that sends the first consumer connection to the producing controller. The first consumer request is based on the order in which the producer and consumer control system powers up, so the first consumer can change if power is cycled to the system. You cannot configure a device to be the first consumer. It can be difficult to tell which consumer will send the first request to the producing controller, so you should plan accordingly when configuring multicast produce tags.

To make sure consumers of produce tags can connect to CompactLogix 5380, CompactLogix 5480, and ControlLogix 5580 controllers, take the following steps:

- For consumers running releases v17 and earlier that are consumers of a producer from CompactLogix 5380, CompactLogix 5480, and ControlLogix 5580 controllers:
  - Verify that all multicast consumed tags of a produce tag are configured with the same RPI. If they are not, some consumers will fail to connect.
- For multicast consumers running releases v18 to v27, take one of these steps:
  - Verify that all multicast consumed tags of a produce tag are configured with the same RPI, or
  - Verify that all multicast consumers are configured to **Allow Consumed Tags To Use RPI Provided By Consumer**.

For more information on replacing a ControlLogix 5560/5570 controller with a ControlLogix 5580 controller or a CompactLogix 5370 controller with a CompactLogix 5380 controller, refer to the following publication in the Rockwell Literature Library:

Replacement Guidelines: Logix5000 Controllers Reference Manual (1756-RM100)

Organize tags for produced or consumed data

As you organize your tags for produced or consumed data (shared data), follow these guidelines.

<table>
<tr><th>Guideline</th><th colspan="4">Details</th></tr>
<tr><td>Create the tags at the controller scope.</td><td colspan="4">You can share only controller-scoped tags.</td></tr>
<tr><td rowspan="2">Use one of these data types:
• DINT
• REAL
• Array of DINTs or REALs
• User-defined</td><td colspan="4">To share other data types, create a user-defined data type that contains the required data.</td></tr>
<tr><td colspan="4">Use the same data type for the produced tag and corresponding consumed tag or tags.</td></tr>
<tr><td rowspan="6">To share tags with a PLC-5C controller, use a user-defined data type.</td><td>To</td><td>This</td><td colspan="2">Then</td></tr>
<tr><td rowspan="2">Produce</td><td>Integers</td><td colspan="2">Create a user-defined data type that contains an array of INTs with an even number of elements, such as INT[2]. (When you produce INTs, you must produce two or more.)</td></tr>
<tr><td>Only one REAL value</td><td colspan="2">Use the REAL data type.</td></tr>
<tr><td>More than one REAL value</td><td colspan="2">Create a user-defined data type that contains an array of REALs.</td></tr>
<tr><td rowspan="3">Consume</td><td rowspan="3">Integers</td><td colspan="2">Create a user-defined data type that contains these members.</td></tr>
<tr><td>Data Type</td><td>Description</td></tr>
<tr><td>DINT</td><td>Status</td></tr>
<tr><td></td><td></td><td>INT[x], where x is the output size of the data from the PLC-5C controller. (If you are consuming only one INT, omit x.)</td><td>Data produced by a PLC-5C controller</td></tr>
<tr><td rowspan="2">Limit the size of the tag to 500 bytes.</td><td colspan="4">If you must transfer more than 500 bytes, create logic to transfer the data in packets.</td></tr>
<tr><td colspan="4">If you produce the tag over a ControlNet network, the tag may need to be less than 500 bytes. See Adjust for Bandwidth Limitations on page 15.</td></tr>
<tr><td>Use the highest permissible RPI for your application.</td><td colspan="4">If the controller consumes the tag over a ControlNet network, use a binary multiple of the ControlNet network update time (NUT). For example, if the NUT is 5 ms, use an RPI of 5, 10, 20, 40 ms, and so forth.</td></tr>
<tr><td>Verify that CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller projects containing multicast produce tags are configured correctly.</td><td colspan="4">Take the following steps to make sure consumers of produce tags can connect:
• For consumers running Logix Designer releases v17 and earlier that are consumers of a producer from a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller:
  • Verify that all multicast consumed tags of a produce tag are configured with the same RPI. If they are not, some consumers will fail to connect.
• For multicast consumers running Logix Designer releases v18 to v27, take one of these steps:
  • Verify that all multicast consumed tags of a produce tag are configured with the same RPI, or
  • Verify that all multicast consumers are configured to Allow Consumed Tags To Use RPI Provided By Consumer.</td></tr>
<tr><td>Combine data that goes to the same controller.</td><td colspan="4">If you are producing several tags for the same controller, group the data:
• Into one or more user-defined data types. (This uses less connections than producing each tag separately.)
• According to similar update intervals. (To conserve network bandwidth, use a greater RPI for less critical data.)
For example, you could create one tag for data that is critical and another tag for data that is not as critical.</td></tr>
</table>

Adjust for bandwidth limitations

When you share a tag over a ControlNet network, the tag must fit within the bandwidth of the network.

- As the number of connections over a ControlNet network increases, several connections, including produced or consumed tags, may need to share a network update time (NUT).
- Since a ControlNet network can pass only 500 bytes in one NUT, the data of each connection must be less than 500 bytes to fit into the NUT.

Depending on the size of your system, you may not have enough bandwidth on your ControlNet network for a tag of 500 bytes. If a tag is too large for your ControlNet network, make one or more of these adjustments.

Adjustment	Description	
Reduce your NUT.	At a faster NUT, less connections have to share an update slot.	
Increase the requested packet interval (RPI) of your connections.	At higher RPIs, connections can take turns sending data during an update slot.	
For a ControlNet bridge module (CNB) in a remote chassis, choose the most efficient communication format for that chassis:	**Are most of the modules in the chassis non-diagnostic, digital I/O modules?**	**Then choose this communication format for the remote CNB module.**
	Yes	Rack Optimization
	No	None
	The Rack Optimization format uses an additional 8 bytes for each slot in its chassis. Analog modules or modules that are sending or getting diagnostic, fuse, timestamp, or schedule data require direct connections and cannot take advantage of the rack optimized form. Choosing **None** frees up the 8 bytes per slot for other uses, such as produced or consumed tags.	
Separate the tag into two or more smaller tags.	1. Group the data according to similar update rates. For example, you could create one tag for data that is critical and another tag for data that is not as critical. 2. Assign a different RPI to each tag.	
Create logic to transfer the data in smaller sections (packets).	See Produce a Large Array. on page 38.	

Produce a tag

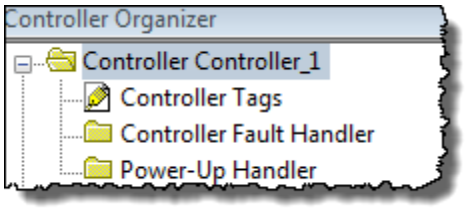
Follow these steps to create a producer tag.

1. Open the Logix Designer application and choose a controller.

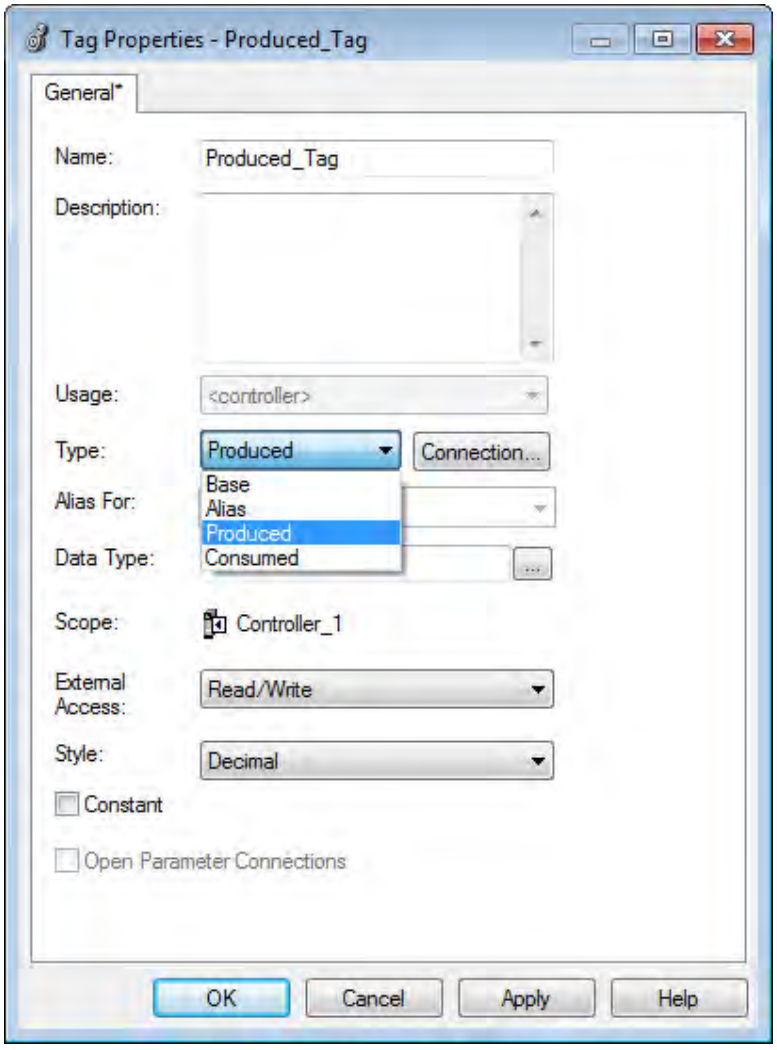
Important:	A controller can have both producing and consuming tags, but a producer cannot consume its own data. The local controller is the consumer, and the remote controller is the producer.

