Logix 5000 Controllers Produced and Consumed Tags

1756 ControlLogix, 1756 GuardLogix, 1769 CompactLogix, 1769 Compact GuardLogix, 1789 SoftLogix, 5069 CompactLogix, Emulate 5570

Rockwell Automation

Programming Manual

Original Instructions

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.

These 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).

The following icon may appear in the text of this document.

Identifies information that is useful and can help to make a process easier to do or easier to understand.

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.

Summary of 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

Subject	Reason
Throughout	Removed references to the CompactLogix 5480 controller or indicated it is not supported in Logix Designer application versions 38 and later.

New or updated information

Subject	Reason
Studio 5000 environment on page 6	Updated the screenshot.
Organize tags for produce-consume data on page 12	Indicated additional information is available in publication 1756-RM094.

New or enhanced features

Subject	Reason
Throughout	Added references to the ControlLogix 5590 controller. The Logix Designer application versions 38 and later support it.

Preface

This manual details how, with a Logix 5000™ controller, to produce and consume standard tags and produce a large array. For information on produced and consumed safety tags, refer to the documentation specific to the controller.

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
Replacement Guidelines: Logix 5000 Controllers Reference Manual, publication 1756-RM100	Provides guidelines for migrating projects from ControlLogix 5560 and 5570 controllers to ControlLogix 5580 controllers, and from CompactLogix 5370 controllers to CompactLogix 5380 controllers.
ControlLogix 5580 and GuardLogix 5580 Controllers User Manual, publication 1756-UM543	Provides information about designing a system, operating a ControlLogix or GuardLogix-based 5580 controllers system, and developing applications.
CompactLogix 5380 and Compact GuardLogix 5380 Controllers User Manual, publication 5069-UM001	Explains how to use the CompactLogix and Compact GuardLogix 5380 Controllers.

Resource	Description
GuardLogix 5570 Controllers User Manual, publication 1756-UM022	Describes the GuardLogix-specific procedures to configure, operate, and troubleshoot the controller.
Compact GuardLogix 5370 Controllers User Manual, publication 1769-UM022	Describes the necessary tasks to install, configure, program, and operate a Compact GuardLogix 5370 controller.
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.

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.

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.

Software and Cloud Services Agreement

Review the Rockwell Automation Software and Cloud Services Agreement here.

Open Source Software Licenses

The software included in this product contains copyrighted software that is licensed under one or more open source licenses.

You can view a full list of all open source software used in this product and their corresponding licenses at this URL:

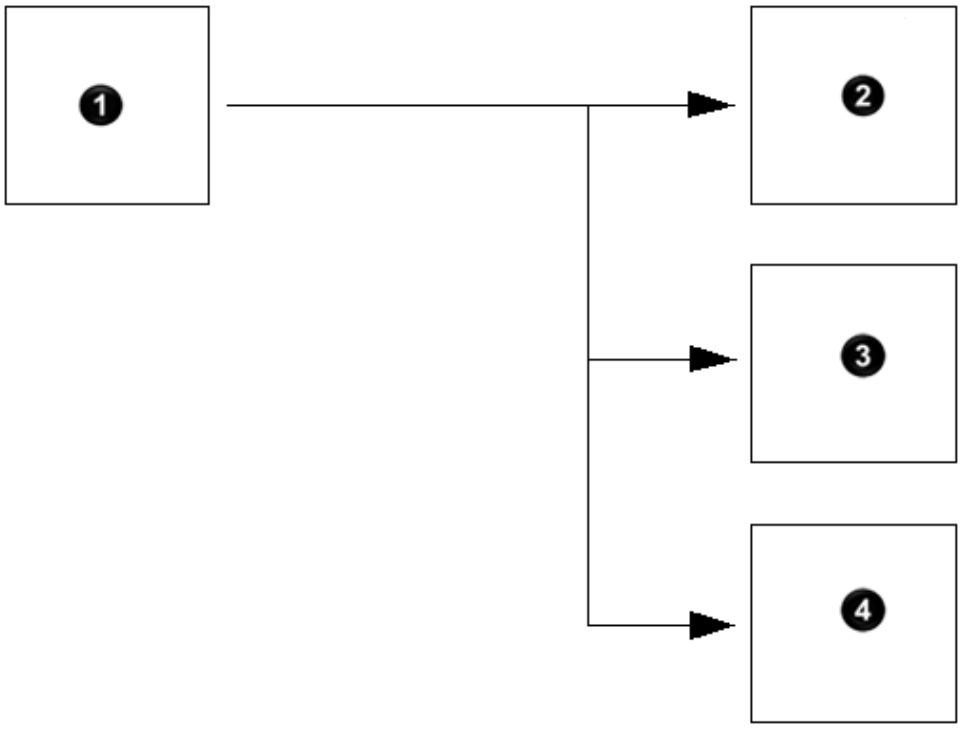
Studio 5000 Logix Designer Open Source Attribution List

You may obtain Corresponding Source code for open source packages included in this product from their respective project web site(s). Alternatively, you may 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.

Produce and consume a tag

Produced/consumed tags

A Logix 5000 controller produces (broadcast) and consumes (receive) system-shared tags.



Number	Description
1	**Controller_1** Produced tag
2	**Controller_2** Consumed tag
3	**Controller_3** Consumed tag
4	**Controller_4** Consumed tag

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

Term	Definition
	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.

Produced-consumed tag considerations

Keep these considerations in mind for produced and consumed tags.

- The tags must be controller-scoped tags.
- The data types must match, including user-defined data type layout.
- The size of the produced and consumed standard tag is limited to less than or equal to 500 bytes.
- A produced/consumed tag can only be created when the controller is offline.
- Connection status information can be included with the produced tag in a standard controller. In order to do this, create a user defined type whose first member is a CONNECTION_STATUS type. Then, the remaining members in the user-defined data type contain the data to be produced.

For produced and consumed safety tags, these requirements are mandatory:

- The size of the produced and consumed safety tags are limited to 128 bytes.
- Connection status information must be included with produced and consumed safety tags. Create a user-defined data type whose first member is a CONNECTION_STATUS type. The remaining members in the user-defined data type contain the data to be produced.
- The Class field must be set to Safety.
- Safety tags can only be modified when the safety application is not in a safety secured state.
- Compact GuardLogix 5380 and GuardLogix 5580 Safety controllers can only produce safety tags as unicast. These controllers can consume safety tags as either unicast or multicast.

Controller and network support

For two controllers to share produced or consumed tags, both controllers must be in the same backplane or attached to the same control network, such as ControlNet or EtherNet/IP network. Although produced and consumed tags can be bridged over two networks, Rockwell Automation® does not support this configuration. Check the documentation specific to your controller to determine which network connections it supports.

Controllers in same chassis

If the controller firmware revision is earlier than revision 9.00, axes in pre-revision 9.00 cannot be consumed in a revision 9.00 or later controller because the axis structure changed in revision 9.00. For example, an axis produced in revision 7.00 cannot be consumed in revision 11.00.

If the firmware revision is version 9.00 or earlier, the revision of the controller for the consuming axis must be the same revision or earlier. For example, a revision 11.00 controller can consume an axis from revision 11.00 or revision 10.00, but it is not possible for a revision 10.00 controller to consume an axis from revision 11.00 or revision 12.00. It is not possible in Studio 5000 Logix Designer® to add a new controller under the I/O configuration with a revision later then the revision that is currently in use.

Separate chassis using SynchLink

When using SynchLink to connect two ControlLogix racks, the produced and consumed axis revision is no longer important, if SynchLink is supported. For example, a revision 11.00 controller produces an axis transmitted from a 1756-SYNCH module. In a different rack, another 1756-SYNCH module receives the axis data. A controller running revision 10.00can then consume this data. In this case, the revision of the consuming end is later then the revision of the producing end. This case is not possible when the controllers are in the same rack. This caseis possible when the controllers are in separate racks using SynchLink to connect the racks.

Connection requirements

Produced and consumed tags each require connections. As the number of controllers that consume a produced tag increases, the number of connections the controller has available for other operations, such as communication and I/O, decreases.

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

Each produced or consumed tag uses these 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 Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, and GuardLogix 5580 controllers

If a controller running release version 24.00 or earlier intends to consume tag data from Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, or GuardLogix 5580 controller, the controller must have a target device in its I/O configuration for the connection. However, releases version 24.00 and earlier do not have profiles for Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, and GuardLogix 5580 controllers and requires a substitute profile.

Rockwell Automation recommends the following substitutes:

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

Migrate projects multicast produce

When migrating projects from an earlier controller model to Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480 (only supported in the Logix Designer application versions 37.00 and earlier), ControlLogix 5580, GuardLogix 5580, or ControlLogix 5590 controller models, verify that the configuration of projects containing multicast produce tags is correct.

Compact GuardLogix 5380 and GuardLogix 5580 controllers in version 31.00 only produce safety tags as unicast

- Compact GuardLogix 5380 and GuardLogix 5580 controllers only produce safety tags as unicast. The controllers consume safety tags as either unicast or multicast. When configuring a consumed safety tag, consider the capabilities of the producer.
  - If the producer in the I/O configuration of this controller is a Compact GuardLogix 5380 or GuardLogix 5580 controller consuming a safety tag, configure the consumed tag to use unicast.
  - If the producer in the I/O configuration of this controller is a Compact GuardLogix 5370, GuardLogix 5570, or GuardLogix 5560 controller, configure the safety consumed tag as either unicast or multicast.

RPI limitations

In releases before version 28.00, produced tags produce data at the requested packet interval (RPI) of the fastest requesting consumer. This RPI allows multiple consumers, with different RPI settings, to connect successfully to a producer. In releases version 28.00 and later, the first consumer of a produced tag determines the RPI of produced data. All subsequent consumers must request the same RPI value as the first consumer or they fail to connect and display error code 0112.

The device that sends the first consumer connection to the producing controller is the first consumer of a produced tag. The first consumer makes a request based on the order in which

the producer and consumer control system powers up. Therefore, cycling power to the system can change the first consumer. A device cannot be configured to be the first consumer. It is difficult to identify which consumer sends the first request to the producing controller, so plan accordingly when configuring multicast produced tags.

