SLC to CompactLogix Programming Migration

Bulletins 1746, 1747, 1769, 5069

Allen-Bradley
by ROCKWELL AUTOMATION

Application Profile

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.

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).

Table of Contents

Chapter 3

Conversion of Programs

Chapter 4

Convert Program Structures

Appendix A

Program Conversion Errors (PCE) Messages

I/O Modules

Appendix B

Performance Expectations

Appendix C

Advanced Modules

Appendix D

Notes:

About This Publication

This document focuses on converting an SLC™ program to a Logix program and migrating the existing SLC I/O to an Ethernet network thereby helping to leverage the existing I/O and minimize cost and risk. This document can be used in the planning stages to help identify issues so that you can anticipate the work that is involved in the conversion. The ideas that are presented in this document require Studio 5000 Logix Designer® software version 21 or later and RSLinx® software version 2.59 or later.

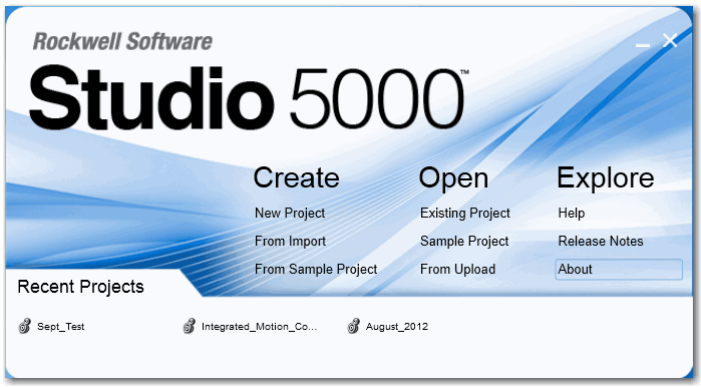
This document focuses on converting an existing SLC program to a CompactLogix™ 5380 controller or CompactLogix 5370 controller. If you want to convert your SLC program to a ControlLogix® 5580 controller or ControlLogix 5570 controller the process is similar to what is presented in this manual.

For information on migrating SLC™ 500 hardware to CompactLogix 5380 controller and Compact 5000™ I/O or CompactLogix 5370 controller and Compact I/O™, see the SLC 500 Hardware Migration Reference Manual, publication 1746-RM003.

Studio 5000 Environment

The Studio 5000® Engineering and Design Environment combines engineering and design elements into a common environment. The first element in the Studio 5000 environment is the 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 was introduced in version 21.



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.

Download Firmware, AOP, EDS, and Other Files

Download firmware, associated files (such as AOP, EDS, and DTM), and access product release notes from the Product Compatibility and Download Center at rok.auto/pcdc.

Summary of Changes

This publication contains the following new or updated information. This list includes substantive updates only and is not intended to reflect all changes.

Topic	Page
Removed mention of MicroLogix™ 1500 migration	7
Updated screen captures of Integrated Architecture Builder	17
Replaced section on Power Consideration with link to 1746-RM003	19
Removed MicroLogix 1500 appendix	82

Additional Resources

These documents contain additional information concerning related products from Rockwell Automation.

Resource	Description
SLC 500 Hardware Migration Reference Manual, publication 1746-RM003.	Provides guidelines for migrating SLC 500 to CompactLogix or Compact 5000 hardware.
SLC Ethernet Adapter User Manual, publication 1747-UM076	Reference guide for the EtherNet/IP Adapter Module.
SLC Ethernet Adapter Installation Instructions, publication 1747-IN521	Installation instructions for the EtherNet/IP Adapter Module.
Converting PLC-5 or SLC 500 Logic to Logix5550 Logic Reference Manual, publication 1756-RM085	Information on converting a PLC-5 or SLC 500 Logic to Logix5550®.
CompactLogix 5380, Compact GuardLogix 5380, and CompactLogix 5480 Controllers Specifications Technical Data, publication 5069-TD002	Provides CompactLogix, Compact GuardLogix, and CompactLogix controllers specifications.
CompactLogix 5380 Controllers User Manual, publication 5069-UM001	Describes how to install, use, and troubleshoot CompactLogix 5380 controllers and Compact GuardLogix 5380 controllers.
Compact 5000 I/O Modules and EtherNet/IP Adapters Technical Data, publication 5069-TD001	Provides Compact 5000 I/O and EtherNet/IP adapter specifications.
Compact 5000 I/O Serial Module User Manual, publication 5069-UM003	Describes how to install, use, and troubleshoot a Compact 5000 I/O serial module.
CompactLogix 5370 Controllers User Manual, publication 1769-UM021	Describes how to install, use, and troubleshoot CompactLogix controllers.
CompactLogix Controllers Specifications Technical Data, publication 1769-TD005	Provides CompactLogix controllers specifications.
Converting PLC-5 or SLC 500 Logic to Logix based Logic, publication 1756-RM085	Provides guidelines for migrating PLC-5 or SLC 500 programming to Logix based programming.
Industrial Automation Wiring and Grounding Guidelines, publication 1770-4.1	Provides general guidelines for installing a Rockwell Automation industrial system.
Product Certifications website, rok.auto/certifications.	Provides declarations of conformity, certificates, and other certification details.

You can view or download publications at rok.auto/literature.

Overview

Why Convert

Migration solutions help you to achieve increased productivity and lessen your risk of maintaining your legacy equipment. Work with a supplier that has the product, service, and industry knowledge to partner with you on an upgrade strategy that will help you maximize your competitive advantage. Rockwell Automation works with you to outline a plan to accomplish the following:

- Lower conversion time and labor costs
- Reduce risk by preserving existing field wiring connections
- Lower engineering costs
- Minimize production downtime

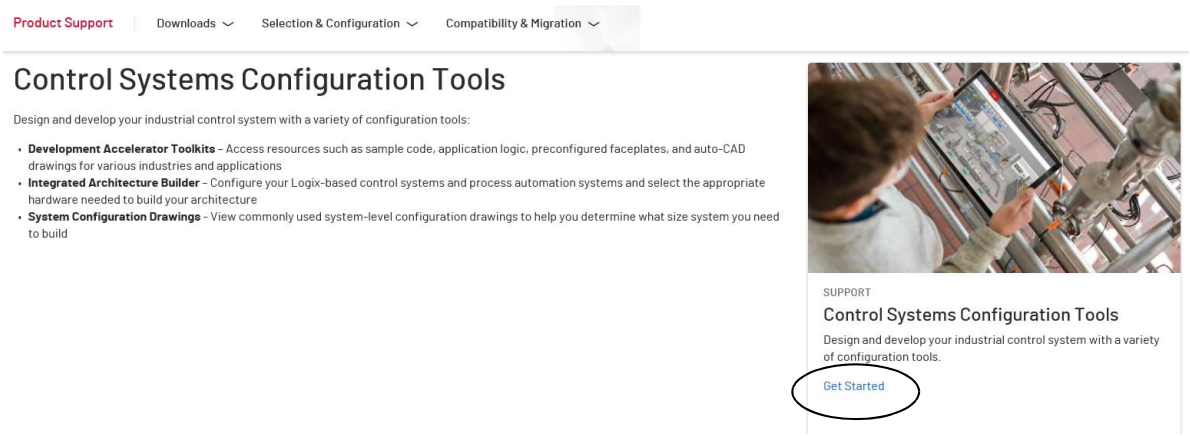
What Is Needed

You need the following items for your migration project:

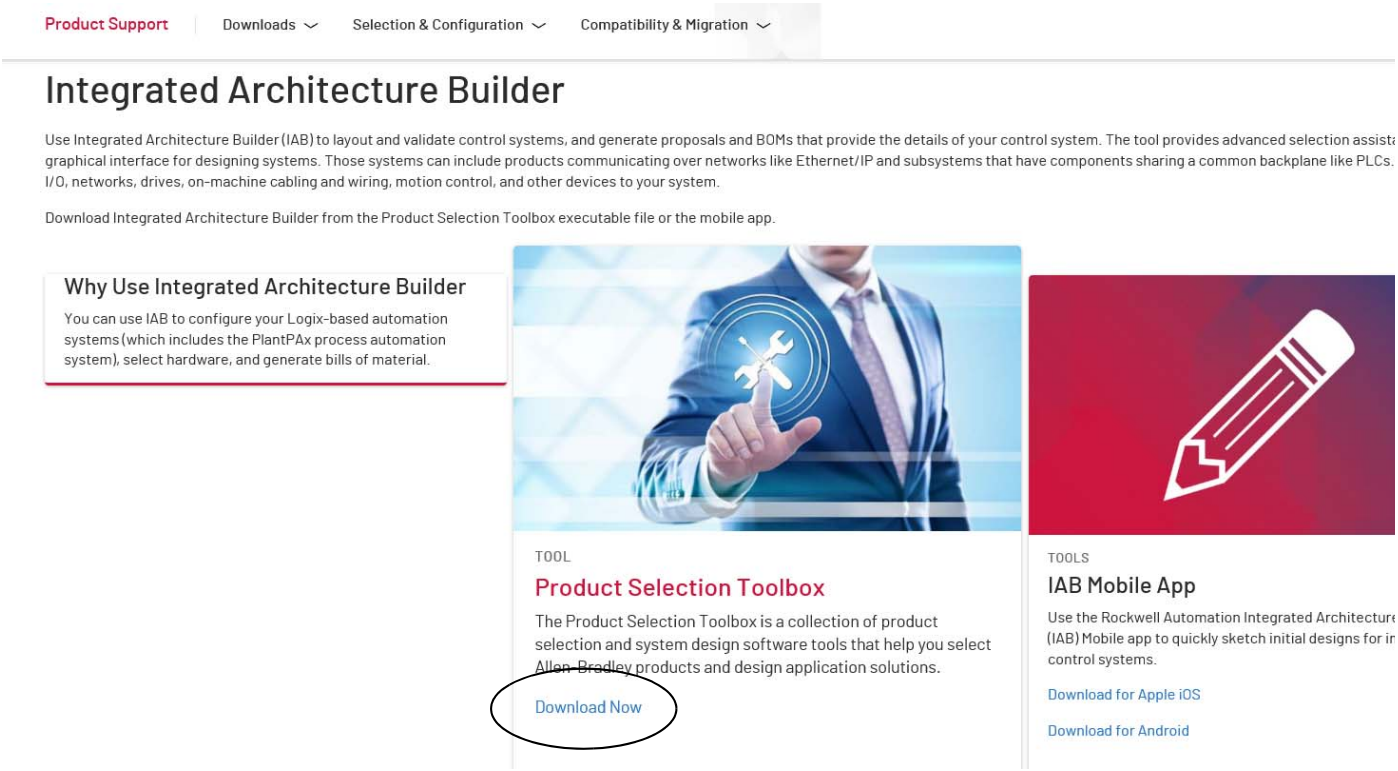
- Current SLC 500 control system
- CompactLogix Controller
- 1747-AENTR SLC EtherNet/IP™ Adapter (If you want to retain your existing SLC I/O)
- Software
  - Studio 5000 Logix Designer application version 30 or higher
  - RSLogix 500® version 12
  - Integrated Architecture® Builder (IAB)

Download the Integrated Architecture Builder

1. Go to http://www.rockwellautomation.com.
2. Mouse over the Support tab and click Selection and Configuration.
3. Scroll down to the 'Control Systems Configuration Tools' section and click 'Get Started'.

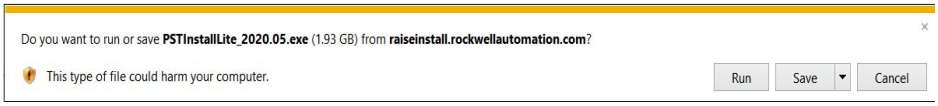


4. Scroll down to the Integrated Architecture Builder section and click 'Download Now'.



The PSTInstallLite_2020.05.exe file pops up at the bottom of the screen.

5. Click Run.



The tool is installed on your computer.

6. Go to Start > Program > Rockwell Automation > Integrated Architecture Builder to Run IAB.

SLC EtherNet/IP Adapter

The 1747-AENTR adapter module enables CompactLogix and ControlLogix processors to control SLC I/O modules. The adapter is primarily designed to enable migration of existing SLC-controlled systems to Logix-based systems.

The adapter mainly acts as a gateway between the SLC backplane and EtherNet/IP. It typically replaces an SLC controller in the SLC rack and upgrades the remote SLC to EtherNet/IP.

Control of the backplane I/O is accomplished with a CompactLogix or ControlLogix controller communicating through an EtherNet/IP router in the Logix backplane, across EtherNet/IP, and into the 1747-AENTR gateway.

As a gateway between the SLC backplane and EtherNet/IP, the 1747-AENTR module is a CIP™ server (for both Explicit Messaging and I/O) on the Ethernet port, and an SLC host on the SLC backplane.

Connections can be made to supported SLC analog, digital, and specialty I/O modules installed in the backplane.

IMPORTANT See Appendix B for a list of supported and unsupported I/O modules.

1747-AENTR Connections

Exclusive Owner, Input Only, and Listen Only connections are Class 1 connections. The 1747-AENTR module supports a combined total of 96 Class 1 connections:

- One Exclusive Owner connection per slot
- A combination of five Input Only or Listen Only connections per slot

The 1747-AENTR module supports a maximum of eight Class 3 connections. Class 3 connections are typically a connected ladder-triggered MSG instruction to the 1747-AENTR module.

1747-AENTR and Rack Optimization

A major difference between using the 1747-AENTR adapter with SLC I/O modules and any other Logix compatible I/O system is that the 1747-AENTR adapter does not support a rack-optimized connection type. This means that each module is an individual connection to the controller. These connections must be taken into account when converting the SLC controller. The main considerations are the effect on the Logix controller's total number of connections and the allowable RPIs to the modules in the 1747-AENTR chassis.

Exclusive Owner, Input Only, Listen Only, and None Connection Types

People familiar with SLC programing and I/O may not be familiar with how Logix handles programming and I/O. In simple terms, all I/O in a Logix controller is placed in an I/O tree. All I/O modules in the I/O tree can have multiple types of connections that transfer the I/O data into the Logix controller's memory. Below are some basic definitions of the types of connections that can be made to various SLC I/O modules.

- None – A method of establishing communication to a 1747-AENTR module that enables connection to individual modules within the 1747-AENTR module chassis by using individual connections. After a None connection is configured to a 1747-AENTR module, each module that you desire to communicate through this connection must also be configured under the 1747-AENTR module by using any type of individual module connection (Exclusive Owner, Input Only, or Listen Only). Only the 1747-AENTR module supports a None connection type. A None connection does not affect the 96 total available Class 1 I/O connections on the 1747-AENTR module.
- Exclusive Owner – There can be only one Exclusive Owner connection to each module. The Exclusive Owner connection is the only connection to send configuration data to that module. An Exclusive Owner connection to a device is the only connection that determines the mode (Prog or Run) of that particular device. Each Exclusive Owner connection does

subtract 1 from the 96 total available Class 1 I/O connections on the 1747-AENTR module.

- Input Only – There can be as many as five Input Only connections to the same module. This connection type lets multiple Logix controllers receive incoming data from the same I/O module. If the connection is to an I/O device that requires configuration or output data, the Input Only connection does not send any configuration or output data. You can get Input Only data with or without an existing Exclusive Owner connection. Each Input Only connection subtracts one from the 96 total available Class 1 I/O connections on the 1747-AENTR module.
- Listen Only – There can be as many as 5 Listen Only connections to the same module. This connection type is identical to an Input Only connection with two differences:
  - Either an Exclusive Owner or Input Only connection must exist and be working to the I/O module before a Listen Only connection can work.
  - The Exclusive Owner or Input Only connection and the Listen Only connection must all be set to Multicast. Each Listen Only connection subtracts 1 from the 96 total available Class 1 I/O connections on the 1747-AENTR adapter.

Add I/O Modules Online

RSLogix 5000 software, version 15.02.00 and later, or Studio 5000 environment, version 21.00.00 and later, the 1747-AENTR module, and all supported SLC I/O modules support adding I/O modules online and Module Discovery. However, this feature is supported only in ControlLogix 5580 and 5570 controllers. CompactLogix 5380 and 5370 controllers do not support adding I/O modules online or Module Discovery.

Using the 1747-AENTR Module in a Redundant Logix System

The 1747-AENTR is not compatible for use in a ControlLogix redundant system using a 1756-SRM, 1756-RM, or 1756-RM2.

PLC-5/SLC 500 Architecture to Logix Architecture Comparison

The following table shows the differences between PLC-5®/SLC 500 and Logix architectures.

Attribute	PLC-5/ SLC 500 Controller	CompactLogix 5380 Controller
CPU	The SLC 500 processor is based on 16-bit operations	Logix controllers use 32-bit operations
Operating system	Process codes based on program files	Process codes based on tasks, programs, and routine organizations
Inputs and Outputs (I/Os)	I/Os are mapped into I and O data table	I/O tags are automatically generated whenever they are created
	I/O data is updated synchronously to the program scan	I/Os are updated asynchronously to the logic scan
Data	Store data in global data table	Support local and global data
	Data are store in table format	Data is shareable with other Logix Controller
		Use array tables to store data in table format

Attribute	PLC-5/ SLC 500 Controller	CompactLogix 5380 Controller
Time bases	Time is based on 16-bit architecture	Time is based on 32-bit architecture
	They can have different time bases (10 ms, 1 s)	Supports 1 ms time base
Communication	PLC-5 processor supports the following: • Block-transfer read and write (BTR and BTW) instructions • ControlNet I/O (CIO) • Message (MSG) instructions	MSG instructions
	SLC 500 supports the following: • Block-transfer read and write (BTR and BTW) instructions • Message (MSG) instructions	

SLC to Logix Memory Comparison

When choosing a Logix controller to migrate to, you must consider the memory size of your existing SLC program and in what type of Logix controller the program will fit after conversion. SLC maximum memory sizes vary from 1 KB to 64 KB. Logix controllers vary in maximum memory size from about 380 KB to about 10 MB. While no two SLC programs' memory usage before and after conversion is the same, a good rule of thumb is that a full 32 KB SLC program converts to a Logix program size of about 360 KB. So in general, SLC programs of less than 32 KB fit into any type of Logix controller while a full 64 KB SLC program fits only in Logix controllers with at least about 800 KB of memory.

Scan Time Comparison

The Logix controller has a significant advantage over the SLC controller in regard to program scan time. While each situation is unique, it is likely that you will see a reduction in overall scan time when you convert to a ControlLogix 5580 or 5570 controller or a CompactLogix 5380 or 5370 controller. In many cases a scan time reduction is beneficial, but there can be instances where this is not ideal and can disrupt an existing process. Consider this in your conversion process and make the necessary adjustments to scan time as needed.

While some conversions involve converting a single SLC controller to a single Logix controller, there are cases where you want to convert multiple SLC controllers and combine them into a single Logix controller. With the significant decrease in scan time, this can be possible without any decrease in throughput or performance.

After the conversion, you could expect a decrease of 50…80% in overall program scan when running in a Logix controller as compared to an SLC controller. For more information on converting an SLC 500 program, refer to the Converting PLC-5 or SLC 500 Logic to Logix based Logic, publication 1756-RM085.

Local Rack SLC 500 I/O Modules

SLC I/O modules contained within the same chassis as the SLC 500 controller are considered local I/O. In simple terms, the SLC controller can be removed

and a 1747-AENTR installed in its place. It is important to note that SLC local I/O, both analog and discrete, was scanned fairly quickly, with the major contributor to throughput being program scan.

IMPORTANT Modules requiring G-file configurations cannot be in a remote rack to a Logix controller.

See Appendix C for information on performance expectations.

Remote Rack SLC I/O Modules

IMPORTANT See Appendix B for a list of supported and unsupported I/O modules.

Many SLC systems have I/O remotely located from the SLC 500 controller. The I/O can be scanned by a 1747-SCNR (ControlNet®) module or a 1747-SN (RIO) module. The adapters used can be a 1747-ACNR (ControlNet) module or a 1747-ASB (RIO) module. The 1747-SCNR/1747-SN modules scan I/O and place the data in a combination of I1 Input, O0 Output, M1 Input, and M0 Output files. Because the 1747-SCNR module and the 1747-SN module are not supported, they are removed from the converted system and the 1747-ACNR module and the 1747-ASB module are replaced with a 1747-AENTR module and scanned directly by the Logix controller. After the replacement of the various modules the data must be MOVed/COPied to/from the original I1, O0, M1, and M0 locations to the new Logix tag locations. The exact process of these moves is beyond the scope of this document.

The scanning of the remotely located I/O in the SLC system was at a slower rate than the scanning of local SLC I/O. After conversion to a Logix system, the scan rate of the remote I/O scanned on Ethernet via the 1747-AENTR module will be similar to that of the original SLC system.

Some SLC systems can perform Block Transfer Reads (BTR) and Block Transfer Writes (BTW) over remote I/O via the 1747-SN module. This is a specialized function and requires a more detailed description. The 1747-AENTR module does not support the 1747-SN module, but the racks communicated with by the 1747-SN module contain a 1747-ASB module that can be replace by a 1747-AENTR module. If your existing SLC system is using BTR and BTW instructions over remote I/O via the 1747-SN module, the BTR/BTW can be replace by a module connection in the Logix Designer application. Expect this part of the conversion to take additional work and time.

Overall Performance Expectations

While each conversion situation is unique, it is likely that the scanning of I/O could be slower in a Logix system while the program scan is faster. Overall performance and throughput are likely to be better than the existing system.

If the current system has stringent performance characteristics, we recommend you perform a more detailed performance analysis to verify the Logix performance in advance of your migration.

Logix Controller Boot Time

As Rockwell Automation moves forward with technology and adds more functionality into the controllers, boot time can be affected. Here are some of the reasons:

- More tasks are performed at startup, such as: memory validity checks, safety diagnostics, controller health, and security tests.
- As memory sizes increase, so does the power required to back up that memory.
- The controller does not enter the 'RUN' condition until there is enough energy stored to tolerate a potential loss of power. Larger capacitors require longer charge-up times.
- Certification requirements continue to drive more startup diagnostics and code validation.
- SLC 500 processors power-up times were relatively small, taking only several seconds. Some Logix controllers can take as long as 40 seconds to power up.

Therefore, you can expect the boot time in the Logix controllers to be greater than in your SLC controllers.

Synchronous Versus Asynchronous I/O Scans

The SLC 500 processor maps I/O data into Input and Output data table files. The I/O data is updated synchronously to the program scan so you know you have current values each time the processor begins a program scan and that the I/O data does not change during the program scan. A Logix controller references I/O that is updated asynchronously to the logic scan.

If you must maintain I/O data integrity throughout the program scan and/or you must maintain synchronous I/O data transfers in the Logic controller you must use a CPS copy instruction. See Knowledgebase answer ID 50235 for more information on how to use the CPS instruction.