2. In the Controller Organizer, right-click the **Controller Tags** folder and click **Edit Tags**.

You can produce only controller-scoped tags.



3. In the Tags Editor, right-click the tag that you want to be a producer, and choose **Edit (name of tag) Properties**.



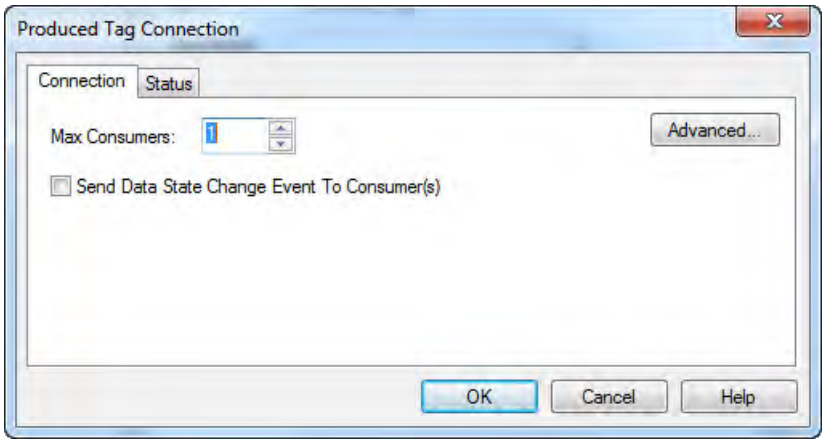
4. In the **Tag Properties** dialog box, in the **Name** box, type the name of the produced tag.

5. From the **Type** list, click **Produced**.

The **Connection** button becomes active when you choose either **Produced** or **Consumed**.

6. Click **Connection**.

   The **Produced Tag Connection** dialog box appears. You can enter the number of controllers to consume (receive) the tag.



7. In the **Max Consumers** box, enter a value.

Important:	The **Advanced** button on the right-hand side of the **Produced Tag Connection** dialog box appears only for controllers using version 18 and later of the application. This button opens the **Advanced Options** dialog box to set the RPI range limitation, negotiated default, or unicast connection. See Additional Steps for a PLC-5C Controller on page 21 for details and procedures.

8. Click **OK**.

Create a consumer tag

Follow these steps to create a consumer tag.

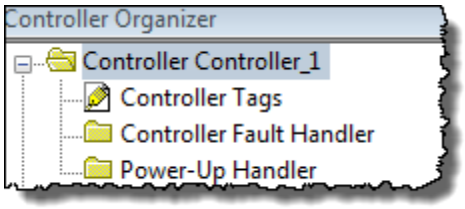
1. Open the Logix Designer application and choose a controller.

Important:	A controller can have both producing and consuming tags, but a producer cannot consume its own data. The local controller is the consumer, and the remote controller is the producer.

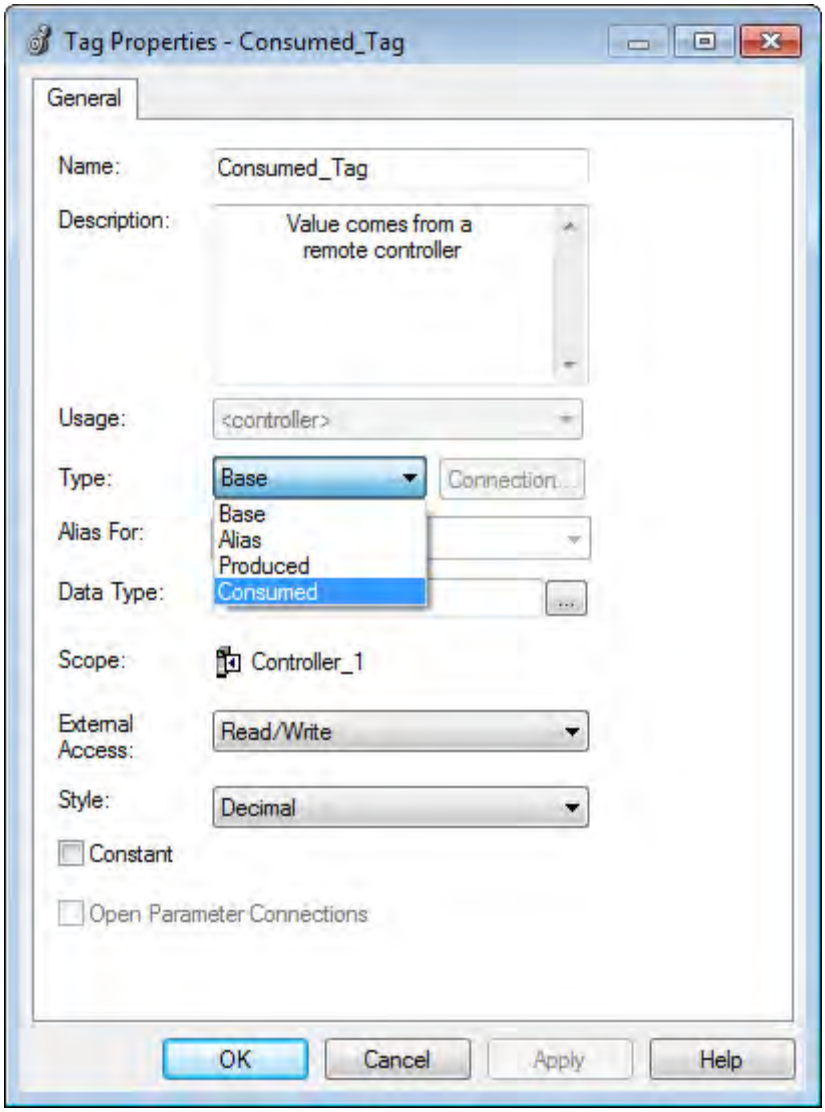
2. In the Controller Organizer I/O Configuration folder, add the controller that is producing the data (the other Logix5000 controller or PLC-5C controller).

3. In the **Controller Organizer**, right-click the **Controller Tags** folder and click **Edit Tags**.

You can produce only controller-scoped tags.



4. In the **Tag Editor** window, right-click the tag that will consume the data, and then choose **Edit (name of tag) Properties**.



5. In the **Name** box, type the name of the consumed tag.

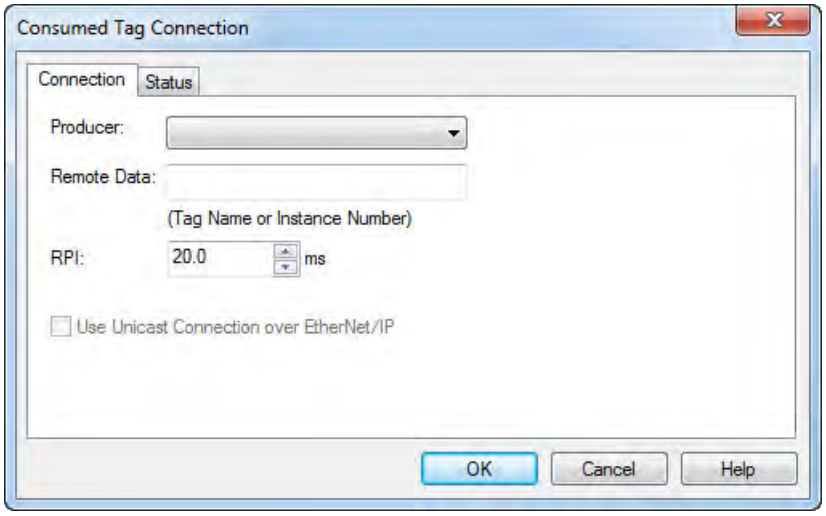
6. From the **Type** list, click **Consumed**.

The **Connection** button becomes active when either Consumed or Produced is chosen as the Type.

7. Specify the data type.

If the producing controller is	Then the data type should be	
Logix5000 controller	Same data type as the produced tag.	
PLC-5C controller	User-defined data type with these members.	
	Data type	**Description**
	DINT	Status
	INT[*x*], where *x* is the output size of the data from the PLC-5C controller. (If you are consuming only one INT, omit *x*.)	Data produced by a PLC-5C controller

8. Click **Connection** to define the consumed tag.



9. Do the following on the **Consumed Tag Connection** dialog box:

   - In the **Producer** list, choose the controller that produces the data.

   If the drop down is empty, you must first add a remote controller to the Controller Organizer I/O Configuration folder. This is the controller that is producing the data. If you have only one controller configured, it cannot be both the consumer and the producer. The producing controller cannot consume its own data.

   - In the **Remote Data** box, type the name or instance number of the remote, produced data.