To make sure consumers of produced tags connect to Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, ControlLogix 5590, and GuardLogix 5580 controllers:

- For consumers running releases versions 17.00 and earlier that are consumers of a produced tag from Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, and GuardLogix 5580 controllers:
  - Verify that all multicast consumed tags of a produced tag have the same RPI. If they do not, some consumers fail to connect.
- For multicast consumers running releases versions 18.00 and later:
  - Verify that all multicast consumed tags of a produce tag have the same RPI, or
  - Verify that all multicast consumers are configured to **Allow Consumed Tags To Use RPI Provided By Producer**.
  - ControlLogix 5590 controllers are supported in the Logix Designer application versions 38.00 and later.
  - CompactLogix 5480 controllers are not supported in the Logix Designer application versions 38.00 and later.

For more information on replacing a ControlLogix 5560/5570 controller with a ControlLogix 5580 or ControlLogix 5590 controller, or a CompactLogix 5370 controller with a CompactLogix 5380 controller, refer to Replacement Guidelines: Logix5000 Controllers Reference Manual (1756-RM100) in the Rockwell Automation Literature Library.

Organize tags for produce-consume data

See the Logix5000 Controllers Design Considerations Reference Manual, publication 1756-RM094, for the following information:

- Guidelines to organize tags for produced or consumed data (shared data) in the Guidelines for Produced and Consumed Tags topic.
- Message instruction comparisons in the Compare Messages and Produced/Consumed Tags topic.

Bandwidth limitations adjustments

When a tag is shared 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 the system, there may not be enough bandwidth on the ControlNet network for a tag of 500 bytes. If a tag is too large for the ControlNet network, make one or more of these adjustments.

Adjustment	Description
Reduce NUT.	At a faster NUT, fewer connections need to share an update slot.

Adjustment	Description	
Increase the requested packet interval (RPI) of the 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, 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*.	

Set up produced and consumed axes

Set up axes to produce and consume axis data in the same chassis or between different chassis.

Prerequisite

- Configure produced and consumer controllers.

To set up produced and consumed axes

1. In the **Controller Organizer**, right-click the motion group and select **New Axis** and select the data tape for the drive.
2. If creating a new axis, in the **New Tag** dialog box, enter a name for the axis.
3. Set the **Type** to **Produced**.
   - If editing an existing tag, in the **Tag Editor**, right-click the tag and click **Edit <name of tag> Properties** and change the **Type** to **Produced**.
4. Make sure the axis **Data Type** to the corresponding data type of the produced axis.

5. Select **Connection** to open the **Produced Tag Connection** dialog box.
   - Change the **Max Consumers** value to limit or expand the number of consumers.
   - Select **Send Data State Change Event to Consumer** to use an IOT Instruction to send event trigger information to the consumers of this tag.
   - Select the **Advanced** tab and to open the **Advanced Options** dialog box and configure the behavior of the multicast and unicast connections to the produced tag, for example, limiting the intervals at which data is produced over a multicast connection.

   IMPORTANT: The minimum and maximum RPI is disabled for axis types. The produced axis produced data at the axis rate. The minimum and maximum RPI is equal to the axis update.
6. Add the producer controller to the I/O configuration of the consumer controller.
7. Create a new consumed axis
   - In the **Controller Organizer**, right-click the motion group and select **New Axis** and select the data tape for the drive.
   - If updating an existing axis, right-click **Controller Tags** and select **New Tag.**
8. If creating a new axis, enter a name for the axis.
9. Set the type to **Consumed** to open the **Consumed Tag Connection** dialog box.
   - On the **Connection tab**, in the **Producer** list, select the controller that has the Produced Axis.
   - In **Remote Data**, enter the name of the produced axis.
   - Specify unicast/multicast connection over Ethernet/IP.
10. In the **Controller Organizer**, expand the **Motion Groups** folder and double click the axis. Adjust axis properties for the application.
    - In the **Units** tab to adjust the **Position Units** value.
    - In the **Conversion** tab, update the **Conversation Constant** value.

Produced and consumed axis example

This example requires these prerequisites:

- ControlLogix® and CompactLogix™ controllers must be from the 5380, 5480, or 5580 family of controllers
- Studio 5000 Logix Designer® v32 or higher
- The Logix controls must be on the same Ethernet/IP.

Assume a machine builder delivered a new filler machine to a customer. The machine builder is also upgrading the capper machine on the same manufacturing line.

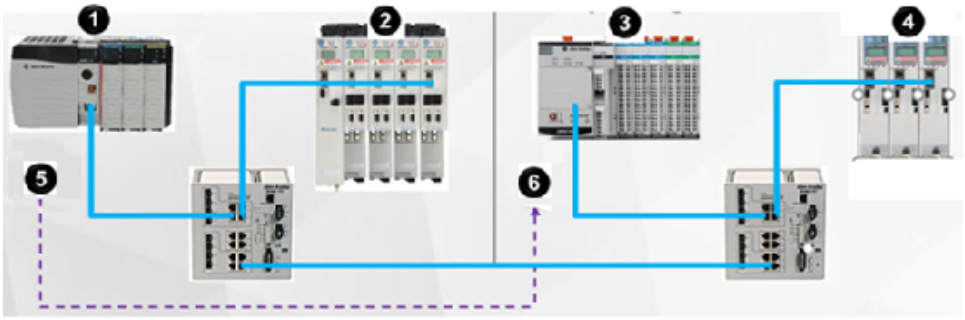
The filler machine is the lead machine on the line:

- Bottles flow from the filler into the capper machine
- The filler machine uses a ControlLogix controller and Kinetix servo drives

The capper machine must follow the output of the filler machine:

- Once synchronized to the filler, the capper must stay in sync
- This machine will use a CompactLogix controller and Kinetix® servo drives

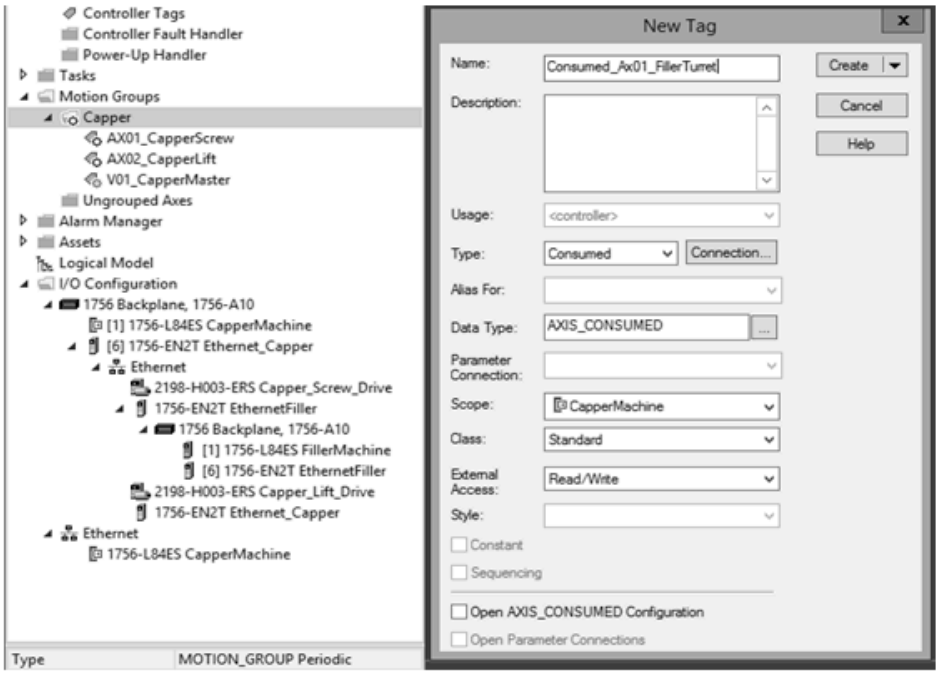
The filler machine produces an axis of motion for the capper motion to consume. This enables the capper and filler machine to maintain coordination.

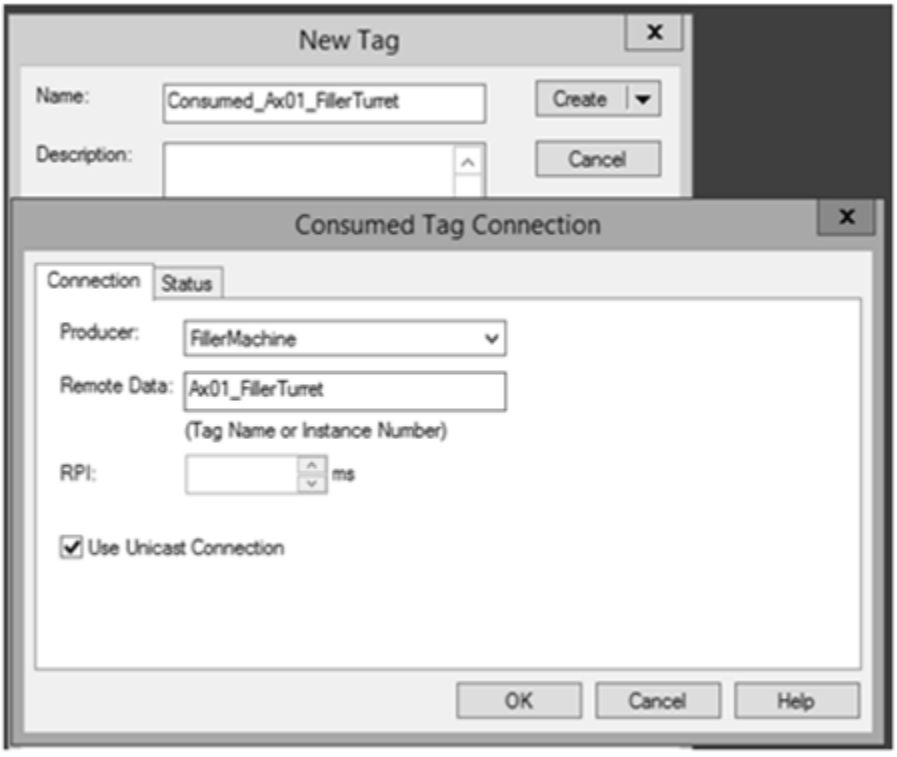


Item	Description
1	ControlLogix 5580
2	Kinetix 5700
3	CompactLogix 5380
4	Kinetix 5500
5	Produced axis
6	Consumed axis

Produce-consume axis config example

This example illustrates how to configure a produced and consumed axis.





Produce and consume program example

This programming example shows syncing the filler (produced) to the capper (consumed) axis.

Create a produced tag

The produced tag sends its data to one or more consumed tags without using logic. Multiple controllers are capable of using the produced tags simultaneously.