SLC I/O Local and Remote Rack Size

The SLC 500 systems had a maximum single-chassis size of 13 slots, by using various cables, multiple chassis could be connected together to create an I/O rack that extended up to 30 slots. RSLogix 5000 software version 20.00.00 with the 1747-AENTR module version 1.1 supports a maximum chassis size of 13 slots and only one chassis. Logix Designer Application version 21.00.00 or later with the 1747-AENTR module version 2.1 supports up to 30 modules and up to 3 chassis.

IMPORTANT	RSLogix 5000 software version 20.00.00 with any version of the 1747-AENTR module supports a single chassis with up to 13 slots. You must have Logix Designer Application version 21.00.00 or later and a 1747-AENTR module version 2.1 or later to support up to 30 modules.

Keeping I/O

With the introduction of the 1747-AENTR module, you can potentially lower the risk, decrease conversion time, and lower the cost of converting an SLC system to a Logix system. Studio 5000 Logix Designer version 21 and later lets

you keep your existing SLC I/O modules while migrating to a Logix controller. The existing SLC I/O can be added to the Logix controllers I/O tree by replacing the SLC controller or SLC adapter with a 1747-AENTR module. No changes are needed to individual SLC I/O modules, but new EDS files have been developed for the existing supported modules. These new EDS files are installed with RSLinx software version 2.59. The EDS files can be verified in RSLinx software by viewing the modules EDS file. The correct EDS files have a ModDate of 2011; incorrect files have a ModDate of 1999.

IMPORTANT See Appendix B for a list of supported and unsupported I/O modules.

RSLinx software version 2.59 installs new EDS files for all 1746 supported I/O modules. However, the 1747-AENTR EDS file installed by RSLinx software version 2.59 is not the latest version. You must update the 1747-AENTR EDS file to the latest version, which is currently version 2.3. Common symptoms of creating an RSLogix 5000 project with an incorrect 1747-AENTR EDS file include the following:

- Incorrectly getting 'Module Configuration Rejected fault code 16#0009' to a properly configured module under the 1747-AENTR module
- Inability to convert an RSLogix 5000 project from one controller type to another
- Inability to open a valid project on another computer

The corrective actions include uninstalling the incorrect EDS file and updating to the correct version EDS file, possibly deleting the 1747-AENTR module and its children from the I/O tree, and exporting and importing the project. If you are using EDS revisions earlier than 2.3 and you right-click on the 1747-AENTR module, you do not see a selection for 'Upload EDS from device'. If you are using EDS revision 2.3 or later and you right-click the 1747-AENTR module, you see a selection for 'Upload EDS from device'.

Migration Considerations

Define Future Requirements

As you investigate the prospect of upgrading current control systems to newer technology, it is important to factor in all aspects of the migration. Deciding how to phase in the new system can be challenging due to conversion time and physical considerations. Rockwell Automation has tools to assist with the conversion of the program code to minimize the engineering design time. But, what about the physical layout of the new system? Does it make more sense to leave the existing legacy I/O and wiring in place and save the I/O conversion for a later date? Or should the entire system be upgraded all at once? How might the hardware costs for each of these scenarios be affected?

The process for converting your SLC program to a ControlLogix 5580 or 5570 controller is similar to what is presented in this manual.

Device Level Ring Topologies

While the 1747-AENTR adapter can be a member of a ring, the adapter does not support being an Active Ring Supervisor or a Back-up Supervisor. For more information on Ethernet network rings and ring supervisors, see the EtherNet/IP Embedded Switch Technology Application Guide, publication ENET-AP005.

Using DeviceNet Network

Existing SLC systems that use 1747-SDN modules must replace the 1747-SDN module with a different DeviceNet® scanner because the 1747-SDN module is not supported by the 1747-AENTR module. Depending on the application and Logix processor that is selected, we recommend a 1756-DNB module, a 1769-SDN module, or a 1788-EN2DNR module.

The CompactLogix 5380 controller does not have a module that supports DeviceNet at the time this publication is released. Use the 1788-EN2DN module to connect the CompactLogix 5380 controller to the DeviceNet devices.

The 1747-SDN module scans I/O and places the data in a combination of I1 Input, O0 Output, M1 Input, and M0 Output files. The 1747-SDN module must be replaced in the converted system with a different DeviceNet scanner. After the replacement, the original 1747-SDN module data must be MOVed/COPied to/from the original I1, O0, M1, and M0 locations to the new Logix tag locations.

IMPORTANT The exact process of these moves is beyond the scope of this document; expect this part of the conversion to take additional work and time.

Use of Advanced Modules

The 1746-HSCE, 1746-QS, 1746-BAS/B, and the 1746-BAS-T modules are considered advanced modules because they transfer Input data, Output data, and in an SLC environment M0 and M1 files. In the Logix Designer application, connections are established to these modules by using an Exclusive Owner Advanced connection. The Exclusive Owner Advanced connection enables the transfer of the additional data that is supplied by the 1746 modules M0 and M1 files. Advanced connections transfer Input, Output, M0, and M1 data from the SLC I/O module via a single connection in the Logix Designer application. This data in the RSLogix 500 environment was obtained in the equivalent of two separate transactions.

Advanced modules do require some additional setup and configuration. Advanced modules also do not have descriptive tag names like most non-advanced SLC I/O modules. All Advanced modules require additional steps and programing to migrate.

See Use Advanced Modules in a Logix System on page 81 for information.

Communication with SLC Controllers Over Serial Networks

Be aware of other networks and take them into account when planning a conversion. This is a brief overview of networks that can be present in the existing architecture.

You can also use products from various Rockwell Automation Encompass™ partners for support on a specific protocol.

If after the conversion you must keep communication with existing nodes, various communication modules can facilitate this messaging.

Network	Modules	See Publication
DH-485 network	1756-DH485 communication module	1756-UM532
DH+™ network	1756-DHRIO communication module	1756-UM514
Ethernet network	1756 Ethernet modules	ENET-UM001
Generic ASCII	5069-SERIAL	5069-UM003
Modbus		
DH-485 network		
DF1 network		

The Compact 5000 I/O serial module provides two independent channels that function as network interfaces to a wide variety of RS-232C, RS-422, and RS-485 devices. The module has two channels that are independent of each other. The channels can transmit data to and receive data from serial devices using the following communication modes:

- Generic ASCII

- Modbus RTU (Master/Slave)
- Modbus ASCII (Master/Slave)

The serial module can reside locally in a CompactLogix 5380 or Compact GuardLogix® 5380 system. The module can also reside in a remote Compact 5000 I/O system accessible using an EtherNet/IP network.

For more information on how Compact 5000 I/O serial modules function in a control system, see the Compact 5000 I/O Serial Module User Manual, publication 5069-UM003.

Power Considerations

The SLC 500 controllers require an SLC power supply module on the leftmost slot to provide power to all modules on the chassis. This powers up the controller and the I/O modules. This also facilitates communication through the backplane.

The CompactLogix 5380 controllers require a 24V DC source supply that is connected to the MOD power terminal to provide power to all modules. Unlike SLC 500 controllers, CompactLogix 5380 controllers do not require a chassis.

For detailed power considerations for controllers, sensor, actuator, and backplane, see the SLC 500 Hardware Migration Reference Manual, publication 1746-RM003.

Using Integrated Architecture Builder to Plan Hardware Migration

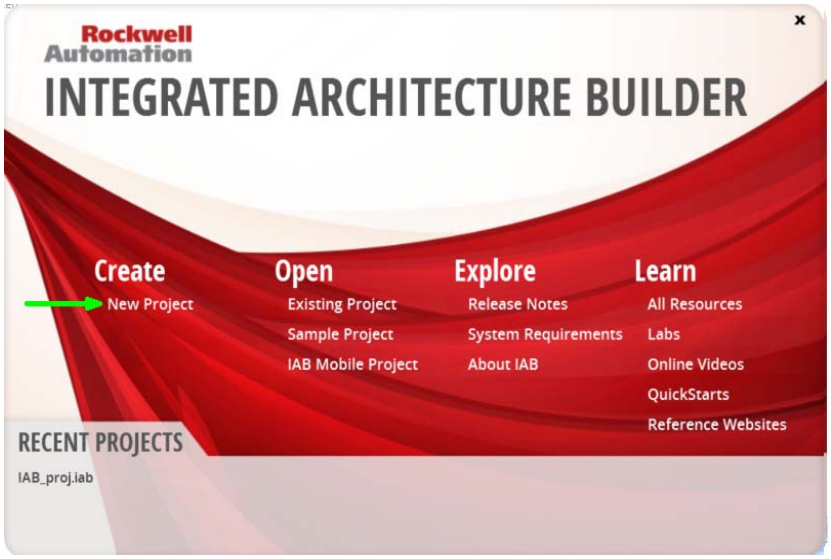
Once you have planned your overall migration approach, let Integrated Architecture Builder (IAB) help plan the details. The SLC migration wizard embedded in IAB steps you through the system configuration process, letting you make the decisions on which components you prefer to keep and reuse and which components you prefer to replace. If you choose to reuse the SLC I/O modules, IAB verifies module support and power supply loading and helps you lay out the new EtherNet/IP network.

This section uses the SLC Migration Wizard within IAB to assist with the conversion of the existing SLC hardware to a CompactLogix system. The process for converting your SLC program to ControlLogix 5580 or 5570 controller is similar to what is presented in this manual.

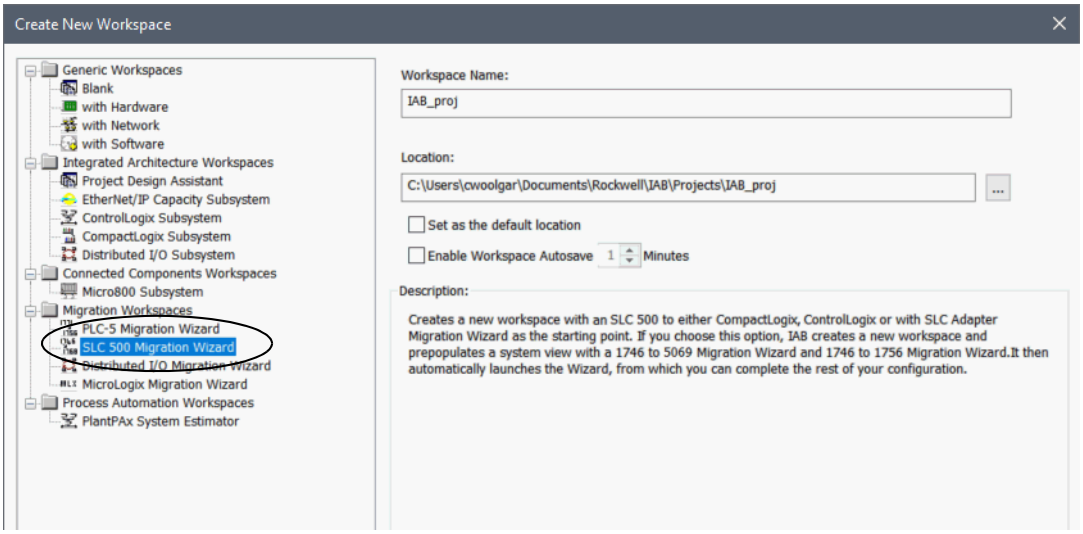
Replace Only the Local SLC Controller

1. Go to Start > Programs > Rockwell Automation > Integrated Architecture Builder > Integrated Architecture Builder
   or double-click the Integrated Architecture Builder icon on the computer desktop to launch IAB.

   The IAB opening dialog box appears.
2. Click New Project.

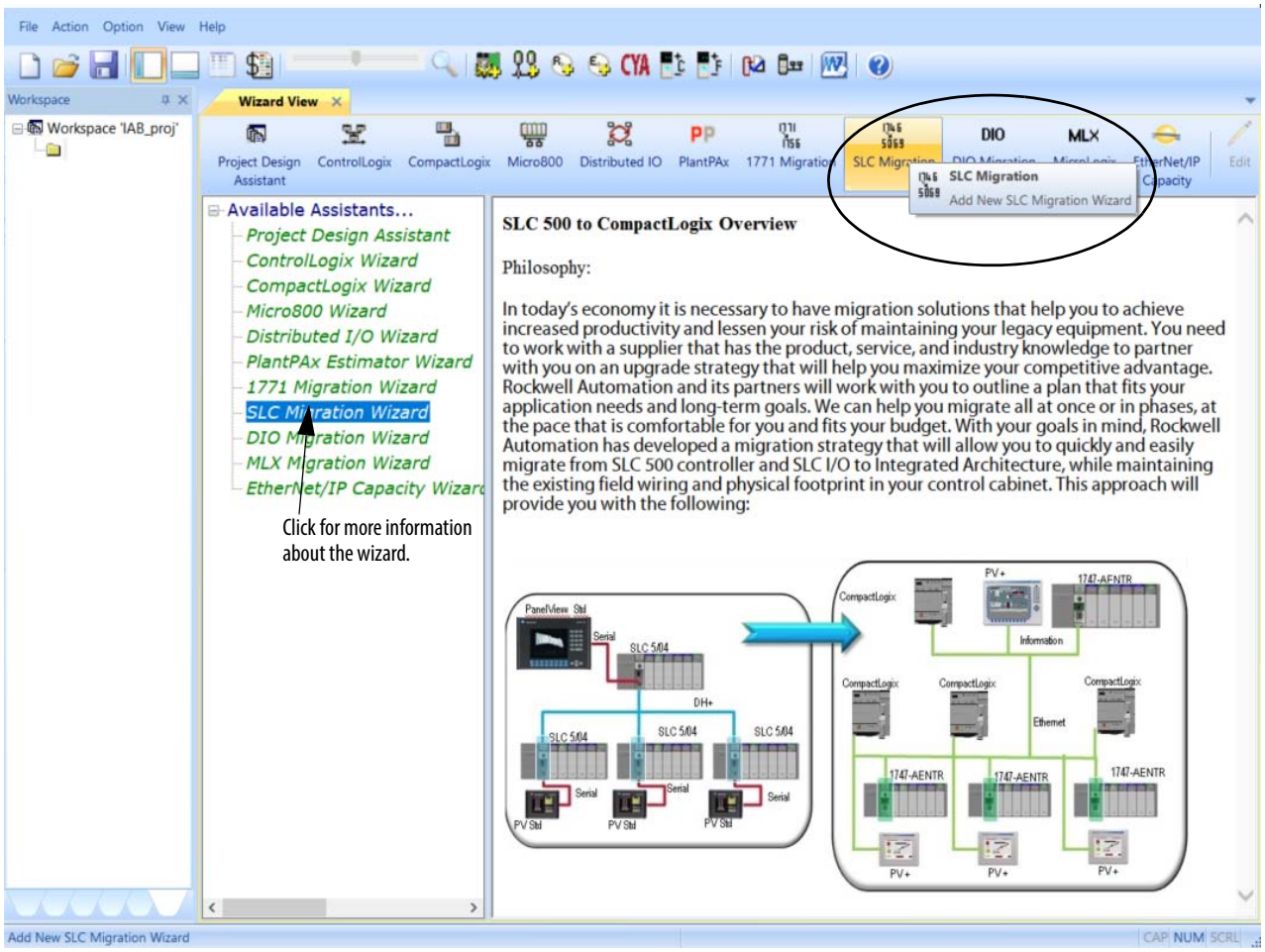


3. In the Workspace Name text box, type an appropriate name, such as 'SLC Migration Wizard', and click OK.



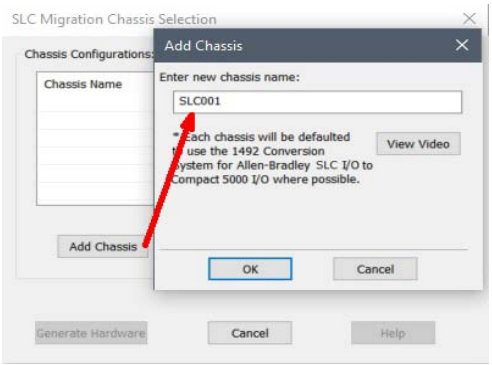
4. Under Wizard View, click SLC Migration.

Clicking the wizards that are listed under Available Assistants provides an overview of the wizard.



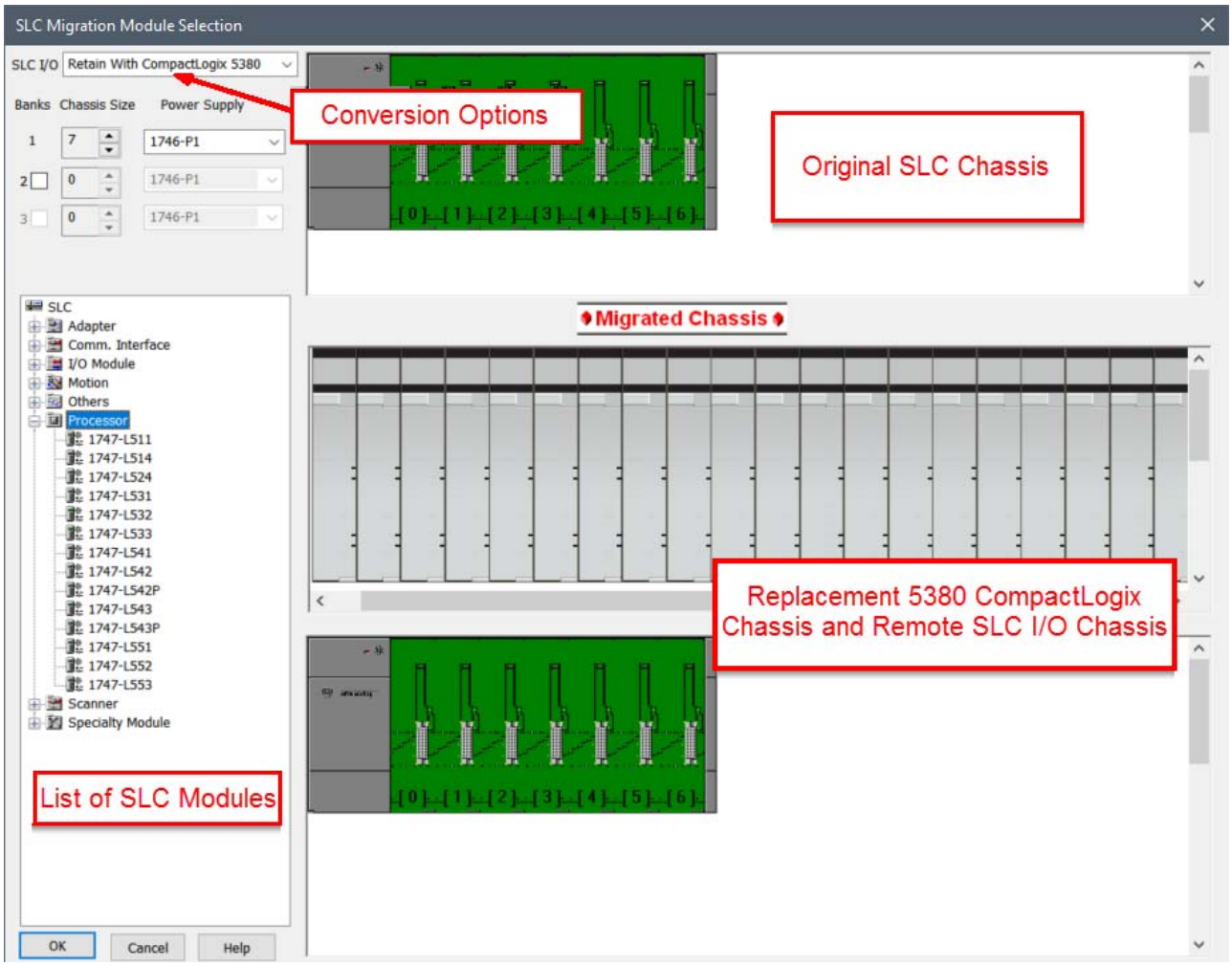
5. In the SLC Migration Chassis Selection dialog box, click Add Chassis.

   IAB opens the Add Chassis dialog box.

