PULSEROLLER

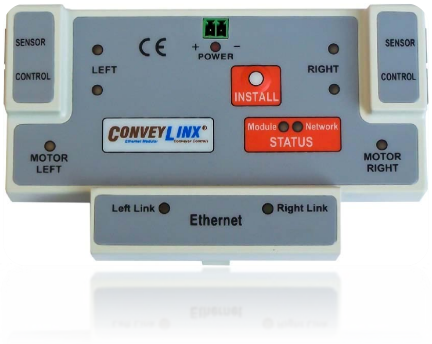
CONVEYLINX

Supplement to PLC Developer's Guide

Connecting to Rockwell PLCs

Version 1.0

September 2014



Publication *ERSC*-1520

Symbol Conventions

⚠ **This symbol indicates that special attention should be paid in order to ensure correct use as well as to avoid danger, incorrect application of product, or potential for unexpected results**

ℹ **This symbol indicates important directions, notes, or other useful information for the proper use of the products and software described herein.**

Important User Information

⚠ *ConveyLinx ERSC* modules contain ESD (Electrostatic Discharge) sensitive parts and components. Static control precautions are required when installing, testing, servicing or replacing these modules. Component damage may result if ESD control procedures are not followed. If you are not familiar with static control procedures, reference any applicable ESD protection handbook. Basic guidelines are:

- Touch a grounded object to discharge potential static
- Wear an approved grounding wrist strap
- Do not touch connectors or pins on component boards
- Do not touch circuit components inside the equipment
- Use a static-safe workstation, if available
- Store the equipment in appropriate static-safe packaging when not in use

⚠ Because of the variety of uses for the products described in this publication, those responsible for the application and use of this control equipment must satisfy themselves that all necessary steps have been taken to assure that each application and use meets all performance and safety requirements, including any applicable laws, regulations, codes, and standards

⚠ The illustrations, charts, sample programs and layout examples shown in this guide are intended solely for purposes of example. Since there are many variables and requirements associated with any particular installation, Insight Automation Inc. does not assume responsibility or liability (to include intellectual property liability) for actual use based on the examples shown in this publication

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

SUMMARY OF CHANGES

The following table summarizes the changes and updates made to this document since the last revision

Revision	Date	Change / Update
1.0	September 2014	Initial Release

GLOBAL CONTACT INFORMATION

PULSE ROLLER

WWW.PULSEROLLER.COM
SALES@PULSEROLLER.COM
SUPPORT@PULSEROLLER.COM

TABLE OF CONTENTS

Preface

Who Should Use This Manual?

This manual is intended for users who need to utilize a Rockwell PLC equipped with Ethernet I/P capability to connect to a *ConveyLinx* Ethernet network to access module status and control conveyor operation.

Prerequisites

You should have reviewed and understood either the *ConveyLinx PLC Developer's Guide* (Insight Automation publication ERSC-1500) or *ConveyLinx-Ai PLC Developer's Guide* (Insight Automation publication ERSC-1510) before utilizing this manual's instructions to physically connect your Rockwell PLC to a ConveyLinx network.

This manual also assumes you have a solid working knowledge of both Rockwell PLC's and the RSLogix 5000 / RSLogix Designer development environments.

Not Included in This Manual

⚠ **Because system applications vary; this manual assumes users and application engineers have properly sized their PLC's Ethernet port capacity to accommodate the quantity of ConveyLinx module connections desired. Please refer to you particular PLC's specifications**.

INTRODUCTION

This manual will provide instructions on how to connect your Rockwell Ethernet I/P capable PLC to a network of ConveyLinx modules. There are three basic methods for connecting ConveyLinx to Rockwell PLCs:

- **Import EDS and AOIs**
- **Use Generic Ethernet Device**
- **Use MSG Instruction**

All three methods can be used for ConveyLinx modules in ZPA mode and in PLC I/O mode. However, the MSG Instruction method does not maintain a constant connection to a ConveyLinx module and should not be used for "time critical" operations.

ETHERNET I/P GUIDELINES

Each Allen-Bradley PLC has 2 metrics for limiting maintained Ethernet I/P communications to remote devices:

- Fixed quantity of TCP connections available on its Ethernet Port
- Fixed quantity of I/O data table memory available for connected devices

If the limit of either of these quantities is reached, the PLC processor will indicate I/O communications fault on one or more instances of device declaration.

For *ERSC* device declarations utilizing either ZPA or PLC I/O Mode instances, in general the PLC limitation on TCP connections will be reached before I/O data table memory limit is realized.

For example, for a CompactLogix L3x series processor, the documented quantity of TCP connections available on its Ethernet Port is 32. The processor always keeps one TCP connection in reserve for programming terminal access, etc. An L3x series processor can accept 31 full-time *ERSC* Connections as generic I/O modules utilizing any combination of ZPA mode and PLC I/O Mode instances.

When an *ERSC* is attached as a "full-time generic I/O module" to the PLC, the connection is continually maintained and data exchanged at a minimum RPI value and if the PLC cannot communicate with the *ERSC* for any reason, the PLC's I/O tree will register a fault.

It is possible for the PLC to communicate via Ethernet I/P with any *ERSC* it can physically reach over its Ethernet port without the *ERSC* being "full-time connected as a generic I/O module". This is accomplished with a Logix5000 MSG instruction.

Reserve Ethernet I/P TCP connections for *ERSC*'s in PLC I/O Mode and for key ZPA Mode *ERSCs* where permanent accumulate/query/release functionality is required.
Use MSG Instruction to gather less time-critical data for things such as status and diagnostics.
For more information on determining the design and capacity of your Ethernet I/P

network; please refer to Allen-Bradley document *EtherNet/IP Performance Application Solution* (publication ENET-AP001D-EN-P).

IMPORT EDS FILES AND USE AOIS

For this method, Insight Automation has pre-packaged the PLC register Assemblies (detailed in the Developer's Guides) into a combination EDS configurations and Add On Instructions (AOIs) to make your program development easier and more straight-forward. What the AOI does is creates a custom data type for your module that puts meaningful tag names for all the I/O going to and from the ERSC module and the PLC. Also, some of these created tags are already in easy to use Boolean data types to make it easier for you to write and for others to follow your logic program.

This method requires you have the following files:

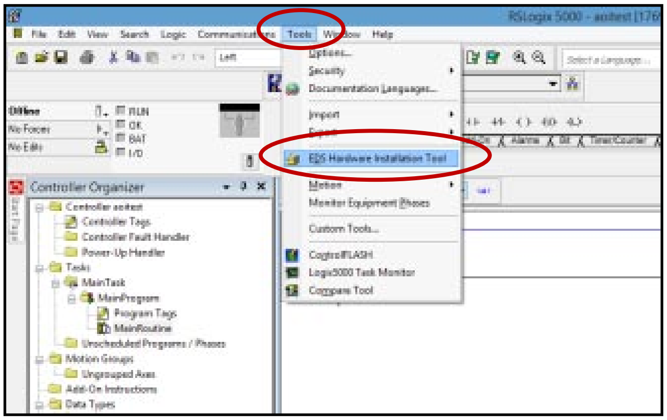
055C002B00010200_ERSC_ZPA_2_1.eds
055C002B00020200_ERSC_IO_2_1.eds
ERSC_ZPA_2_1.L5X
ERSC_PLCIO_2_2.L5X

These are the filenames as of the publication of this document. Please go to pulseroller.com to download the latest versions of these files.

You do not have to install both EDS files and both AOI files into your RSLogix 5000 environment unless you want to use ConveyLinx modules in both ZPA mode and PLC I/O mode. We recommend you install both so they are available for future projects.

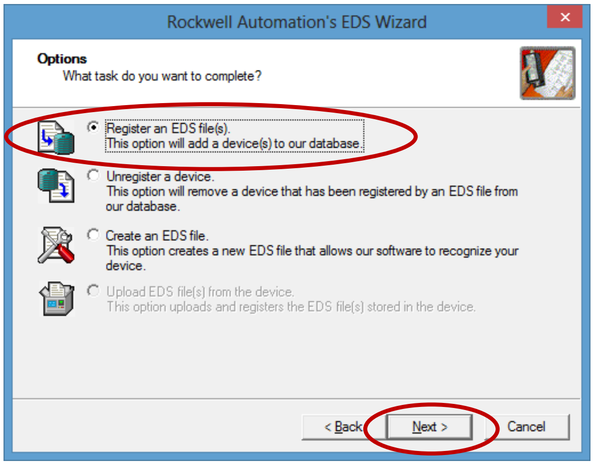
By default, ConveyLinx modules install themselves in ZPA mode. You have to utilize the EasyRoll software tool to change the mode of any and all ConveyLinx modules that need to be in PLC I/O mode.

INSTALLING EDS FILES INTO RSLOGIX5000



Step 1

With RSLogix5000 open, select Tools from the menu and EDS Hardware Installation Tool



Rockwell Automation's EDS Wizard

Options

What task do you want to complete?

● Register an EDS file(s).
This option will add a device(s) to our database.

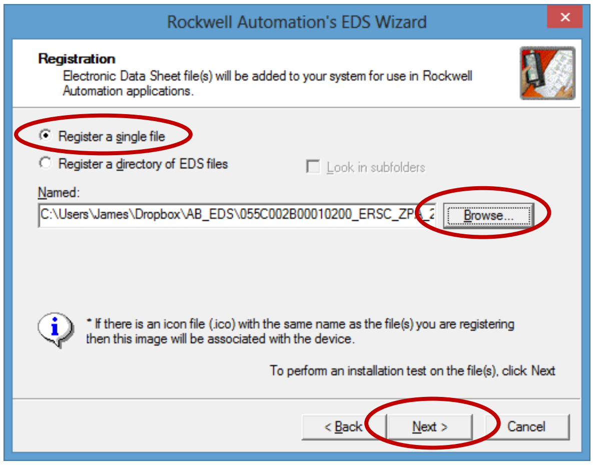
○ Unregister a device.
This option will remove a device that has been registered by an EDS file from our database.

○ Create an EDS file.
This option creates a new EDS file that allows our software to recognize your device.

○ Upload EDS file(s) from the device.
This option uploads and registers the EDS file(s) stored in the device.

Step 2

Select the Register an EDS file(s) radio button and click next



Rockwell Automation's EDS Wizard

Registration

Electronic Data Sheet file(s) will be added to your system for use in Rockwell Automation applications.

● Register a single file

○ Register a directory of EDS files □ Look in subfolders

Named:

C:\Users\James\Dropbox\AB_EDS\055C002B00010200_ERSC_ZPA_2

* If there is an icon file (.ico) with the same name as the file(s) you are registering then this image will be associated with the device.

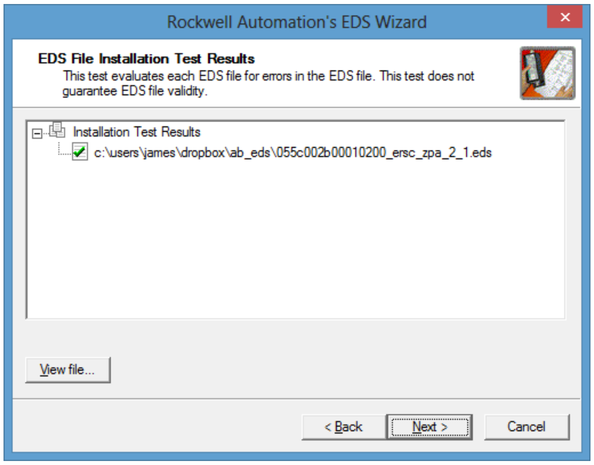
To perform an installation test on the file(s), click Next

Step 3

Select the register a single file radio button and click Browse and then browse to the location on your PC where you downloaded the EDS file. In this example we are installing the ZPA version. Click Next to continue.