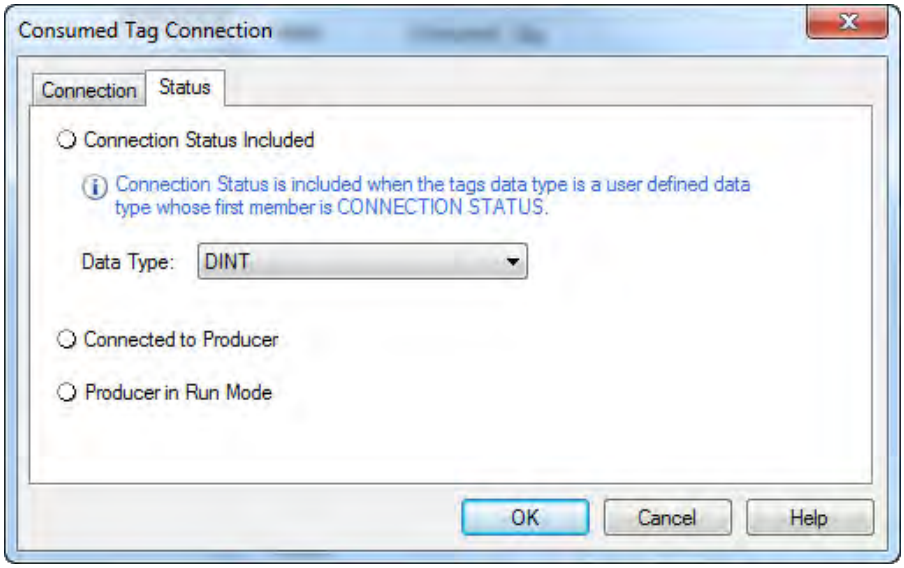
If the producing controller is	Then type or choose
Logix5000 controller	Tag name of the produced tag

If the producing controller is	Then type or choose
PLC-5C controller	Message number from the ControlNet configuration of the PLC-5C controller

- In the **RPI** field, type or select the requested packet interval (RPI) for the connection.

Important: Consuming controllers have additional setup to allow its consuming tags to use an RPI provided by a producing controller. See Set RPI Limits, Default for Producer Tag on page 24 for details.

10. Click the **Status** tab to configure the status properties for the consumed tags.



11. In the **Data Type** list, click the data type that allows the connection status.

12. Click **OK**.

Tip: If you consume the tag over a ControlNet network, use RSNetWorx for ControlNet software to schedule the network.

Additional steps for a PLC-5C controller

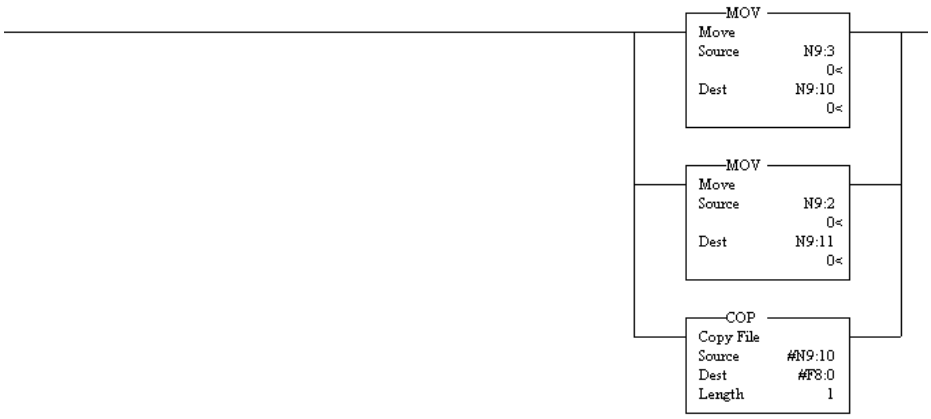
If you are sharing data with a PLC-5C controller, do the following.

Action	Details		
In the ControlNet configuration of the PLC-5C controller, schedule a message.	**If the PLC-5C controller**	**This**	**Then in RSNetWorx software**
	Produces	Integers	In the ControlNet configuration of the PLC-5C controller, insert a Send Scheduled Message.
	Consumes	Integers	Do these steps in the ControlNet configuration of the PLC-5C controller. Insert a Receive Scheduled Message. In the Message size, enter the number of integers in the produced tag.
		REALs	Do these steps in the ControlNet configuration of the PLC-5C controller. Insert a Receive Scheduled Message. In the Message size, enter two times the number of REALs in the produced tag. For example, if the produced tag contains 10 REALs, enter 20 for the Message size.
If the PLC-5C controller consumes REALs, reconstruct the values.	When you produce REALs (32-bit floating-point values) for a PLC-5C controller, the PLC-5C controller stores the data in consecutive 16-bit integers. • The first integer contains the upper (leftmost) bits of the value. • The second integer contains the lower (rightmost) bits of the value. This pattern continues for each floating-point value.		
If the PLC-5C controller consumes REALs, reconstruct the values.	When you produce REALs (32-bit floating-point values) for a PLC-5C controller, the PLC-5C controller stores the data in consecutive 16-bit integers. • The first integer contains the upper (leftmost) bits of the value. • The second integer contains the lower (rightmost) bits of the value. This pattern continues for each floating-point value.		

The following example shows how to reconstruct a REAL (floating-point value) in the PLC-5C controller.

Example: Reconstruct a floating point value.

The two MOV instructions reverse the order of the integers as the integers move to a new location. Because the destination of the COP instruction is a floating-point address, it takes two consecutive integers, for a total of 32 bits, and converts them to a single floating-point value.



RPI limitations and negotiated default

RPI limitations and negotiated behavior vary depending on the version of the Logix Designer application that is running on your controllers.

Behavior in releases v28 and later

In releases v28 and later, the first consumer of a produce tag determines the RPI at which data is produced. All subsequent consumers must request the same RPI value as the first consumer or they fail to connect and display error code 0112.

The first consumer of a produce tag is the device that sends the first consumer connection to the producing controller. The first consumer request is based on the order in which the producer and consumer control system powers up, so the first consumer can change if power is cycled to the system. You cannot configure a device to be the first consumer. There is no way to tell which consumer will send the first request to the producing controller, so you must plan accordingly when configuring multicast produce tags. See Considerations when migrating projects that contain multicast produce tags on page 13 for more information.

Tip: The controller's backplane and Ethernet ports can support different RPIs. If multiple consumers need to connect to the controller at different RPIs, you can configure them to connect through the backplane or the Ethernet port. See Connection requirements of a produced or consumed tag on page 12 for more information.

Behavior in releases v18 to v27

In versions 18 and earlier, controllers can be programmed to enhance the requested packet interval (RPI) acceptance between producing and consuming tags within specified limitations for multicast connections. A producing controller verifies that the RPI of incoming connections are within the produced tag settings. If the consuming tag's RPI falls outside the configured range, a producing

controller will reject the incoming RPI and may provide an RPI (default) to the consuming controller.

This optional default RPI applies only if the consuming controller is set up to allow its consumed tags to use an RPI provided by the producer. You can choose to not use the default value. However, you will have to manually change a rejected RPI to make the value within the limitation settings for the produced tag.

Tip:	1769-L2x and 1769-L3x controllers have additional RPI information. See 1769-L2x and 1769-L3x RPI Limits on page 33 for details.

If you have multicast consumers running Logix Designer releases v18 to v27, take the following steps to make sure consumers of produce tags can connect:

- Verify that all multicast consumed tags of a produce tag are configured with the same RPI, or
- Verify that all multicast consumers are configured to **Allow Consumed Tags To Use RPI Provided By Consumer**.

Behavior in releases v17 and earlier

When using releases v17 and earlier of the application, an incoming RPI of a consuming controller will be accepted by the producer. No default RPI is offered by the producing controller. The packet intervals of existing connections from other consumers could be sped up on multicast connections.

The RPI limitations (the range set on the produced tag) and the default process applies only to multicast connections on unscheduled networks. This feature is unavailable for controllers on the ControlNet network or unicast connections. Scheduled bandwidths on the ControlNet network transmit critical data at pre-determined intervals of time.

Important:	RPI limits and default values are not supported for safety tags and AXIS tags. AXIS produced tags are produced at a rate equal to the Coarse Update Period of the associated motion group. Safety produced tags are produced at a rate specified by the Safety Task Period.

If you have controllers running releases v17 or earlier that are consumers of a producer from a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller, take the following step to make sure consumers of produce tags can connect:

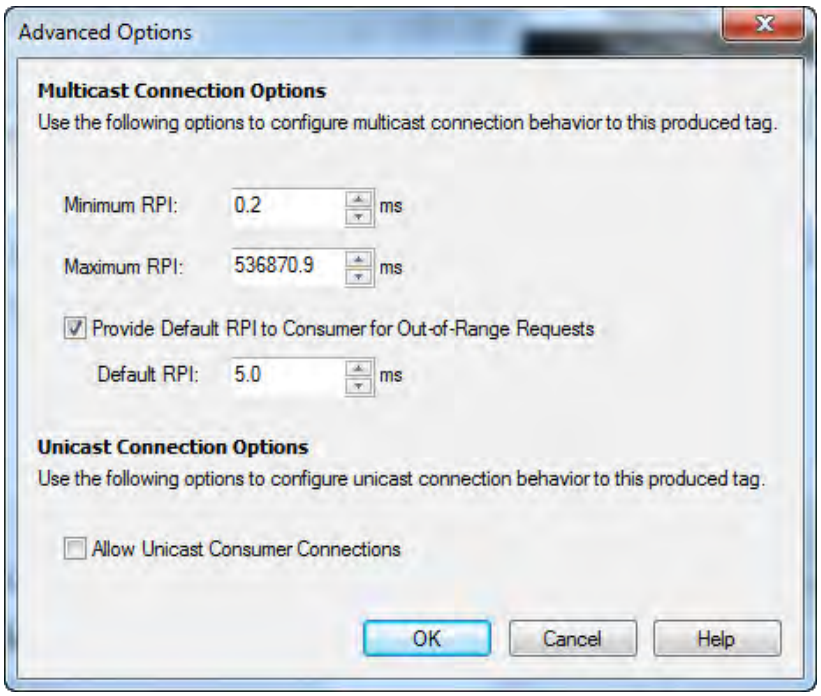
- Verify that all multicast consumed tags of a produce tag are configured with the same RPI. If they are not, some consumers will fail to connect.