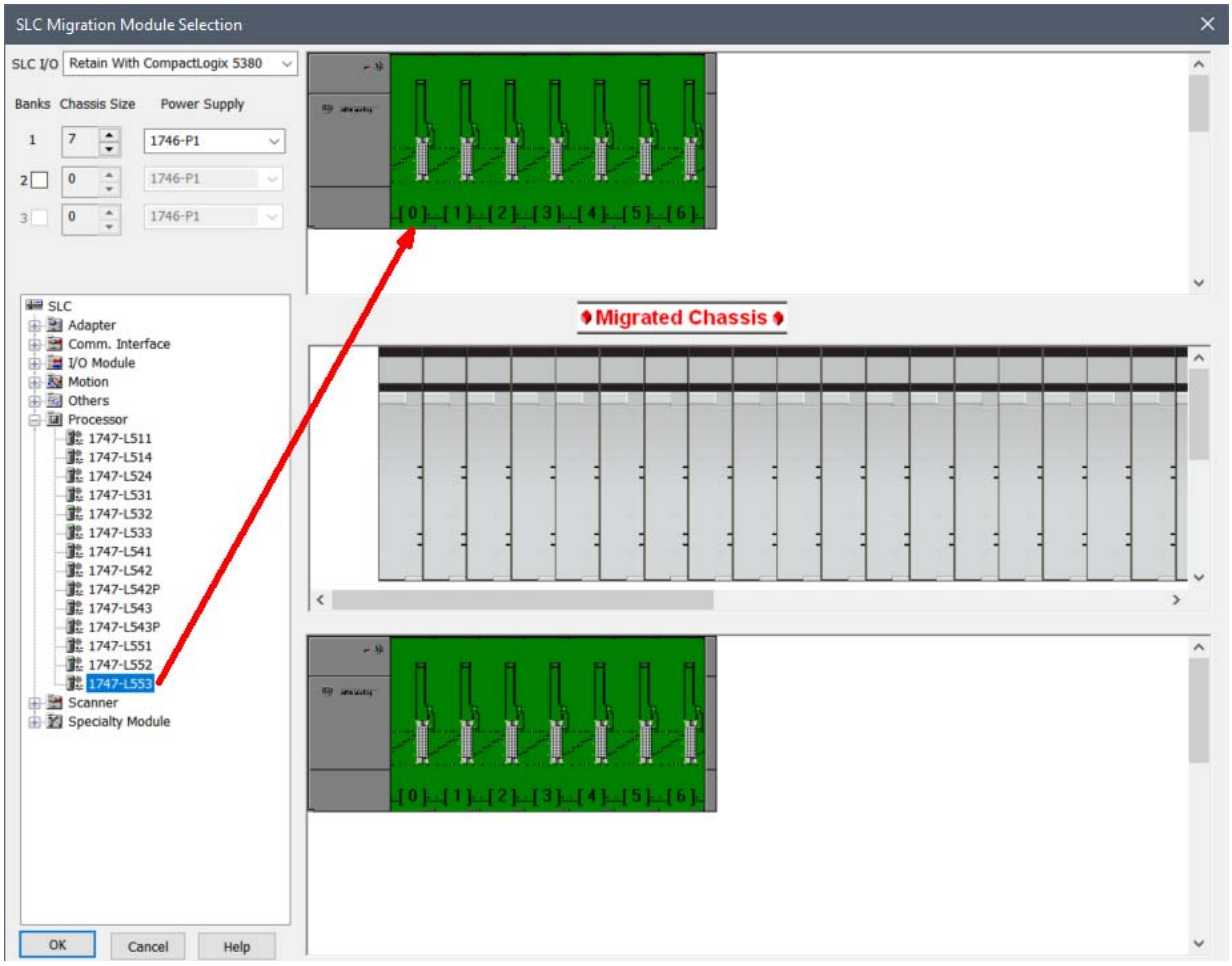


6. Click OK to accept the default name (SLC001) for the new chassis.

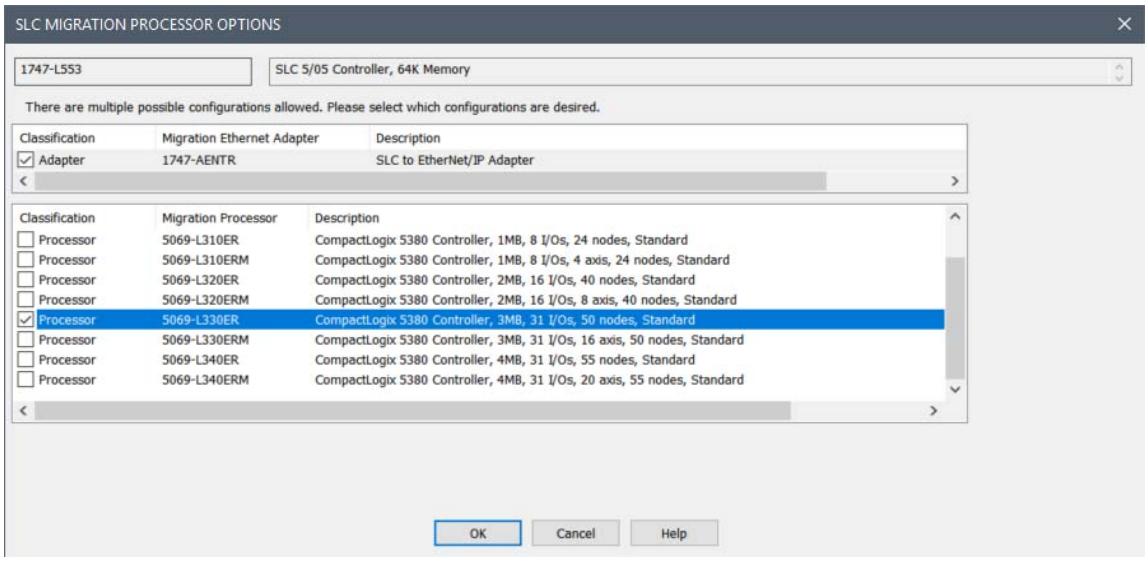
   IAB opens the SLC Migration Module Selection dialog box. This is where you make the conversion selections for this chassis. Notice the different areas of this window.



7. Select your chassis size and choose power supply.
8. From the Processor module list, expand the Processor heading and drag your processor module to slot 0 of the SLC chassis.



Because IAB has found multiple possible CompactLogix controller migration options, the 'SLC Migration Conflict Resolution Dialog' box appears.

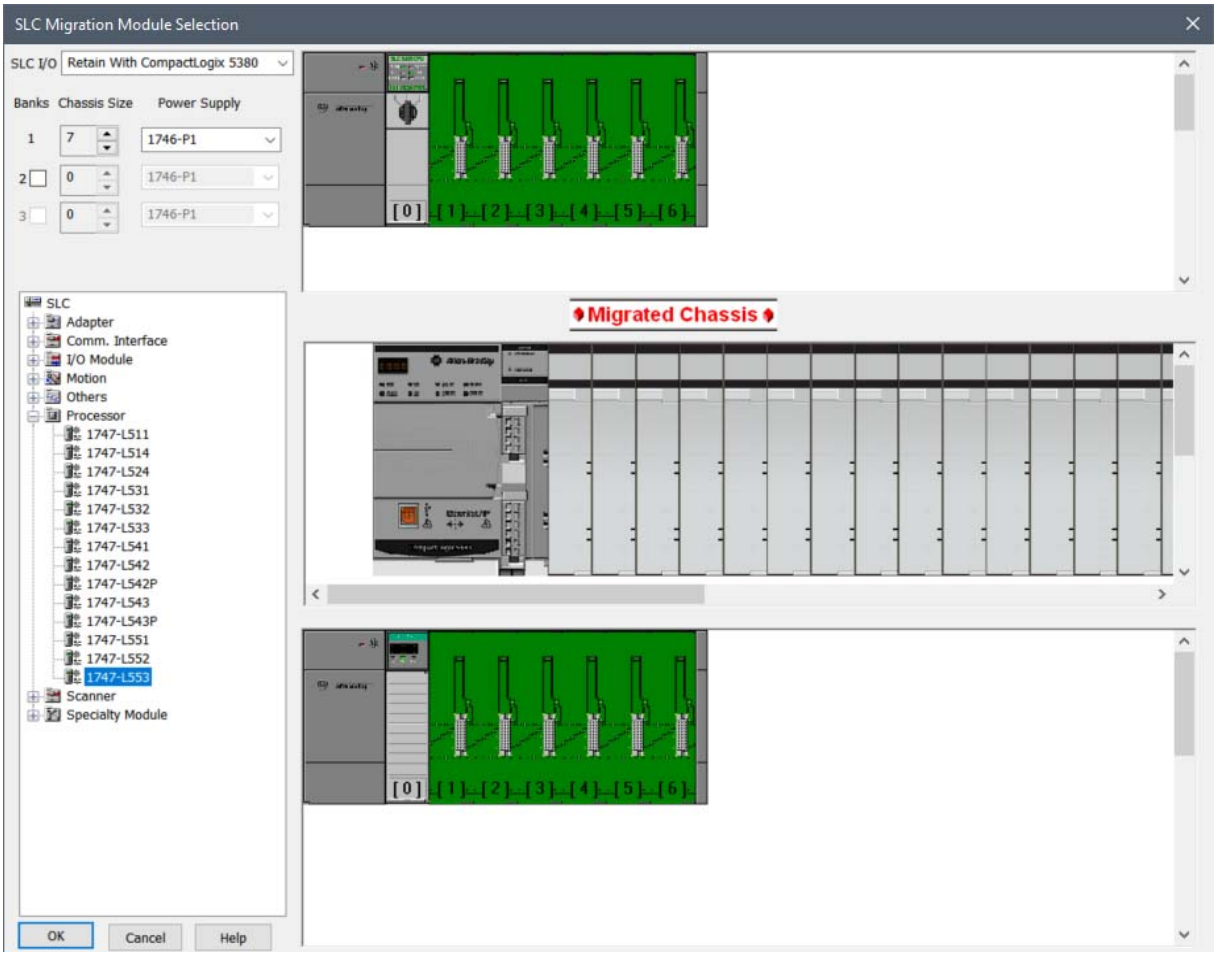


Conflict resolution dialogs appear when you must make a decision about the conversion. The information in the dialog box is specific to the action you are performing. In this case, we must select the CompactLogix processor that we wish to use.

9. Select a processor from the list and click OK.

   In this example, we show the CompactLogix 5380 5069-L330ER controller.

   IAB adds processors to both the SLC chassis and the replacement CompactLogix 5380 chassis. Additionally, IAB also adds a 1747-AENTR Ethernet adapter to the retained I/O on the SLC chassis at the bottom of the display.



10. Repeat steps 8 and 9 to fill remaining slots.

If you incorrectly place a module, simply right-click the module and click Remove Module to try again.

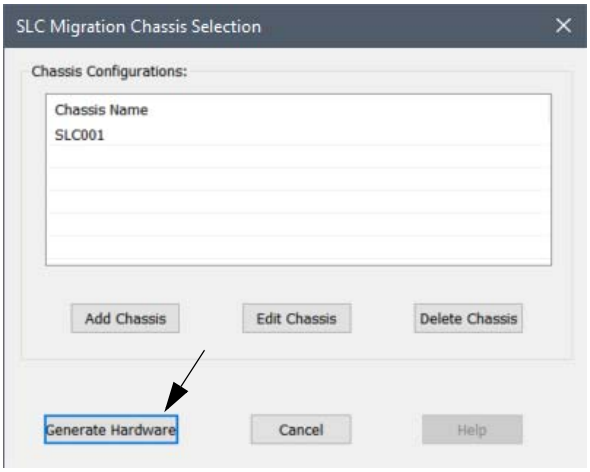
Most SLC I/O discrete, analog, and specialty modules are compatible with the 1747-AENTR adapter.

For a list of supported and unsupported modules, see I/O Modules on page 75.

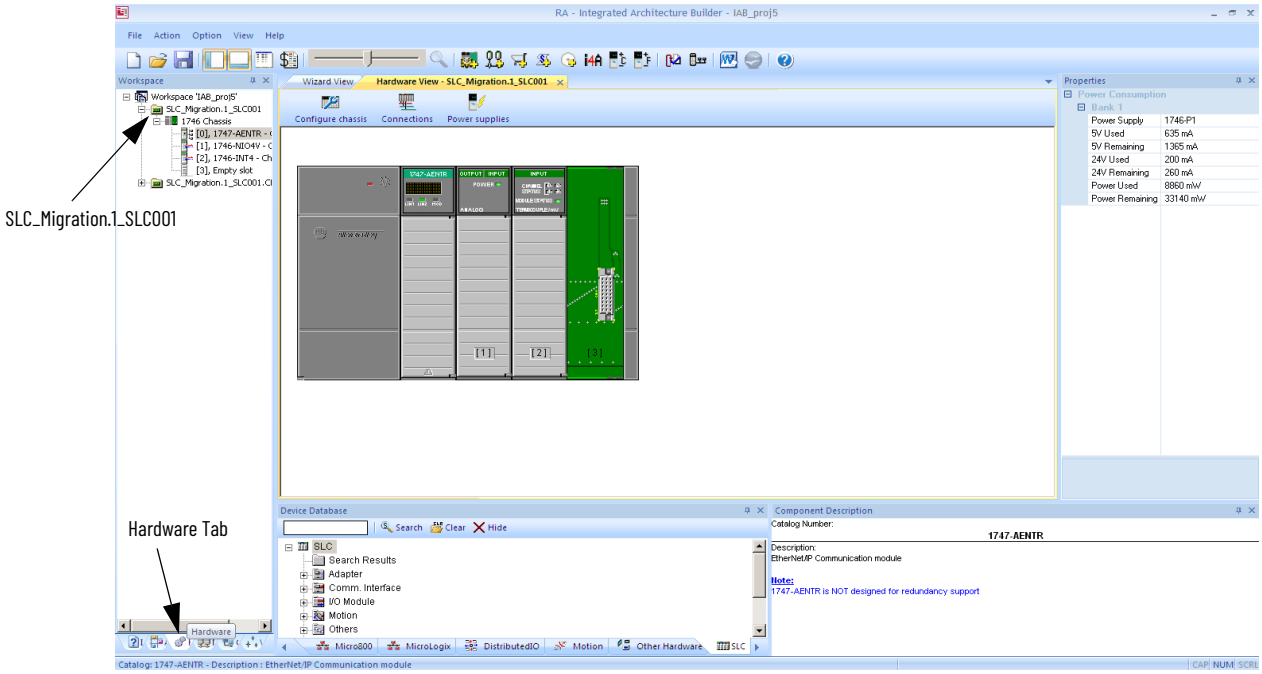
If you have a module that is not compatible, the following warning occurs.

IMPORTANT Modules not supported as part of a retained I/O solution, when connected to a Logix controller, are not placed into the lower chassis in IAB.

11. Once the local SLC chassis is complete, click OK.
12. In the SLC Migration Chassis Selection dialog box, click Generate Hardware to create the wizard-defined CompactLogix configuration in IAB.



13. Click the Hardware tab in the lower left corner of the IAB window.
14. Click the SLC_Migration.1_SLC001 chassis to see the hardware.



This chassis contains all SLC I/O from the local rack that we replaced. A 1747-AENTR module has been added to connect this I/O remotely over Ethernet to the CompactLogix controller that is replacing our SLC processor.

15. Click the save icon to save your project.

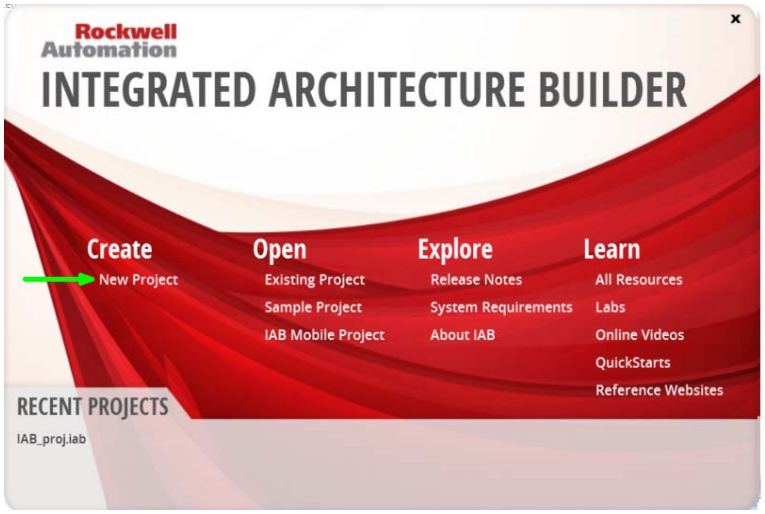
Replace the Local SLC System with CompactLogix System

Although retaining the SLC I/O when converting to a Logix system can save on rewiring costs, adding a controller and a power supply to an existing control panel can prove to be impossible due to physical space limitations. In such optional cases, SLC conversion solution can actually involve converting the I/O to the Compact I/O or POINT I/O™ platforms in addition to converting to a Logix controller.

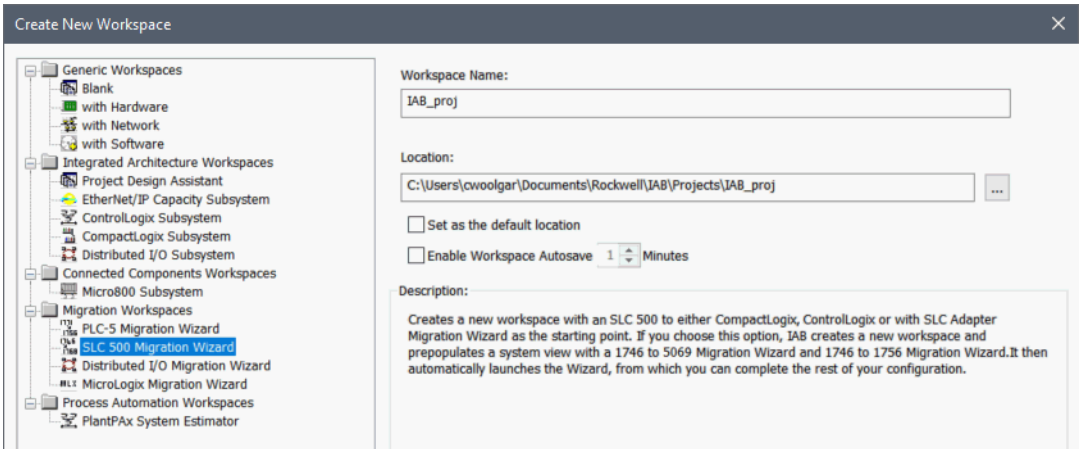
1. Go to Start > Programs > Rockwell Automation > Integrated Architecture Builder > Integrated Architecture Builder
   or double-click the Integrated Architecture Builder icon on the computer desktop to launch IAB.

   The IAB opening dialog box appears.
2. Click New Project.

   The Create New Workspace dialog box appears.

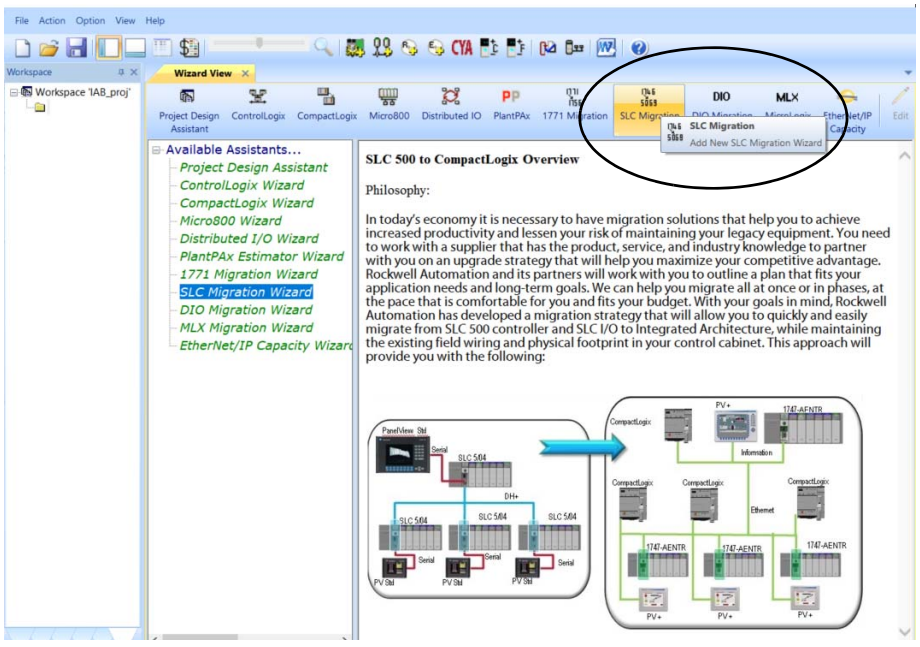


3. In the Workspace Name text box, type an appropriate name, such as 'SLC Migration Wizard', and click OK.



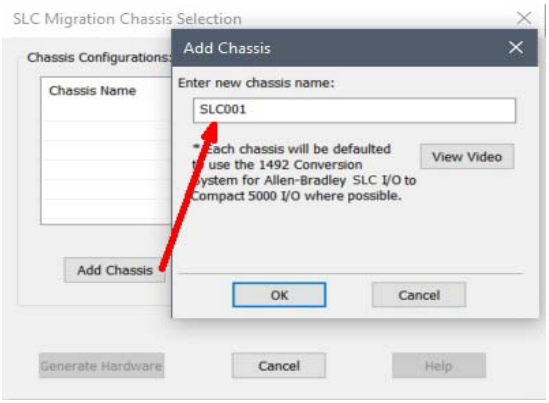
4. Click SLC Migration in the Wizard View.

Click Available Assistants for more information about each assistant.



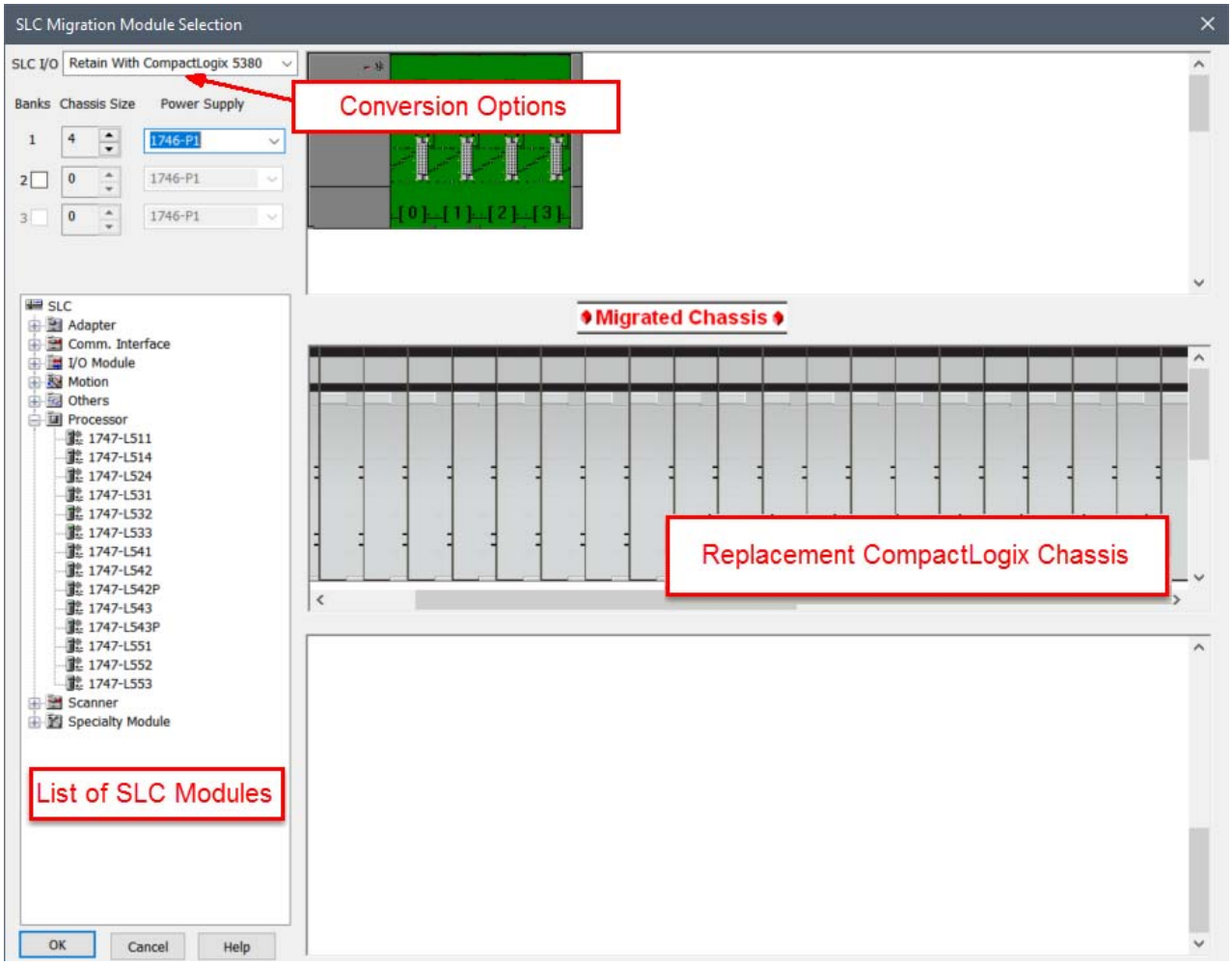
5. In the SLC Migration Chassis Selection dialog box, click Add Chassis.

   IAB opens the Add Chassis dialog box.