Step 4

This window should appear with the green check indicating there were no errors. Click Next to continue


Rockwell Automation's EDS Wizard

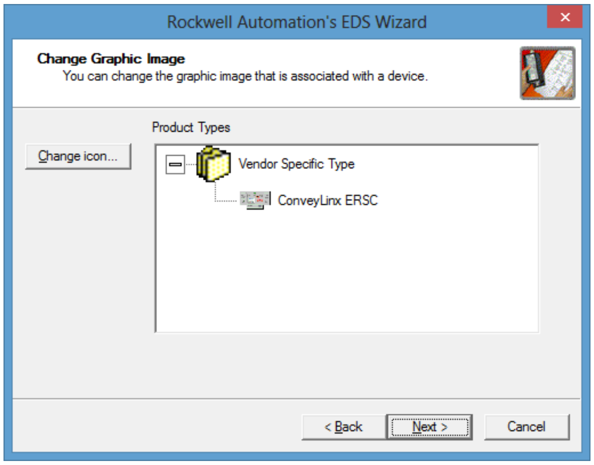
EDS File Installation Test Results

This test evaluates each EDS file for errors in the EDS file. This test does not guarantee EDS file validity.

Installation Test Results
c:\users\james\dropbox\ab_eds\055c002b00010200_ersc_zpa_2_1.eds

Step 5

A window appears indicating the graphic image included in the EDS file. This image will be used if you want to show network topology in RSNetworx. You can change to your own icon if you wish. Click Next to Continue


Rockwell Automation's EDS Wizard

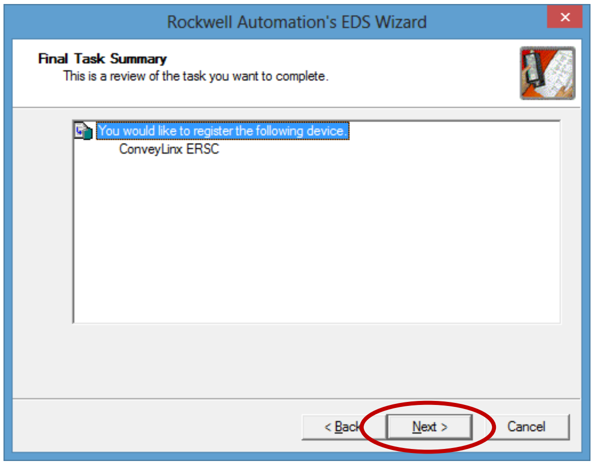
Change Graphic Image

You can change the graphic image that is associated with a device.

Product Types
Vendor Specific Type
ConveyLinx ERSC

Step 6

RSLogix5000 asks if you want to complete the installation. Click Next to proceed.


Rockwell Automation's EDS Wizard

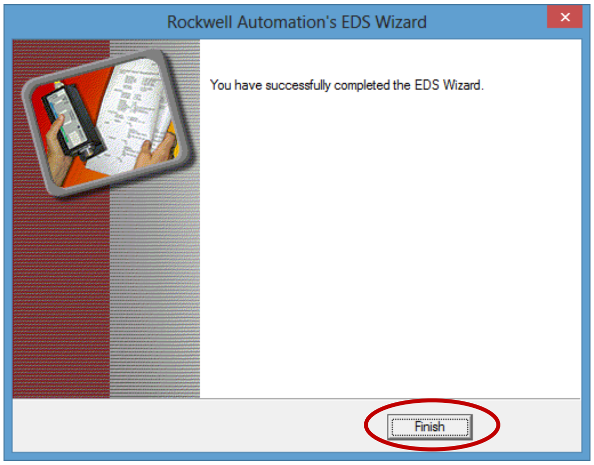
Final Task Summary

This is a review of the task you want to complete.

You would like to register the following device.
ConveyLinx ERSC

Step 7

RSLogix5000 lets you know when it is done by showing this window. Click Finish.



You have successfully completed the EDS Wizard.

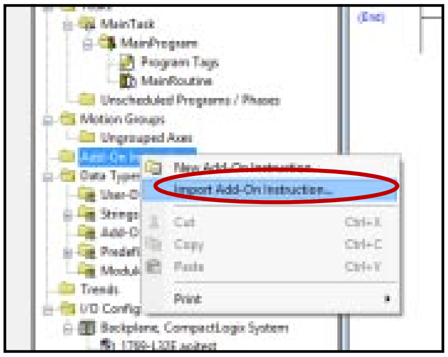
Simply repeat this process to install another EDS file into your RSLogix5000 environment.

INSTALLING AOI INTO RSLOGIX 5000

After your EDS files have been installed; the next procedure is to install the Add On Instruction (AOI) files that you downloaded.

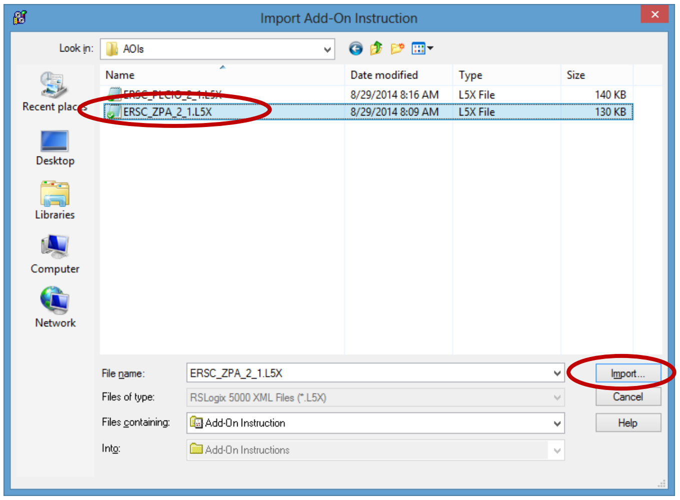
Step 1

Right click on the Add On Instruction folder in the explorer tree. From the pop-up menu select Import Add On Instruction…



Step 2

Navigate to the folder location where you downloaded your AOIs, select the file then click import. In this example we are importing the AOI for ZPA mode.



Look in: AOIs

Name	Date modified	Type	Size
ERSC_PLCIO_2_1.L5X	8/29/2014 8:16 AM	L5X File	140 KB
ERSC_ZPA_2_1.L5X	8/29/2014 8:09 AM	L5X File	130 KB

File name: ERSC_ZPA_2_1.L5X

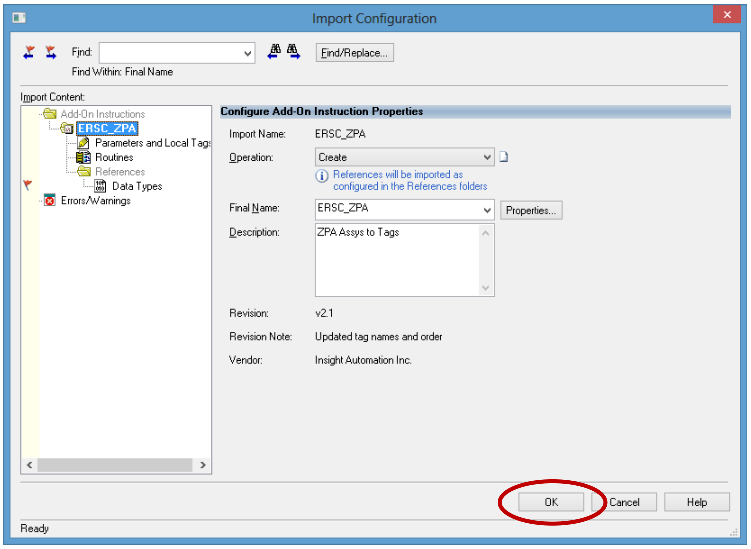
Files of type: RSLogix 5000 XML Files (*.L5X)

Files containing: Add-On Instruction

Into: Add-On Instructions

Step 3

A window will appear indicating the details about the AOI you are about to import. There should be no errors or warnings. Click OK to proceed with the import.



Configure Add-On Instruction Properties

Import Name: ERSC_ZPA

Operation: Create

References will be imported as configured in the References folders

Final Name: ERSC_ZPA

Description: ZPA Assys to Tags

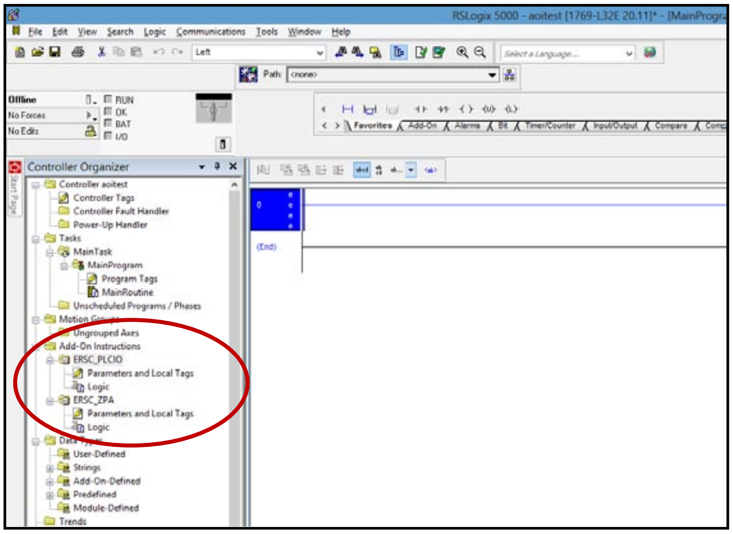
Revision: v2.1

Revision Note: Updated tag names and order

Vendor: Insight Automation Inc.

Simply repeat this process to import the other AOI for the ERSC in PLC I/O mode.

When you are done, these AOIs will appear in the explorer tree as shown.

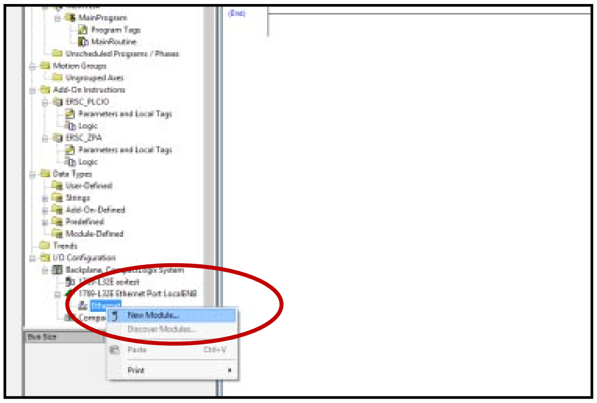


Adding Physical Modules to Your Project

For our example, we are going to add one ZPA module and one PLC I/O module to our current project. For our example we are assuming these have been configured with I.P. addresses 192.168.27.20 for the ZPA module and 192.168.27.21 for the PLC I/O module. The EDS files for each module type already include all the assembly and connection information. All you need to do is add the module type you want to use.