Set RPI limits, default for producer tag

Follow these steps to establish RPI limitations (range) and default values if you are setting up a producing tag for multicast connections.

See Unicast Connection Options on page 26 if you are setting up a unicast connection.

1. On the **Produced Tag Connections** dialog box, click **Advanced**.



Important:	A unicast connection between a consumer and producer controller is the default on the **Advanced Options** dialog check box. Multicast connections can be changed to a produced tag whether or not the box is checked. The check box dictates only whether this produced tag can accept a unicast connection. The consuming controller determines if the connection is unicast.

2. Complete the **Advanced Options** dialog box to set up the multicast connection.

   For CompactLogix 5380, CompactLogix 5480, and ControlLogix 5580 controllers, the first consumer of a produce tag determines the RPI at which data is produced. All subsequent consumers must request the same RPI value as the first consumer or they fail to connect. Take the following steps to make sure consumers of produce tags can connect:

   - For consumers running Logix Designer releases v17 and earlier that are consumers of a producer from a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller:
     - Verify that all multicast consumed tags of a produce tag are configured with the same RPI. If they are not, some consumers will fail to connect.

- For multicast consumers running Logix Designer releases v18 to v27:
  - Verify that all multicast consumed tags of a produce tag are configured with the same RPI, or
  - Verify that all multicast consumers are configured to **Allow Consumed Tags To Use RPI Provided By Consumer**.

The following table describes the connection parameters.

Parameter	Description
Minimum RPI	The smallest requested packet interval (fastest rate) at which consumers may consume data from the tag. For ControlLogix controllers, the minimum RPI is 0.2 ms. For CompactLogix controllers, the minimum RPI is 1.0 ms. The **Minimum RPI** value must be less than or equal to the **Maximum RPI** value. This field may be modified in offline mode.
Maximum RPI	The largest packet interval (slowest rate) at which consumers may consume data from the tag. Rockwell recommends using the default setting of **536870.9 ms**. The **Maximum RPI** value must be greater than or equal to the **Minimum RPI** value. This field may be modified in offline mode.
Provide Default RPI to Consumer for Out-of-Range Requests	Enables a producing tag to provide an RPI to a consuming tag whenever a consuming tag has an RPI outside the range of the producer's Minimum and Maximum RPI. A consumer that supports negotiations then connects with the Default RPI provided by the producer. This field may be modified in offline mode.
Default RPI	The value that the producer provides to the consumer when the consumer's RPI falls outside the range of the producer's Minimum and Maximum RPI. This value must be within the minimum and maximum RPI range of the producer. This field may be modified in offline mode and when the Provide Default RPI to Consumer for Out-of-Range Requests box is checked. **Important:** You must select **Allow Consumed Tags to Use RPI Provided by Producer** on the **Controller Properties dialog box - Advanced tab** to allow negotiations to use the RPI provided by the producer.
Allow Unicast Consumer Connections	Allows multiple Unicast consumers to consume from the produced tag. This check box is selected by default. This check box is enabled unless the safety controller is locked. **Tip:** Selecting **Allow Unicast Consumer Connections** results in additional producer resources being allocated for each potential consumer. You should clear the check box if you are not using Unicast and you are running low on controller memory. This field may be modified in offline mode.

3. Click **OK**. When the **Produced Tag Connections** dialog box appears, click **OK**.

Important:	The RPI Limits configuration, which is disabled when online, must be within: • Minimum RPI: 0.2 ms for ControlLogix controllers and 1.0 ms for CompactLogix controllers. • Maximum RPI: 536870.911 ms. • Default RPI: configurable within the minimum/maximum range. • 1769-L2*x* and 1769-L3*x* controllers: see RPI Limits on page 22 for additional instructions. The RPI limits apply only to multicast connections on unscheduled networks. This functionality is not supported on the ControlNet network.

Unicast connection options

Unicast connections are point-to-point connections between a producing controller and consuming controller; no multiple connections. You do not have to enter a minimum or maximum RPI range or default value for this type of connection.

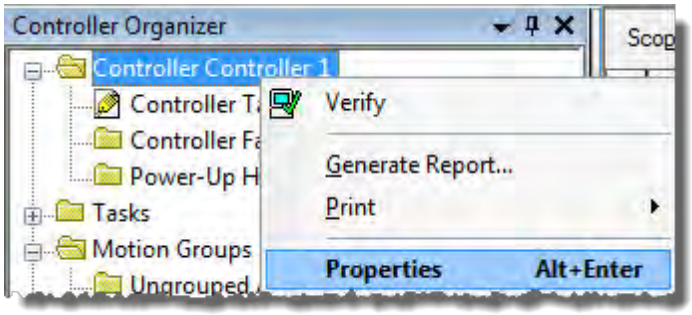
The **Allow Unicast Consumer Connections** check box at the bottom of the **Advanced Options** dialog box defaults with a check mark. Click **OK** to complete setting up the producing tag.

Set up the consumer tag

The Consuming controller must be set up to allow its consumed tags to accept an RPI provided by the producer. Follow these steps to access the **Controller Properties** dialog box.

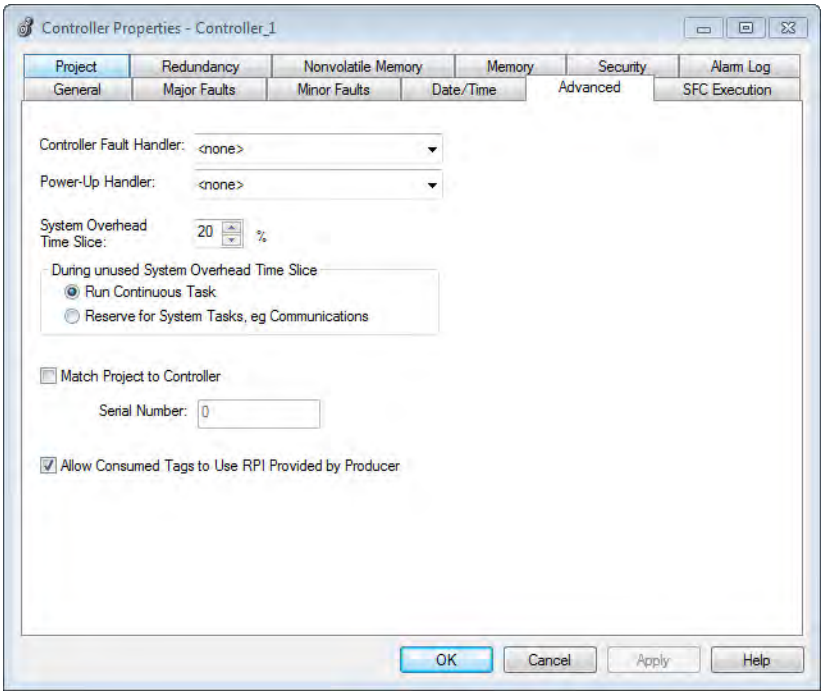
Important:	For controllers running Logix Designer releases v17 or earlier that contain consumers of a producer from a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller, verify that all multicast consumed tags of a produce tag are configured with the same RPI. If they are not, some consumers will fail to connect. See RPI limitations and negotiated default on page 22 for more information.

1. In the **Controller Organizer**, right-click a controller that has been set up with a consuming tag and choose **Properties**.



The **Controller Properties** dialog box appears with the General tab as the default.

2. Click the **Advanced** tab.



3. Check **Allow Consumed Tags to Use RPI Provided by Producer**.

Important:	The check box that activates **RPI Provided by Producer** is disabled online. This check box must be checked for the RPI negotiation to succeed. If the check box is not checked, the consuming controller does not accept the RPI provided by the producer if the consuming controller requests an RPI that is out of the producer's configured RPI range. As a result, an out-of-range RPI request results in a failed connection.