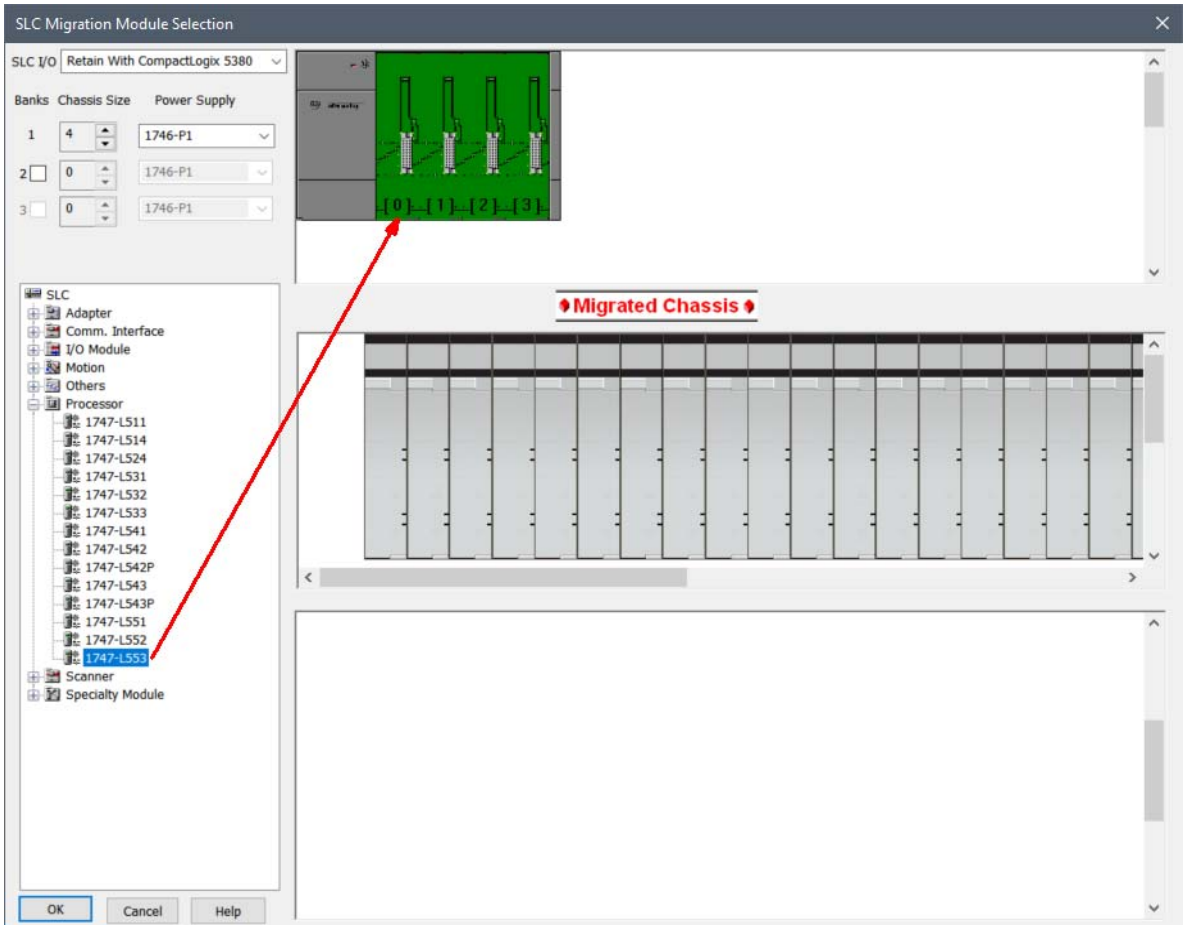


6. Click OK in the Add Chassis dialog box to accept the default name for the new chassis (SLC001).

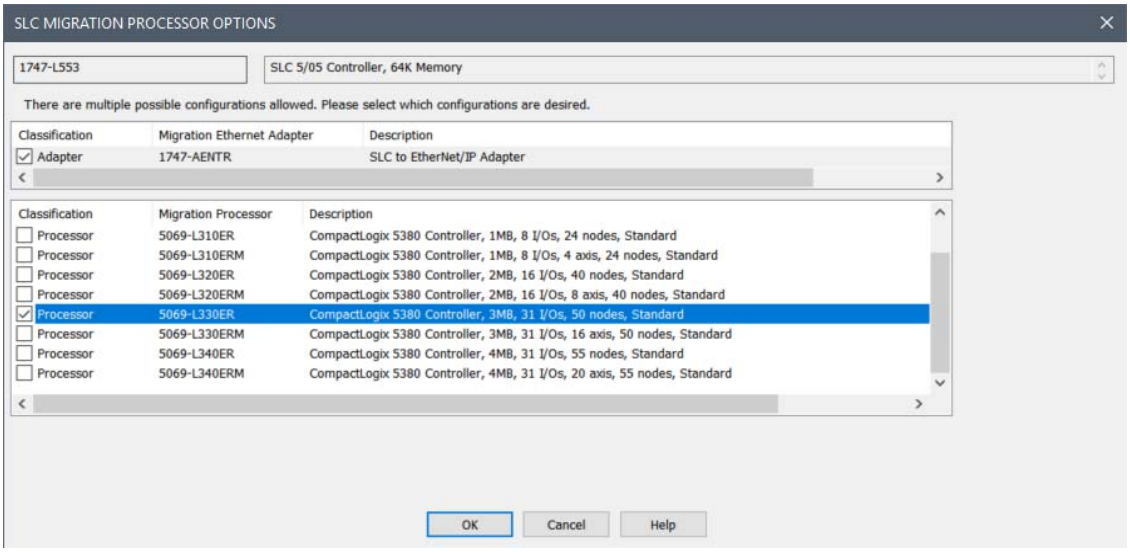
   IAB opens the SLC Migration Module Selection dialog box. This is where you make the conversion selections for this chassis. Notice the different areas of this window.



7. Select your chassis size and choose power supply.
8. From the Processor module list, expand the Processor heading and drag your processor module to slot 0 of the SLC chassis.



Because IAB has found multiple possible CompactLogix controller migration options, the SLC Migration Conflict Resolution Dialog box appears.

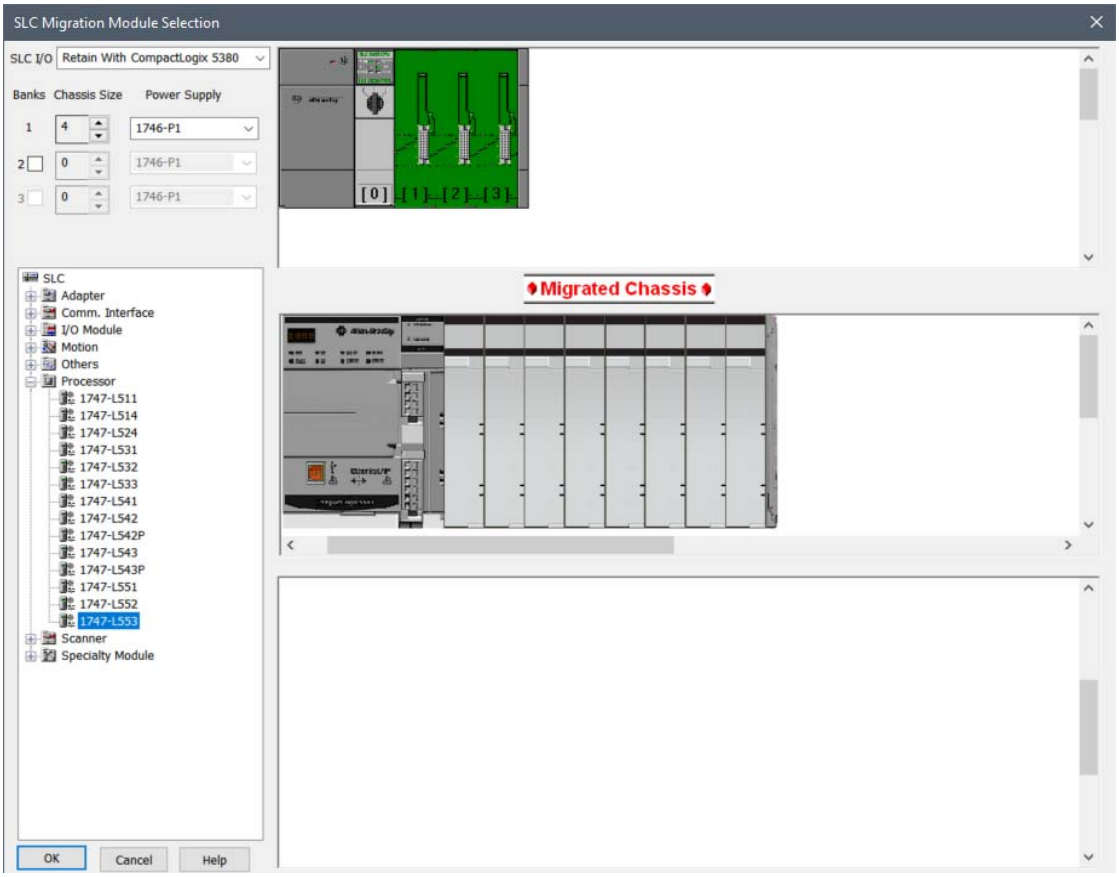


Conflict resolution dialog boxes appear when you must make a decision about the conversion. The information in the dialog box is specific to the action you are performing. In this case, we must select the CompactLogix processor that we wish to use.

9. Select a processor in the list and click OK.

   In this example, we show the 5069-L330ER module.

   IAB adds processors to both the SLC chassis and the replacement CompactLogix chassis. Additionally, IAB also adds a 1747-AENTR Ethernet adapter to the retained I/O on the SLC chassis at the bottom of the display.



10. Repeat steps 8 and 9 to fill remaining slots.

> If you incorrectly place a module, right-click the module and click Remove Module to try again.

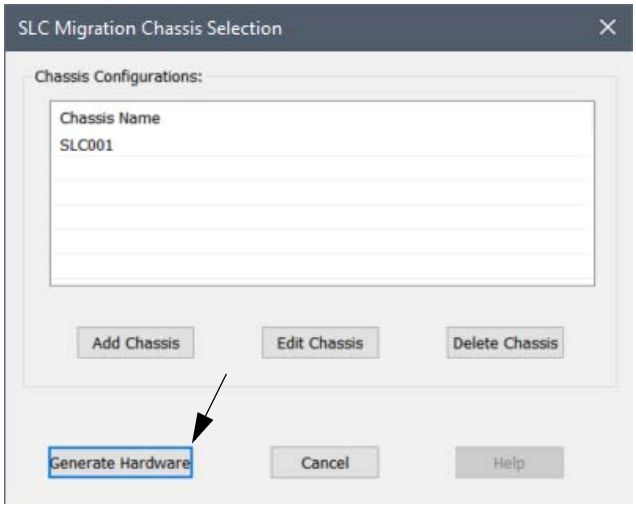
Most SLC I/O discrete, analog, and specialty modules are compatible with the 1747-AENTR adapter.

For a list of supported and unsupported modules, see I/O Modules on page 75.

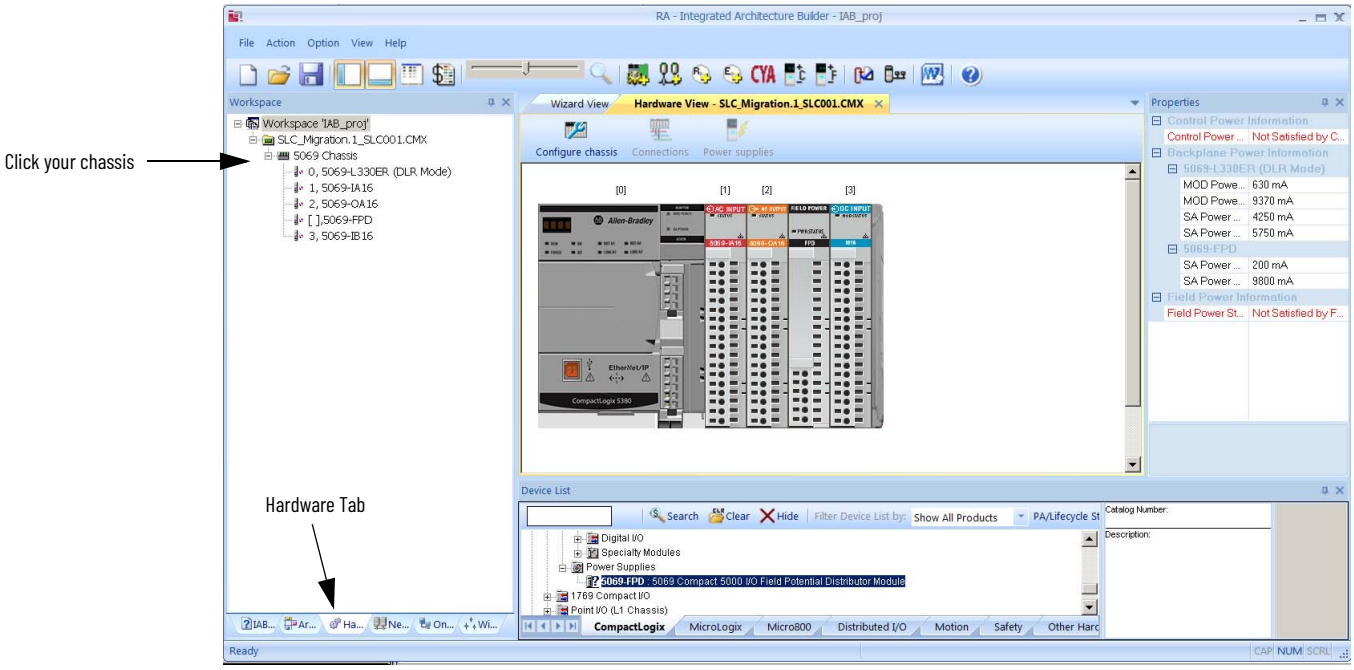
If you have a module that is not compatible, the following warning occurs.

IMPORTANT Modules not supported as part of a retained I/O solution, when connected to a Logix controller, are not placed into the lower chassis in IAB.

11. Once the local SLC chassis is complete, click OK.
12. In the SLC Migration Chassis Selection dialog box, click Generate Hardware to create the wizard-defined CompactLogix configuration in IAB.



13. Click the Hardware tab in the lower left corner of the IAB window and click the chassis to see the hardware.



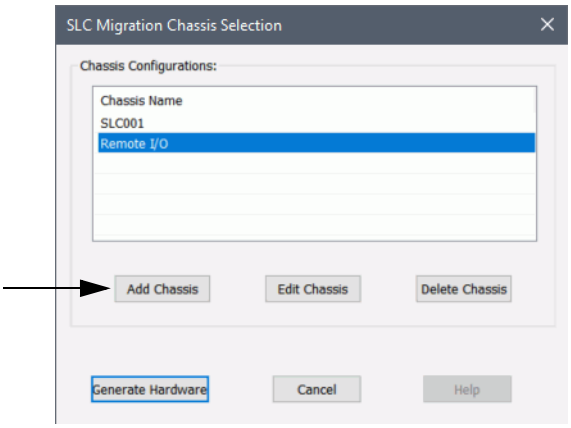
This chassis contains all SLC I/O from the local rack that was replaced.

14. Click the save icon to save your project.

Adding a Chassis

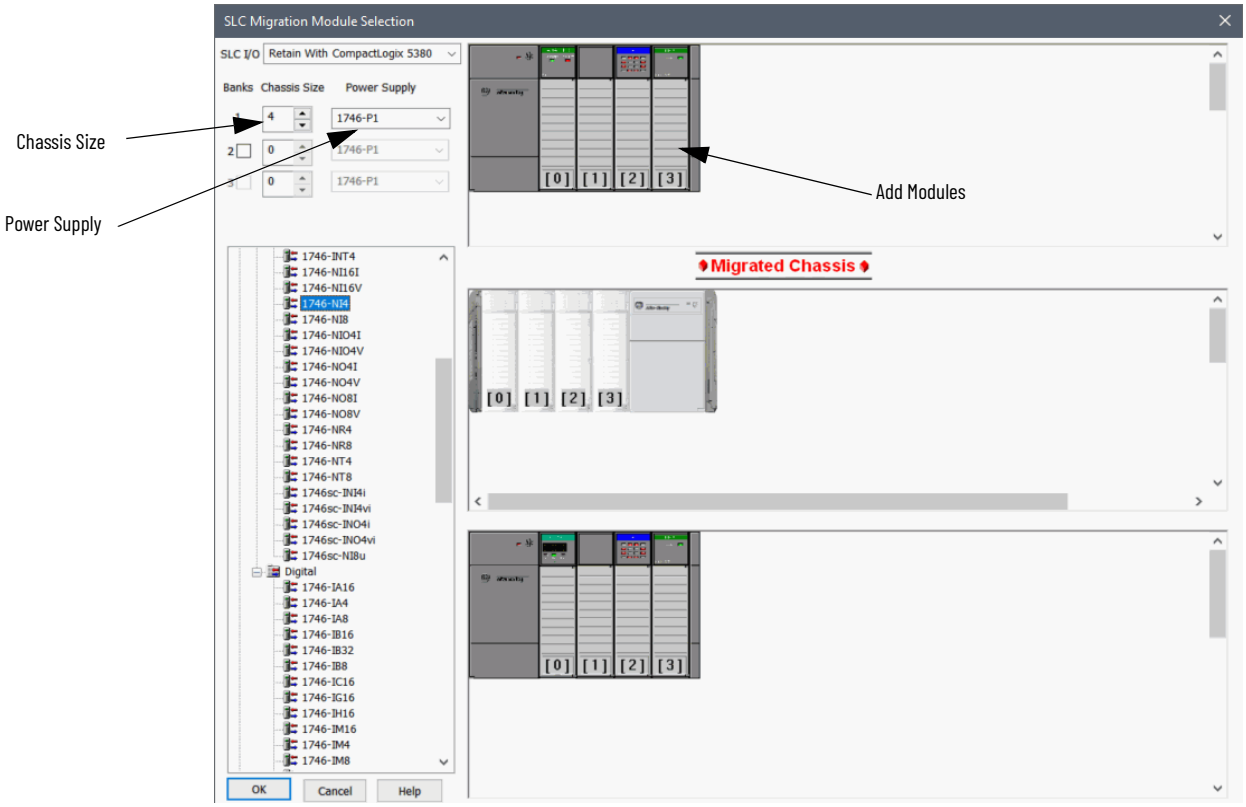
If your system does contain additional remote chassis, you can add to the existing chassis.

1. On the SLC Migration Chassis Selection dialog box, click Add Chassis.



This chassis replaces the remote SLC I/O chassis in your existing system.

2. Name this chassis, for example, SLC002_Remote, and click OK.
3. Select the chassis and power supply.
4. Choose to retain the SLC I/O for this chassis.



5. Add your remote I/O adapter to slot 0 of the SLC chassis.

IMPORTANT IAB replaces the 1747-ASB adapter with a 1747-AENTR Ethernet adapter in the replacement SLC remote I/O chassis.

6. Add in the rest of your I/O modules.

If you incorrectly place a module, simply right-click the module and click Remove Module to try again.

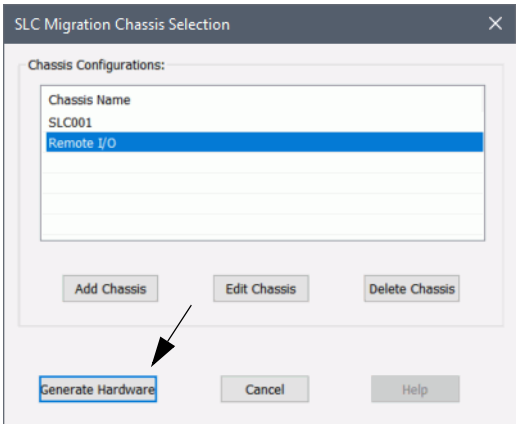
Most 1746 and 1747 discrete, analog, and specialty modules are compatible with the 1747-AENTR adapter.

For a list of supported and unsupported modules, see I/O Modules on page 75.

If you have a module that is not compatible, the following warning occurs.

IMPORTANT Modules not supported as part of a retained I/O solution, when connected to a Logix controller, are not placed into the lower chassis in IAB.

7. Once the remote SLC chassis is complete, click OK.
8. In the SLC Migration Chassis Selection dialog box, click Generate Hardware to create the wizard-defined CompactLogix configuration in IAB.



9. Click the Hardware tab in the lower left corner of the IAB window.
10. Click the SLC_Migration.1_SLC002 - Remote chassis to see the hardware.

    This chassis contains all SLC I/O from the remote rack that you replaced. A 1747-AENTR module has been added to connect this I/O remotely over Ethernet to the CompactLogix controller that is replacing our SLC processor.
11. Click the save icon to save your project.

You have three options when converting the second, remote chassis:

- To convert the local SLC I/O, leave the remote SLC chassis in place, add the Ethernet wiring, and rebuild the BOM.

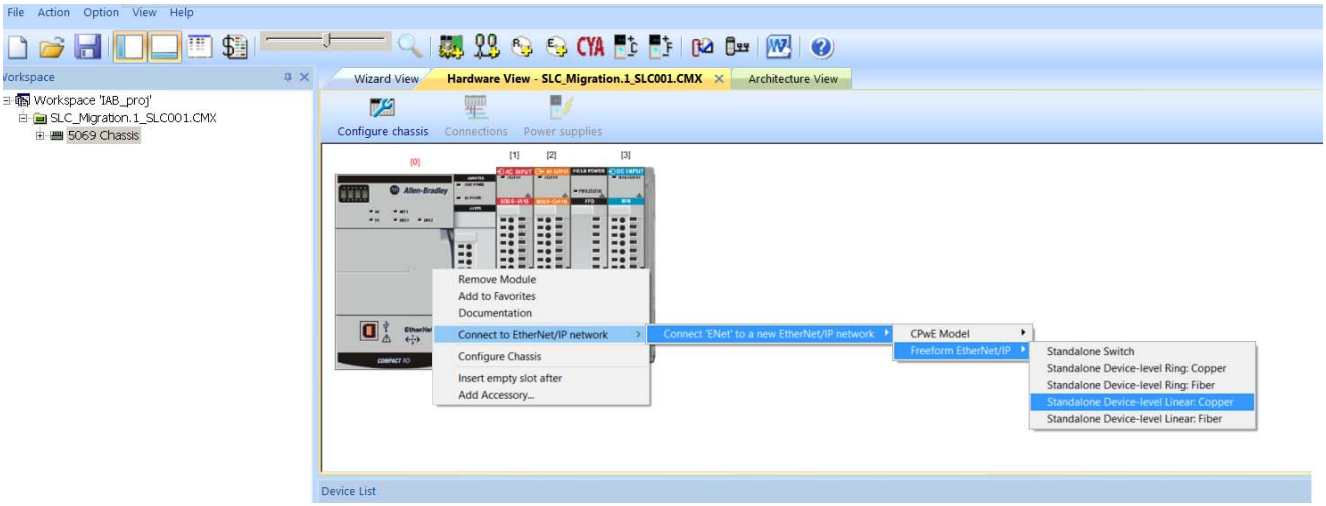
- Add the I/O modules from the second SLC chassis to the new local CompactLogix system.
- Swap out the remote SLC I/O chassis with a more cost-effective FLEX™ I/O or POINT I/O system.