Step 1

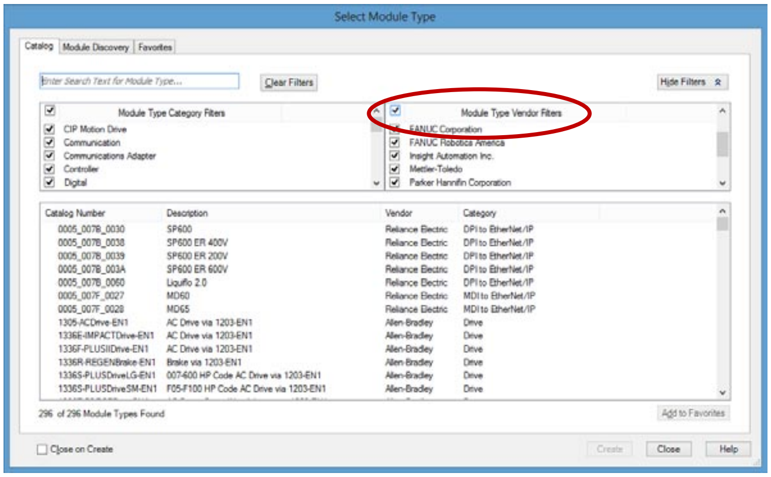
Right Click the Ethernet icon in the explorer tree and select New Module



Step 2

The Select Module Type window appears. We want to use filters so we don't have to scroll through a long list. Uncheck the Module Type Vendor Filters box to clear the list

Note your list may look different depending on what you have already installed in your RSLogix5000 environment.

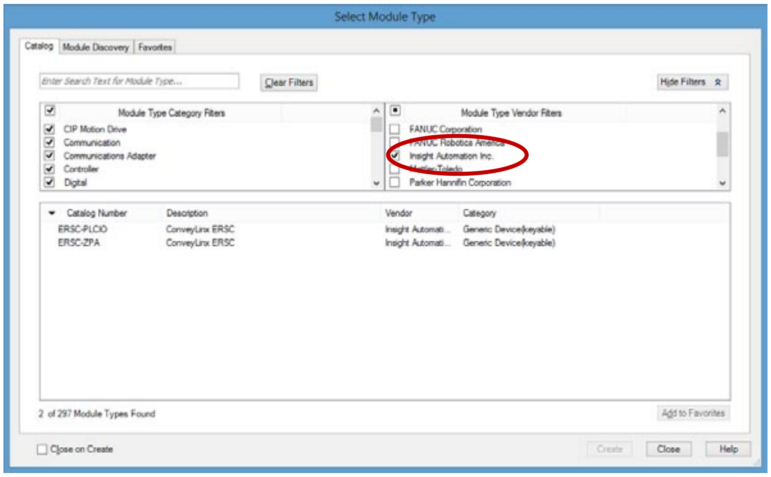


Catalog Number	Description	Vendor	Category
0005_007B_0030	SP600	Reliance Electric	DPI to EtherNet/IP
0005_007B_0038	SP600 ER 400V	Reliance Electric	DPI to EtherNet/IP
0005_007B_0039	SP600 ER 200V	Reliance Electric	DPI to EtherNet/IP
0005_007B_003A	SP600 ER 600V	Reliance Electric	DPI to EtherNet/IP
0005_007B_0060	Liquiflo 2.0	Reliance Electric	DPI to EtherNet/IP
0005_007F_0027	MD60	Reliance Electric	MDI to EtherNet/IP
0005_007F_0028	MD65	Reliance Electric	MDI to EtherNet/IP
1305-ACDrive-EN1	AC Drive via 1203-EN1	Allen-Bradley	Drive
1336E-IMPACTDrive-EN1	AC Drive via 1203-EN1	Allen-Bradley	Drive
1336F-PLUSIIDrive-EN1	AC Drive via 1203-EN1	Allen-Bradley	Drive
1336R-REGENBrake-EN1	Brake via 1203-EN1	Allen-Bradley	Drive
1336S-PLUSDriveLG-EN1	007-600 HP Code AC Drive via 1203-EN1	Allen-Bradley	Drive
1336S-PLUSDriveSM-EN1	F05-F100 HP Code AC Drive via 1203-EN1	Allen-Bradley	Drive

296 of 296 Module Types Found

Step 3

Scroll down the Vendor List until you see Insight Automation, then click its checkbox. All modules whose vendor is Insight Automation should appear in the list.



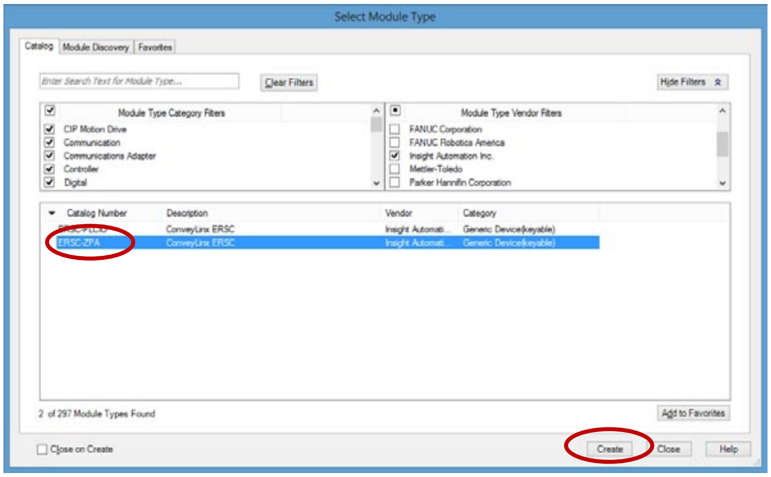
☑ Module Type Category Filters	■ Module Type Vendor Filters
☑ CIP Motion Drive	□ FANUC Corporation
☑ Communication	□ FANUC Robotics America
☑ Communications Adapter	☑ Insight Automation Inc.
☑ Controller	□ Mettler-Toledo
☑ Digital	□ Parker Hannifin Corporation

Catalog Number	Description	Vendor	Category
ERSC-PLCIO	ConveyLinx ERSC	Insight Automati...	Generic Device(keyable)
ERSC-ZPA	ConveyLinx ERSC	Insight Automati...	Generic Device(keyable)

2 of 297 Module Types Found

Step 4

For our example, we are going to select the ERSC-ZPA item from the list and click Create.



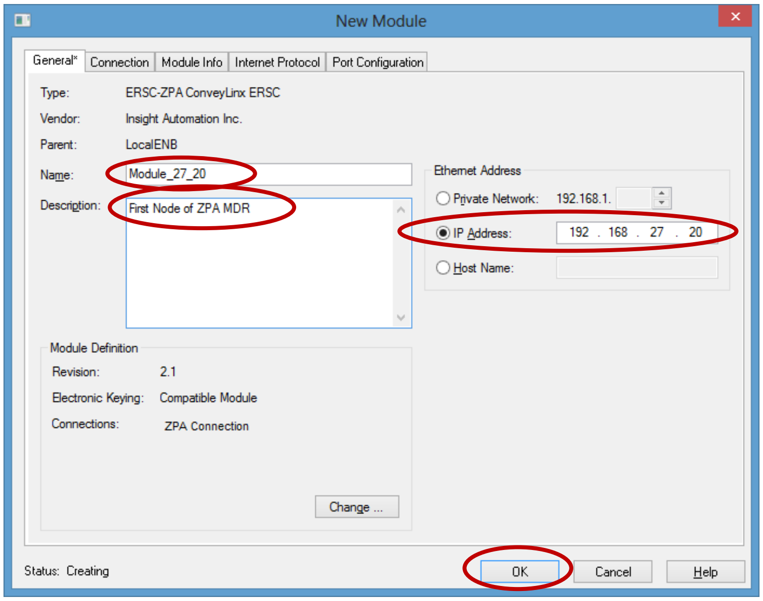
☑ Module Type Category Filters	■ Module Type Vendor Filters
☑ CIP Motion Drive	□ FANUC Corporation
☑ Communication	□ FANUC Robotics America
☑ Communications Adapter	☑ Insight Automation Inc.
☑ Controller	□ Mettler-Toledo
☑ Digital	□ Parker Hannifin Corporation

Catalog Number	Description	Vendor	Category
ERSC-PLCIO	ConveyLinx ERSC	Insight Automati...	Generic Device(keyable)
ERSC-ZPA	ConveyLinx ERSC	Insight Automati...	Generic Device(keyable)

2 of 297 Module Types Found

Step 5

The New Module window appears. For our example, we are creating the ZPA module for our project. Enter the name, description (optional), and I.P. address and then click OK



Type: ERSC-ZPA ConveyLinx ERSC
Vendor: Insight Automation Inc.
Parent: LocalENB
Name: Module_27_20
Description: First Node of ZPA MDR

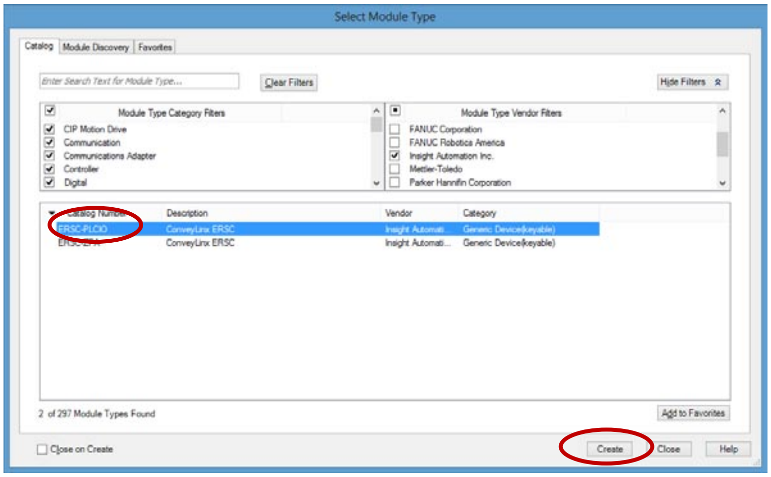
Ethernet Address
○ Private Network: 192.168.1.
● IP Address: 192 . 168 . 27 . 20
○ Host Name:

Module Definition
Revision: 2.1
Electronic Keying: Compatible Module
Connections: ZPA Connection

Status: Creating

Step 6

To add another module, repeat the process to get to the Select Module Type window. For our example, we want to create the PLC I/O module for our project. Click on the ERSC-PLCIO selection and click Create.