Prerequisite

- Create a user-defined data type defining the structure of the produced data. The first data member must be the predefined CONNECTION_STATUS data type.

To create a produced tag

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

   IMPORTANT: A controller is capable of having producing and consuming tags. A produced cannot consume its own data. The local controller is the consumer. The remote controller is the producer.

2. In the **Controller Organizer**, right-click **Controller Tags** and select **New Tag**. Only controller-scoped tags can be shared.
3. On the **New Tag** dialog box, in **Name**, type the name of the tag.
4. In **Type**, select **Produced**.
5. In **Data Type**, select the data type that includes the connection status.
6. If the tag is a produced safety tag, in **Class**, select **Safety**.
7. Select **Connection** to define the produced tag. The **Produced Tag Connection** dialog box opens.
8. On the **Connection** tab, in **Max Consumers**, type the number of controllers to consume (receive) this tag.

IMPORTANT:

- Producing a tag requires a connection for each consumer. Connections are a limited resource in the controller, so only produce tags that are needed in other controllers.
- Unicast produced tags can only connect to one controller.

A controller is capable of having producing and consuming tags. A produced cannot consume its own data. The local controller is the consumer. The remote controller is the producer.

9. (optional) Select **Advanced** to change the RPI range limitation, negotiated default, or unicast connection.
10. Select **OK** to close the **Produced Tag Connection** dialog box.
11. Select **Create** to create the tag and close the **New Tag** dialog box.

Related information

Configure connection properties for produced tags on page 18
Organize tags for produced or consumed data on page
Considerations for produced and consumed tags on page

Configure connection properties for produced tags

Use the **Produced Tag Connection** dialog box to configure connection properties for produced tags:

- Specify the maximum number of consumers.
- Send event triggers to consumers, using an IOT instruction.

To configure connection properties for produced tags

1. In **Max Consumers**, specify the maximum number of tags that can consume this tag. The valid range for maximum number of tags is 1 to 255.
2. Select **Send Data State Change Event to Consumer** to use an IOT instruction to send event trigger information to the consumers of this tag. The check box is selected and disabled for Safety controllers.
3. Select **Advanced** to open the **Advanced Options** dialog box and configure the behavior of the multicast and unicast connections to the produced tag, for example, limiting the intervals at which data is produced over a multicast connection.

Related information

Set Advanced Options dialog box parameters on page 26
Produce and consume array types on page 48
Set an RPI Range for Produced Tags on page 28

Create a produced tag

The produced tag sends its data to one or more consumed tags without using logic. Multiple controllers are capable of using the produced tags simultaneously.

Prerequisite

- Create a user-defined data type defining the structure of the produced data. The first data member must be the predefined CONNECTION_STATUS data type.

To create a produced tag

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

   IMPORTANT: A controller is capable of having producing and consuming tags. A produced cannot consume its own data. The local controller is the consumer. The remote controller is the producer.

2. In the **Controller Organizer**, right-click **Controller Tags** and select **New Tag**. Only controller-scoped tags can be shared.
3. On the **New Tag** dialog box, in **Name**, type the name of the tag.
4. In **Type**, select **Produced**.
5. In **Data Type**, select the data type that includes the connection status.
6. If the tag is a produced safety tag, in **Class**, select **Safety**.
7. Select **Connection** to define the produced tag. The **Produced Tag Connection** dialog box opens.
8. On the **Connection** tab, in **Max Consumers**, type the number of controllers to consume (receive) this tag.

   IMPORTANT:
   - Producing a tag requires a connection for each consumer. Connections are a limited resource in the controller, so only produce tags that are needed in other controllers.
   - Unicast produced tags can only connect to one controller.

   A controller is capable of having producing and consuming tags. A produced cannot consume its own data. The local controller is the consumer. The remote controller is the producer.

9. (optional) Select **Advanced** to change the RPI range limitation, negotiated default, or unicast connection.
10. Select **OK** to close the **Produced Tag Connection** dialog box.
11. Select **Create** to create the tag and close the **New Tag** dialog box.

Related information

Configure connection properties for produced tags on page 18
Organize tags for produced or consumed data on page
Considerations for produced and consumed tags on page

Configure connection properties for produced tags

Use the **Produced Tag Connection** dialog box to configure connection properties for produced tags:

- Specify the maximum number of consumers.
- Send event triggers to consumers, using an IOT instruction.

To configure connection properties for produced tags

1. In **Max Consumers**, specify the maximum number of tags that can consume this tag. The valid range for maximum number of tags is 1 to 255.
2. Select **Send Data State Change Event to Consumer** to use an IOT instruction to send event trigger information to the consumers of this tag. The check box is selected and disabled for Safety controllers.
3. Select **Advanced** to open the **Advanced Options** dialog box and configure the behavior of the multicast and unicast connections to the produced tag, for example, limiting the intervals at which data is produced over a multicast connection.

Related information

Set Advanced Options dialog box parameters on page 26
Produce and consume array types on page 48
Set an RPI Range for Produced Tags on page 28

Configure status properties for produced tags

Use the **Produced Tag Connection** dialog box **Status** tab to configure status properties for produced tags, including determining:

- The connection status included.
- The data type that allows for connection status.

To configure status properties for produced tags

1. Use **Connection Status Included** to determine the connection status.

   The LED is ON when the tag associated with the produced connection is a User-defined data type whose first member is CONNECTION STATUS.

   In Safety applications: tags associated with Safety I/O and produced or consumed Safety data must be controller-scoped Safety tags. For produced/consumed Safety tags, create a User-Defined data type with the first member of the tag structure (CONNECTION_STATUS) containing the status of the connection.

   In Standard applications: tags associated produced or consumed data must be controller-scoped tags. For produced/consumed tags, create a User-Defined data type, with the first member of the tag structure (CONNECTION_STATUS) containing the status of the connection.
2. Use the pull-down menu to pick a data type that allows for connection status.

   The pull-down box contains the current data type and a list of all data types whose first member is of type CONNECTION_STATUS. The data type chosen is displayed in the data type field the tag property dialog box or the Create Tag dialog box.

Create a consumed tag

The consumed tag receives data of a produced tag. Match the data type of the consumed tag to the data type (including any array dimensions) of the produced tag. The Requested Packet Interval (RPI) of the consumed tag determines the period that the data updates.

IMPORTANT: Set up only one consumed tag to get data from the same producing tag in another controller. Setting up more than one consumed tag results in unpredictable controller to controller behavior.

If a consumed tag connection faults, all other consumed tags from that remote controller also stop receiving data.

Prerequisite

- Add the controller that is producing the data to the **Controller Organizer I/O Configuration** folder.

To create a consumed tag

1. In the **Controller Organizer**, right-click **Controller Tags** and select **New Tag**. Only controller-scoped tags can be shared.
2. In the **New Tag** dialog box, in **Name**, type the name of the tag.
3. In **Type**, select **Consumed**.
4. In **Data Type**, select the data type that includes the connection status. Make the first member of the user-defined data type CONNECTION_STATUS type.
5. If the consumed tag is a consumed safety tab, in **Class**, select **Safety**.
6. Select **Connection** to define the consumed tag. The **Consumed Tag Connection** dialog box opens.
7. On the **Connection** tab, in **Producer**, select the controller that produces the data.

   The available controllers depend on if the tag is a Safety Tag or Standard Tag.

   For a Standard Tag, the list of controller is restricted to controllers that produce standard data. This may include safety controllers.

   For a Safety Tag, the list of controller is restricted to controllers that produce safety data.

   - If there is only one controller in the I/O Configuration tree, the list is empty. A controller cannot be both the consumer and the producer.
   - If the list is empty, add a remote controller to the **Controller Organizer I/O Configuration** tree.

8. In **Remote Data**, type the name of the produced tag. If the controller is a PLC-5, then type the instance number rather than the tag name. Valid range for the remote instance is 1...128.

   The controller must be offline to change the remote tag name or instance number.
9. In **Requested Packet Interval**, enter the RPI for the connection. The amount of time (in ms) between updates of the data from the remote controller. This is the minimum rate at which the local controller receives data.

   For a Standard Tag, valid ranges are:

   - 0.196 ms...536870.911 ms for ControlLogix controllers
   - 1 ms...536870.911 ms for CompactLogix controllers

For a Safety Tag, the parameter is unavailable. The range is 1...500 ms and is set on the **Consumed Tag Connection** dialog box, **Safety** tab. The RPI must match the rate specified by the Safety Task Period of the producing controller.

IMPORTANT: Changing the RPI while online temporarily disables the connection.

10. If applicable, set the RPI value used by the consumed tag to connect to the producer . This notification appears when a consuming controller accepts an RPI provided by the producer.

    The notification ( ) appears only when the controller is online and the consumed tag is connected with a producer provided RPI. The RPI parameter appears dimmed.

    Use this value to view the time interval the consuming controller had requested and the RPI provided by the producing controller.

11. If applicable, clear the **Use Unicast Connection** check box to use a Multicast connection.

    Use Unicast Connection is selected by default.

12. Select **OK** to close the **Consumed Tag Connection** dialog box.
13. Select **Create** to create the tag and to close the **New Tag** dialog box.

The controller must be offline or online in Program mode to change the RPI.

Related information

Consumed Tag Connection dialog box - Connection tab on page 24
Organize tags for produced or consumed data on page
Considerations for produced and consumed tags on page
Produce and consume array types on page 48

Additional steps for a PLC-5C controller

When sharing data with a PLC-5C controller:

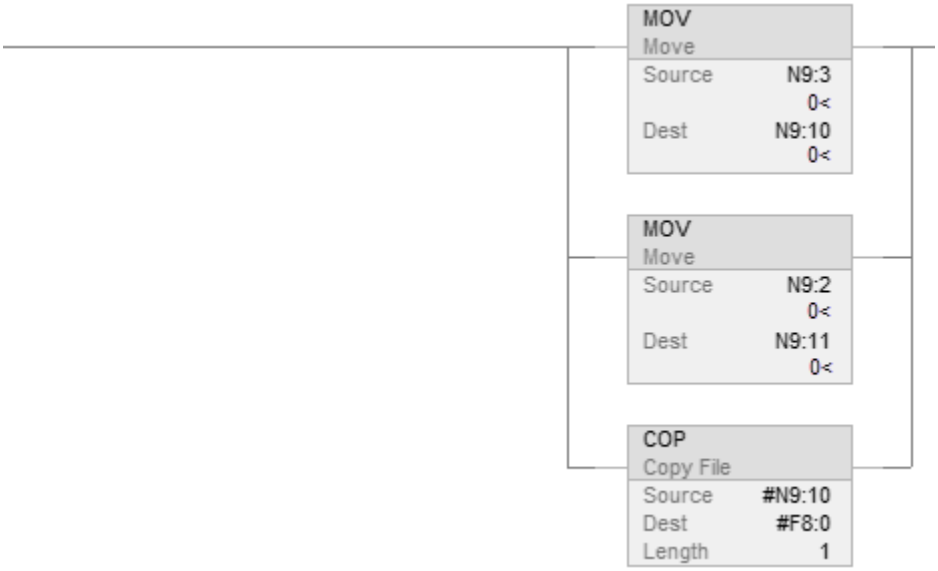
Action	Details		
In the ControlNet configuration of the PLC-5C controller, schedule a message.	If the PLC-5C controller	This	Then in RSNetWorx software
	Produces	Integers	In the ControlNet configuration of the PLC-5C controller, insert a Send Scheduled Message.