4. Click **OK**.

Verify consumed tag acceptance

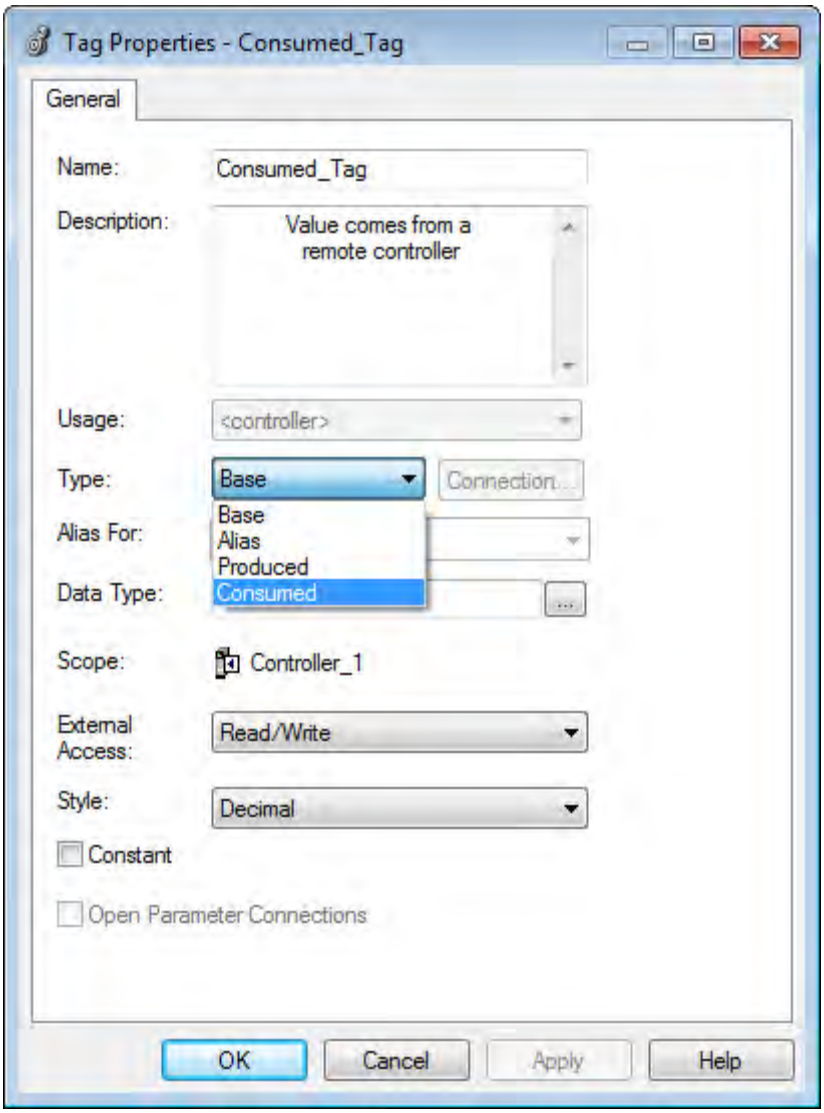
When a consuming controller accepts a default RPI by a producing controller, you can verify the acceptance and the RPI value on the **Consumed Tag Connection** dialog box. You must be online to complete the following procedures.

1. Open the Logix Designer application and choose a controller that is set up with a consuming tag.

2. In the **Controller Organizer**, right-click the **Controller Tags** folder and choose **Edit Tags**.

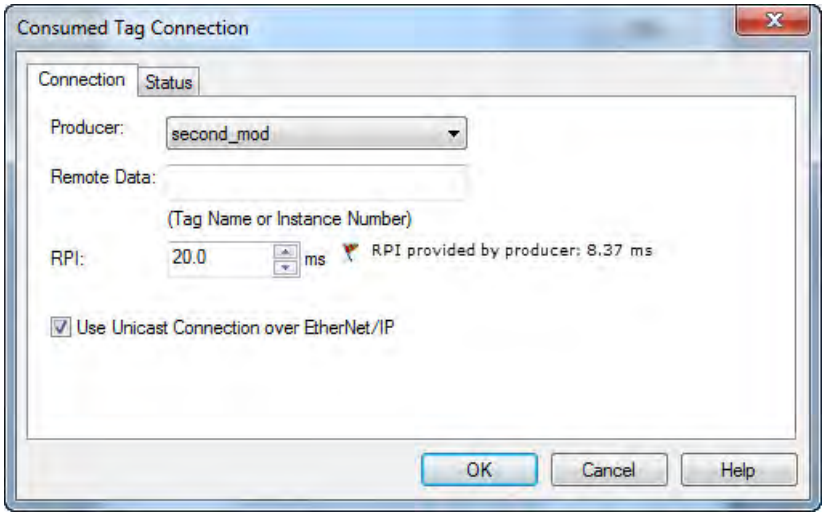
3. In the **Controller Tags** window, right-click a consuming tag and choose **Edit (name of tag) Properties**.

The **Tag Properties** dialog box appears.



4. Click **Connection**, at the right-hand side of the **Type** box.

The **Consumed Tag Connection** dialog box appears with a flag to the right of the **RPI** box to indicate the consuming controller has accepted an RPI provider by the producer. The time interval of the RPI also is indicated.



The RPI that had been requested by the consuming controller is dimmed. This lets you view the time interval the consuming controller had requested and the RPI provided by the producing controller.

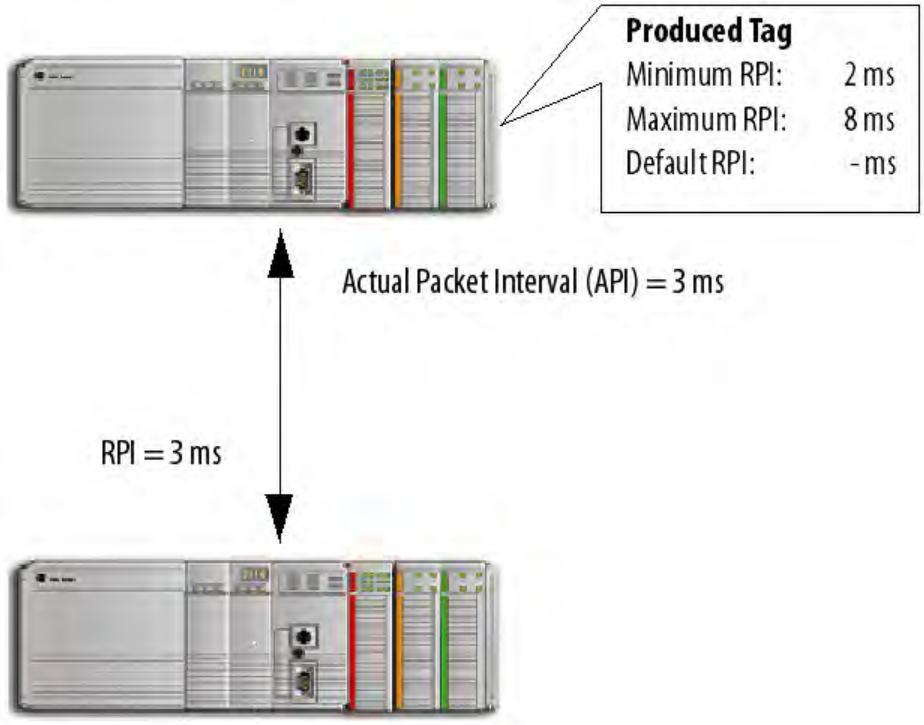
5. Click **Cancel** to exit.

Produced and consumed RPI scenarios

Scenario 1

The following scenarios explain how producing and consuming tags exchange RPI for controllers.

The RPI is within range of the producing controller's RPI Limits.
There is no default RPI that is set up for the producing controller.



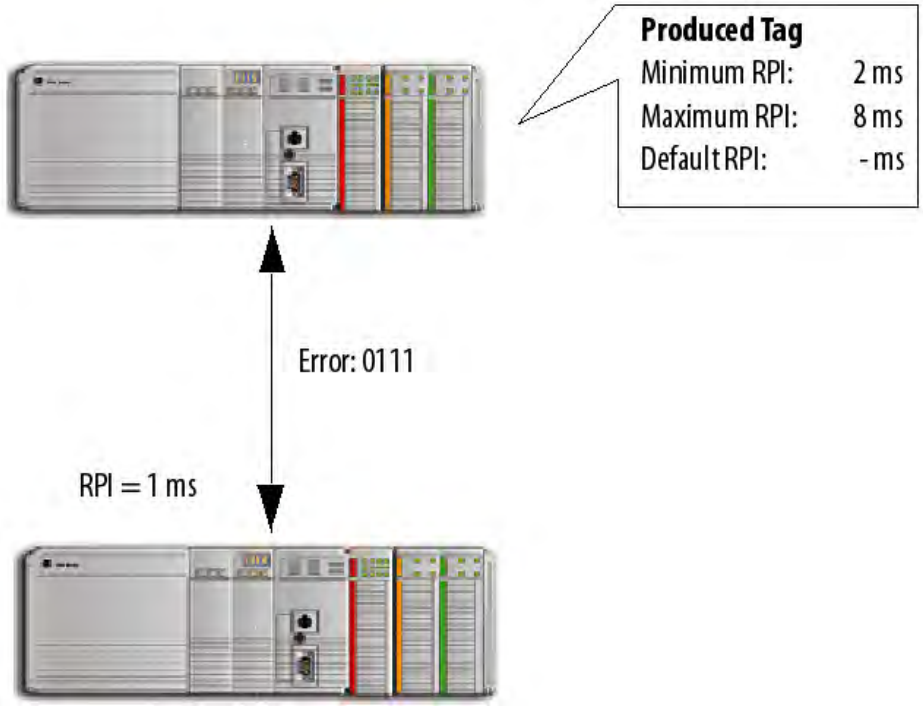
Version 17 and earlier: The consuming controller is brought online with an RPI configured at 3 ms. The producer accepts the RPI requested by the consuming tag. All controllers on the multicast connection will be increased to 3 ms for the API, which is the interval the data is actually being produced.