In either case, this is best done outside the wizard.

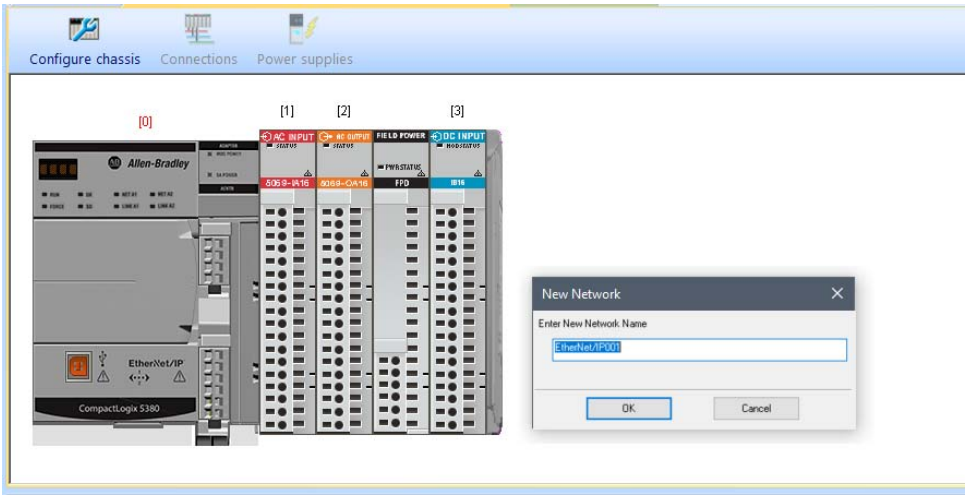
Network Connection Options for the Remote (SLC System) I/O Chassis

Connecting I/O systems to a controller is best accomplished by first creating a network connection on the controller itself. Because the remote SLC I/O chassis is configured with a 1747-AENTR Ethernet adapter, you can connect it to the CompactLogix chassis by using an Ethernet network.

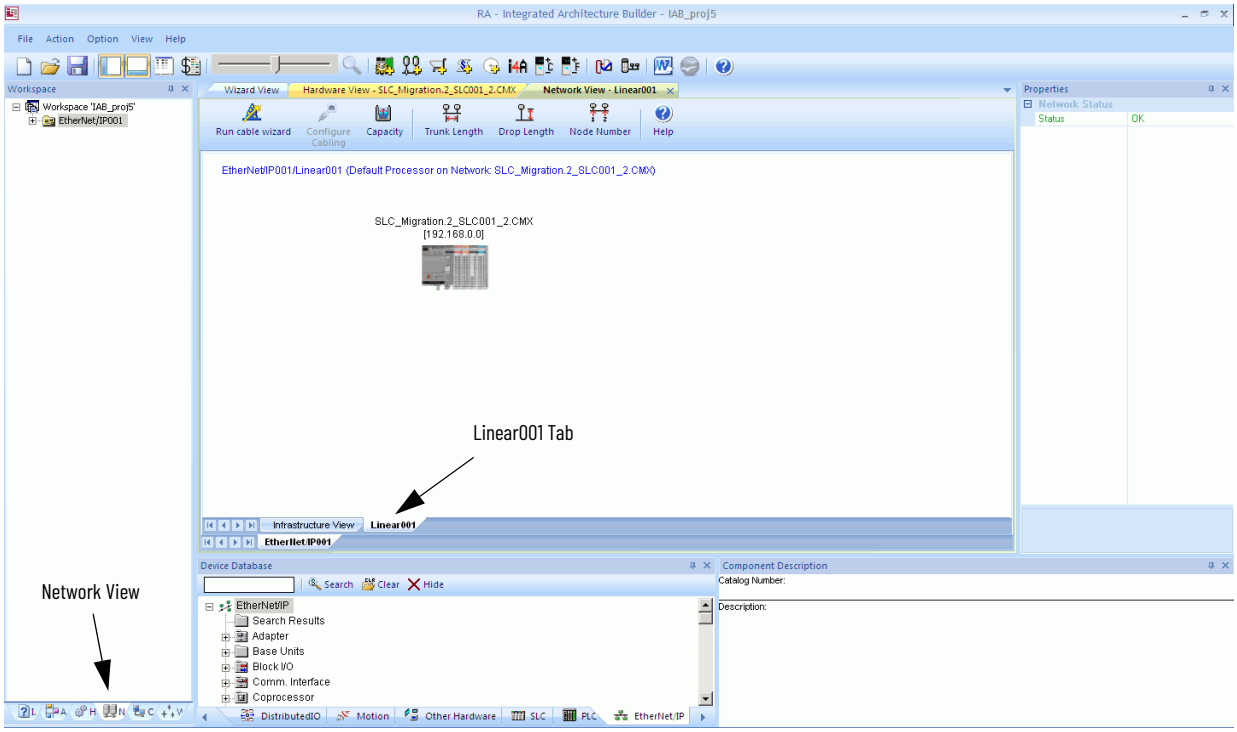
1. In the Hardware View for the SLC_Migration_SLC001.CMX chassis, right-click the controller and choose Connect to EtherNet/IP network > Connect 'Enet' to new EtherNet/IP network > Freeform EtherNet/IP > Standalone Device-Level Linear: Copper.



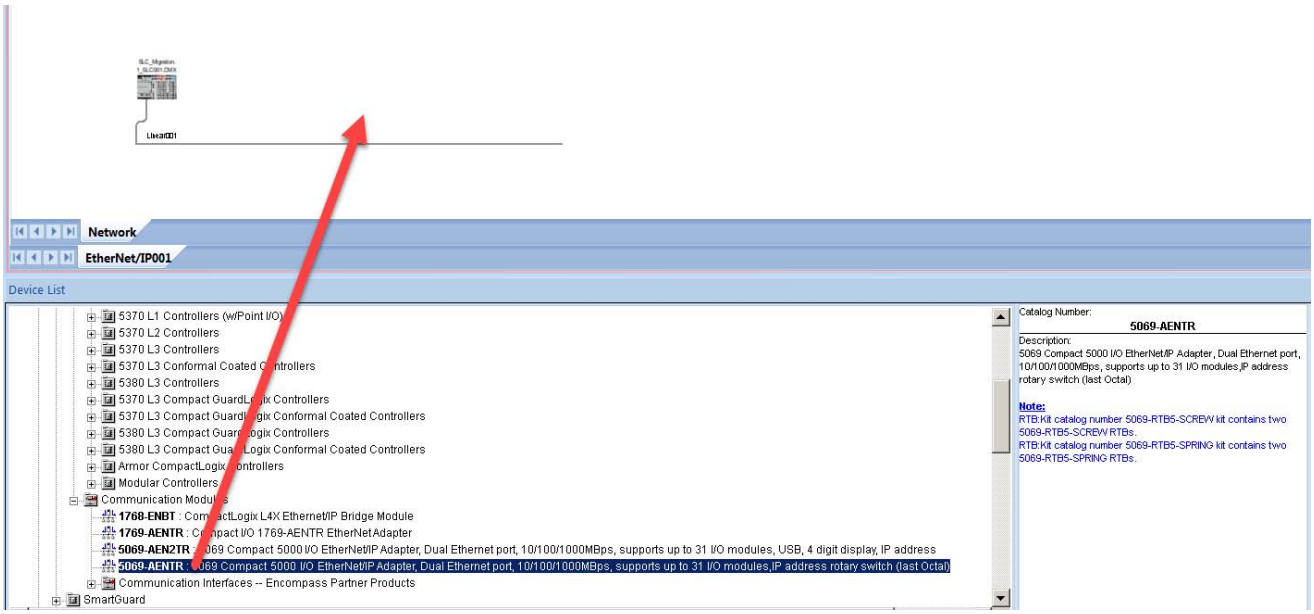
3. Click OK to accept the default network name.



4. Choose the Network tab.

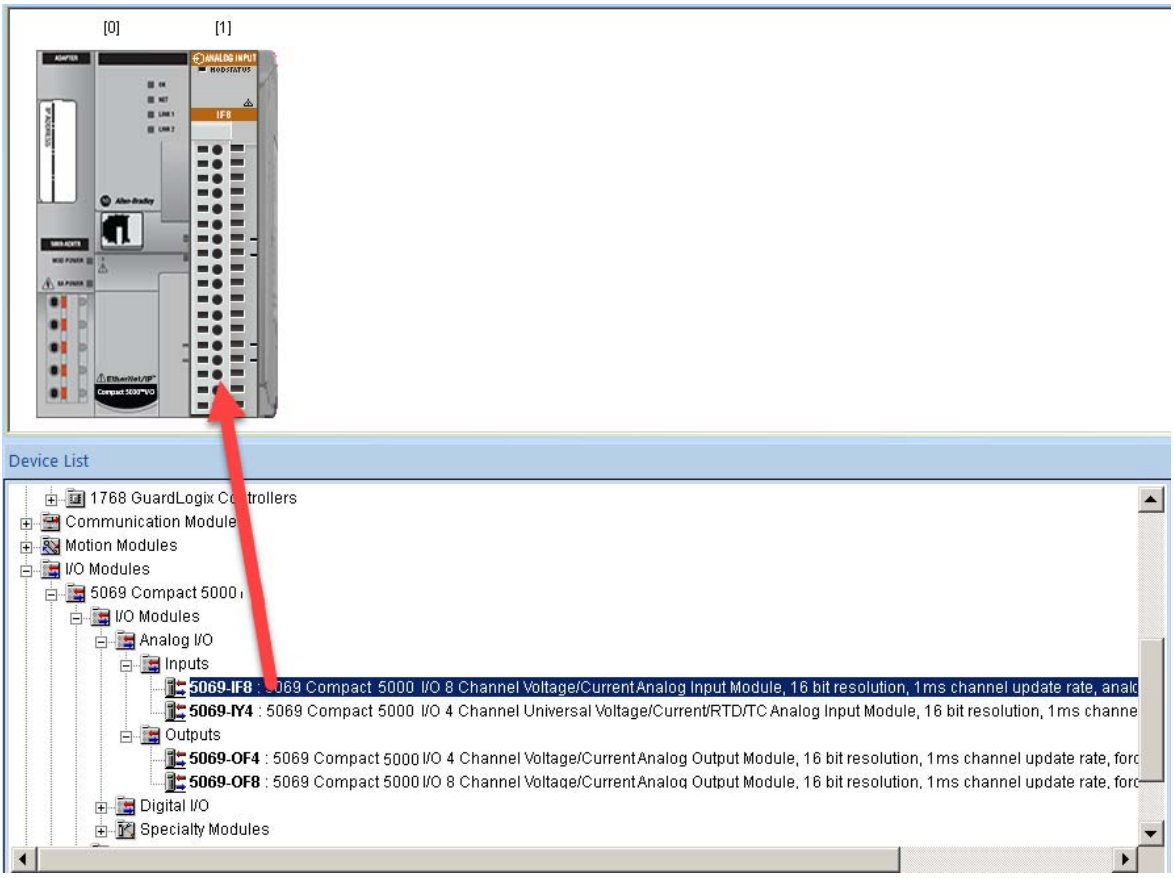


5. Drag-and-drop a remote I/O adapter into the network space.



In this example, we use the Compact 5000 I/O 5069-AENTR adapter.

6. Add an I/O module to the remote I/O adapter.



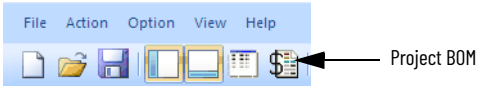
In this example, we add an 8-point analog input terminal.

7. The example SLC architecture is complete. You can now save your project.

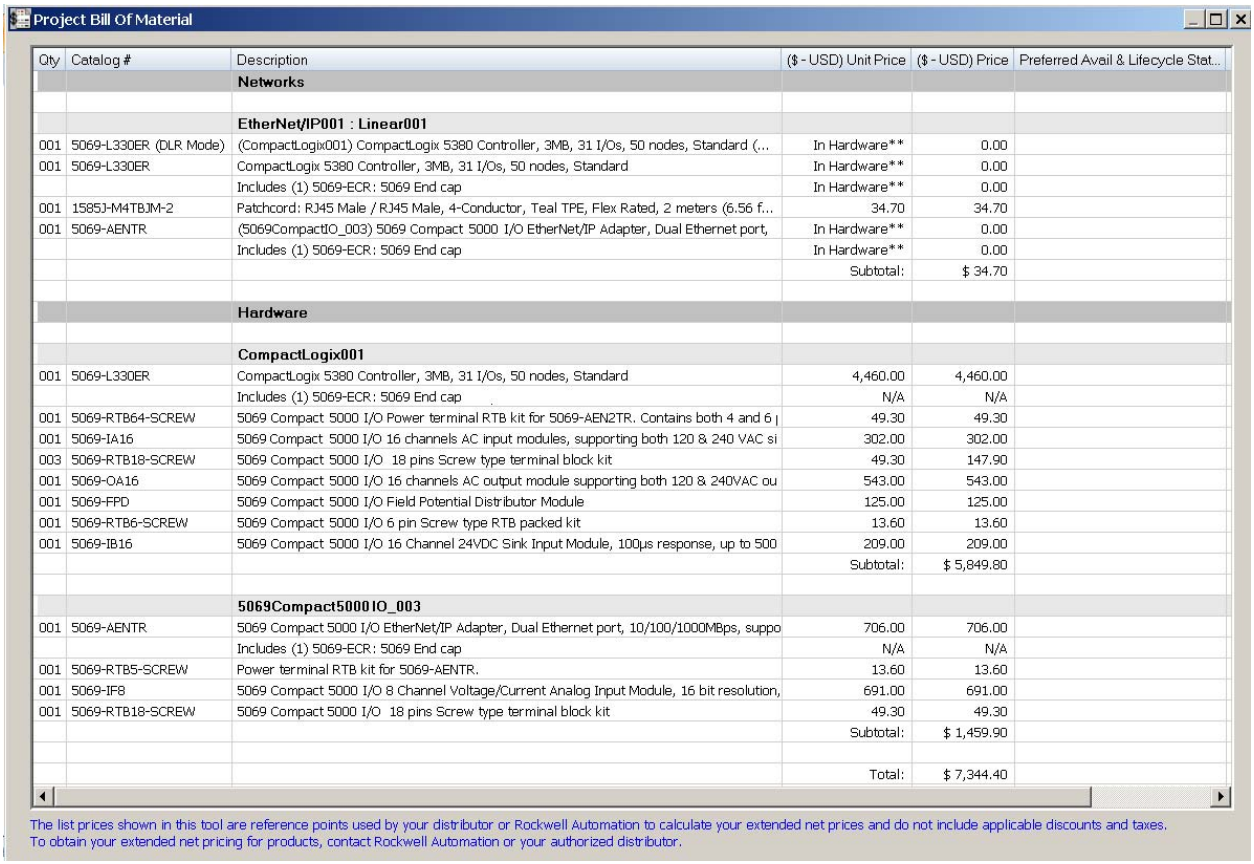
Create Project Bill of Materials

Follow these steps to get an idea of how much your conversion is going to cost and what is needed.

1. From the Menu Bar, click the Project Bill of Materials (BOM) icon.



From this dialog box, we can get a clear view of the material necessary to make the conversion based on the chassis layouts.



Project Bill Of Material

Qty	Catalog #	Description	($ - USD) Unit Price	($ - USD) Price	Preferred Avail & Lifecycle Stat...
		Networks			
		EtherNet/IP001 : Linear001			
001	5069-L330ER (DLR Mode)	(CompactLogix001) CompactLogix 5380 Controller, 3MB, 31 I/Os, 50 nodes, Standard (...	In Hardware**	0.00	
001	5069-L330ER	CompactLogix 5380 Controller, 3MB, 31 I/Os, 50 nodes, Standard	In Hardware**	0.00	
		Includes (1) 5069-ECR: 5069 End cap	In Hardware**	0.00	
001	1585J-M4TBJM-2	Patchcord: RJ45 Male / RJ45 Male, 4-Conductor, Teal TPE, Flex Rated, 2 meters (6.56 f...	34.70	34.70	
001	5069-AENTR	(5069CompactIO_003) 5069 Compact 5000 I/O EtherNet/IP Adapter, Dual Ethernet port,	In Hardware**	0.00	
		Includes (1) 5069-ECR: 5069 End cap	In Hardware**	0.00	
			Subtotal:	$ 34.70	
		Hardware			
		CompactLogix001			
001	5069-L330ER	CompactLogix 5380 Controller, 3MB, 31 I/Os, 50 nodes, Standard	4,460.00	4,460.00	
		Includes (1) 5069-ECR: 5069 End cap	N/A	N/A	
001	5069-RTB64-SCREW	5069 Compact 5000 I/O Power terminal RTB kit for 5069-AEN2TR. Contains both 4 and 6	49.30	49.30	
001	5069-IA16	5069 Compact 5000 I/O 16 channels AC input modules, supporting both 120 & 240 VAC si	302.00	302.00	
003	5069-RTB18-SCREW	5069 Compact 5000 I/O 18 pins Screw type terminal block kit	49.30	147.90	
001	5069-OA16	5069 Compact 5000 I/O 16 channels AC output module supporting both 120 & 240VAC ou	543.00	543.00	
001	5069-FPD	5069 Compact 5000 I/O Field Potential Distributor Module	125.00	125.00	
001	5069-RTB6-SCREW	5069 Compact 5000 I/O 6 pin Screw type RTB packed kit	13.60	13.60	
001	5069-IB16	5069 Compact 5000 I/O 16 Channel 24VDC Sink Input Module, 100µs response, up to 500	209.00	209.00	
			Subtotal:	$ 5,849.80	
		5069Compact5000IO_003			
001	5069-AENTR	5069 Compact 5000 I/O EtherNet/IP Adapter, Dual Ethernet port, 10/100/1000MBps, suppo	706.00	706.00	
		Includes (1) 5069-ECR: 5069 End cap	N/A	N/A	
001	5069-RTB5-SCREW	Power terminal RTB kit for 5069-AENTR.	13.60	13.60	
001	5069-IF8	5069 Compact 5000 I/O 8 Channel Voltage/Current Analog Input Module, 16 bit resolution,	691.00	691.00	
001	5069-RTB18-SCREW	5069 Compact 5000 I/O 18 pins Screw type terminal block kit	49.30	49.30	
			Subtotal:	$ 1,459.90	
			Total:	$ 7,344.40	

The list prices shown in this tool are reference points used by your distributor or Rockwell Automation to calculate your extended net prices and do not include applicable discounts and taxes.
To obtain your extended net pricing for products, contact Rockwell Automation or your authorized distributor.

In addition, the radio buttons along the bottom of the dialog box let you manipulate the information either as a consolidated spreadsheet or by slot location. All of these arrangements incorporate pricing either with List or Custom pricing models.

2. Click Close to close the BOM window.

Notes:

Conversion of Programs

Introduction

You can use RSLogix 500 version 12 to convert a .RSS program into .ACD program. This means that you can migrate an SLC 500 project into a Studio 5000 project with its in-built migration capability.

The Studio 5000 environment, which includes the Logix Designer application, was introduced in version 21. If you are using RSLogix 5000 software version 20, the steps are nearly identical.

This section will step through the program migration with RSLogix 500 version 12 for an SLC 500 controller.

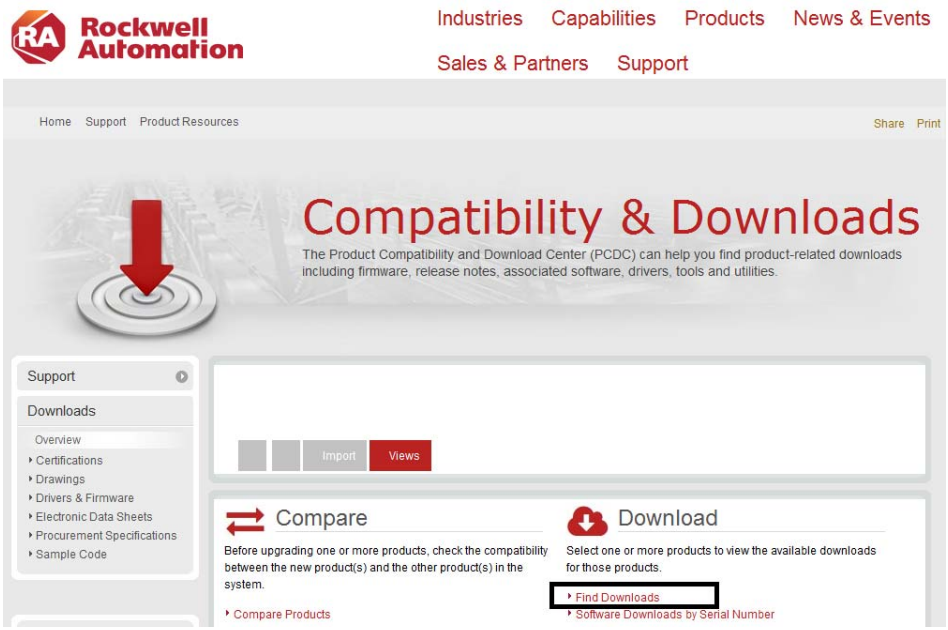
For hardware migration, see the Logix 5000 Controllers General Instructions Reference Manual, publication 1756-RM003.

For a more detailed explanation of the Project Migrator, see SLC 500 Hardware Migration Reference Manual, publication 1746-RM003.

IMPORTANT The process for converting an SLC system to a ControlLogix system is similar to converting to a CompactLogix system.

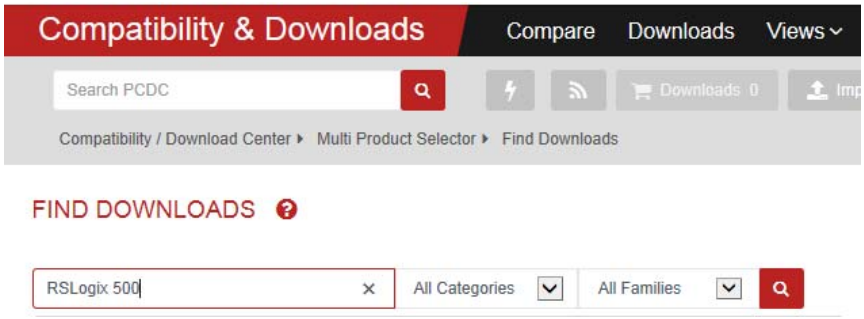
Download the Standalone RSLogix 500

You can download a standalone copy of the RSLogix™ Project Migrator from the Rockwell Automation Compatibility & Download page.