Select Module Type

Module Type Category Filters: ☑ CIP Motion Drive, ☑ Communication, ☑ Communications Adapter, ☑ Controller, ☑ Digital

Module Type Vendor Filters: ☐ FANUC Corporation, ☐ FANUC Robotics America, ☑ Insight Automation Inc., ☐ Mettler-Toledo, ☐ Parker Hannifin Corporation

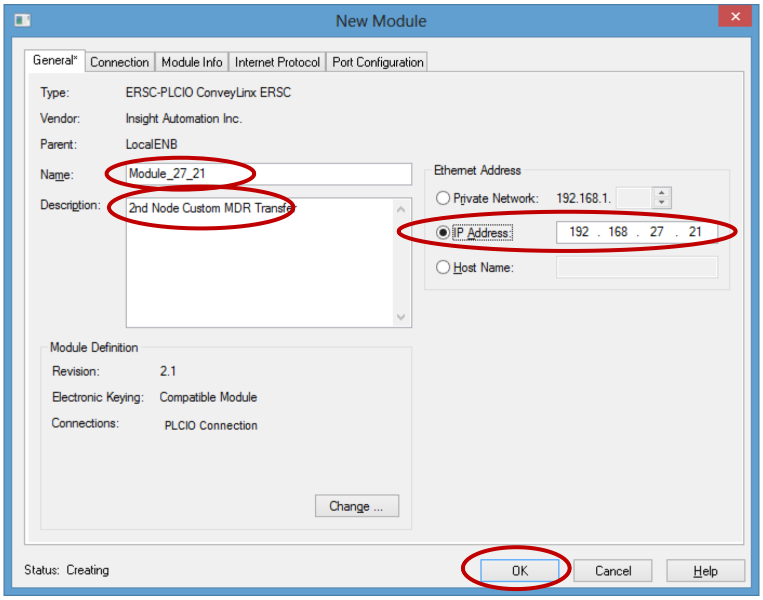
Catalog Number	Description	Vendor	Category
ERSC-PLCIO	ConveyLinx ERSC	Insight Automati...	Generic Device(keyable)
ERSC-ZPA	ConveyLinx ERSC	Insight Automati...	Generic Device(keyable)

2 of 297 Module Types Found

☐ Close on Create

Step 7

This is a repeat of Step 5 except the information entered is for the PLC I/O module for our project.


New Module

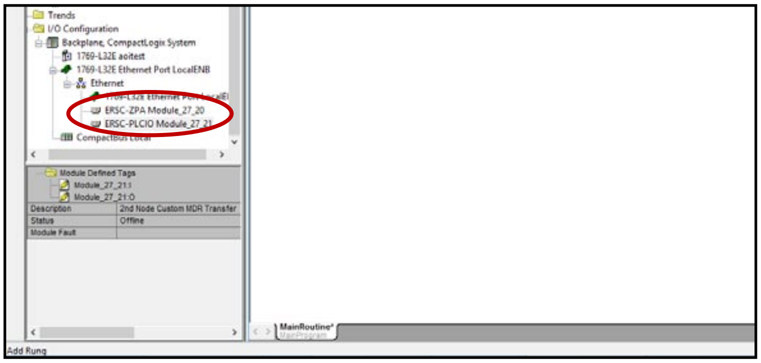
Type: ERSC-PLCIO ConveyLinx ERSC
Vendor: Insight Automation Inc.
Parent: LocalENB
Name: Module_27_21
Description: 2nd Node Custom MDR Transfe

Ethernet Address
○ Private Network: 192.168.1.
● IP Address: 192 . 168 . 27 . 21
○ Host Name:

Module Definition
Revision: 2.1
Electronic Keying: Compatible Module
Connections: PLCIO Connection

Status: Creating

When you are done, you should see both new modules in the explorer tree


Trends
I/O Configuration
Backplane, CompactLogix System
1769-L32E aoitest
1769-L32E Ethernet Port LocalENB
Ethernet
1769-L32E Ethernet Port LocalENB
ERSC-ZPA Module_27_20
ERSC-PLCIO Module_27_21
CompactBus Local

Module Defined Tags
Module_27_21:I
Module_27_21:O

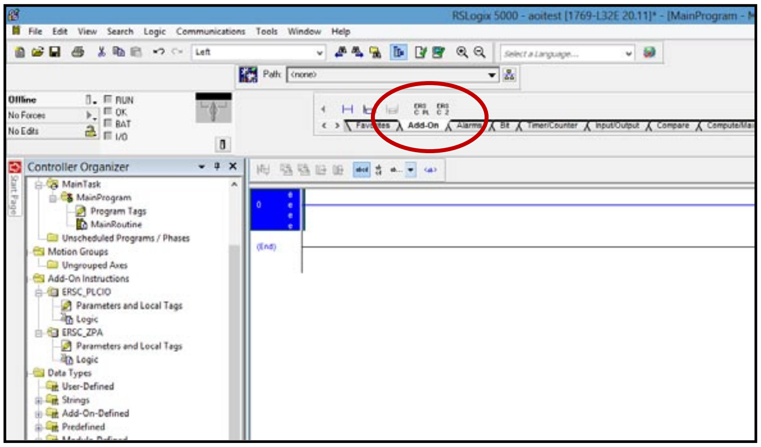
Description	2nd Node Custom MDR Transfer
Status	Offline
Module Fault	

ASSIGNING NEW MODULES TO AOI

The next step in the procedure is to create an instance of the appropriate AOI for each physical module that you created.

Step 1

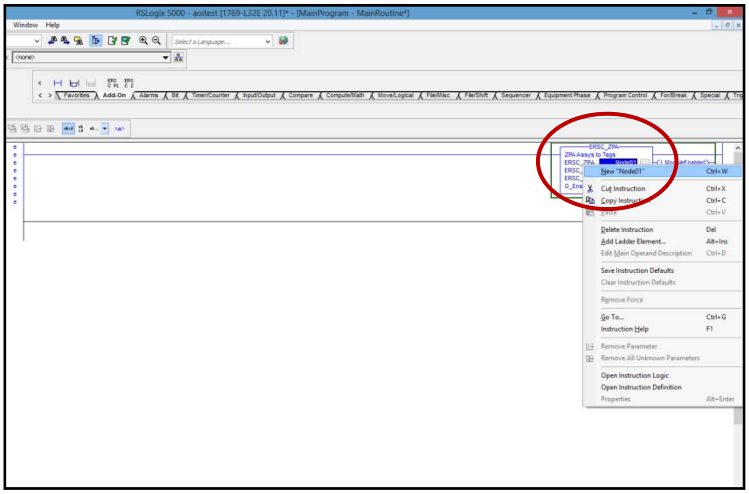
Locate the AOIs and place in your ladder diagram. For our example we are selecting the ERSC_ZPA instruction



Step 2

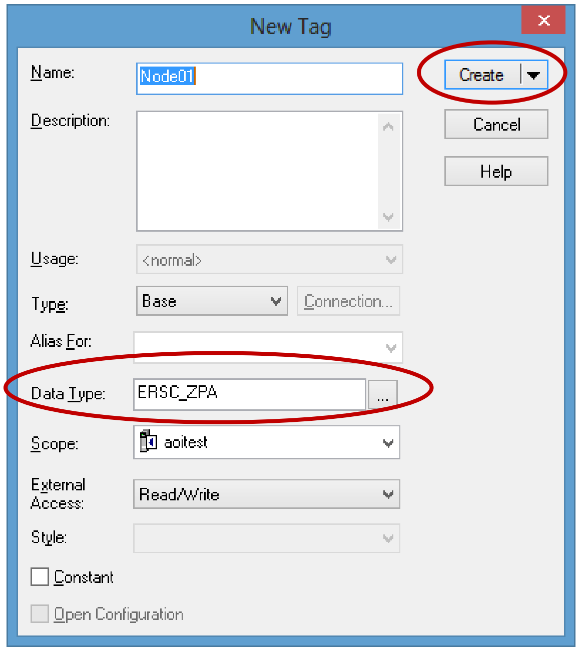
Once the instruction has been added to the ladder, we need to create a tag that will be how you access the modules data.

For our example we entered "Node01" and then created the new tag



Step 3

This is the typical New Tag window you invoke from the ladder diagram screen. Note that the DataType defaults to the AOI's data type. Click Create to create the new tag



New Tag

Name: Node01

Description:

Usage: <normal>

Type: Base

Alias For:

Data Type: ERSC_ZPA

Scope: aoitest

External Access: Read/Write

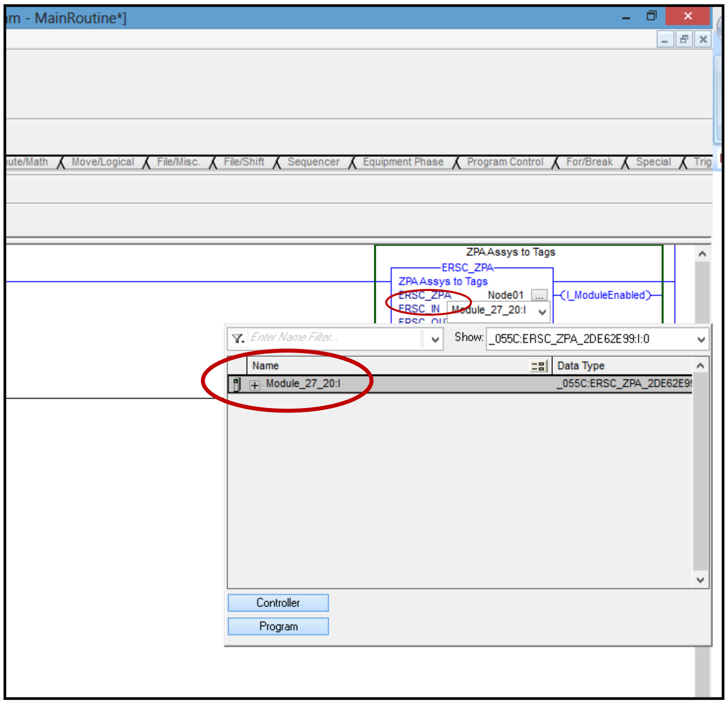
Style:

□ Constant

□ Open Configuration

Step 4

The AOI requires two other parameters; "ERSC_IN" for the data coming from the module and "ERSC_OUT for data coming from the PLC to the module. These will be from the physical modules we previously added. Here we will add the ERSC_IN parameter by clicking the drop down box arrow will automatically show all tags that match the data type for the ERSC_IN parameter. In this case, "Module_27_20" is the only ZPA module we created, so it is the only selection. Double click this and it will be assigned to the Node01 instance of our ZPA AOI's "ERSC_IN" parameter.



m - MainRoutine*]

ZPA Assys to Tags

ERSC_ZPA

ZPA Assys to Tags

ERSC_ZPA Node01

ERSC_IN Module_27_20:I

ERSC_OUT

(I_ModuleEnabled)

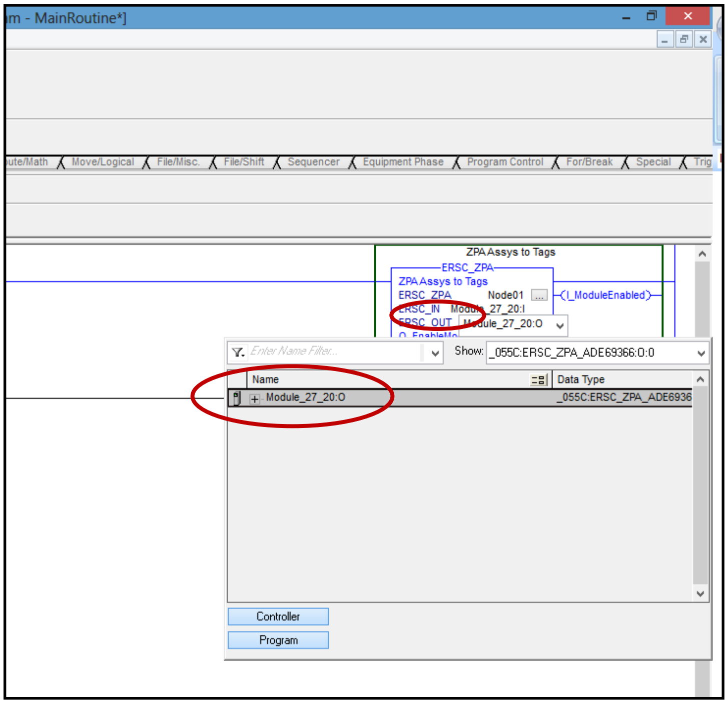
Enter Name Filter... Show: _055C:ERSC_ZPA_2DE62E99:I:0

Name	Data Type
Module_27_20:I	_055C:ERSC_ZPA_2DE62E9

Step 5

Similarly to Step 4, we need to do the select the physical module for the ERSC_OUT parameter. Clicking the drop down box arrow will show all physical modules that have the matching data type for the ERSC_OUT parameter.