	Consumes	Integers	Perform these steps in the ControlNet configuration of the PLC-5C controller. • Insert a Receive Scheduled Message. • In **Message size**, enter the number of integers in the produced tag.
		REALs	Perform these steps in the ControlNet configuration of the PLC-5C controller. • Insert a Receive Scheduled Message. • In **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 producing 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 producing 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.

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, the COP instruction takes two consecutive integers, for a total of 32 bits, and converts them to a single floating-point value.



Consumed Tag Connection dialog box - Connection tab

Use the settings in the **Consumed Tag Connection** dialog box, **Connection** tab to configure these connection properties for the Consumed tag:

- Select the controller that produces the data.
- Identify the remote tag name or instance number of the produced tag.
- Set the RPI value.
- View the RPI value when the consuming tag is using a producer RPI.
- Select a Unicast or Multicast connection.

To open the Connection tab:

- In the **Tag Editor**, right-click the tag that will consume the data and click **Edit Properties**.
- In the **Type** box in the **Edit (name of tag) Properties** dialog box, click **Consumed** and then click **Connection**.

Consumed Tag Connection dialog box - Status tab

Use the settings on the **Status** tab in the **Consumed Tag Connection** dialog box to configure and view the connection status of the consumed tag. Use the tab to:

- Check whether the tag associated with the consumed connection includes the connection status.
- Select the data type that allows the connection status.
- Check if the controller is connected to the producing controller.
- Check if the producing controller is in Run mode.

To open the Status tab:

- In the **Tag Editor**, right-click the tag that will consume the data and click **Edit Properties**.
- In the **Type** box in the **Edit (name of tag) Properties** dialog box, click **Consumed** and then click **Connection**.
- Click the **Status** tab.

Parameter	Description
Connection Status Included	The LED is ON when the tag associated with the consumed connection is a user defined data type whose first member is CONNECTION_STATUS.
Data Type	The data type that allows for connection status. The list includes all data types whose first member is CONNECTION_STATUS.
Connected to Producer	The LED is ON when the controller is connected to the producing controller. It also displays the name of the producing controller. This control is only active online.
Producer in Run Mode	The LED is ON when the producing controller is in Run mode. The LED is OFF for all other modes. This control is only active online.

Consumed Tag Connection dialog box - Safety tab

Use the settings on the **Safety** tab in the **Consumed Tag Connection** dialog box to configure safety connection properties for the consumed tag. The **Safety** tab is only available for consumed safety tags.

To open the Safety tab:

- In the **Tag Editor**, right-click the tag that will consume the data and click **Edit Properties**.
- In the **Type** box in the **Edit (name of tag) Properties** dialog box, click **Consumed** and then click **Connection**.
- Click the **Safety** tab.

Parameter	Description
Requested Packet Interval (RPI)	The amount of time (in ms) between updates of the data from the remote controller. The valid range is 1...500 ms for Safety controllers. The default is 20 ms. The RPI of the consumed safety tag must match the safety task period of the producing safety project. The controller must be offline or online in Program mode to change the RPI.

Parameter	Description
Advanced	Opens the **Advanced Connection Reaction Time Limit Configuration** dialog box to configure the Connection Reaction Time Limit.
Connection Reaction Time Limit	Displays the predicted period of safety packets on the associated connection. If the **Max Network Delay** exceeds the **Connection Reaction Time Limit**, a connection fault occurs. This value is recalculated immediately upon changes to the RPI value.
Max Network Delay	The maximum observed transport delay from the time the data was produced until the time the data was received. The default value is 0.
Reset Max	Resets the Max Network Delay value in the controller. Available when the controller is online.

Set Advanced Options dialog box parameters

Use the **Advanced Options** dialog box to configure the behavior of the Multicast and Unicast connections to the produced tag.

Prerequisites

For CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380, GuardLogix 5580, and ControlLogix 5590 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 these steps to make sure consumers of produce tags can connect:

- For consumers running Logix Designer releases v17.00 and earlier that are consumers of a producer from CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, Compact GuardLogix 5380, and GuardLogix 5580, controllers:
  - Verify that all multicast consumed tags of a produce tag are configured with the same RPI. If they are not, some consumers fail to connect.
- For multicast consumers running Logix Designer releases v18.00 and later:
  - 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 Producer**.

IMPORTANT: The Multicast Connection Options are not available for AXIS tags or safety 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.

The RPI limits apply only to Multicast connections on unscheduled networks. This feature is unavailable for controllers on the ControlNet network or unicast connections.

- For consumers running the Logix Designer application release versions 38.00 and later:
  - CompactLogix 5480 is not supported.
  - ControlLogix 5590 is supported.

To set Advanced Options dialog box parameters

1. Set the **Minimum RPI** to 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.

   Set the **Minimum RPI** value to less than or equal to the **Maximum RPI** value.

   Modify this field in online or offline mode.

   The **Minimum RPI** is disabled for axis types. The produced axis produced data at the axis rate. The **Minimum RPI** is equal to the axis update.

2. Set the **Maximum RPI** to the largest packet interval (slowest rate) at which consumers may consume data from the tag. Rockwell Automation recommends using the default setting of **536870.9 ms**.

   Set the **Maximum RPI** value to greater than or equal to the **Minimum RPI** value.

   Modify this field in online or offline mode.

   The **Maximum RPI** is disabled for axis types. The produced axis produced data at the axis rate. The **Maximum RPI** is equal to the axis update.

3. Select **Provide Default RPI to Consumer for Out-of-Range Requests** to enable 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 RPI** and **Maximum RPI** values. A consumer that supports negotiations then connects with the **Default RPI** provided by the producer.

   Modify this field in online or offline mode.

   The **Project Default RPI to Consumer for Out-of-Range Requests** is disabled for all axis types. The produced axis produced data at the axis rate. The **Project Default RPI to Consumer for Out-of-Range Requests** is equal to the axis update.

4. Set the **Default RPI** to the value that the producer provides to the consumer when the consumer's RPI falls outside the range of the producer's **Minimum RPI** and **Maximum RPI** values. This value must be within the **Minimum RPI** and **Maximum RPI** ranges 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 selected.

   The **Default RPI** is disabled for all axis types. The produced axis produced data at the axis rate. The **Default RPI** is equal to the axis update.

IMPORTANT: On the **Controller Properties dialog box - Advanced tab**, select **Allow Consumed Tags to Use RPI Provided by Producer** to allow negotiations to use the RPI provided by the producer.

5. **Allow Unicast Consumer Connections** allows multiple Unicast consumers to consume from the produced tag. This check box is enabled and selected by default, unless the safety controller is locked.

   Selecting **Allow Unicast Consumer Connections** results in additional producer resources allocation for each potential consumer. Clear the check box when not using Unicast and when running low on controller resources.

   Modify this field in online or offline mode.

Related information

Create a produced tag on page 17
Create a consumed tag on page 20

Set an RPI Range for Produced Tags

Program controllers 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 rejects the incoming RPI and may provide an RPI (default) to the consuming controller. This allows a consuming controller to be modified or added to an existing multicast connection without interruptions.

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. If choosing not to use the default value, manually change a rejected RPI to make the value within the limitation settings for the produced tag.

To set an RPI Range for Produced Tags

1. Set RPI Limits and defaults for the producer tag.

   RPI Limits and default values are not supported for safety tags and AXIS tags, and apply only to multicast connections on unscheduled networks. Unavailable for controllers on the ControlNet network or Unicast connections.

   a. In **Produced Tag Connection**, click **Advanced**.

   b. Configure the:

      - **Minimum RPI**
      - **Maximum RPI**
      - Select **Provide Default RPI to Consumer for Out-of-Range Requests**
      - **Default RPI**

   c. Select **OK**.

2. 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.

a. In the **Controller Organizer**, right-click the controller and choose **Properties**.

b. Select the **Advanced** tab.

c. Select **Allow Consumed Tags to Use RPI Provided by Producer**.

d. Select **OK**.

3. Verify consumed tag acceptance.

When a consuming controller accepts a default RPI by a producing controller, verify the acceptance and the RPI value on the **Consumed Tag Connection** dialog box. The controller must be online.

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

b. In **Controller Tags**, right-click a consuming tag and choose **Edit <tag name> Properties**. The **Tag Properties** dialog box opens.

c. In **Type**, choose **Consumed**.

d. Select **Connection** to open the **Consumed Tag Connection** dialog box with a flag to the right of the RPI field, indicating that the consuming controller has accepted an RPI provided by the producer.

Related information

Set Advanced Options dialog box parameters on page 26

Unicast connection option

Unicast connections are point-to-point connections between a producing controller and consuming controller; no multiple connections. Unicast connections do not require a minimum or maximum RPI range or default RPI value.

The **Allow Unicast Consumer Connections** check box at the bottom of the **Advanced Options** dialog box is selected by default to allow multiple unicast consumers to consume from the produced tag.

When the controller is running low on controller memory, clear the **Allow Unicast Consumer Connections** check box to discontinue using the option. Additional producer resources allocate for each potential consumer when the check box is selected.

RPI limitations - negotiated default

Requested packet interval (RPI) limitations and negotiated behavior vary depending on theLogix Designer application version that is running on the controllers.