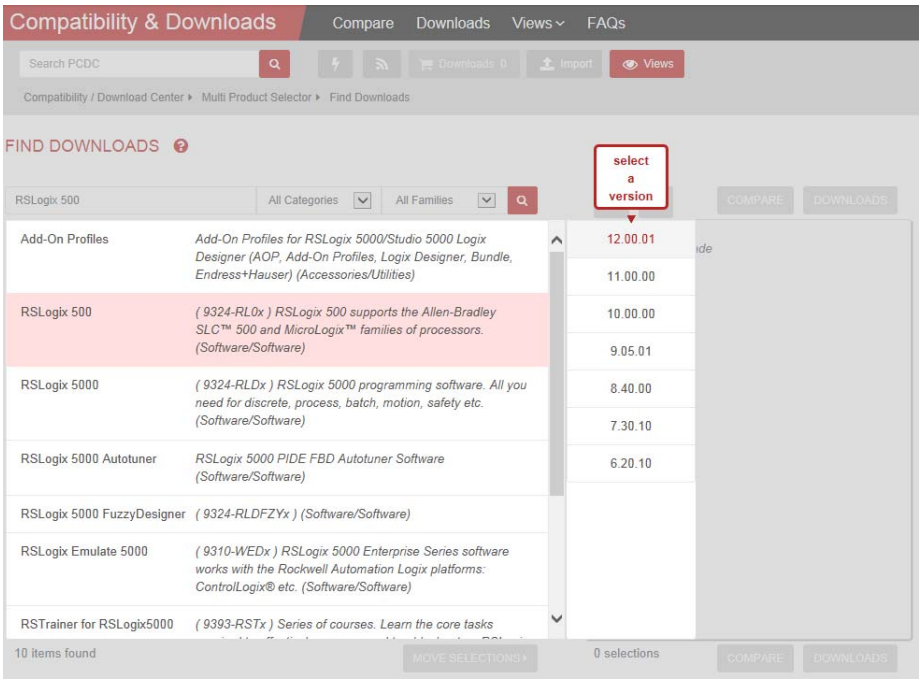


IMPORTANT You must log in with your Rockwell Automation account to be able to download the software.

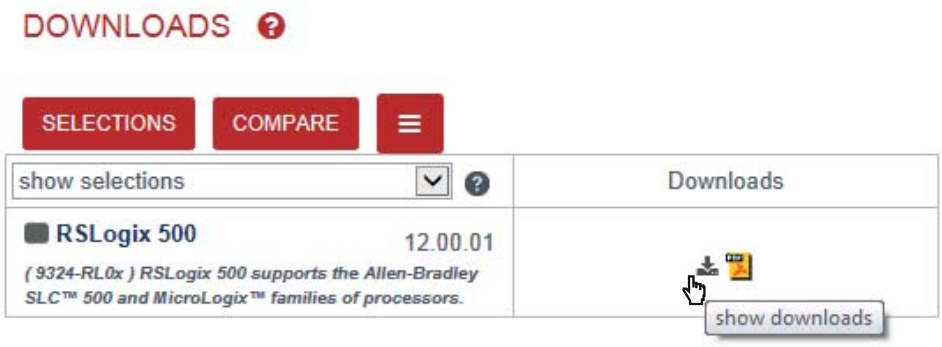
1. Go to Rockwell Automation Compatibility & Download page.
2. Under the Download section, click 'Find Downloads'.
3. In the Product Search box of the page appears, type RSLogix 500.



4. From the product list, select **RSLogix 500** > **version 12**.



5. From the selected versions on the right side of the screen, click RSLogix 500 version 12 and click Downloads.
6. On the Downloads page, click the show downloads icon.



The RSLOGIX 500 dialog box appears with the links of all files that you can download for version 12.

7. Under the General options, click 'Download Software'.

What to Expect from the RSLogix Project Migrator

The goal of the RSLogix Project Migrator is to reduce the amount of work involved in migrating a PLC-5 or SLC 500 program to a Logix project. The RSLogix Project Migrator automatically converts the program logic, but it is not the complete solution. Depending on the application, you may need to do additional work to make the converted logic work properly.

The RSLogix Project Migrator produces a syntactically correct import/export file, but the exact intent of the original application could be lost. This loss could be due to differences in rules. (For example, rules of precedence, rules of indexed addressing, or rules of I/O addressing). When there is an error in the translation, the RSLogix Project Migrator records the error in the rung of the Logix routine in which it occurred. You can use that error message to analyze and fix the error.

ATTENTION: After running the conversion process, the resulting import/export file still requires further manipulation. You must map the I/O and use BTD, MOV, or CPS instructions to place this mapped data into the structures created by the conversion process.

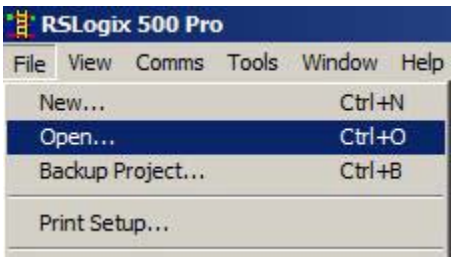
Application Code Conversion

The first step in a procedure of this type is to export the current SLC project into an ASCII text format.

1. From the desktop, double-click the RSLogix 500 programming software icon.

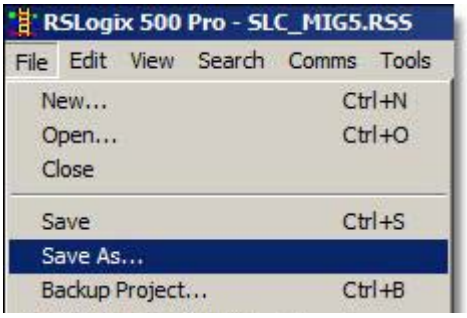
Or, choose Start > All Programs > Rockwell Software > RSLogix 500 > RSLogix 500.

2. From the File menu, choose Open to open the file you want to convert.

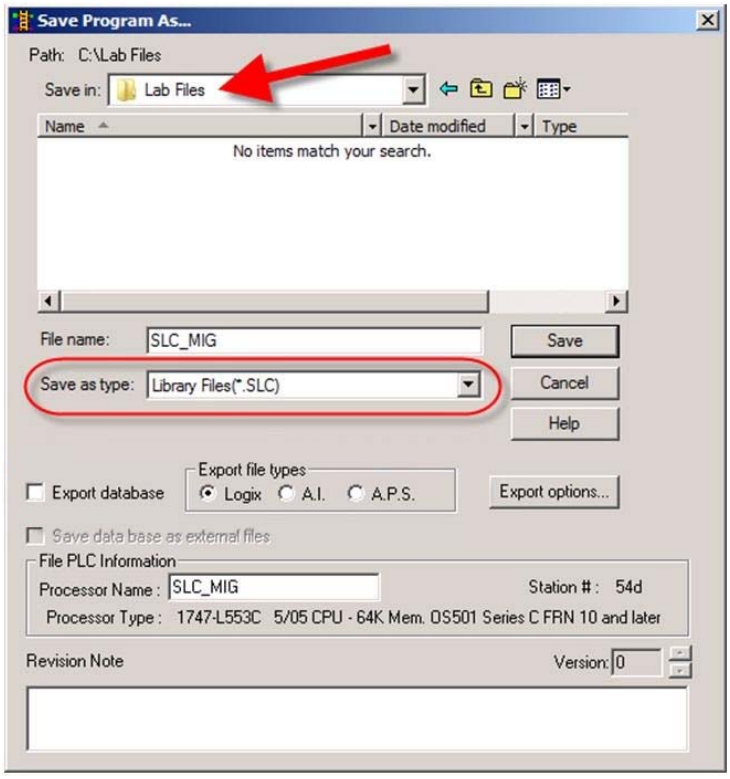


The first step is to export the current SLC project into an ASCII text format.

3. From the File menu, choose Save As.



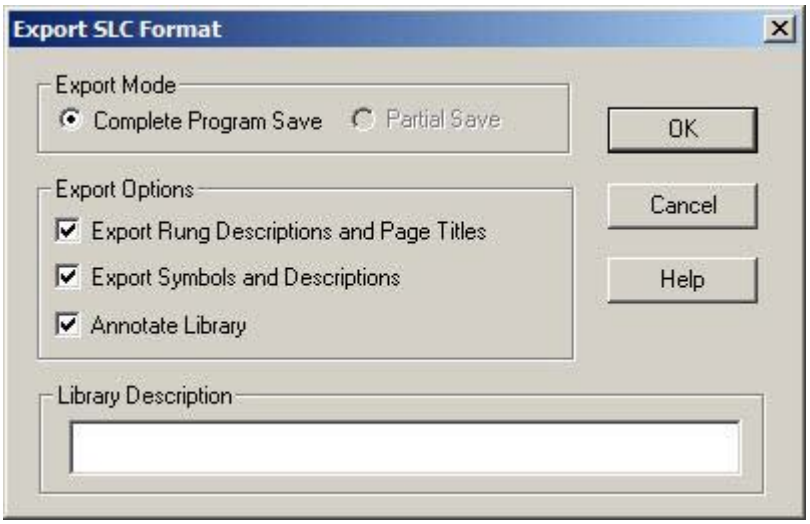
4. Browse to the folder where you are saving the converted program.
5. Set the file type to '.SLC.



The RSLogix Project Migrator accepts projects that have been saved as .SLC file type.

6. Click Save to continue.

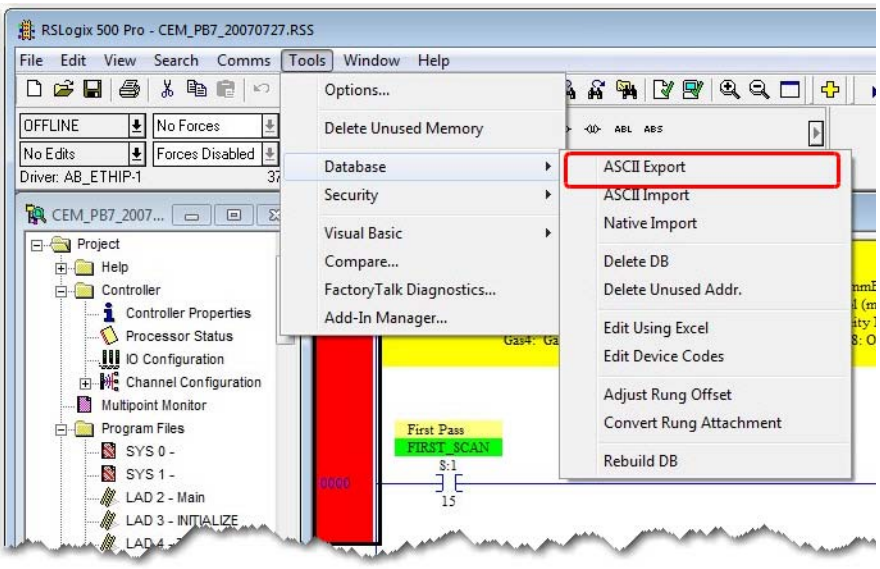
   Additional file export options are presented. For this example, we want to export the entire project, so the default settings here are fine.



7. Click OK.

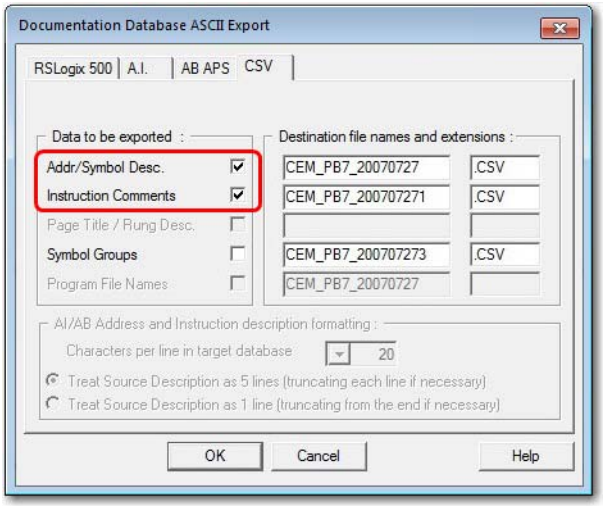
   After the file is converted, select the export options.

   From the Tools menu, select Database >ASCII Export.



   The Document Database ASCII Export window displays.

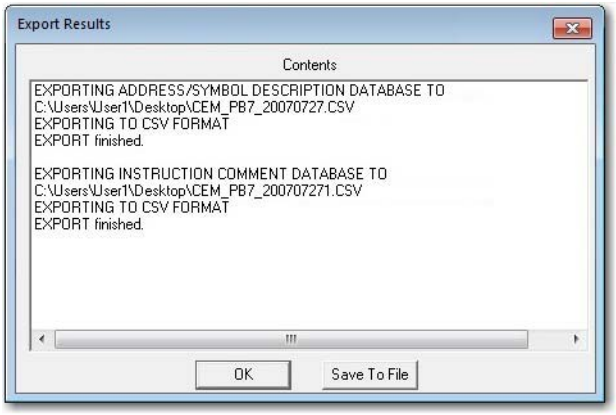
8. From the CSV tab, select Export Addr/Symbol Desc. and Instruction Comments, and then click OK.



The Select Export Destination Directory dialog displays.

9. Browse to the folder where you saved the .slc file, and click OK.

   The Export Result dialog displays.



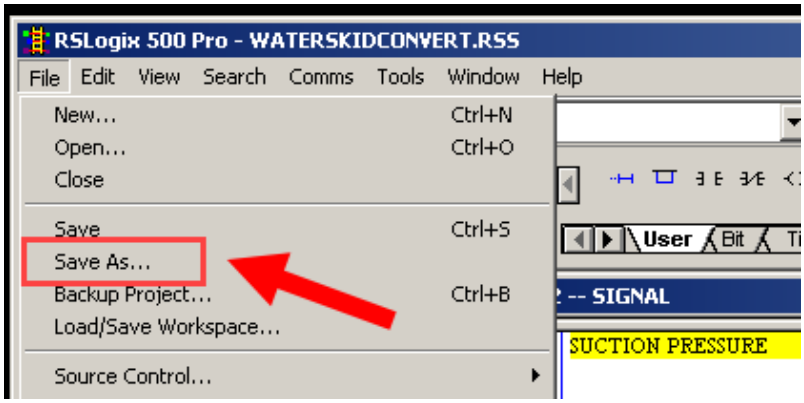
10. Click OK to close the RSLogix 500 software.

Convert RSLogix 500 Projects Into Studio 5000 Projects

Before you start the migration process, delete or remove unused memories, addresses, and routines:

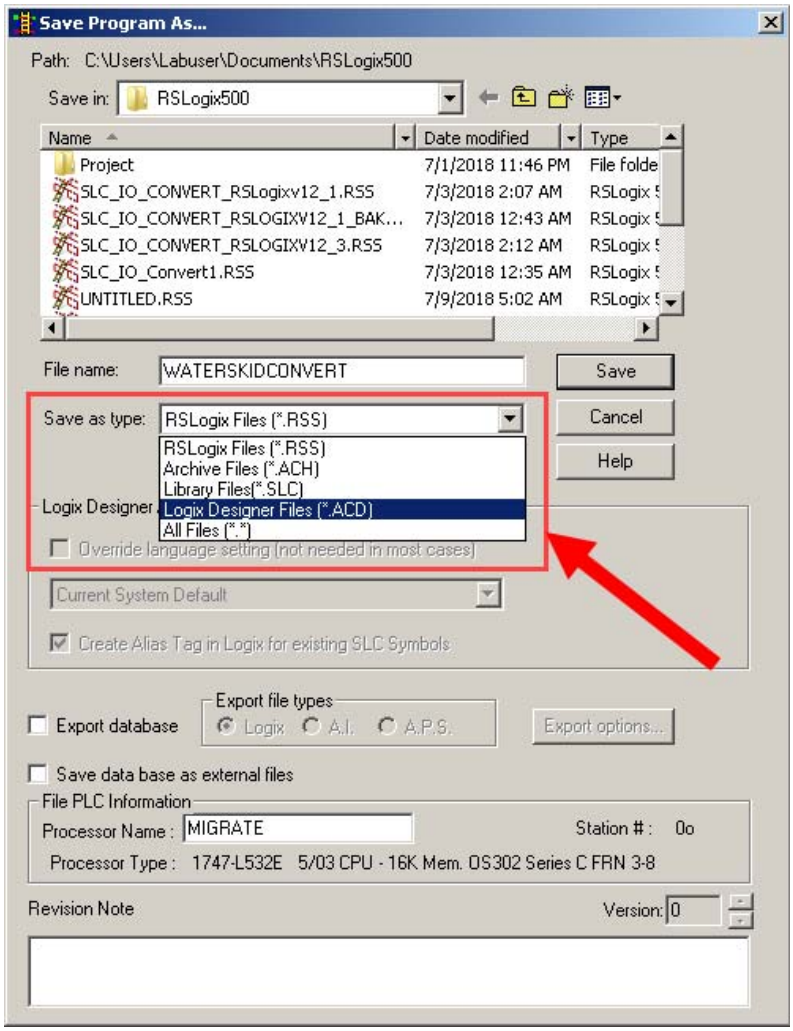
- Delete unused memory from Tools > Delete Unused Memory.
- Delete unused addresses from Tools > Database > Delete Unused Addresses.
- Remove SFC and STX routines to help avoid syntax errors that RSLogix Project Migrator fails to convert if encountered in the PC5 file.

1. From the RSLogix 500 version 12 software, open the SLC 500 project that you want to convert.
2. Click File > Save as...



The Save Program As... dialog box appears.

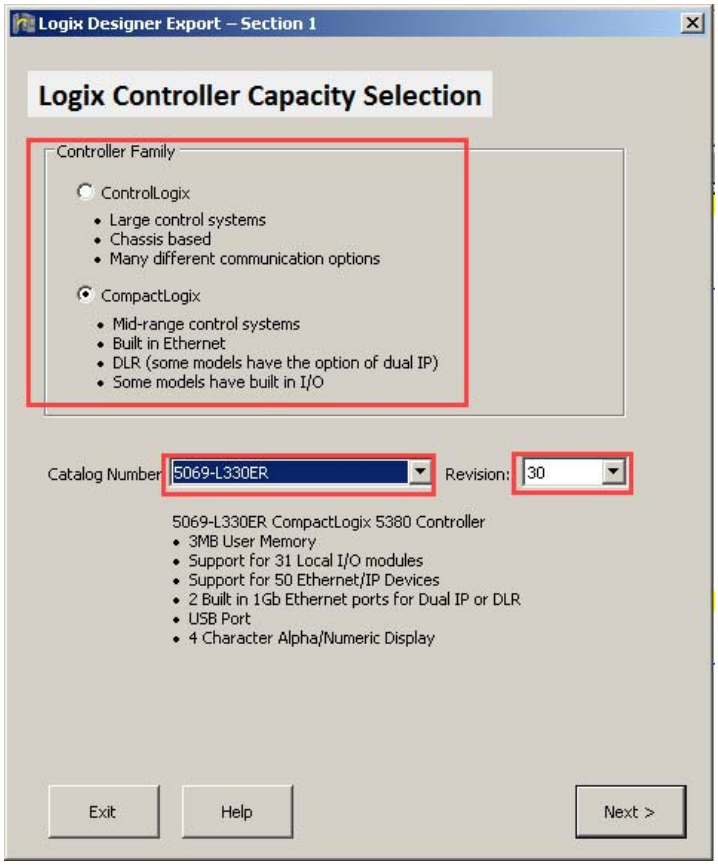
3. From the 'Save as type' pull-down menu, select Logix Designer application Files (*.ACD) and click Save.



The Logix Designer application Export dialog box appears.

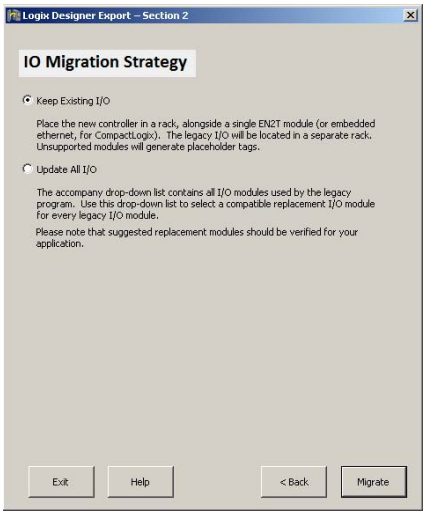
4. Select from the following options and click Next.

- Controller Family that you are migrating to
- Catalog Number
- Revision Number

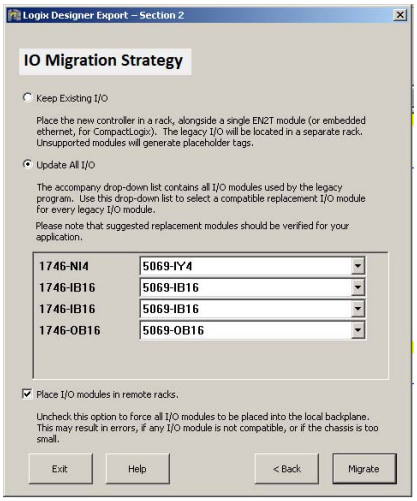


IMPORTANT This tool supports direct conversion to revision 30 or later.

5. Select your I/O Migration Strategy.
   a. Select 'Keep Existing I/O' to indicate that you are using a 1747-AENTR with the existing SLC I/O modules from your system.

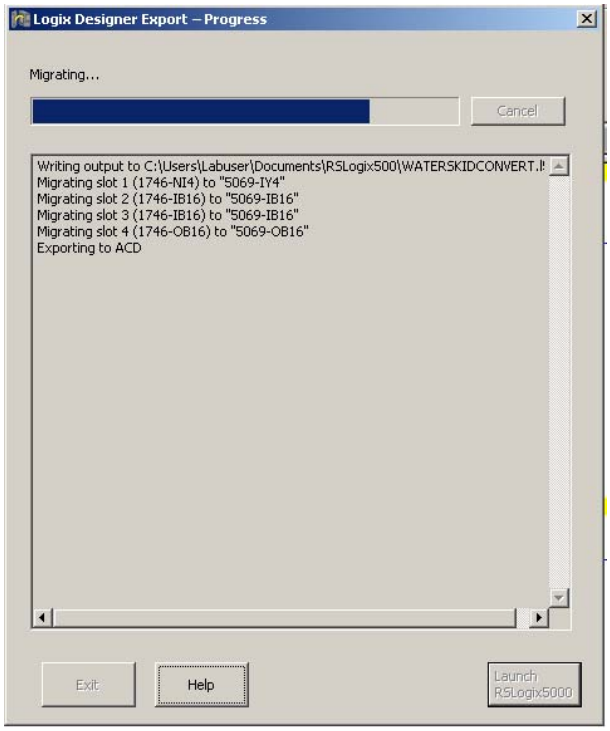


   b. Select 'Update All I/O' to allow the tool to select a compatible module. In this example, we place the new I/O module in a remote rack.



6. Click Migrate.

   The 'Logix Designer Export - Progress ' dialog box shows the progress of the migration process.