Double click this and it will be assigned to the Node01 instance of our ZPA AOI's "ERSC_OUT" parameter.

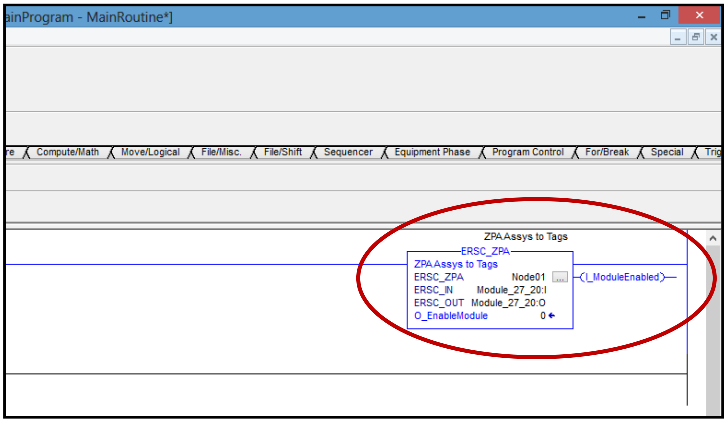


ZPA Assys to Tags	
ERSC_ZPA	Node01
ERSC_IN	Module_27_20:I
ERSC_OUT	Module_27_20:O

Show: _055C:ERSC_ZPA_ADE69366:0:0

Name	Data Type
Module_27_20:O	_055C:ERSC_ZPA_ADE6936

At this point, the AOI has been set up to use in your logic program. All of the tags associated with using the Module_27_20 in ZPA mode are in the structured tag "Node01".



ZPA Assys to Tags	
ERSC_ZPA	Node01
ERSC_IN	Module_27_20:I
ERSC_OUT	Module_27_20:O
O_EnableModule	0

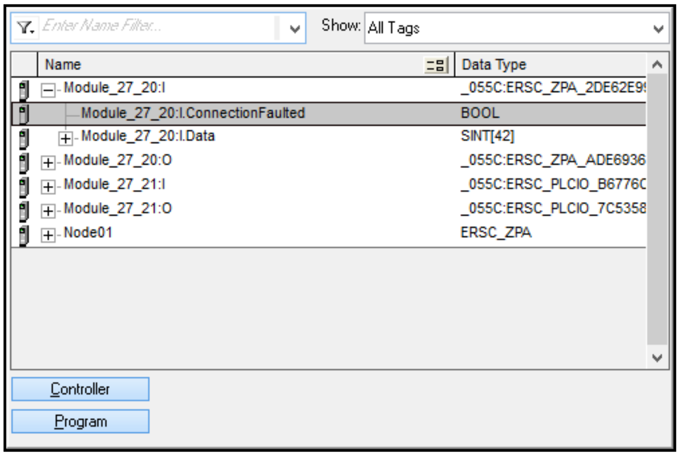
Enabling the Module for Operation

Before using the AOI in your program, you need to add some logic to enable the outputs on the physical module. Both EDS files (ZPA and PLC I/O) use the assemblies that require the PLC Enable function to instruct the ERSC module to process output data from the PLC.

This topic is covered in PLC Developer's Guide under the heading of ZPA Mode with PLC Enable.

Another function that is built-in when you created the module is indication of whether the PLC is communicating with the module. For example, for the ZPA module we created (Module_27_20); if you look in the Controller tags for the input data coming from the module, there is a Boolean value that indicates "Connection Faulted".

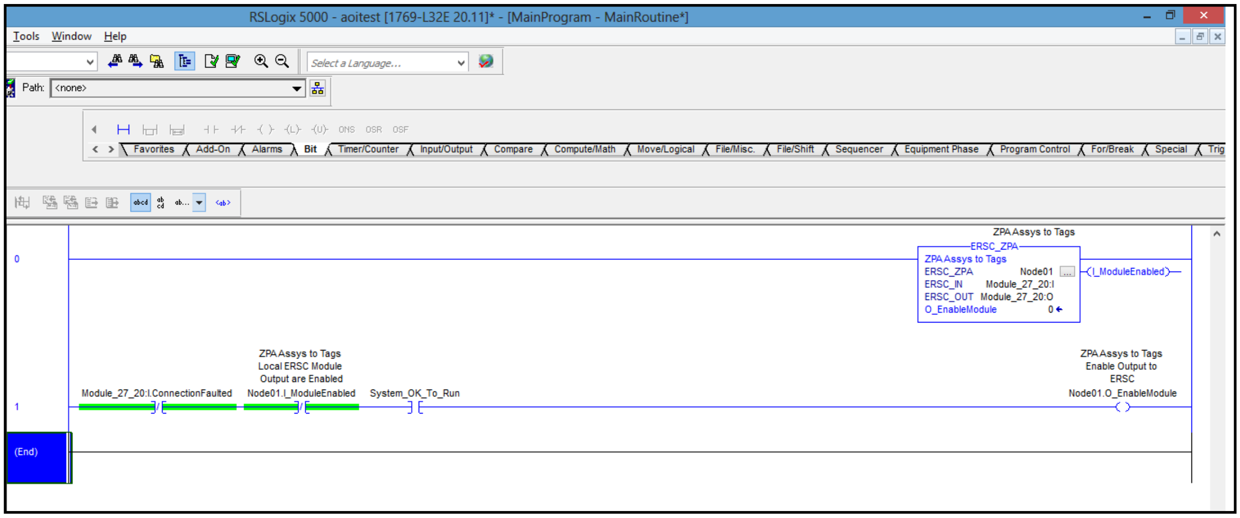
From our example, when you expand the "Module_27_20:I" structure, there is a BOOL that indicates "Connection Faulted". This tag can be used in your logic to assure connection is OK prior to enabling the module.



Show: All Tags

Name	Data Type
Module_27_20:I	_055C:ERSC_ZPA_2DE62E9
Module_27_20:I.ConnectionFaulted	BOOL
Module_27_20:I.Data	SINT[42]
Module_27_20:O	_055C:ERSC_ZPA_ADE6936
Module_27_21:I	_055C:ERSC_PLCIO_B6776C
Module_27_21:O	_055C:ERSC_PLCIO_7C5358
Node01	ERSC_ZPA

We recommend a simple rung of logic for each module that will allow the module to be enabled when its connection is OK and when "the system" is OK to start. This "system OK" state is wholly up to you as the programmer to determine.


RSLogix 5000 - aoitest [1769-L32E 20.11]* - [MainProgram - MainRoutine*]
Tools Window Help
Select a Language...
Path: <none>
ONS OSR OSF
Favorites Add-On Alarms Bit Timer/Counter Input/Output Compare Compute/Math Move/Logical File/Misc. File/Shift Sequencer Equipment Phase Program Control For/Break Special Trig

```
0   ZPA Assys to Tags
    ERSC_ZPA
    ZPA Assys to Tags
    ERSC_ZPA        Node01           -(I_ModuleEnabled)-
    ERSC_IN         Module_27_20:I
    ERSC_OUT        Module_27_20:O
    O_EnableModule  0

                                            ZPA Assys to Tags                                ZPA Assys to Tags
                                            Local ERSC Module                                Enable Output to
                                            Output are Enabled                               ERSC
1   Module_27_20:I.ConnectionFaulted   Node01.I_ModuleEnabled   System_OK_To_Run   Node01.O_EnableModule
    ]/[                                ]/[                      ] [                ( )

(End)
```

For our example, we added a N.C. contact for the module’s “ConnectionFaulted” tag, a N.C. contact for the AOI’s tag that indicates that the module is not enabled, and a N.O. contact for the programmer’s “System_OK_to_Run” condition as previously described. When this logic becomes true, then AOI’s input to “EnableModule” is energized. When the module becomes enabled, the “ModuleEnabled” contact becomes true so that the “EnableModule” input to the AOI does not need to be held ON to keep the module enabled.

ZPA AOI Tag Descriptions

The following chart lists each tag made available in the ERSC-ZPA AOI along with the register reference from the PLC Developer's Guide 4.7 and later or ConveyLinx-Ai PLC Developer's Guide version 1.0 and later.

Tag Name	Data Type	Developer's Guide Register	Bit	Description
I_ArrivalUPZn	BOOL	5-0	AOI Logic	Arrival at Local Upstream Zone – Only active if zone becomes occupied and it has been set to accumulate
I_ArrivalDNZn	BOOL	5-1	AOI Logic	Arrival at Local Downstream Zone – Only active if zone becomes occupied and it has been set to accumulate
I_ControlPortPin3_Left	BOOL	5-18	1	Left Control Port Pin 3 Energized
I_ControlPortPin3_Right	BOOL	5-18	3	Right Control Port Pin 3 Energized
I_ControlPortPin4_Left	BOOL	5-18	5	Left Control Port Pin 4 Energized
I_ControlPortPin4_Right	BOOL	5-18	7	Right Control Port Pin 4 Energized
I_ConveyStopByLeftControlPort	BOOL	5-20	8	ConveyStop Activated at Local Left Control Port
I_ConveyStopByLostConnection	BOOL	5-20	6	ConveyStop Activated because of Lost Connection
I_ConveyStopByPLCCmd	BOOL	5-20	7	ConveyStop Activated because of PLC Command
I_ConveyStopByPLCDisconnect	BOOL	5-20	10	ConveyStop Activated because of Lost PLC Connection
I_ConveyStopByRemoteModule	BOOL	5-20	5	ConveyStop Activated by another module in Stop Group
I_ConveyStopByRightControlPort	BOOL	5-20	9	ConveyStop Activated at Local Right Control Port
I_GetForwardTracking	DINT	5-14 (MSW) 5-15 (LSW)	-	Current Forward Tracking Value at Induct to Local Upstream Zone
I_Heartbeat	BOOL	5-18	15	Module Heartbeat
I_JamAtUPZn	BOOL	5-6	5	Sensor Jam at Local Upstream Zone
I_JamAtDNZn	BOOL	5-7	5	Sensor Jam at Local Downstream Zone
I_ModuleEnabled	BOOL	-	-	Local ERSC Module Output are Enabled
I_ModuleFault	BOOL	5-6 5-7	AOI Logic	Module Fault Active (Logical OR of bits 2,4,and 7 from 5-6 and bits 2 and 7 from 5-7
I_ModuleStatus	DINT	5-6 (MSW) 5-7 (LSW)		Modules Status Words 1 and 2
I_MtrError_Left	BOOL	5-6	3	Left Motor Error is Active
I_MtrError_Right	BOOL	5-7	3	Right Motor Error is Active
I_SensorPortPin3_Left	BOOL	5-18	0	Left Sensor Port Pin 3 Energized
I_SensorPortPin3_Right	BOOL	5-18	2	Right Sensor Port Pin 3 Energized
I_SensorPortPin4_Left	BOOL	5-18	4	Left Sensor Port Pin 4 Energized
I_SensorPortPin4_Right	BOOL	5-18	6	Right Sensor Port Pin 4 Energized
I_TrackingDNZn	DINT	5-10 (MSW) 5-11 (LSW)	-	Current Tracking Value for Arrival at Local Downstream Zone
I_TrackingUPZn	DINT	5-8 (MSW) 5-9 (LSW)	-	Current Tracking Value for Arrival at Local Upstream Zone
I_ZoneStatusDnZn	SINT	5-1	Lo Byte	Zone Status Local Downstream Zone Forward Direction
I_ZoneStatusUpZn	SINT	5-0	Lo Byte	Zone Status Local Uptream Zone Forward Direction
O_AccForArrivalDNZn	BOOL	6-5	0	Set Local Downstream Zone to Accumulate
O_AccForArrivalUPZn	BOOL	6-4	0	Set Local Upstream Zone to Accumulate
O_AccumAdjUpstreamToDNZn	BOOL	6-5	8	Accumulate Adjacent Upstream to Local Downstream Zone
O_AccumAdjUpstreamToUPZn	BOOL	6-4	8	Accumulate Adjacent Upstream to Local Upstream Zone
O_ClearJamDNZn	BOOL	6-21	0	Clear Jam at Local Upstream Zone
O_ClearJamUPZn	BOOL	6-20	0	Clear Jam at Local Downstream Zone
O_ClearMotorError	BOOL	6-16	0	Clear Motor Error Left & Right
O_ConfArrivalAdjDownstreamToDNZn	BOOL	6-5	9	Confirm Downstream Arrival for Local Downstream Zone
O_ConfArrivalAdjDownstreamToUPZn	BOOL	6-4	9	Confirm Downstream Arrival for Local Upstream Zone
O_ControlPortOutputLeft	BOOL	6-17	1	Set Left Control Port Output
O_ControlPortOutputRight	BOOL	6-17	3	Set Right COntrol Port Output