Behavior in release versions 28.00 and later

In release versions 28.00 and later, the first consumer of a produce tag determines the RPI of produced data. All subsequent consumers must request the same RPI value as the first consumer or the subsequent consumers 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 order in which the producer and consumer control system powers up determines the first consumer request. Therefore, cycling power to the system can change the first consumer. A device cannot be configured to be the first consumer. Determining which

consumer sends the first request to the producing controlleris not possible. Plan accordingly when configuring multicast produce tags. See **Considerations when migrating projects that contain multicast produce tags** for more information.

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

Behavior in release versions 18.00 to version 27.00

In version 18.00 and earlier controllers can be programmed to enhance the 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 rejects 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. When choosing to not use the default value, manually change a rejected RPI to make the value within the limitation settings for the produced tag.

1769-L2x and 1769-L3x controllers have additional RPI information. See **1769-Lx2 and 1769-L3x RPI Limits** for details.

With multicast consumers running Logix Designer releases version 18.00 and later, performthese 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 Producer**.

Behavior in releases version 17.00 and earlier

When using releases version 17.00 and earlier of the application, the producer accepts an incoming RPI of a consuming controller. The producing controller does not offer a default RPI. Use multicast connections to speed up the packet intervals of existing connections from other consumers.

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: Safety tags and AXIS tags do not support RPI limits and default values. 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.

With controllers running releases version 17.00 or earlier that are consumers of a producer from a Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, or GuardLogix 5580 controller, verify that all multicast consumed tags of a produce tag have the same RPI. If the consumed tagsdo not, some consumers fail to connect.

Produce and Consume ControlNet Safety Tags

The ControlNet communications modules support the following.

- Messaging, produced/consumed tags, and distributed I/O.
- Use of coaxial cable and fiber repeaters for isolation and increased distance.

Produce and Consume Data via the ControlNet Network

The GuardLogix controller supports the ability to produce (broadcast) and consume (receive) system-shared tags over the ControlNet network. Produced and consumed tags each require connections.

The total number of tags that can be produced or consumed is limited by the number of available connections in the GuardLogix controller.

Connections Over the ControlNet Network

You indirectly determine the number of connections the controller uses by configuring the controller to communicate with other devices in the system. Connections are allocations of resources that provide more reliable communications between devices compared to unconnected messages.

ControlNet connections can be either scheduled or unscheduled.

ControlNet Connections

Connection Type	Description
Scheduled (unique to the ControlNet network)	A scheduled connection is unique to ControlNet communications. A scheduled connection lets you send and receive data repeatedly at a predetermined interval, which is the requested packet interval (RPI). For example, a connection to an I/O module is a scheduled connection because you repeatedly receive data from the module at a specified interval. Other scheduled connections include connections to: • Communication devices • Produced/consumed tags On a ControlNet network, you must use RSNetWorx for ControlNet software to enable all scheduled connections and establish a network update time (NUT). Scheduling a connection reserves network bandwidth to specifically handle the connection.
Unscheduled	An unscheduled connection is a message transfer between controllers that is triggered by the requested packet interval (RPI) or the program (such as a MSG instruction). Unscheduled messaging lets you send and receive data when needed. Unscheduled connections use the remainder of network bandwidth after scheduled connections are allocated.

	Safety produced/consumed connections are unscheduled, by default.

The 1756-CNB and 1756-CNBR communication modules support 64 CIP connections over a ControlNet network.

The 1756-CN2 module supports 100 CIP connections over the ControlNet network. However, we recommend that you configure only 97 connections for each module to maintain optimal performance.

Related information

Set an RPI Range for Produced Tags on page 28

Edit Connection Information

For Produced and Consumed tags, you can edit connection information. Using the **Tag Properties** dialog box, follow the steps below based on the type of tag you have chosen.

To edit information for a Produced Tag

1. In the Maximum Consumers field, enter the maximum number of consumers you can have for this tag.
2. Check the Programmatically Send Event Triggers to Consumers if you want to use an IOT instruction to send event trigger information to the consumers of this tag.
3. Select **OK** to accept your edits and close the Tag Properties dialog box.

To edit information for a Consumed Tag

1. From the Producer pull-down menu, choose the name of produced tag you want this tag to consume.

   The pull-down menu lists only those options that are available.
2. In the Remote Data field, enter the name or instance number of the tag in the remote controller that you wish to consume.

   If your controller is a PLC-5, then you will enter the instance number rather than the tag name. The valid range for the remote instance is 1...128.

   You must be offline to change the remote tag name or the remote instance number.
3. In the RPI field, enter the Requested Packet Interval.

   This is the amount of time (in ms) between updates of the data from the remote controller. This is the minimum rate at which the local controller will receive data. For ControlLogix controllers, the valid RPI range is 0.196 ms...536870.911 ms. For CompactLogix controllers, the valid range is 1 ms...536870.911 ms.

   You must be offline or online in Program mode to change the RPI.
4. Select **OK** to accept your edits and close the **Tag Properties** dialog box.

Related information

Set an RPI Range for Produced Tags on page 28

Set up the consuming controller

Set up the Consuming controller to allow its consumed tags to accept an RPI from the producer. Use the **Advanced** tab in the **Controller Properties** dialog box to set up the controller.

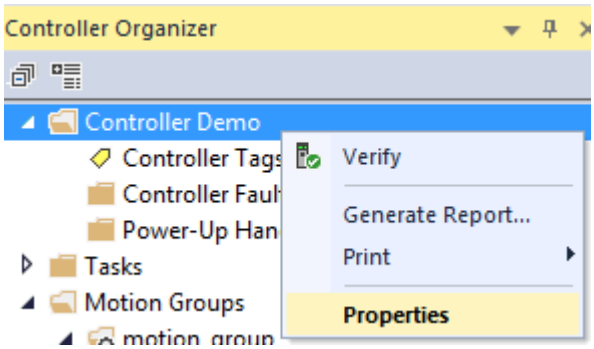
IMPORTANT:

For controllers running Logix Designer releases version 17.00 or earlier that contain consumers of a producer from Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, or GuardLogix 5580 controller

, verify that all multicast consumed tags of a produce tag have the same RPI. If they do not, some consumers fail to connect. See **RPI limitations and negotiated default** for more information.

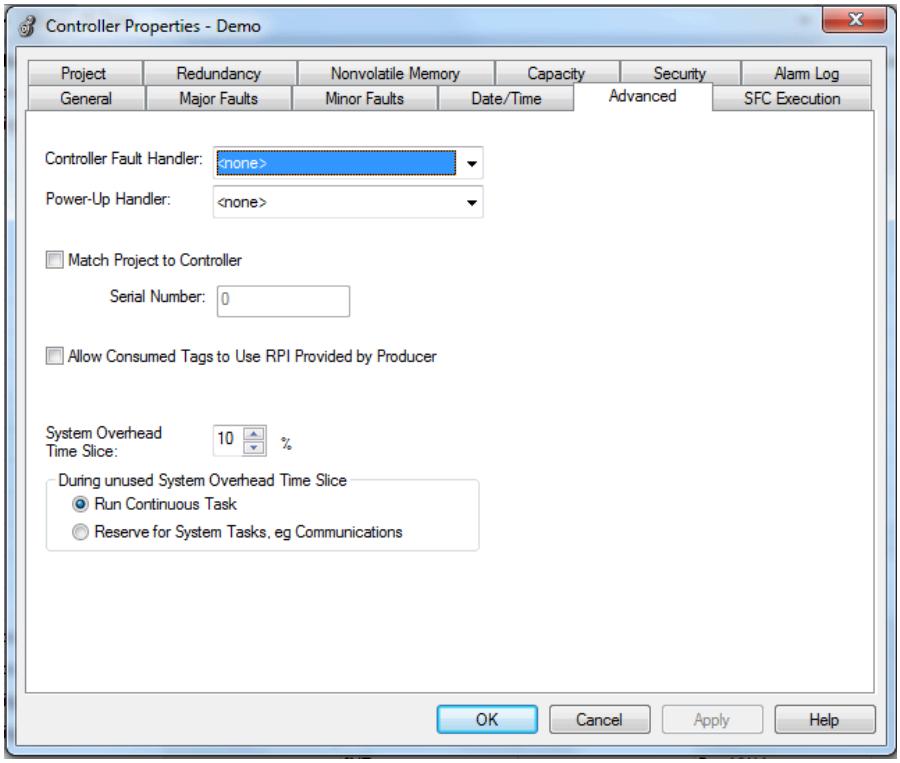
To set up the consuming controller:

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



The **Controller Properties** dialog box opens.

2. Select the **Advanced** tab.



3. Select the **Allow Consumed Tags to Use RPI Provided by Producer** check box to allow for the RPI negotiation.

IMPORTANT: The check box is only available when the controller is offline. If the check box is not selected, 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. Select **OK**.

Verify consumed tag acceptance

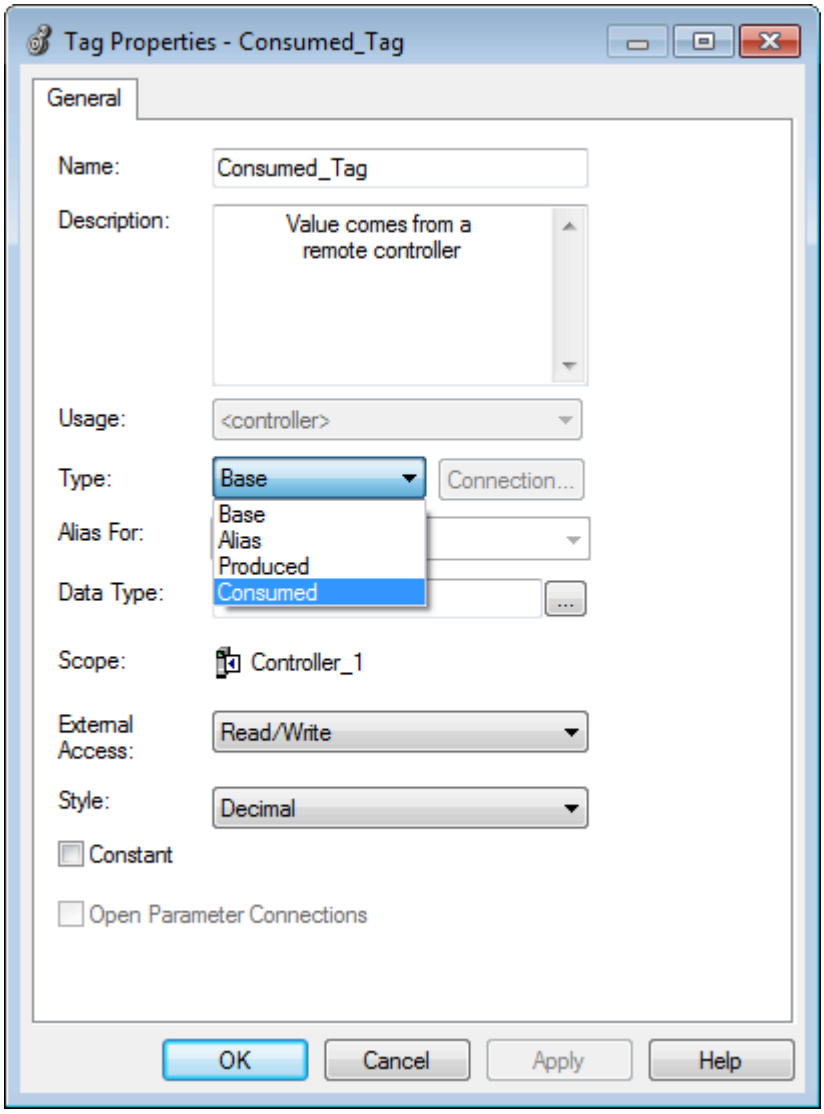
When a consuming controller accepts a default RPI by a producing controller, verify the acceptance and the RPI value on the **Consumed Tag Connection** dialog box.