Version 18 and later: The consuming controller is brought online with an RPI configured at 3 ms. The producing controller verifies the requested RPI by the consumer is within the RPI limits set up for the producer. Since the request is within the range, the producing controller accepts the RPI of the consumer.

Tip: If you use all the default settings of version 18 and later, you have the same behavior as version 17 and earlier.

Scenario 2

The RPI is outside the range of the producing controller's RPI Limits. There is no default RPI that is set up for the producing controller.



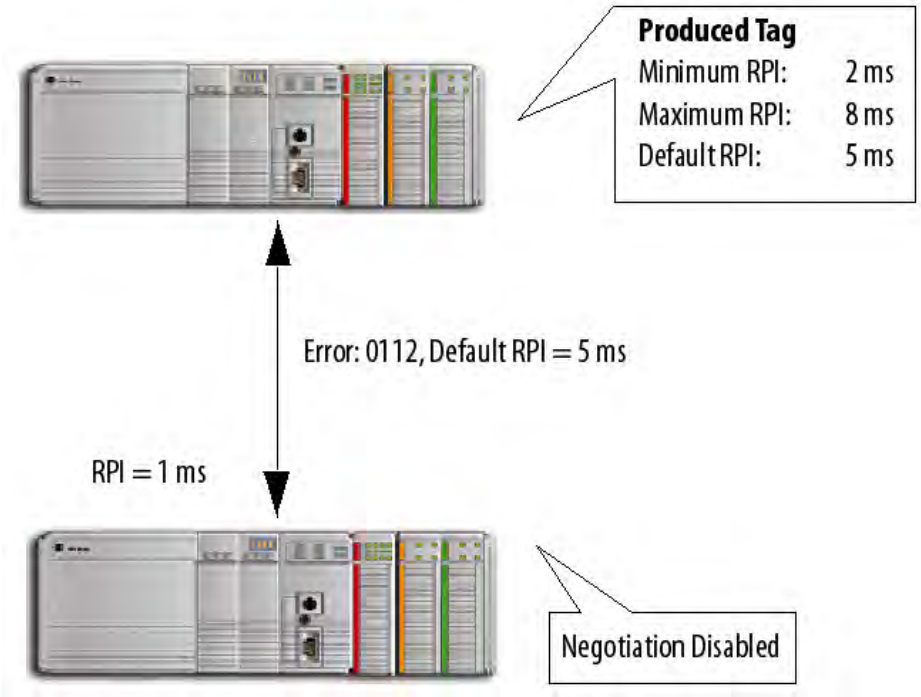
Version 17 and earlier: The RPI requested by the consumed tag is not within the range of the producer. The connection is rejected and a 0111 error is reported that the connection failed. You must re-enter an RPI that is within the range configured on the produced tag. If a version 17 consumer is trying to connect with a version 18 producer and the RPI is outside the range of the producer and there is no default set up, a 0111 error reports the connection failed.

Tip: See RPI I/O Faults on page 35 for an explanation of the error messages.

Version 18 and later: The RPI requested by the consumed tag is not within the RPI limitation range of the producer. The producer is not set up to provide a default RPI, so a 0111 error message is reported that the connection failed. You must re-enter an RPI that is within the range configured for the producer because there is no RPI default negotiation with the consumer.

Scenario 3

The RPI is outside the range of the producing controller's RPI Limits. A default RPI is set up for the producing controller. But, the network path contains a bridge that blocks the negotiated RPI to prevent an interruption to other controllers on the network.



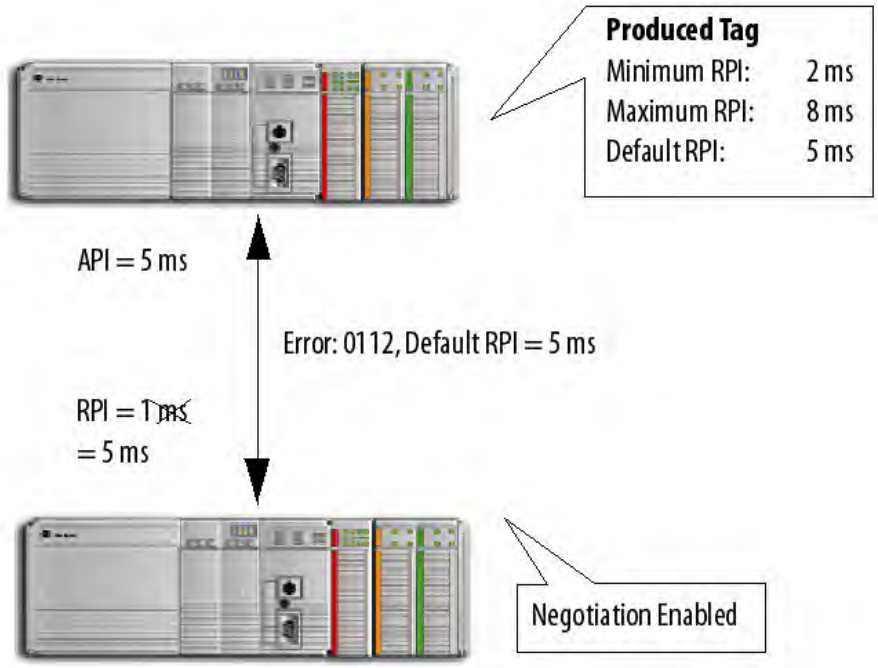
Version 17 and earlier: The RPI requested by the consumed tag is not within the range of the producer. The connection is rejected and a 0112 error is reported that the connection failed. You must re-enter an RPI that is within the range configured on the produced tag. If a version 17 consumer is trying to connect with a version 18 producer and the RPI is outside the range of the producer, a 0112 error reports the connection failed.

Tip: See RPI I/O Faults on page 35 for an explanation of the error messages.

Version 18 and later: The RPI requested by the consumed tag is not within the RPI limitation range of the producer. The producer is configured to provide an RPI default. The producer sends the default RPI to the consumer, but the consumer controller is not set up to accept a default RPI from the producer. The negotiated RPI is disabled to prevent an interruption to the multicast connection. A 0112 error message is reported and you must re-enter an RPI within the range of the producer's limits.

Scenario 4

The RPI is outside the range of the producing controller's RPI Limits. A default RPI is set up for the producing controller. The consumer is set up to accept the RPI provided by the producer.



Version 17 and earlier: The RPI requested by the consumed tag is not within the range of the producer. The connection is rejected and a 0112 error is reported that the connection failed. You must re-enter an RPI that is within the range configured on the produced tag. If a version 17 consumer is trying to connect with a version 18 producer and the RPI is outside the range of the producer, a 0112 error reports the connection failed.

Version 18 and later: The RPI requested by the consumed tag is not within the RPI limitation range of the producer. The producer is configured to provide an RPI default. The producer sends the default RPI to the consumer. The consumer is set up to accept the RPI provided by the producer. The connection is successful and the multicast connection accepts the new RPI of 5 ms.

1769-L2x and 1769-L3x RPI limits

All controllers, with the exception of the CompactLogix 1769-L2*x* and 1769-L3*x* controllers, produce data in packet intervals ranging from 0.196...536870.911 ms. The 1769-L2*x* and 1769-L3*x* controllers produce only in packet intervals of 2, 4, 8, 16, 32, 64, 128, 256, 512, or 1024 ms.

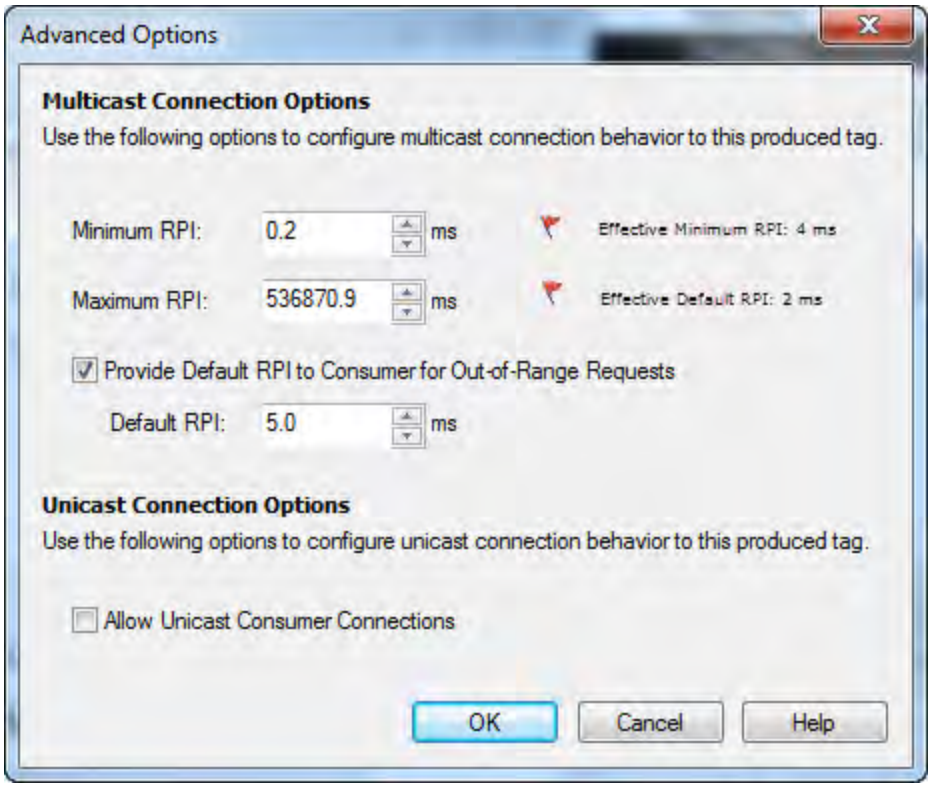
You are not confined to entering these exact values when configuring an RPI limitation range and default RPI for the 1769-L2*x* and 1769-L3*x* controllers. However, if an unsupported packet interval is entered, then the Logix Designer application may override the unsupported packet interval with a supported packet interval. When this occurs, the new 'effective' value is displayed separately next to the unsupported value that was entered.