Tag Name	Data Type	Developer's Guide Register	Bit	Description
O_ConveyStopCommand	INT	6-19	-	Set Local ConveyStop Command Word
O_DAModeCmdDNZn	SINT	6-23	Lo Byte	Direction & Accumulation Mode Command Byte for Downstream Zone
O_DAModeCmdUPZn	SINT	6-22	Lo Byte	Direction & Accumulation Mode Command Byte for Upstream Zone
O_DAModeValueDNZn	SINT	6-23	Hi Byte	Direction & Accumulation Mode Data Byte for Downstream Zone
O_DAModeValueUPZn	SINT	6-22	Hi Byte	Direction & Accumulation Mode Data Byte for Upstream Zone
O_EnableModule	BOOL	-	AOI Logic	Enable Output to ERSC
O_JogFwdDNZn	BOOL	6-5	10	Jog Forward for Local Downstream Zone
O_JogFwdUpZn	BOOL	6-4	10	Jog Forward for Local Upstream Zone
O_JogRevDNZn	BOOL	6-5	11	Jog Reverse for Local Downstream Zone
O_JogRevUPZn	BOOL	6-4	11	Jog Reverse for Local Upstream Zone
O_ReleaseDNZn	BOOL	6-8	AOI Logic	Release and Accumulate on Next at Downstream Zone – Automatically increments release counter
O_ReleaseUPZn	BOOL	6-9	AOI Logic	Release and Accumulate on Next at Upstream Zone – Automatically increments release counter
O_SpeedLeftMtr	INT	6-6	-	Set Left Motor Speed Reference
O_SpeedRightMtr	INT	6-7	-	Set Right Motor Speed Reference
O_StatusDownstreamDischarge	INT	6-11	-	Set Downstream Discharge Zone Status Value
O_StatusUpstreamInduct	INT	6-10	-	Set Upstream Induct Zone Status Value
O_TrackingDNZn	DINT	6-2 (MSW) 6-3 (LSW)	-	Set Tracking Value for Local Downstream Zone
O_TrackingInductFwd	DINT	6-12 (MSW) 6-13 (LSW)	-	Set Forward Induct Tracking Value
O_TrackingUPZn	DINT	6-0 (MSW) 6-1 (LSW)	-	Set Tracking Value for Local Upstream Zone
O_WakeUpDNZn	BOOL	6-5	12	Wakeup Local Downstream Zone
O_WakeUpUPZn	BOOL	6-4	12	Wakeup Local Upstream Zone

PLC I/O AOI Tag Descriptions

The following chart lists each tag made available in the ERSC-PLCIO AOI along with the register reference from the PLC Developer's Guide 4.7 and later or ConveyLinx-Ai PLC Developer's Guide version 1.0 and later.

Tag Name	Data Type	Developer's Guide Register	Bit	Decription
I_ControlPortPin3_Left	BOOL	7-1	1	Port Inputs
I_ControlPortPin3_Right	BOOL	7-1	3	Port Inputs
I_ControlPortPin4_Left	BOOL	7-1	5	Port Inputs
I_ControlPortPin4_Right	BOOL	7-1	7	Port Inputs
I_ConveyStopByLeftControlPort	BOOL	7-0	8	ConveyStop
I_ConveyStopByLostConnection	BOOL	7-0	6	ConveyStop
I_ConveyStopByPLCDisconnect	BOOL	7-0	7	ConveyStop
I_ConveyStopByPLCCmd	BOOL	7-0	10	ConveyStop
I_ConveyStopByRemoteModule	BOOL	7-0	5	ConveyStop
I_ConveyStopByRightControlPort	BOOL	7-0	9	ConveyStop
I_DigitalMtrOverCurrent_Left	BOOL	7-12	14	Left Motor Port as Digital
I_DigitalMtrOverCurrent_Right	BOOL	7-13	14	Right Motor Port as Digital
I_DigitalMtrShortCkt_Left	BOOL	7-12	12	Left Motor Port as Digital
I_DigitalMtrtShortCkt_Right	BOOL	7-13	12	Right Motor Port as Digital
I_DownstreamModuleStatus	SINT	7-15	Lo Byte	Module Status
I_Heartbeat	BOOL	7-1	15	Port Inputs
I_ModuleEnabled	BOOL	-	-	Module Status
I_ModuleVoltage	REAL	7-3	-	Module Status
I_MtrCurrent_Left	REAL	7-4	-	Left Motor Status
I_MtrCurrent_Right	REAL	7-8	-	Right Motor Status
I_MtrFreq_Left	INT	7-5	-	Left Motor Status
I_MtrFreq_Right	INT	7-9	-	Right Motor Status
I_MtrRunningCCW_Left	BOOL	7-7	AOI Logic	Left Motor Status
I_MtrRunningCCW_Right	BOOL	7-11	AOI Logic	Right Motor Status
I_MtrRunningCW_Left	BOOL	7-7	AOI Logic	Left Motor Status
I_MtrRunningCW_Right	BOOL	7-11	AOI Logic	Right Motor Status
I_MtrStatus_Left	INT	7-7	-	Left Motor Status
I_MtrStatus_Right	INT	7-11	-	Right Motor Status
I_SensorDetectLeftPort	BOOL	7-2	1	Sensor Port Status
I_SensorDetectRightPort	BOOL	7-2	0	Sensor Port Status
I_SensorPortPin3_Left	BOOL	7-1	0	Port Inputs
I_SensorPortPin3_Right	BOOL	7-1	2	Port Inputs
I_SensorPortPin4_Left	BOOL	7-1	4	Port Inputs
I_SensorPortPin4_Right	BOOL	7-1	6	Port Inputs
I_ServoCmdStatus_Left	BOOL	7-21	2	Left Servo Function
I_ServoCmdStatus_Right	BOOL	7-22	2	Right Servo Function
I_ServoLastCmdComplete_Left	BOOL	7-21	0	Left Servo Function
I_ServoLastCmdComplete_Right	BOOL	7-22	0	Right Servo Function
I_ServoPosition_Left	INT	7-19		Left Servo Function
I_ServoPosition_Right	INT	7-20		Right Servo Function
I_ServoResetStatus_Left	BOOL	7-21	1	Left Servo Function
I_ServoResetStatus_Right	BOOL	7-22	1	Right Servo Function
I_TemperatureCalculated_Left	SINT	7-6	Hi Byte	Left Motor Status
I_TemperatureCalculated_Right	SINT	7-10	Hi Byte	Right Motor Status
I_TemperatureOnBoard_Left	SINT	7-6	Lo Byte	Left Motor Status
I_TemperatureOnBoard_Right	SINT	7-10	Lo Byte	Right Motor Status
I_UpstreamModuleStatus	SINT	7-14	Lo Byte	ZPA Status

I_UpstreamTracking	DINT	7-16 (MSW) 7-17 (LSW)		ZPA Tracking
O_BrakeMethod_Left	SINT	8-5	Lo Byte	Left Motor Control
O_BrakeMethod_Right	SINT	8-8	Lo Byte	Right Motor Control
O_ClearMotorError	BOOL	8-16		Motor Control
O_ControlPortOutput_Left	BOOL	8-3	1	Left Motor Control
O_ControlPortOutput_Right	BOOL	8-3	3	Right Motor Control
O_ControlPortPin3Mask_Left	BOOL	8-19	1	Sensor/Control Port Configuration
O_ControlPortPin3Mask_Right	BOOL	8-19	3	Sensor/Control Port Configuration
O_ControlPortPin4Mask_Left	BOOL	8-19	5	Sensor/Control Port Configuration
O_ControlPortPin4Mask_Right	BOOL	8-19	7	Sensor/Control Port Configuration
O_ConveyStopCommand	INT	8-0		ConveyStop
O_DischargeTracking	DINT	8-20 (MSW) 8-21 (LSW)		ZPA Tracking
O_DownstreamStatus	SINT	8-17	Lo Byte	ZPA Status
O_EnableModule	BOOL	-	-	Module Control
O_LeftMtrDigitalPin3	BOOL	8-1	0	Left Motor Port Digital Control
O_LeftMtrDigitalPin4	BOOL	8-1	1	Left Motor Port Digital Control
O_LeftMtrDigitalPin5	BOOL	8-1	2	Left Motor Port Digital Control
O_MtrAccel_Left	INT	8-12		Left Motor Control
O_MtrAccel_Right	INT	8-14		Right Motor Control
O_MtrDecel_Left	INT	8-13		Left Motor Control
O_MtrDecel_Right	INT	8-15		Right Motor Control
O_RightMtrDigitalPin3	BOOL	8-2	0	Right Motor Port Digital Control
O_RightMtrDigitalPin4	BOOL	8-2	1	Right Motor Port Digital Control
O_RightMtrDigitalPin5	BOOL	8-2	2	Right Motor Port Digital Control
O_RunMtrFwd_Left	BOOL	8-4	1	Left Motor Control
O_RunMtrFwd_Right	BOOL	8-7	1	Right Motor Control
O_RunMtrRev_Left	BOOL	8-4	8	Left Motor Control
O_RunMtrRev_Right	BOOL	8-7	8	Right Motor Control
O_SensorPortPin3Mask_Left	BOOL	8-19	0	Sensor/Control Port Configuration
O_SensorPortPin3Mask_Right	BOOL	8-19	2	Sensor/Control Port Configuration
O_SensorPortPin4Mask_Left	BOOL	8-19	4	Sensor/Control Port Configuration
O_SensorPortPin4Mask_Right	BOOL	8-19	6	Sensor/Control Port Configuration
O_ServoCmdPulses_Left	INT	8-23		Left Servo Function
O_ServoCmdPulses_Right	INT	8-25		Right Servo Function
O_ServoGoCmd_Left	BOOL	8-24	1	Left Servo Function
O_ServoGoCmd_Right	BOOL	8-26	1	Right Servo Function
O_ServoZero_Left	BOOL	8-24	0	Left Servo Function
O_ServoZero_Right	BOOL	8-26	0	Right Servo Function
O_SetLeftMtrPortDigital	BOOL	8-1	15	Left Motor Port Digital Control
O_SetRightMtrPortDigital	BOOL	8-2	15	Right Motor Port Digital Control
O_SpeedMethod_Left	SINT	8-6	Lo Byte	Left Motor Control
O_SpeedMethod_Right	SINT	8-9	Lo Byte	Right Motor Control
O_SpeedReference_Left	INT	8-10		Left Motor Control
O_SpeedReference_Right	INT	8-11		Right Motor Control
O_UpstreamStatus	SINT	8-18	Lo Byte	ZPA Status