Prerequisite:

- Verify the controller is online and the consumed tag is connected with the producer providing the RPI.

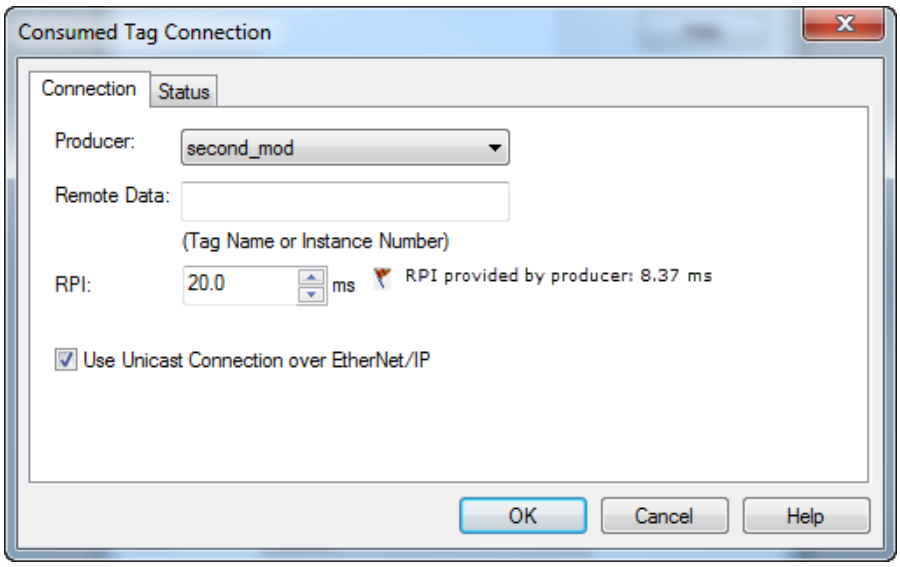
To verify consumed tag acceptance:

1. In the **Controller Organizer**, right-click the **Controller Tags** folder and select **Edit Tags**.
2. In the **Tag Editor**, right-click a consuming tag and select **Edit <name of tag> Properties**. The **Tag Properties** dialog box opens.



3. In the **Type** box, select **Connection**.

The **Consumed Tag Connection** dialog box opens. On the **Connection** tab, a flag to the right of the **RPI** box indicates the consuming controller accepted an RPI provided by the producer and the time interval of the RPI.



4. Select **Cancel** to exit.

Advanced Connection Reaction Time Limit Configuration Dialog

Use the Advanced Connection Reaction Time Limit Configuration dialog box to perform the following actions.

Parameters

Requested Packet Interval (RPI)

Enter the Requested Packet Interval. This is the amount of time (in ms) between updates of the data from the remote controller. This is the minimum rate at which the local controller will receive data. The valid range for the RPI is 1...500 ms for Safety controllers.

You must be offline or online in Program mode to change the RPI.

Timeout Multiplier

Determines the number of RPIs to wait for a packet before declaring a connection timeout.

Network Delay Multiplier

Defines the message transport time that is enforced by the CIP Safety protocol. The Network Delay Multiplier specifies the round trip delay from the producer to the consumer and back to the producer. You can use the Network Delay Multiplier to increase or decrease the Connection Reaction Time Limit.

Connection Reaction Time Limit

Displays the predicted period of safety packets on the associated connection. If the Max connection reaction time ever exceeds the Reaction Time Limit, a connection fault occurs.

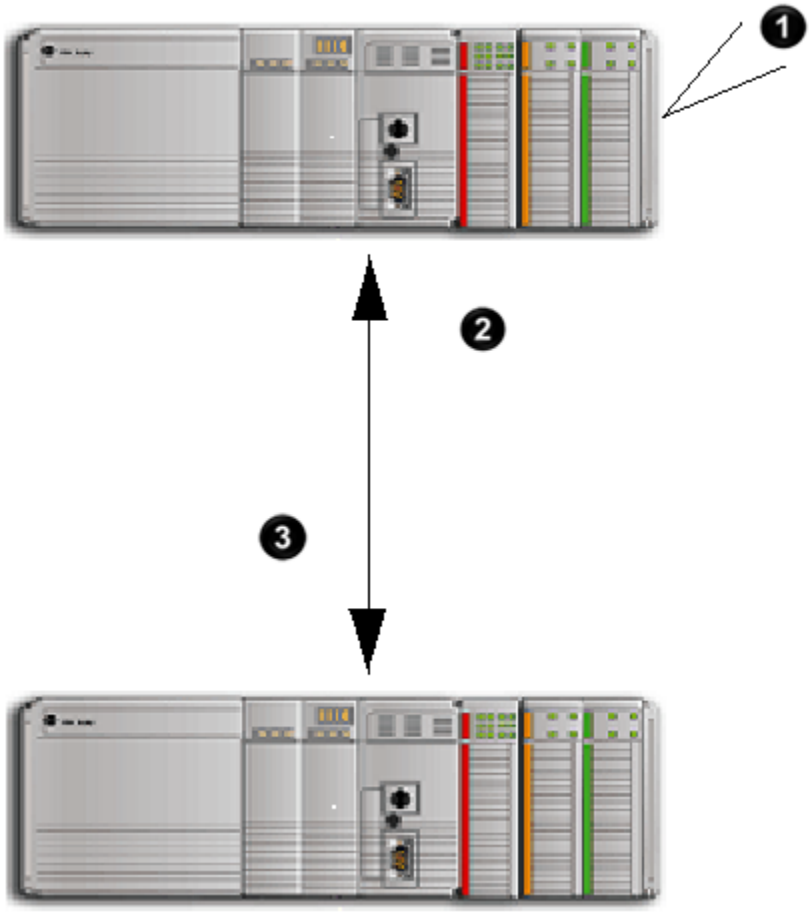
This value is calculated as a function of the RPI value set on this tab, and the Advanced Connection Reaction Time Limit Configuration dialog box. This value is recalculated immediately upon changes to the RPI value.

Produced and consumed RPI scenarios

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

Scenario 1

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



Number	Description
1	**Produced Tag** Minimum RPI: 2 ms Maximize RPI: 8 ms Default RPI: - ms
2	Actual Packet Interval (API) = 3 ms

Number	Description
3	RPI = ms

Version 17.00 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 increase to 3 ms for the API, which is the interval the data is actually being produced.

Version 18.00 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.

When using all the default settings of version 18.00 and later, the behavior matches that of version 17.00 and earlier.

Scenario 2

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

Number	Description
1	**Produced Tag** Minimum RPI: 2 ms Maximize RPI: 8 ms Default RPI: - ms
2	Error: 0111
3	RPI = 1 ms

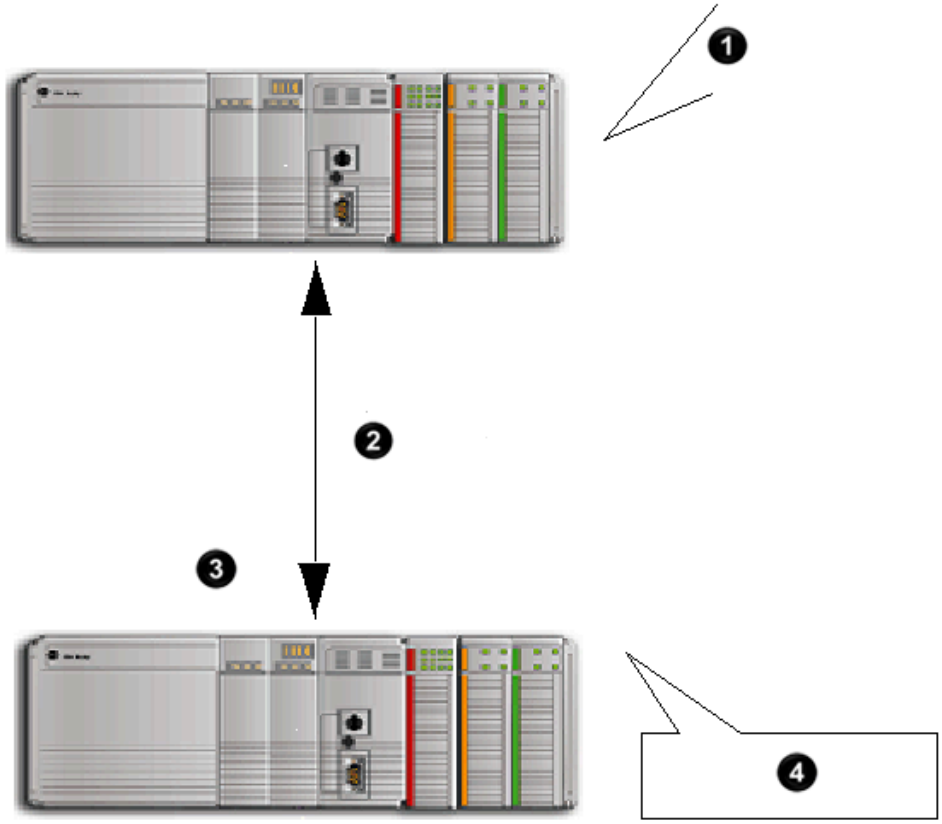
Version 17.00 and earlier: The RPI requested by the consumed tag is not within the range of the producer. The producer rejects the connection and a 0111 error reports that the connection failed. Re-enter an RPI that is within the range configured on the produced tag. If a version 17.00 consumer tries to connect with a version 18.00 producer, and the RPI is outside the range of the producer, and no default is default set up, error 0111 error reports the connection as failed.