When the migration process is complete, open the migrated project in Studio 5000.

The migration tool supports CompactLogix version 30 or later. Ensure that Studio 5000 is pre-installed with the revision of CompactLogix equal to what you specified in step 4.

Resolve the Differences in the New Logix Program

Now that the SLC 500 program has been initially converted to a Logix program, you need to look at some of the most common elements that must be addressed for the CompactLogix project to properly control the installed SLC I/O modules.

See Appendix B for supported and unsupported I/O modules.

SLC Controllers Data Tables and Logix Controller Tags

The SLC 500 processors store all data in global data tables. You access this data by specifying the address of the data you want. A Logix controller supports data that is local to a program and data that is global to all tasks within the controller. A Logix controller can also share data with other controllers, and instead of addresses, you use tags to access the data you want. Each SLC 500 data table file can store several words of related data. A Logix controller uses arrays to store related data. The RSLogix Project Migrator converts the SLC 500 data table files into Logix arrays.

With a Logix controller, you use a tag (alphanumeric name) to address data (variables). The controller uses the tag name internally and does not need to cross-reference a physical address.

- In conventional programmable controllers, a physical address identifies each item of data.
  - Addresses follow a fixed, numeric format that depends on the type of data, such as N7:8, F8:3.
  - Symbols are required to make logic easier to interpret.
- In Logix controllers, there is no fixed, numeric format. The tag name itself identifies the data.
  - Organize your data to mirror your machinery.
  - Document (through tag names) your application as you develop it.

Resolve Program Code Issues

The RSLogix Project Migrator inserts a Program Conversion Error (PCE) instruction within the appropriate ladder rung to help you identify possible errors with the conversion. To complete the conversion process, locate, analyze, and fix any discrepancies involving the PCE instructions.

For a complete list of the PCE instruction Message IDs and their descriptions, please refer to Appendix A.

IMPORTANT After the correction of any errors, you must still spend time running and debugging the machine or process.

Work with PCE Instructions

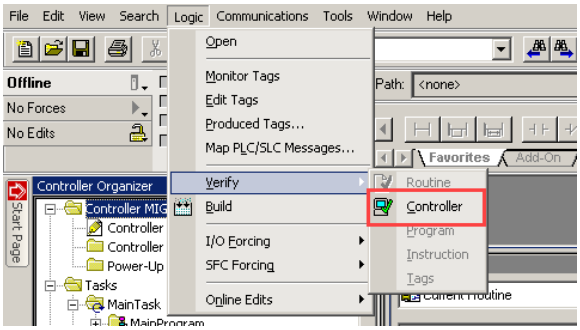
The RSLogix Project Migrator inserts a PCE instruction within the appropriate ladder rung to help you identify possible errors with the conversion. To complete the conversion process, locate, analyze, and fix any discrepancies using the PCE instructions.

For a list of PCE instruction errors, see Appendix A Program Conversion Errors (PCE) Messages on page 73.

Locate PCE Instructions

You can locate all PCE instructions by verifying the logic. The Verify > Controller task compiles the Logix program and checks for errors. This is an easy way to see where all of the PCE instructions are because the error checking points them out. To locate the PCE instructions, follow these steps.

1. From the Logic menu, select Verify > Controller.



   Or, from the menu bar, click the Verify Routine icon.

   The bottom of the dialog box displays results.

2. Double-click the error shown in the error window to go directly to the rung where the error occurred.

Some Warnings reference bits that are used as outputs in more than one rung, Duplicate Destructive Bits. While using this type of coding is not recommended, with careful programming, using the same outputs on several different rungs can be done.

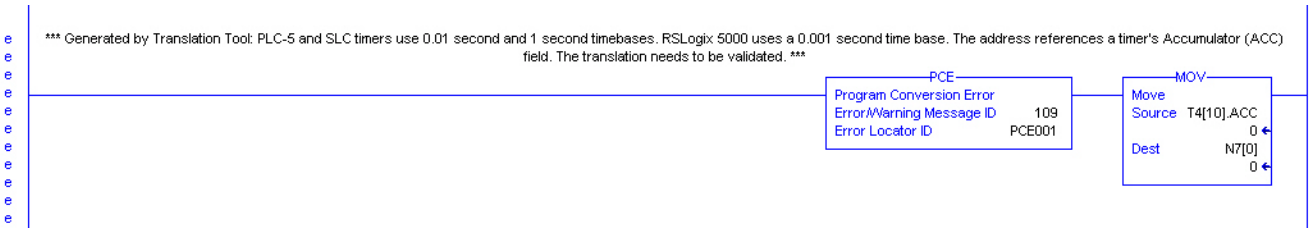
Recognize Instructions

Text is appended to the rung comments that have the PCE instruction. The message text begins with asterisks(*) and the words “Generated by RSLogix Project Migrator”, and ends with asterisks.

An example of a PCE instruction follows:

*** Generated by RSLogix Project Migrator: Source and destination types may differ *** ";

N: PCE(120, PCE011) COP(I1_008, N23[0], 4);



Resolve PCE Instructions

Once you import the converted Logix project, find each PCE instruction. A PCE instruction highlights a possible conversion error. Delete each PCE instruction and replace it with the appropriate, corrected logic.

Work with UNK Instructions

The RSLogix Project Migrator converts some PLC-5 and SLC 500 instructions that have no equivalent in the Logix architecture. Once you import these instructions into the Logix project, they appear as UNK instructions. You must delete each UNK instruction and replace it with the appropriate corrected logic.

Common PCE Issues

Translation greatly reduces the amount of work in a conversion; however, you may still get conversion errors that must be addressed individually.

- A common error occurs with all Timer related instructions. The timer instruction and its associated elements are compatible between the SLC 500 controller and the Logix platform. However, the SLC 500 controller supports only a .01 or 1 second time base for timers. Logix controllers support a 1 ms time base. The code conversion resulted in an increase of the timer Preset value by an order of magnitude. That is, the original Preset in the SLC 500 controller for this timer was 32767 and now it has been changed to 327670. All related references to this timer have been adjusted automatically, except those that reference a specific bit within the Accum or Preset itself. This can lead to errors on any rung that addresses a Timer.PRE or Timer.ACC because the scaling can be off and can be corrected only by user intervention.
- Another common conversion tool error is related to MSG instructions. Not all SLC 500 MSG instructions convert completely and, after the conversion, you need to verify that the data and path in all MSG instructions are correct.
- Several other SLC instructions may not convert properly or may not have the intended behavior. Among the more significant SLC instructions that can have issues are serial port instructions, Block Transfer instructions, FBC, and PID.

Map PLC/SLC Messages

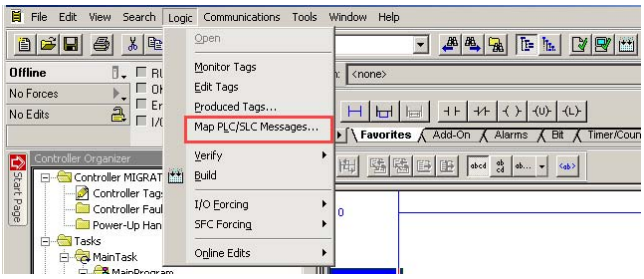
If your application does not require you to communicate with a legacy controller, you can skip this section.

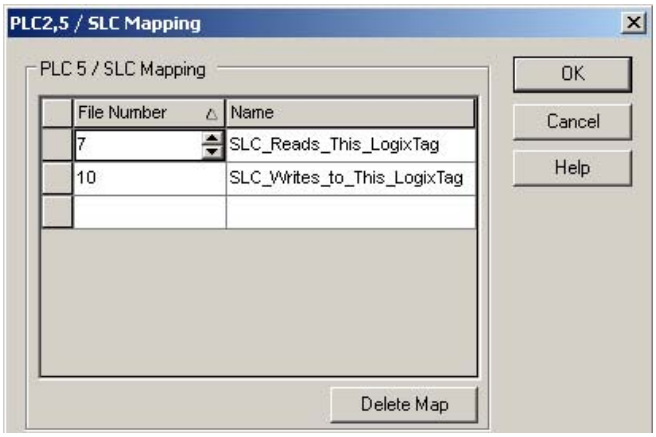
Map PLC/SLC Messages is a built-in feature of Logix controllers that lets older products that support a data table memory architecture, like PLC-2®, PLC-5, and SLC controllers, read/write to a Logix controller that has a tag-based memory architecture.

If after the conversion, legacy SLC controllers need to communicate to the converted Logix controller, Map PLC/SLC Messages can facilitate this. See the example below for an overview explaining PLC/SLC Mapping.

EXAMPLE If an incoming message from an SLC controller requests to read data from file N7:x, the Logix controller replies with data from tag SLC_Reads_This_LogixTag[x].

If an incoming message from an SLC controller requests to write data to file N10:x, the Logix controller places that data in tag SLC_Writes_to_This_LogixTag[x].





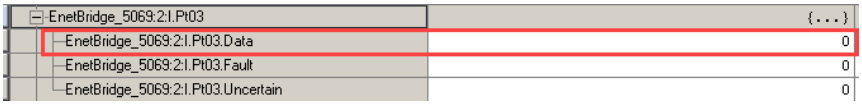
Map I/Os

The file structure in a Logix controller is tag-based. To facilitate the conversion, the RSLogix Project Migrator creates tags and arrays of tags to align and map the PLC-5 or SLC 500 files. See the following table for some examples:

SLC 500 Address	Map to the Logix Address
N7:500	N7[500]
N17:25	N17[25]
R6:100	R6[100]
C5:0	C5[0]
T4:6	T4[6]

SLC 500 Address	Map to the Logix Address
I:1.0	Local IO: Local:1.I.Ch00.Data Remote IO: [AdapterName]: 1.I.Ch00.Data
I:2/3	Local IO: Local: 2:I.Pt03.Data Remote IO: [AdapterName]:2:I.Pt03.Data
O:3.0	Local IO: Local:3:O.Ch00.Data Remote IO: [AdapterName]: 3:O.Ch00.Data
O:4/7	Local IO: Local:4:O.Pt07.Data Remote IO: [AdapterName] :4:O.Pt07.Data

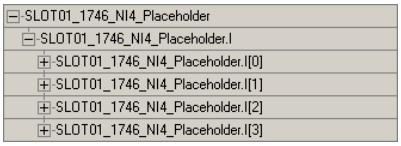
By default, the I/O data in CompactLogix 5380 controller is grouped by “channel”, and is no longer shown at the word level. For example, 'I:2/3' is a word-level digital input data. In the CompactLogix 5380 controller, the same data is accessible by 'Local: 2:I.Pt03.Data'. This data is grouped under 'Local: 2:I.Pt03', along with 'Fault' and 'Uncertain' statuses for you to verify the data.



Input module data group for CompactLogix 5380 controller.

In case there is no direct replacement for the I/O module, the migration wizard allows you to create an array variable placeholder and maintain all your existing assignment in your original program.

EXAMPLE Here is an example of placeholder tag for 1746-NI4:



Existing input assignment will be replaced by the element of the array.

Resolve Issues with Physical I/Os

When your migration strategy is to 'Keep Existing I/O', the SLC I/O is considered remote. Each SLC chassis is connected through the 1747-AENTR Ethernet adapters. This eliminates quite a bit of extra code that is otherwise required to communicate to the remote I/O.

There are three options to resolve physical I/O issues:

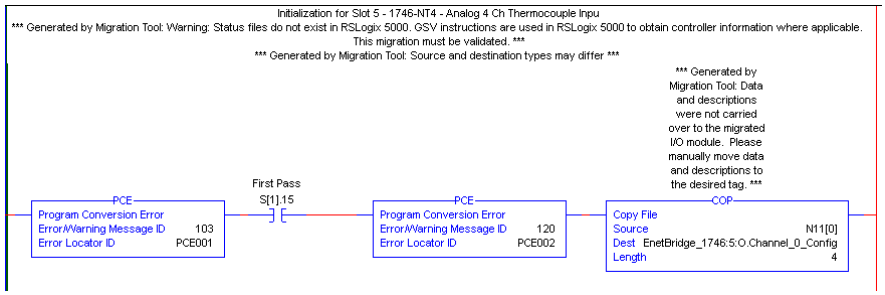
- MOV instructions
- CPS instructions
- Aliasing instructions

Each has its advantages and drawbacks depending on the type of data. Ideally, the RSLogix Project Migrator identifies I/O to be converted and offers you options during the conversion process rather than leaving this to you afterwards.

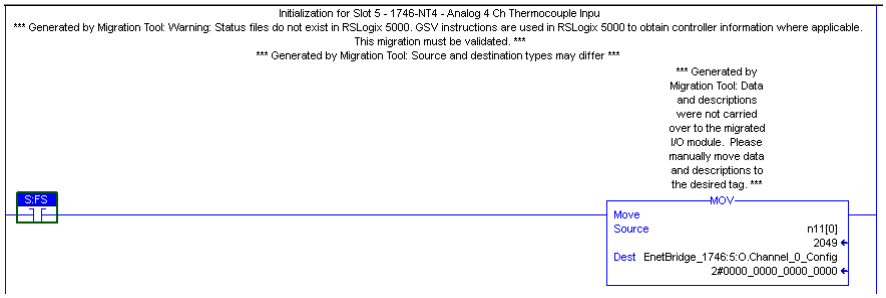
These examples illustrate situations that you can encounter when translating your files.

MOV Example

This first rung is used to initialize the configuration for a 1746-NT4 module in slot 5 of the local SLC 500 chassis.



The memory structure in the new environment is not the same as RSLogix 500. As a result, the Dest data location is not an array variable. We need to replace the COP instruction with MOV instruction to move the configuration for the channel to the corresponding module config tag.



This resolves the issue for Channel 0. Additional MOV instructions are required if other channels are used.

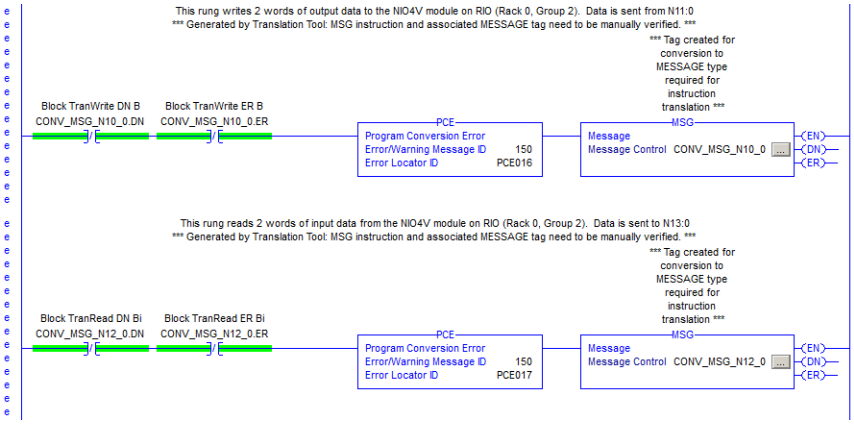
IMPORTANT The configuration for the individual channel resides in the C word of the module. These words are separated and a COP instruction cannot copy all configuration into the respective C word. The simplest way is to use MOV instruction 4 times to move each configuration data into the respective channel configuration.

SLC_ETN:1:C	{...}	{...}
SLC_ETN:1:C.Program_Mode_Output_Action_0	0	
SLC_ETN:1:C.Program_Mode_Output_Action_1	0	
SLC_ETN:1:C.Lost_Communications_Output_Action_0	0	
SLC_ETN:1:C.Lost_Communications_Output_Action_1	0	
SLC_ETN:1:C.Safe_State_Channel_0_Config	2#0000_0000_0000_0000	
SLC_ETN:1:C.Safe_State_Channel_0_Config_Input_Type_0	0	
SLC_ETN:1:C.Safe_State_Channel_0_Config_Input_Type_1	0	
SLC_ETN:1:C.Safe_State_Channel_0_Config_Input_Type_2	0	
SLC_ETN:1:C.Safe_State_Channel_0_Config_Input_Type_3	0	
SLC_ETN:1:C.Safe_State_Channel_0_Confi_Data_Format_0	0	
SLC_ETN:1:C.Safe_State_Channel_0_Confi_Data_Format_1	0	
SLC_ETN:1:C.Safe_State_Channel_0_Conf_Open_Circuit_0	0	
SLC_ETN:1:C.Safe_State_Channel_0_Conf_Open_Circuit_1	0	
SLC_ETN:1:C.Safe_State_Channel_0_C_Temperature_Units	0	
SLC_ETN:1:C.Safe_State_Channel_0_Filter_Frequency_0	0	
SLC_ETN:1:C.Safe_State_Channel_0_Filter_Frequency_1	0	
SLC_ETN:1:C.Safe_State_Channel_0_Conf_Channel_Enable	0	
SLC_ETN:1:C.Safe_State_Channel_1_Config	2#0000_0000_0000_0000	
SLC_ETN:1:C.Safe_State_Channel_1_Config_Input_Type_0	0	
SLC_ETN:1:C.Safe_State_Channel_1_Config_Input_Type_1	0	
SLC_ETN:1:C.Safe_State_Channel_1_Config_Input_Type_2	0	

CPS Example

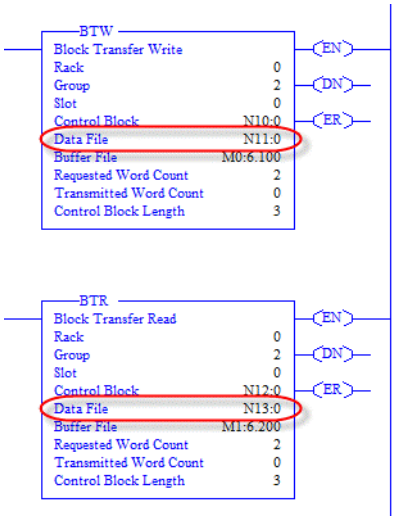
In RSLogix 500 software in order for many specialty modules to transfer data over RIO, sophisticated instructions called Block Transfers were used. In the

Logix Designer application, these Block Transfers were replaced with the MSG instructions shown below.



The need for messaging to/from these modules is no longer needed because the controller is communicating with them directly through the 1747-AENTR module.

Original RSLogix 500 Software Block Transfer Instructions

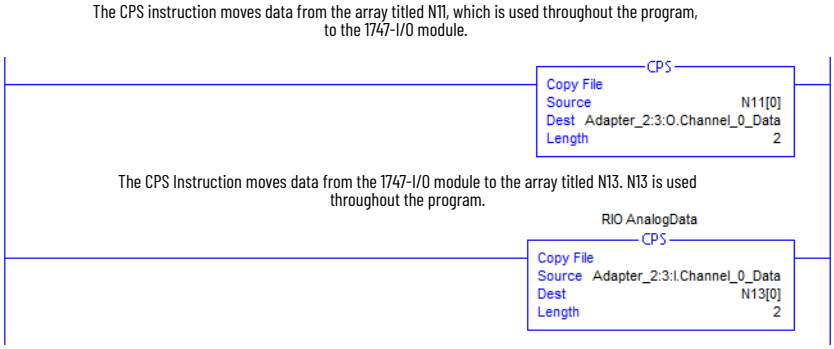


- Data to be written to the remote module in our example, was entered into N11:0 (2 words).
- Data to be read from the remote module into the controller was placed into N13:0 (2 words).

If consecutive I/O groups map to consecutive elements in an array, a CPS instruction must be used. Use a CPS instruction when you copy I/O or Produced/Consumed peer data of more than one DINT. This is the only information you need to create the instructions necessary to replicate the original program functionality.

IMPORTANT Extensive use of the CPS instruction can lock the tag database, which could affect other processes.

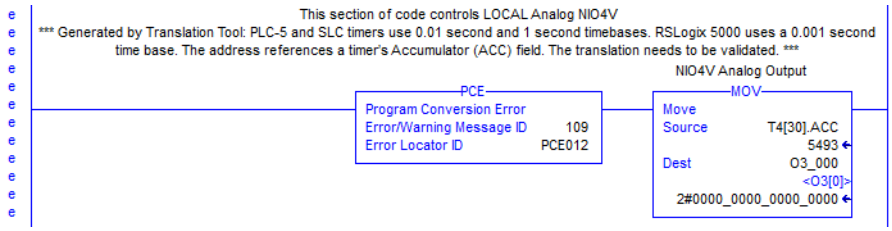
You can delete the rungs containing the PCE and MSG instructions and replace them with CPS rungs.



IMPORTANT The CPS instruction is intended to be used when copying I/O data or Produced/Consumed peer data to/from controller tags. For more information on the use of the CPS instruction, see Knowledgebase See Knowledgebase answer ID 50235, contact your local Allen-Bradley distributor, or Rockwell Automation sales representative.

Alias Example

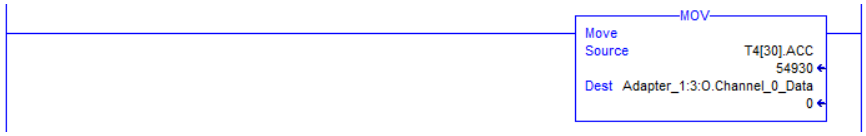
The rung in this routine uses a timer accumulator to simulate an analog output value for an SLC I/O module. Although this is not realistic, it does provide an instance in which aliasing can be most used.



In the example rung, a value is being moved into the O3_000 'alias' tag generated by the RSLogix Project Migrator. However, you need to actually tie this tag to a physical output address, namely the output location associated with the SLC I/O module.

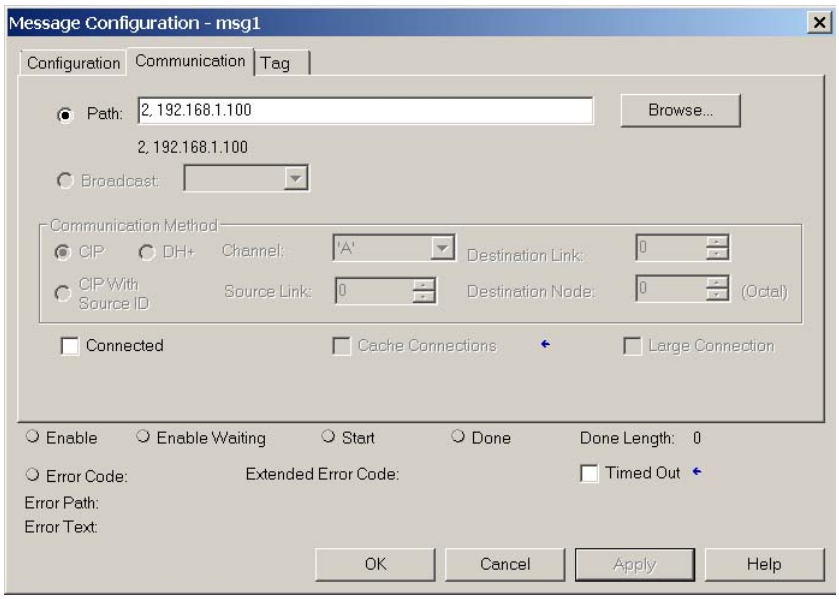
Aliasing the T4[30].ACC to the physical output address can seem like the right choice, but members of an array cannot be aliased in Logix. Because this instruction manipulates just a single word of data, you can modify the MOV instruction. Based on the I/O configuration, the base output data for the local SLC I/O module is contained in the 'Adapter_1:x:O' tag.