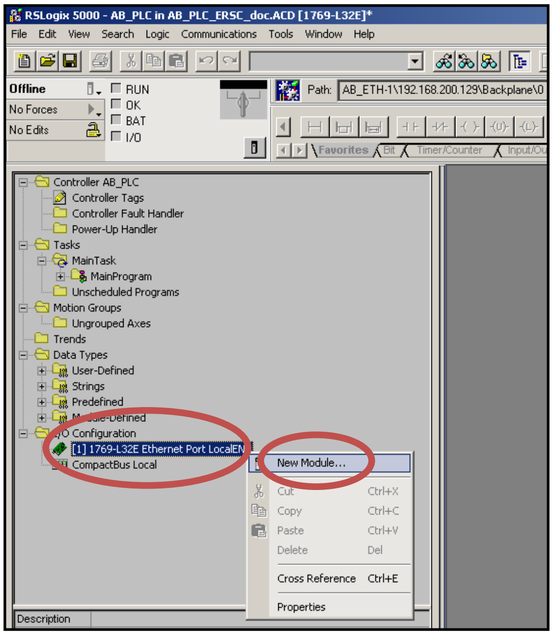
Use generic Ethenret Module Method

This section previously appeared in PLC Developer's Guide Rev 4.6 and earlier.

When using the Generic Ethernet Module construct in RSLogix 5000, you must supply configuration information about the device you are trying to connect. The following sections show the step by step procedure to connect a module for each set of I/O Assemblies described in the *PLC Developer's Guides.*

New Module for Assemblies 5 and 6

This section will provide the set-by-step procedure for creating an instance of an *ERSC* into the I/O configuration for an Allen-Bradley CompactLogix processor in RSLogix 5000 software.



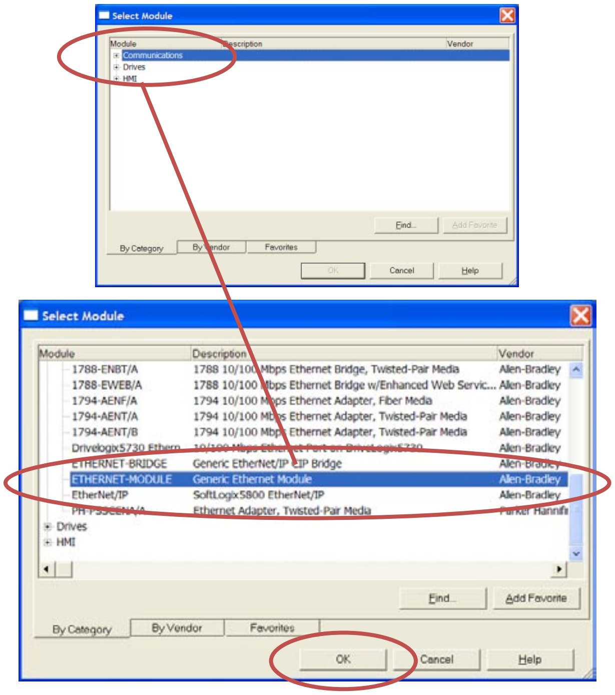
Step #1

Add a New Module to the processor's I/O configuration by highlighting the processor's local Ethernet port in the I/O configuration tree.

Right-clicking will show the context menu. Select "New Module…"

Step #2

From the Select Module pop-up window, expand the Communications tree and select “Generic Ethernet Module” and click OK



Select Module

Module	Description	Vendor
⊞ Communications		
⊞ Drives		
⊞ HMI		

Select Module

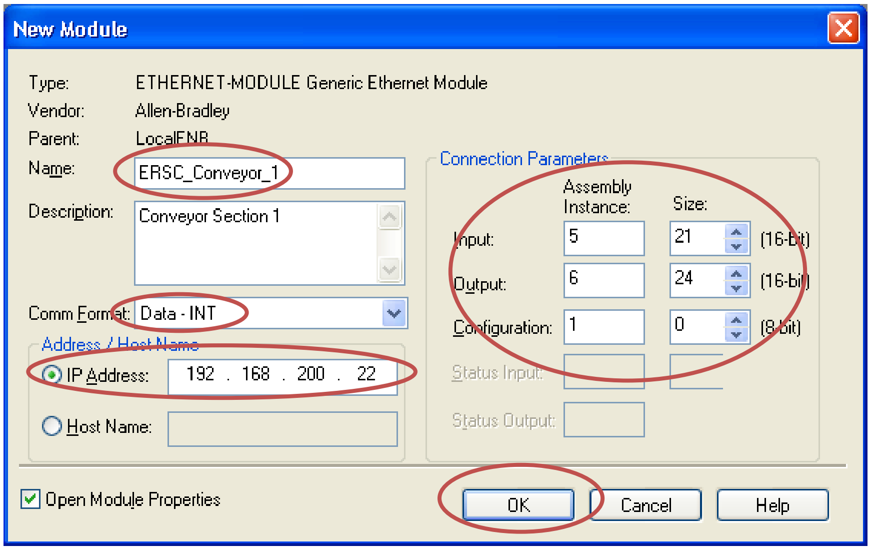
Module	Description	Vendor
1788-ENBT/A	1788 10/100 Mbps Ethernet Bridge, Twisted-Pair Media	Allen-Bradley
1788-EWEB/A	1788 10/100 Mbps Ethernet Bridge w/Enhanced Web Servic...	Allen-Bradley
1794-AENF/A	1794 10/100 Mbps Ethernet Adapter, Fiber Media	Allen-Bradley
1794-AENT/A	1794 10/100 Mbps Ethernet Adapter, Twisted-Pair Media	Allen-Bradley
1794-AENT/B	1794 10/100 Mbps Ethernet Adapter, Twisted-Pair Media	Allen-Bradley
Drivelogix5730 Ethern	10/100 Mbps Ethernet Port on DriveLogix5730	Allen-Bradley
ETHERNET-BRIDGE	Generic EtherNet/IP CIP Bridge	Allen-Bradley
ETHERNET-MODULE	Generic Ethernet Module	Allen-Bradley
EtherNet/IP	SoftLogix5800 EtherNet/IP	Allen-Bradley
PH-PSSCENA/A	Ethernet Adapter, Twisted-Pair Media	Parker Hannifi
⊞ Drives		
⊞ HMI		

Step #3

Fill in the Name field. This will be the *ModuleName* that will appear in your program Tag Database for any addressing.

Select Comm Format to be “Data – INT” and fill in the I.P. address of the *ERSC*.

Fill in the Connection Parameters as shown.



New Module

Type: ETHERNET-MODULE Generic Ethernet Module

Vendor: Allen-Bradley

Parent: LocalENB

Name: ERSC_Conveyor_1

Description: Conveyor Section 1

Comm Format: Data - INT

Address / Host Name

● IP Address: 192 . 168 . 200 . 22

○ Host Name:

Connection Parameters

	Assembly Instance:	Size:	
Input:	5	21	(16-bit)
Output:	6	24	(16-bit)
Configuration:	1	0	(8-bit)
Status Input:			
Status Output:			

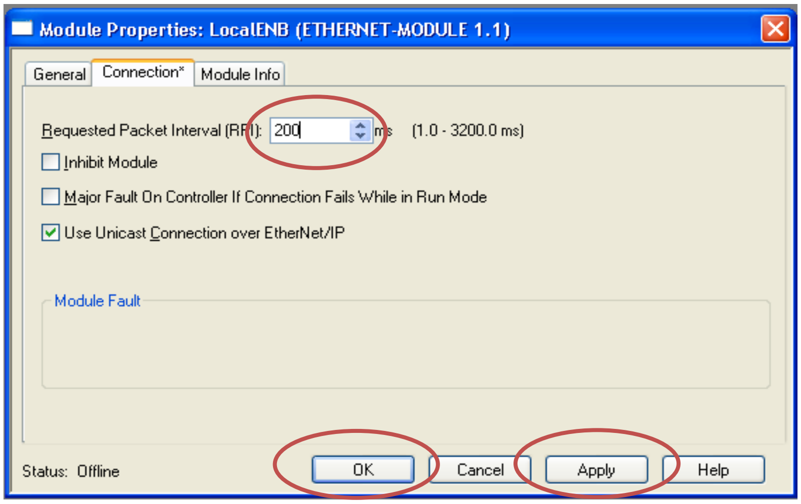
☑ Open Module Properties

It is <u>very important</u> to select *Comm Format* data type to be INT or interface to *ERSC* will not operate correctly!

Step #4

Set RPI to a value no lower than 10ms. 200 ms is typical for ZPA Interface. You may also optionally select Unicast Connection.

Click “Apply” to update the value and then “OK” to exit the window.



Module Properties: LocalENB (ETHERNET-MODULE 1.1)

Requested Packet Interval (RPI): 200 ms (1.0 - 3200.0 ms)

☐ Inhibit Module

☐ Major Fault On Controller If Connection Fails While in Run Mode

☑ Use Unicast Connection over EtherNet/IP

Module Fault

Status: Offline

New Module for Assemblies 7 and 8

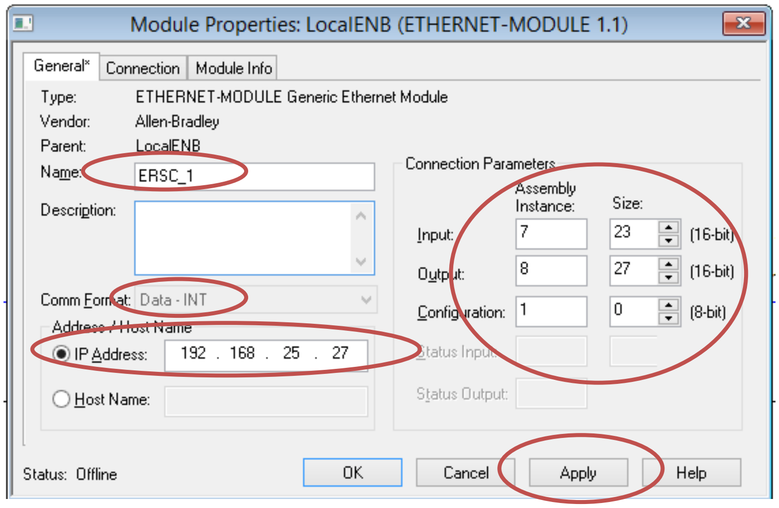
Follow Steps #1 and # 2 from **Error! Reference source not found.** on page **Error! Bookmark not defined.**

Step #3

Fill in the Name field. This will be the *ModuleName* that will appear in your program Tag Database for any addressing.