See *RPI I/O Faults* for an explanation of the error messages.

Version 18.00 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 error 0111 reports that the connection failed. Re-enter an RPI 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.



Number	Description
1	**Produced Tag** Minimum RPI: 2 ms Maximize RPI: 8 ms Default RPI: 5 ms
2	Error: 0112, Default RPI = 5 ms
3	RPI = 1 ms
4	Negotiation Disabled

Version 17.00 and earlier: The RPI requested by the consumed tag is not within the range of the producer. The connection is rejected and error 0112 reports that the connection failed. Re-enter an RPI within the range configured on the produced tag. If a version 17.00 consumer tries

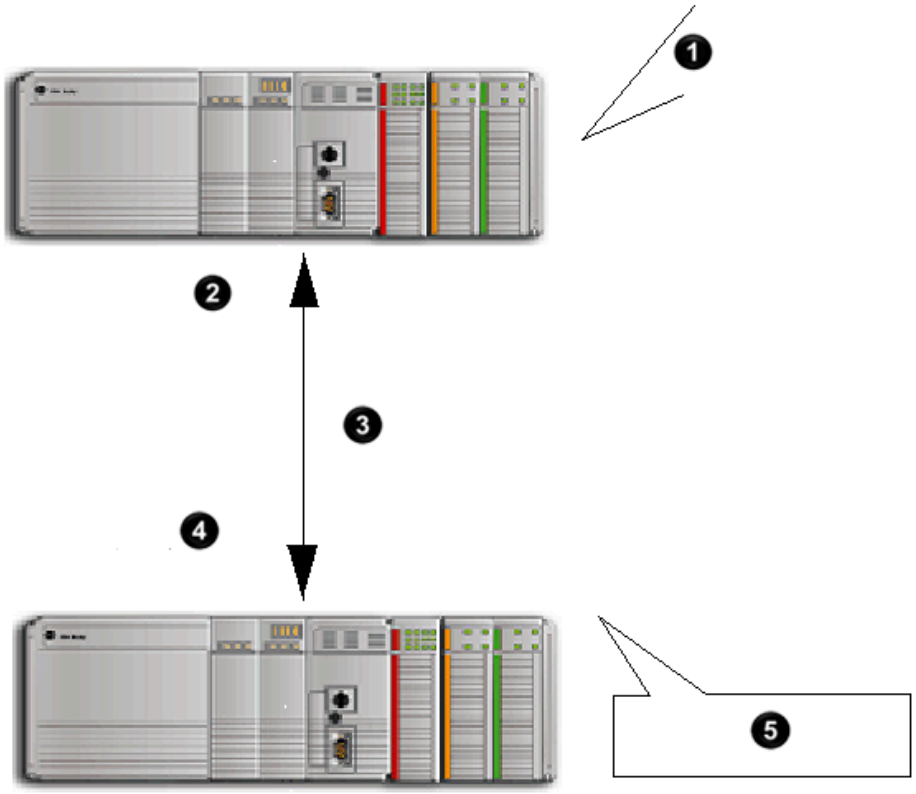
to connect with a version 18.00 producer, and the RPI is outside the range of the producer, error 0112 reports the connection as failed.

See *RPI I/O Faults* for an explanation of the error messages.

Version 18.00 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. Error message 0112 is reported. 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.



Number	Description
1	**Produced Tag** Minimum RPI: 2 ms Maximize RPI: 8 ms Default RPI: 5 ms
2	API = 5 ms

Number	Description
3	Error: 0112, Default RPI = 5 ms
4	RPI = 5 ms (changed from 1 ms)
5	Negotiation Enabled

Version 17.00 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 reports that the connection failed. Re-enter an RPI within the range configured on the produced tag. If a version 17.00 consumer tries to connect with a version 18.00 producer, and the RPI is outside the range of the producer, error 0112 reports the connection as failed.

Version 18.00 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 through 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.

Entering these exact values is not required when configuring an RPI limitation range and default RPI for the 1769-L2*x* and 1769-L3*x* controllers. However, when entering an unsupported packet interval, the Logix Designer application may override the unsupported packet interval with a supported packet interval. When this occurs, the new effective value displays separately next to the unsupported value.

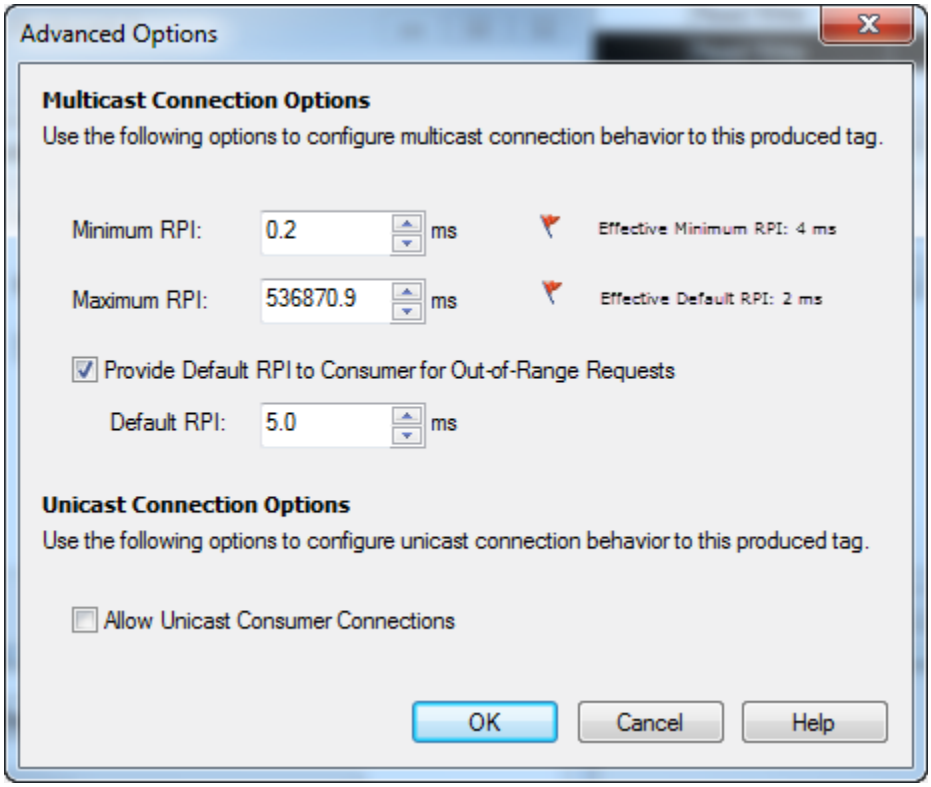
Similar to other controllers, the 1769-L2*x* and 1769-L3*x* controllers verify that the RPI of incoming connections are within the produced tag settings. If the consuming tag's RPI falls outside the configured range, a producing controller rejects the incoming RPI and then provides an RPI (default) to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

IMPORTANT: Set up the producing and consuming controllers to allow the consumed tags to use an RPI provided by the producer.

See *RPI Limitations and Negotiated Default* and *Set up the consuming controller* 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 use the **Effective Minimum RPI**, when present, to determine the fastest packet interval allowed by the tag. When present, the **Effective Default RPI** is the largest packet interval (slowest rate) at which negotiated connections are produced for the tag.

A flag on the **Advanced Options** dialog box presents the effective RPI limit values to indicate that RPI limits are set to values that are different from the entered values.



RPI I/O faults

The 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.	18.00 and later	(Code 0112) Requested Packet Interval (RPI) out of range.
	17.00 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	18.00 and later	(Code 0112) Requested Packet Interval (RPI) out of range. Producer provided RPI blocked by the network path.

Scenario Description	Software Version	I/O Fault Message
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 the network, refer to the following publication in the Rockwell Automation Literature Library: *EtherNet/IP Network Configuration User Manual* (ENET-UM001).		
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).	18.00 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 Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, and GuardLogix 5580 controllers

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

- For consumers running Logix Designer releases version 17.00 and earlier that are consumers of a producer from a Compact GuardLogix 5380, CompactLogix 5380, CompactLogix 5480, ControlLogix 5580, or GuardLogix 5580 controller:
  - Verify that all multicast consumed tags of a produce tag have the same RPI. If they do not, some consumers fail to connect.
- For multicast consumers running Logix Designer releases version 18.00 and later, perform one of these steps:
  - Verify that all multicast consumed tags of a produce tag have the same RPI, or
  - Verify that all multicast consumers are configured to **Allow Consumed Tags To Use RPI Provided By Producer**.

Produce a large array

Transferring a large array as smaller packets improves system performance.

- Large arrays use fewer connections than breaking the data into multiple arrays and sending each as a produced tag. For example, an array with 5000 elements takes40 connections (5000/125=40) by using individual arrays.
- Large arrays achieve faster transmission times than using a message instruction to send the entire array.
  - Messages are unscheduled and executed only during the system overhead portion of the Logix5550 execution. Therefore, messages can take a fairly long time to complete the data transfer.
  - Improved transfer time by increasing the system overhead time slice, but this diminishes the performance of the continuous task.

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 the array_ack tag 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 contains the logic for the transfer, create these tags.