Similar to other controllers, the 1769-L2*x* and 1769-L3*x* controllers will verify that the RPI of incoming connections are within the produced tag settings. If the consuming tag's RPI falls outside the configured range, a producing controller will reject the incoming RPI and then provide an RPI (default) to the consuming controller.

Important:	The producing and consuming controllers must be set up to allow the consumed tags to use an RPI provided by the producer. See RPI Limitations and Negotiated Default on page 22 and Set Up the Consumer Tag on page 26 for procedures to set up the producer and consumer tags to accept a negotiated default RPI.

For the 1769-L2*x* and 1769-L3*x* controllers, the **Effective Minimum RPI**, when present, is used to determine the fastest packet interval allowed by the tag. When the **Effective Default RPI** is present, it is the largest packet interval (slowest rate) at which negotiated connections will be produced for the tag.

The effective RPI limit values are presented with a flag on the **Advanced Options** dialog box to indicate that RPI limits are set to values other than what you entered.



RPI I/O faults

The following table provides a description of error messages for situations where an RPI is not accepted.

Scenario Description	Software Version	I/O Fault Message
RPI of consumed tag is not within the range of the producer. Producer does not support RPI negotiation or is not configured to provide a default RPI to the consumer.	Any	(Code 0111) Requested Packet Interval (RPI) out of range.
RPI of consumed tag is not within the range of the producer. Producer is returning default RPI. Consumer does not support negotiation or is not configured to accept RPI from producer.	18 and later	(Code 0112) Requested Packet Interval (RPI) out of range.
	17 and earlier	(Code 0112) Unknown Error.
RPI of consumer is out of range, but producer provides a default RPI and consumer can accept default RPI. Network path contains bridge that does not support extended error information. For I/O connectivity between two controllers, such as consuming a tag from a producing controller, both controllers must be attached to the same EtherNet/IP subnet. Two controllers cannot bridge produced or consumed tags over two subnets. For more information on setting up your network, refer to the following publication in the Rockwell Literature Library: *EtherNet/IP Network Configuration User Manual* (ENET-UM001)	18 and later	(Code 0112) Requested Packet Interval (RPI) out of range. Producer provided RPI blocked by the network path.
RPI of consumer is out of range, but producer provides a default RPI and consumer can accept RPI provided by producer. Network path contains scheduled network update time (NUT).	18 and later	(Code 0112) Requested Packet Interval (RPI) out of range. Cannot use producer provided RPI over a path containing a scheduled network.

Prevent connection failures with CompactLogix 5380, CompactLogix 5480, and ControlLogix 5580 controllers

If a consumer displays error code 0112 and fails to connect to a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller that provides produce tags, take the following steps to make sure consumers of produce tags can connect:

- For consumers running Logix Designer releases v17 and earlier that are consumers of a producer from a CompactLogix 5380, CompactLogix 5480, or ControlLogix 5580 controller:
  - Verify that all multicast consumed tags of a produce tag are configured with the same RPI. If they are not, some consumers will fail to connect.
- For multicast consumers running Logix Designer releases v18 to v27, take one of these steps:
  - Verify that all multicast consumed tags of a produce tag are configured with the same RPI, or
  - Verify that all multicast consumers are configured to Allow Consumed Tags To Use RPI Provided By Consumer.

Produce a Large Array

Introduction

The Logix5000 controller can send as many as 500 bytes of data over a single scheduled connection. This corresponds to 125 DINT or REAL elements of an array. To transfer an array of more than 125 DINTs or REALs, use a produced/consumed tag of 125 elements to create a packet of data. You can then use the packet to send the array piecemeal to another controller.

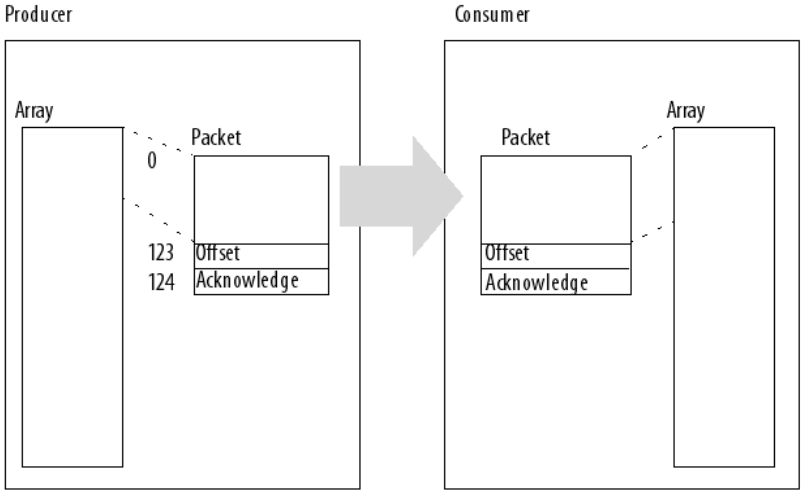
When you send a large array of data in smaller packets, you must be sure that the transmission of a packet is complete before the data is moved into the destination array.

- Produced data over the ControlLogix backplane is sent in 50 byte segments.
- Data transmission occurs asynchronous to a program scan.

The logic that this section includes uses an acknowledge word to make sure that each packet contains new data before the data moves to the destination array. The logic also uses an offset value to indicate the starting element of the packet within the array.

Because of the offset and acknowledge elements, each packet carries 123 elements of data from the array, as shown in the following.

Figure 1 - Producer/Consumer Data Elements



In addition, the array must contain an extra 122 elements. In other words, it must be 122 elements greater than the greatest number of elements that you want to transfer.

- These elements serve as a buffer.
- Because each packet contains the same number of elements, the buffer prevents the controller from copying beyond the boundaries of the array.
- Without the buffer, this would occur if the last packet contained fewer than 123 elements of actual data.

Produce a large array

Follow these steps to produce a large array.

1. In the **Controller Tags** folder of the controller project that produces the array, create these tags.

P	Tag Name	Type
	array_ack	DINT[2]
X	array_packet	DINT[125]

2. Convert array_ack to a consumed tag.

For	Specify
Controller	Name of the controller that is receiving the packet.
Remote Tag Name	array_ack Both controllers use the same name for this data.

3. In either the **Controller Tags** folder or the tags folder of the program that will contain the logic for the transfer, create these tags.

Tag Name	Type
array	DINT[x] where x equals the number of elements to transfer plus 122 elements
array_offset	DINT
array_size	DINT
array_transfer_time	DINT
array_transfer_time_max	DINT
array_transfer_timer	TIMER

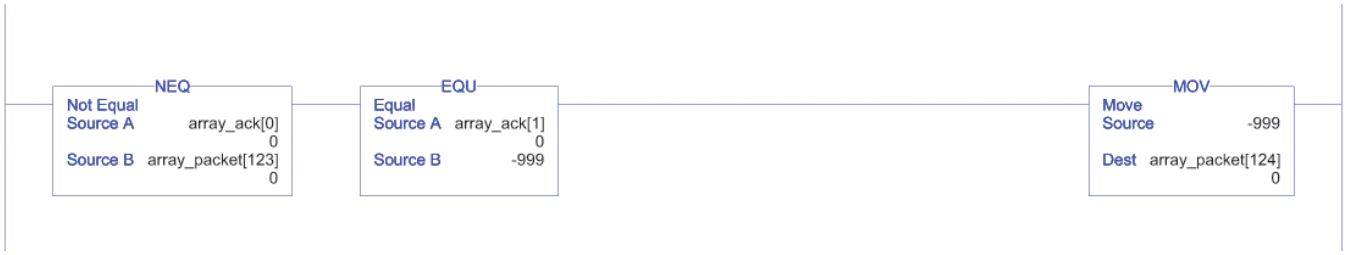
4. In the array_size tag, enter the number of elements of real data.

   The value of x from step 3 minus the 122 elements of buffer.