Select Comm Format to be "Data – INT" and fill in the I.P. address of the *ERSC*.

Fill in the Connection Parameters as shown.



Module Properties: LocalENB (ETHERNET-MODULE 1.1)

Type: ETHERNET-MODULE Generic Ethernet Module
Vendor: Allen-Bradley
Parent: LocalENB
Name: ERSC_1
Description:
Comm Format: Data - INT
Address / Host Name
● IP Address: 192 . 168 . 25 . 27
○ Host Name:

Connection Parameters

	Assembly Instance:	Size:	
Input:	7	23	(16-bit)
Output:	8	27	(16-bit)
Configuration:	1	0	(8-bit)
Status Input:			
Status Output:			

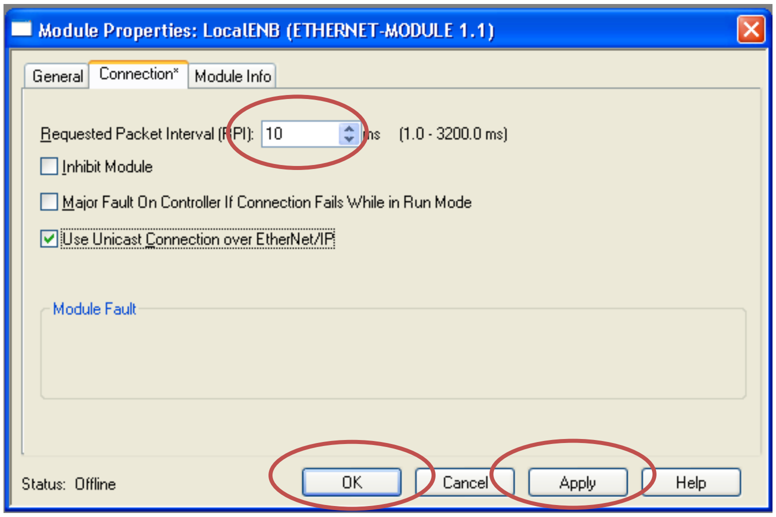
Status: Offline

It is <u>very important</u> to select *Comm Format* data type to be INT or interface to *ERSC* will not operate correctly!

Step #4

Set RPI to a value no lower than 10ms. You may also optionally select Unicast Connection.

Click "Apply" to update the value and then "OK" to exit the window.



Module Properties: LocalENB (ETHERNET-MODULE 1.1)

Requested Packet Interval (RPI): 10 ms (1.0 - 3200.0 ms)
□ Inhibit Module
□ Major Fault On Controller If Connection Fails While in Run Mode
☑ Use Unicast Connection over EtherNet/IP

Module Fault

Status: Offline

ERSC will not accept a connection with an RPI value of less than 10 ms

New Module for Assemblies 25 and 26

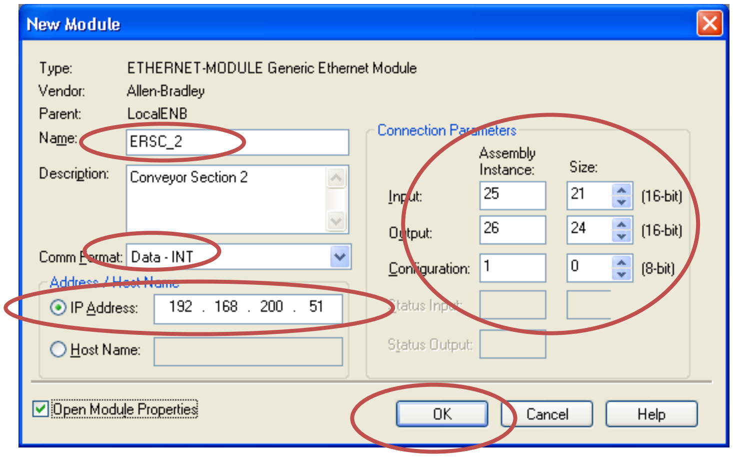
Follow Steps #1 and # 2 from **Error! Reference source not found.** on page **Error! Bookmark not defined.**

Step #3

Fill in the Name field. This will be the *ModuleName* that will appear in your program Tag Database for any addressing.

Select Comm Format to be "Data – INT" and fill in the I.P. address of the *ERSC*.

Fill in the Connection Parameters as shown.



New Module

Type: ETHERNET-MODULE Generic Ethernet Module
Vendor: Allen-Bradley
Parent: LocalENB
Name: ERSC_2
Description: Conveyor Section 2
Comm Format: Data - INT

Address / Host Name
● IP Address: 192 . 168 . 200 . 51
○ Host Name:

Connection Parameters

	Assembly Instance:	Size:	
Input:	25	21	(16-bit)
Output:	26	24	(16-bit)
Configuration:	1	0	(8-bit)
Status Input:			
Status Output:			

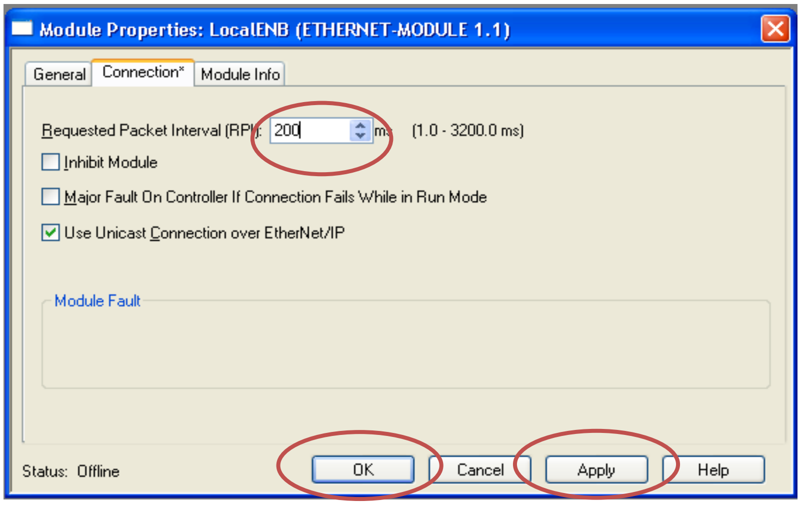
☑ Open Module Properties

It is <u>very important</u> to select *Comm Format* data type to be INT or interface to *ERSC* will not operate correctly!

Step #4

Set RPI to a value no lower than 10ms. 200 ms is typical for ZPA interface. You may also optionally select Unicast Connection.

Click "Apply" to update the value and then "OK" to exit the window.



Module Properties: LocalENB (ETHERNET-MODULE 1.1)

Requested Packet Interval (RPI): 200 ms (1.0 - 3200.0 ms)
□ Inhibit Module
□ Major Fault On Controller If Connection Fails While in Run Mode
☑ Use Unicast Connection over EtherNet/IP

Module Fault

Status: Offline

Using Logix5000 MSG Instruction

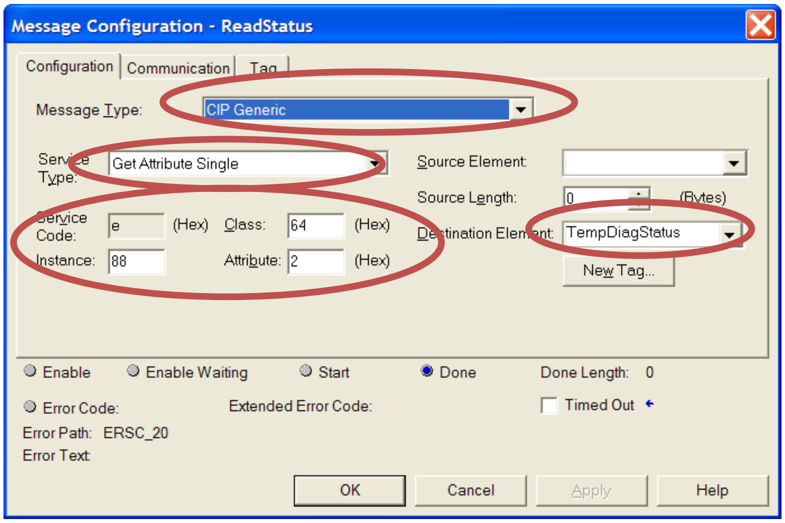
This section appeared as Appendix C in PLC Developer's Guide Version 4.6 and earlier

Any *ERSC* on the network will respond to an appropriately configured Logix5000 MSG instruction without the *ERSC* being attached as a Generic I/O instance to the PLC. The *ERSC* will allow a MSG instruction to read up to 30 contiguous Modbus registers in a single instruction. The *ERSC* will allow a MSG instruction to write 1 Modbus register in a single instruction.

Message Configuration for Reading Data from ERSC

Read MSG Setup

- Select "CIP Generic" as the Message Type
- Select "Get Attribute Single" and the Service Type
- Class is always set to 64
- Instance is the Modbus register address. In this example the Instance is 88 indicating register 4:0088
- Attribute is the number of registers to read. In this example it is set = 2. This means the MSG instruction will read Modbus registers 4:0088 and 4:0089
- Destination Element is the user defined tag for the MSG instruction to place the data it reads from the *ERSC*. In this example, "TempDiagStatus" is the user defined tag.



Message Configuration - ReadStatus

Message Type: CIP Generic

Service Type: Get Attribute Single

Service Code: e (Hex) Class: 64 (Hex)

Instance: 88 Attribute: 2 (Hex)

Source Element:

Source Length: 0 (Bytes)

Destination Element: TempDiagStatus

○ Enable ○ Enable Waiting ○ Start ● Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path: ERSC_20

Error Text:

The acceptable values for "Attribute" are from 0x1 to 0x1E which is 1 to 30 contiguous registers. In the above example, the data being read is Module Status #1 and Module Status #2 registers (4:0088 and 4:0089). This same MSG instruction could be duplicated for each *ERSC* in ZPA mode in a given conveyor system and used to populate an array of *ERSC* status data that could in turn be used for example to feed an HMI diagnostic application.

Please note that the data type of each Modbus register is integer (INT). The user defined controller tag used for "Destination Element" must of appropriate data type to accept the MSG instruction data. Please consult Allen-Bradley documentation for full description of MSG instruction usage.

Although a read MSG instruction can be used on an *ERSC* in PLC I/O mode, it is assumed that any *ERSC* in PLC I/O will already be utilizing a permanent TCP connection and should not ever need to be accessed with a read MSG instruction.

Refer to Allen-Bradley reference documentation for the particular PLC processor being used as to the proper usage and expected performance loading on the processor communication channels due to multiple MSG instructions executing simultaneously.

NOTES:

PULSE ROLLER

WWW.PULSEROLLER.COM
SALES@PULSEROLLER.COM
SUPPORT@PULSEROLLER.COM