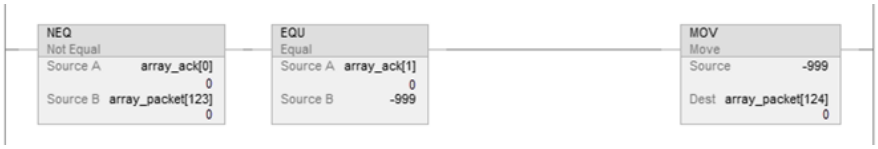
Tag Name	Type
array	DINT[x] where x equals the number of elements to transfer plus 122 elements
array_offset	DINT
array_size	DINT
array_transfer_time	DINT

Tag Name	Type
array_transfer_time_max	DINT
array_transfer_timer	TIMER

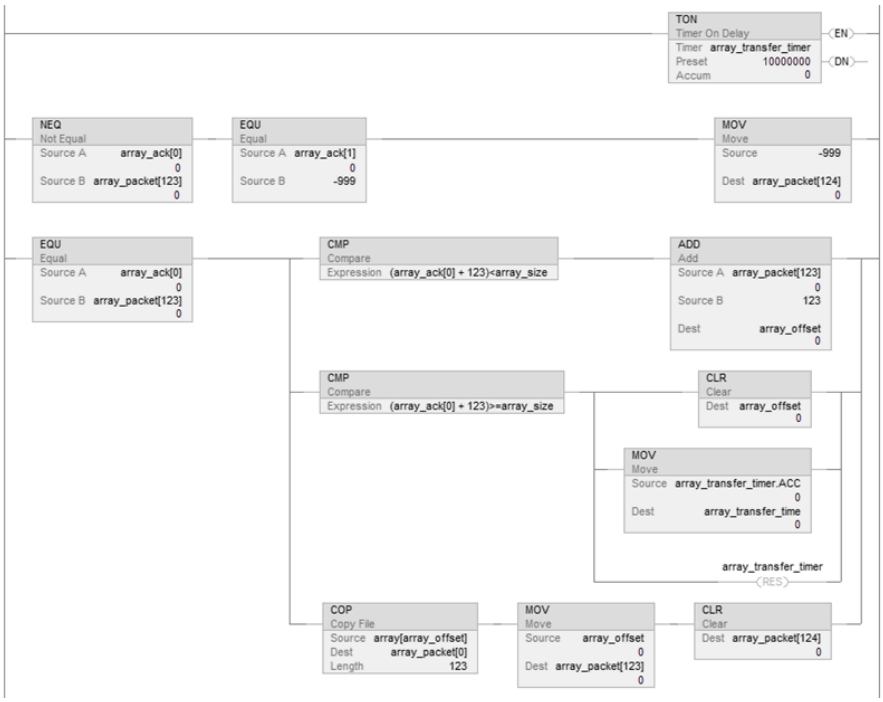
4. In the array_size tag, enter the number of elements of real data, which is the value of x from step 3 minus the 122 elements of buffer.
5. Create or open a routine for the logic that creates packets of data.

   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 begins receiving 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 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.

6. 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]

7. Convert the array_packet tag 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.

8. 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

9. Create or open a routine for the logic that moves the data from the packets to the destination array.
10. Enter this logic.

    When the offset value in array_packet[123] is different than the offset value in array_ack[0], the controller begins receiving 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 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.

Produce and consume array types

DINT, REAL and STRUCTURE

These types of arrays can be directly produced and consumed. In order to do this, make sure that the sizes of the arrays are the same in both the consumer and producer controllers.

BOOL, SINT and INT

These types of arrays can be indirectly produced and consumed. In order to do this:

- Define a data type (structure) containing a single member.
- Make the single member an array of the desired type.
- Duplicate the type definition to both controllers.
- Produce and consume tags of this datatype.

In addition, to consume an array of INTs from a PLC-5C:

- Define a datatype with two members:
  - Name the first member **Status** and make a type DINT.
  - Make the second member an array of INTs. Ensure the size of the array matches the size specified in RSNetWorx.

Related information

Create a consumed tag on page 20
Create a produced tag on page 17

Work with large arrays

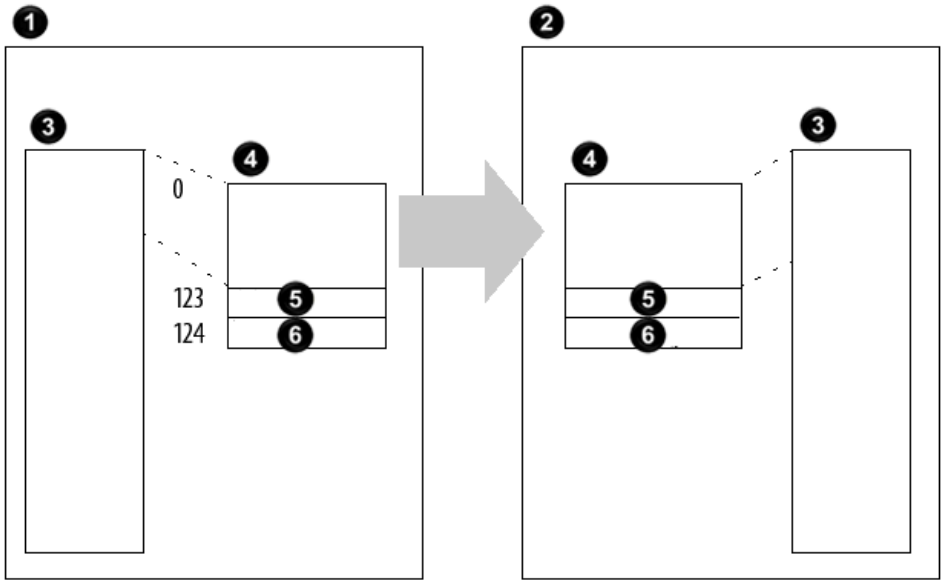
The Logix 5000 controller can send as many as 500 bytes of data over a single scheduled connection. This corresponds to 125 DINT or REAL elements of an array. To transfer an array of more than 125 DINTs or REALs, use a produced/consumed tag of 125 elements to create a packet of data. Then use the packet to send the array piecemeal to another controller.

When sending a large array of data in smaller packets, be sure that the packet transmission is complete before moving the data 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 Producer/Consumer Data Elements image.



Number	Description
1	Producer
2	Consumer
3	Array
4	Packet
5	Offset
6	Acknowledge

In addition, the array must contain an extra 122 elements. In other words, the array must be 122 elements greater than the greatest number of elements 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, transferring occurs if the last packet contains fewer than 123 elements of actual data.

Produce a large array

Transferring a large array as smaller packets improves system performance.

- Large arrays use fewer connections than breaking the data into multiple arrays and sending each as a produced tag. For example, an array with 5000 elements takes40 connections (5000/125=40) by using individual arrays.
- Large arrays achieve faster transmission times than using a message instruction to send the entire array.
  - Messages are unscheduled and executed only during the system overhead portion of the Logix5550 execution. Therefore, messages can take a fairly long time to complete the data transfer.
  - Improved transfer time by increasing the system overhead time slice, but this diminishes the performance of the continuous task.

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 the array_ack tag 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 contains the logic for the transfer, create these tags.

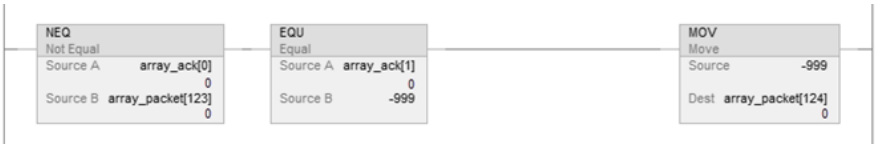
Tag Name	Type
array	DINT[x] where x equals the number of elements to transfer plus 122 elements
array_offset	DINT
array_size	DINT
array_transfer_time	DINT
array_transfer_time_max	DINT

Tag Name	Type
array_transfer_timer	TIMER

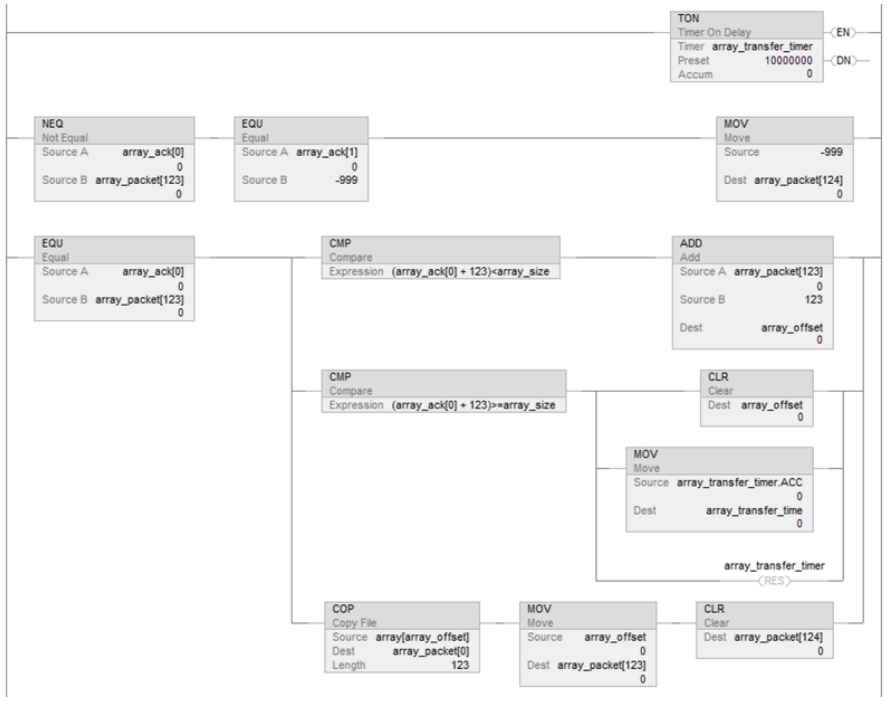
4. In the array_size tag, enter the number of elements of real data, which is the value of x from step 3 minus the 122 elements of buffer.
5. Create or open a routine for the logic that creates packets of data.

   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 begins receiving 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 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.

6. 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]

7. Convert the array_packet tag 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.

8. 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

9. Create or open a routine for the logic that moves the data from the packets to the destination array.
10. Enter this logic.

    When the offset value in array_packet[123] is different than the offset value in array_ack[0], the controller begins receiving 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 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.

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
Local Technical Support Phone Numbers	Locate the telephone number for your country.	rok.auto/phonesupport
Technical Documentation Center	Quickly access and download technical specifications, installation instructions, and user manuals.	rok.auto/techdocs
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 Otomasyon 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
EUROPE/MIDDLE EAST/AFRICA: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2663 0600
ASIA PACIFIC: Rockwell Automation SEA Pte Ltd, 2 Corporation Road, #04-05, Main Lobby, Corporation Place, Singapore 618494, Tel: (65) 6510 6608
UNITED KINGDOM: Rockwell Automation Ltd., Pitfield, Kiln Farm, Milton Keynes, MK11 3DR, United Kingdom, Tel: (44)(1908) 838-800

Publication 1756-PM011N-EN-P - September 2025
Supersedes Publication 1756-PM011M-EN-P - November 2023