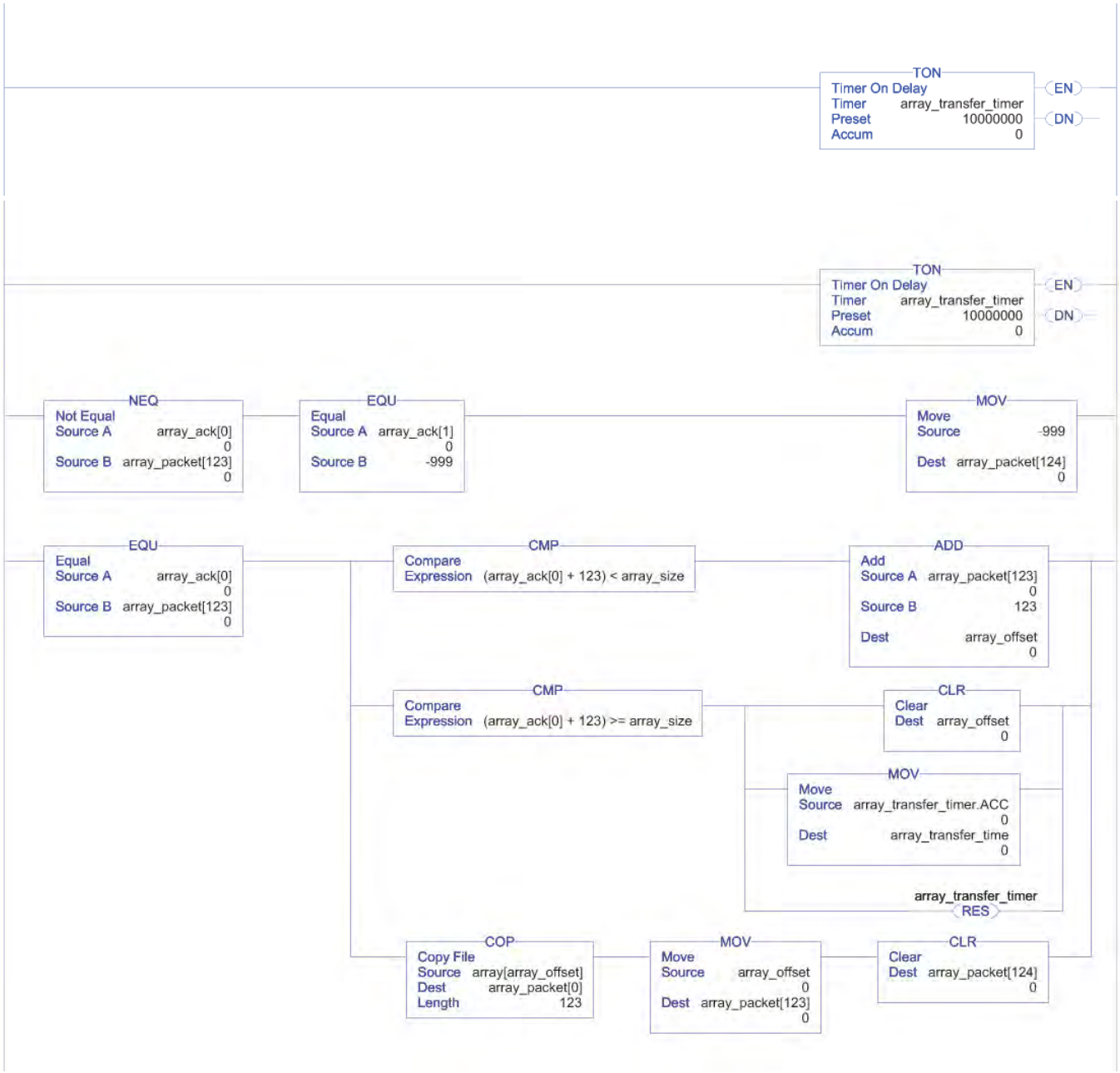
5. Create or open a routine for the logic that creates packets of data.

6. Enter this logic.

When the offset value in array_ack[0] is not equal to the current offset value but array_ack[1] equals -999, the consumer has begun to receive a new packet, so the rung moves -999 into the last element of the packet. The consumer waits until it receives the value -999 before it copies the packet to the array. This guarantees that the consumer has new data.



When the offset value in array_ack[0] is equal to the current offset value, the consumer has copied the packet to the array; so the rung checks for more data to transfer. If the offset value plus 123 is less than the size of the array, there is more data to transfer; so the rung increases the offset by 123. Otherwise, there is no more data to transfer; so the rung resets the offset value, logs the transfer time, and resets the timer. In either case, the rung uses the new offset value to create a new packet of data, appends the new offset value to the packet, and clears the acknowledge element of the packet (packet[124]).



If the current transfer time is greater than the maximum transfer time, update the maximum transfer time. This maintains a record of the longest time to transfer data.

7. In the **Controller Tags** folder of the controller project that consumes the array, create these tags.

P	Tag Name	Type
X	array_ack	DINT[2]
	array_packet	DINT[125]

8. Convert array_packet to a consumed tag.

For	Specify
Controller	Name of the controller that is sending the packet.
Remote tag name	array_packet Both controllers use the same name for this data.

9. In either the **Controller Tags** folder or the tags folder of the program that will contain the logic for the transfer, create these tags.

Tag Name	Type
array	DINT[x] where x equals the number of elements to transfer plus 122 elements
array_offset	DINT

10. Create or open a routine for the logic that will move the data from the packets to the destination array.

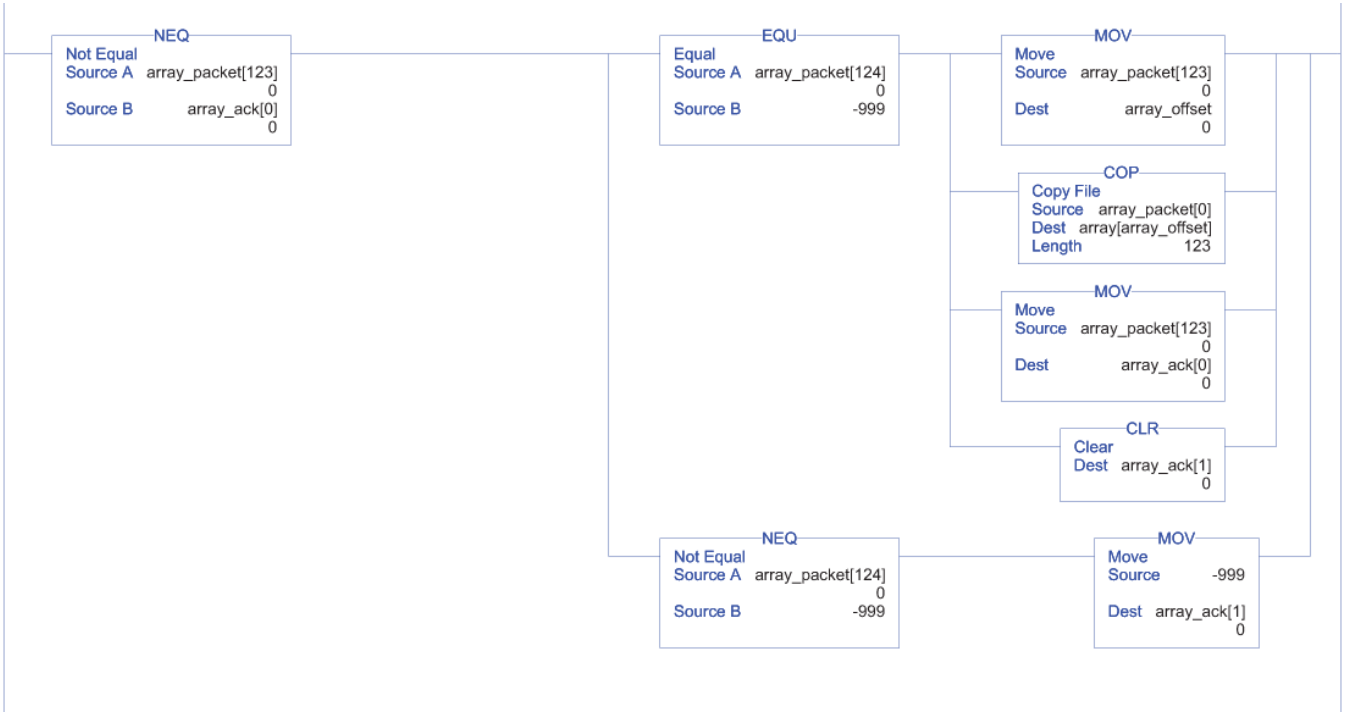
11. Enter this logic.

When the offset value in array_packet[123] is different than the offset value in array_ack[0], the controller has begun to receive a new packet of data; so the rung checks for the value of -999 in the last element of the packet.

If the last element of the packet equals -999, the controller has received an entire packet of new data and begins the copy operation.

- The offset value moves from the packet to array_offset.
- The COP instructions copy the data from the packet to the destination array, starting at the offset value.
- The offset value moves to array_ack[0], which signals that the copy is complete.
- Array_ack[1] resets to zero and waits to signal the arrival of a new packet.

If the last element of the packet is not equal to -999, the transfer of the packet to the controller may not be complete; so -999 moves to array_ack[1]. This signals the producer to return the value of -999 in the last element of the packet to verify the transmission of the packet.



Transferring a large array as smaller packets improves system performance over other methods of transferring the data.

- Fewer connections are used than if you broke the data into multiple arrays and sent each as a produced tag. For example, an array with 5000 elements would take 40 connections (5000/125=40) by using individual arrays.
- Faster transmission times are achieved than if you used a message instruction to send the entire array.

- Messages are unscheduled and are executed only during the 'system overhead' portion of the Logix5550 execution. Therefore, messages can take a fairly long time to complete the data transfer.
- You can improve the transfer time by increasing the system overhead time slice, but this diminishes the performance of the continuous task.

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-PM011I-EN-P - December 2016

Supercedes Publication 1756-PM011H-EN-P - June 2016