You need to modify the Dest element of the MOV instruction and delete the PCE instruction.



Complete the MSG Configuration

The RSLogix Project Migrator only partially converts MSG instructions. Use the Logix Designer application to configure each MSG instruction by completing the information on the Communication tab.



IMPORTANT For more information on how to configure MSG instructions, see the Logix 5000 Controllers General Instructions Reference Manual, publication 1756-RM003.

Replace SLC Processor and Adapters

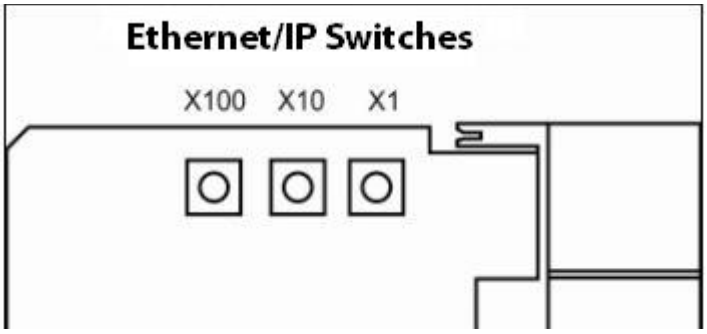
Mount and wire the CompactLogix system and replace the SLC first slot modules (SLC 500 processor or SLC I/O communication adapter module) with the SLC I/O Ethernet adapter, catalog number 1747-AENTR.

Set the Network Address Switches

The network address switches are set to 999 and DHCP enabled, by default. You can set the network Internet Protocol (IP) address in the following ways:

- Use the network address switches on the module.
- Use a Dynamic Host Configuration Protocol (DHCP) server, such as Rockwell Automation BootP/DHCP.
- Retrieve the IP address from nonvolatile memory.

The adapter reads the network address switches first to determine if the switches are set to a valid number. You set the node address by using the network address switches. Valid settings range from 001…254.



When the switches are set to a valid number, the adapter's IP address is 192.168.1.xxx (where xxx represents the number set on the switches).

The adapter subnet mask is 255.255.255.0 and the gateway address is set to 0.0.0.0. The adapter does not have a host name assigned, or use any Domain Name System when using the network address switch settings.

If the switches are set to an invalid number (for example, 000 or a value greater than 254 excluding 888), the adapter checks to see if DHCP is enabled. Setting the switches to 888 restores default factory settings.

IMPORTANT If you set the value of the adapter switch to 888 and then power cycle the module, the following occurs:

- The DHCP Enabled function is enabled (set to True).
- The Ethernet link is negotiated automatically. The Auto Negotiate function is set to True.
- The web server is enabled. The Disabled Web Server function is disabled.
- The Ethernet ports are disabled. Both ports are re-enabled once the switches are returned to their previous value and power is cycled.

DHCP Enabled and Not Enabled

DHCP Enabled and Not Enabled	
If DHCP is	**Then the Adapter**
Enabled	Asks for an address from a DHCP server. The DHCP server also assigns other Transport Control Protocol (TCP) parameters. The 1747-AENTR factory default is DHCP enabled. When you apply power, the module sends a message containing its hardware address to any DHCP server on the network. The server(s) replies by sending a message with an appropriate IP address for the adapter. The adapter responds by acknowledging to a server that the adapter will use the offered IP address.
Not enabled	Uses the IP address (along with other TCP configurable parameters) stored in nonvolatile memory. When the IP address assigned to the module, as indicated in the four-character dot-matrix status display, is changed through the DHCP configuration utility, the DHCP is disabled. When power is cycled to the device, the device uses the new configuration and implements the new IP address.

Determine Power Requirements

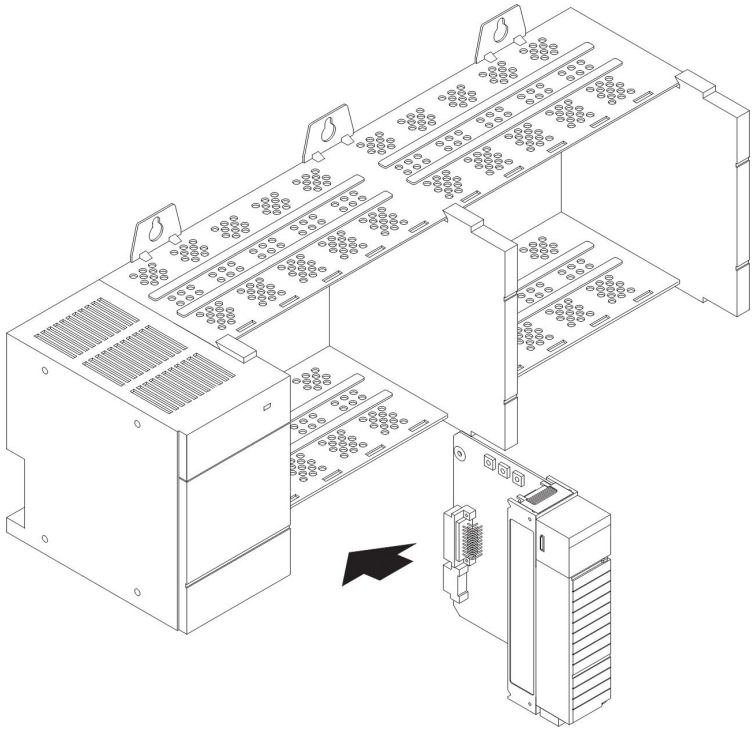
The Ethernet adapter requires 5V DC with current consumption of 470 mA. The power is supplied through backplane from SLC power supply. Remember to consider this requirement when planning your system configuration.

Install the Adapter Module in the Chassis

After you set the appropriate switch assemblies for your adapter module, follow these procedures for installation.

See the Industrial Controller Wiring and Grounding Guidelines publication 1770-4.1 for proper grounding and wiring methods to use when installing your module.

1. Remove power from the I/O chassis before inserting (or removing) the module.
2. Align the circuit board with the chassis card guide in the left slot.



3. Install the module in slot 0 of the chassis by aligning the circuit board with the chassis card guide.

   The 1747-AENTR module must be installed only in slot 0 (leftmost slot) of the chassis.
4. Press firmly and evenly to seat the module in its backplane connectors.

   To remove the module, press the releases at the top and bottom of the module and pull it out.

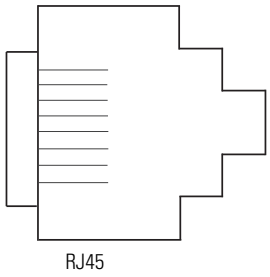
ATTENTION: Do not force the module into the backplane connector. If you cannot seat the module with firm pressure, check the alignment. Forcing the module can damage the backplane connector or the module.

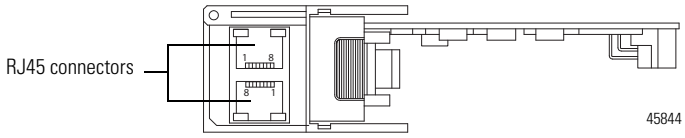
Connect Your Adapter to the Ethernet/IP Network through RJ45 Connection

Connect your 1747-AENTR adapter module to an Ethernet/IP network as shown below.

Wire the RJ45 connectors as shown.

	Signal
1	TxData+
2	TxData-
3	Recv Data+
4	Reserved
5	Reserved
6	Recv Data-
7	Reserved
8	Reserved



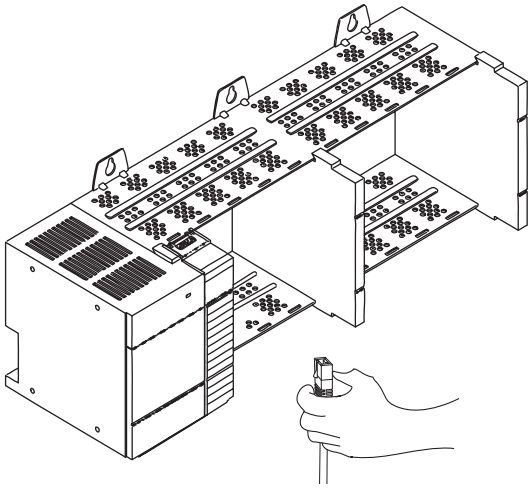


1747-AENTR module bottom view

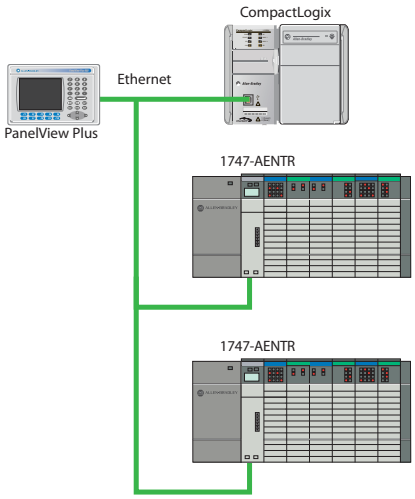
To connect the module to the network, follow these steps.

WARNING: If you connect or disconnect the communication cable with power applied to this module or any device on the network, an electrical arc can occur. This could cause an explosion in hazardous location installations. Be sure that power is removed or the area is nonhazardous before proceeding.

1. Attach the cables with the RJ45 connectors to the two Ethernet ports on the bottom of the module.



2. Attach the other end of the cables to the devices in your network.

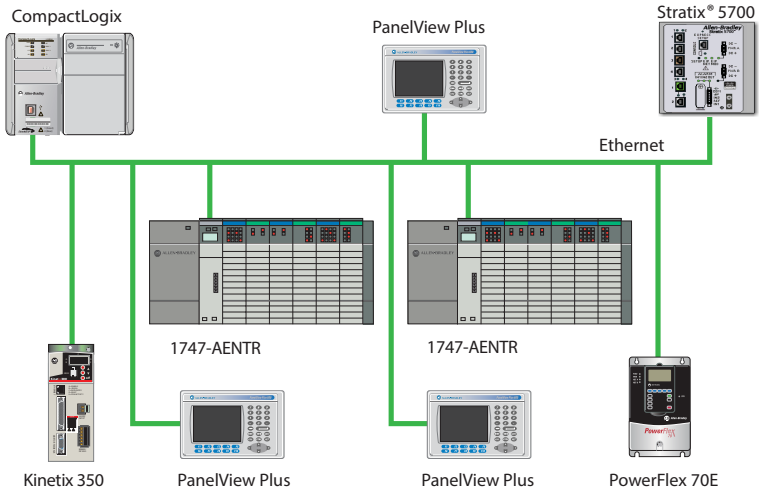


3. Configure your 1747-AENTR adapter.

   See publication 1747-UM076 for information on configuring and using your SLC I/O adapter.

Replace Other Components

If your control system has legacy or competitive operator interface, variable-frequency drives, motion control, sensors, or motor control centers, you can migrate those products as well.



See the following publications for more information:

- PanelView™ Plus Terminals
  - For selecting your terminal: VIEW-SG001
  - For installing and operating your terminal: 2711P-UM001
- PowerFlex® Drives
  - For selecting your drive: PFLEX-SG002
  - PowerFlex 700S Adjustable Frequency AC Drive - Phase II Control (Frames 1...6) Installation Instructions, publication 20D-IN024. Provide information needed to install and wire a PowerFlex 700S Phase II Adjustable Frequency AC drive.
  - PowerFlex 700H Adjustable Frequency AC Drive / PowerFlex 700S High-Performance AC Drive, Frames 9...14 Installation Instructions,

publication PFLEX-IN006. Provides drive mounting and wiring information.

- PowerFlex 700S High-Performance AC Drive- Phase I Control Reference Manual, publication PFLEX-RM002 Provides information on specifications and dimensions, mounting, and detailed drive operation for the PowerFlex 700S with Phase I control.
- PowerFlex 700S High Performance AC Drive - Phase II Control Reference Manual, publication PFLEX-RM003 Provides information on detailed drive operation for the PowerFlex 700S with Phase II control.
- PowerFlex 700S High Performance AC Drive - Phase II Control Programming Manual, publication 20D-PM001 Provides information needed to startup, program, and troubleshoot PowerFlex 700S Phase II Adjustable Frequency AC drives.
- PowerFlex 700S with Phase II Control Technical Data, publication 20D-TD002 Provides information on options, specifications, ratings, dimensions, derating data, and other product information.
- PowerFlex 700H Adjustable Frequency AC Drive Programming Manual, publication 20C-PM001 Provides basic information needed to startup, program, and troubleshoot the PowerFlex 700H Adjustable Frequency AC Drive.
- Wiring and Grounding Guidelines for Pulse Width Modulated (PWM) AC Drives, publication DRIVES-IN001 Provides basic information needed to properly wire and ground PWM AC drives.

- Kinetix® Motion Control
  - For information about motion control: GMC-SG001

Other Considerations

The following are additional issues to keep in mind:

- The time base for instructions is fixed at 1 ms for a Logix controller. The conversion process scales PLC-5 and SLC 500 presets and accumulators accordingly. For example, a PLC-5 with a time base of 0.01 s and a preset of 20 is converted to a time base of 1 ms and a preset of 200.
- Instruction comments are not converted.
- A Logix controller is a 32-bit based controller. This means that most of the Logix instructions use 32-bit words, as opposed to the 16-bit words in PLC-5 processors. This means that instructions that use masks might work differently after the conversion.
- The conversion process creates alias tags for address comments. These aliases are then used in place of the converted tags.

  Alias tags use additional memory in a Logix controller, so delete alias tags that you do not plan to use. Use the Logix Designer application to delete aliases after you import the project.

Notes:

Convert Program Structures

Introduction

A Logix 5000 controller (CompactLogix 5380, Compact GuardLogix 5380, CompactLogix 5370, Compact GuardLogix 5370, ControlLogix 5580, GuardLogix 5580, or ControlLogix 5570 controllers) uses a different execution model than either the PLC-5 processor or the SLC 500 processor. The Logix 5000 controller operating system is a preemptive multitasking system that is IEC 61131-3 compliant and uses:

- Tasks
- Programs
- Routines

This chapter describes controller programs to explain migration results.

Divide Logic Into Tasks, Programs, and Routines

The tasks, programs, and routines work together as follows:

- **Tasks:** Tasks are used to configure controller execution. A task provides scheduling and priority information for a set of one or more programs. You can configure tasks as either continuous, periodic, or event tasks.
- **Programs:** Programs are used to group data and logic. A task contains programs, each with its own routines and program-scoped tags. Once a task is triggered (activated), all programs that are assigned to the task execute in the order in which they are listed in the Controller Organizer.

  Programs are useful for projects that are developed by multiple programmers. During development, the code in one program that makes use of program-scoped tags can be duplicated in a second program, which minimizes the possibility of tag name collisions.
- **Routines:** Routines are used to encapsulate executable code written in a single programming language.

  Routines contain the executable code. Each program has a main routine that is the first routine to execute within a program. You can use logic, such as the Jump to Subroutine (JSR) instruction, to call other routines. You can also specify an optional program fault routine.

IMPORTANT	Currently, the RSLogix Project Migrator converts only ladder instructions. SFC and structured text files are not converted.

As the RSLogix Project Migrator converts the PLC-5 or SLC 500 logic, consider the program structures in the following table.

IMPORTANT For more information on Logix 5000 Controllers, refer to the Logix 5000 Controllers Design Considerations Reference Manual, publication 1756-RM094.

Create Continuous Tasks

A Logix controller supports one continuous task that operates in a self-triggered mode. It restarts itself after each completion. The continuous task operates as the lowest priority task in the controller (one priority level lower than the lowest periodic task). This means that all periodic tasks will interrupt the continuous task.

The RSLogix Project Migrator automatically creates one continuous task that is named MainTask with a default watchdog setting of 500 ms. It contains a program that is named MainProgram, and uses a main routine that is named MainRoutine.

The RSLogix Project Migrator creates a continuous task, but it uses the EVENT instruction to better simulate the PLC-5/SLC 500 behavior.

Create Event Tasks

The RSLogix Project Migrator also creates Event tasks for each program file that is configured in the PLC-5 Main Control Program (MCP).

To call each Event task, the RSLogix Project Migrator creates EVENT instructions within the continuous task. It uses the PLC-5 status file to determine which is the first MCP and orders them accordingly, in the MainRoutine.

The SLC 500 processors do not contain an MCP, so ladder program 2, which is the main ladder program, becomes the main routine.

Create Periodic Tasks for Selectable Timed Interrupts (STIs)

Processor status word 31 contains the number of the ladder program, if any, that is designated for use as a selectable timed interrupt (STI). The RSLogix Project Migrator creates a Periodic task and converts this program file that is named file number STI into its main routine.

The RSLogix Project Migrator retrieves the STI interval from the processor status file. If necessary, the RSLogix Project Migrator converts the interval to a 1 ms time base. After the conversion, you have to edit the task properties to specify its priority.

Processor status bit S:2/1 allows enabling and disabling of the STI. A Logix controller does not support this. The RSLogix Project Migrator generates a PCE instruction if it encounters any references to S:2/1.

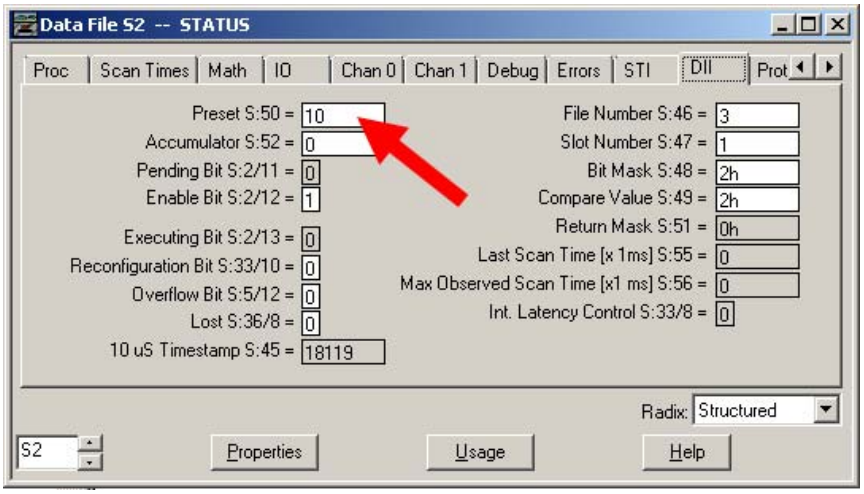
Convert Input Interrupts (DIIs/PIIs)

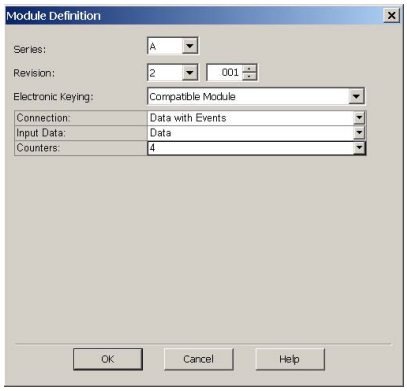
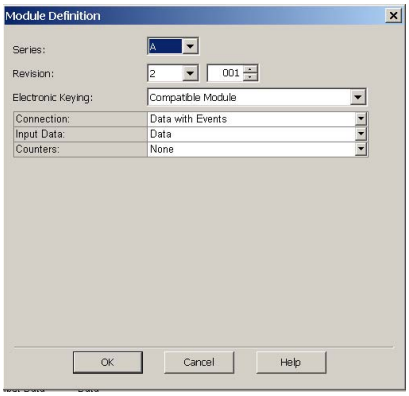
Overview

The CompactLogix 5380 and ControlLogix 5580 controllers support input interrupt via the CompactLogix 5069-IB16F and 5069-IB6F-3W high-speed input modules. You need to perform additional configuration from their default settings.

Follow these steps to set up the interrupt for your routine:

1. To use the input interrupt feature of the CompactLogix 5380 controller, you must use the 5069-IB16F or IB6F-3W (3-wire) module. Create either one of the modules in the local rack. You can also use a remote rack if you use a 5069-AENTR/AEN2TR module to mount the IB16F remotely.
2. If your application triggers an interrupt via a counter, you must configure your module with a counter. Check the 'Preset S:50' value of the SLC DII configuration.



If Preset S:50 Is Greater Than 1, Configure Your 5069-IB16F/IB6F-3W with Counters.	If Preset S:50 Is Equal to 0 or 1, Configure Your 5069-IB16F/IB6F-3W without Counters.
	

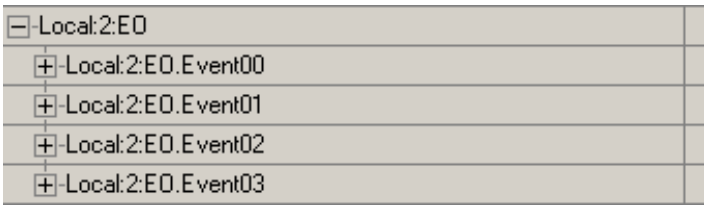
When the module is created, the following tags become available in the controller tag list. The data in these tags is created according to your configuration of the Module Definition.

Interrupt Trigger by Input

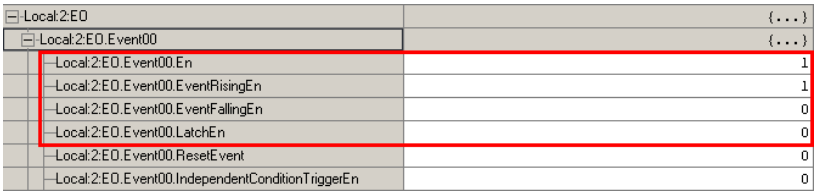
If the SLC configuration in Preset S:50 is lesser or equal to 1, then the SLC interrupt routine is triggered by inputs. The inputs are configured in Bit Mask S:48 and Compare Value S:49.

The following steps show you how to configure the 5069-IB16F/IB6F-3W module as an input interrupt for your routine. Here you configure an event to trigger an interrupt.

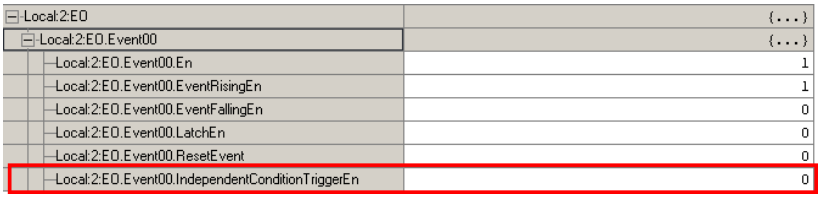
1. Select the event that you want to participate for the interrupt.



2. Expand the event and set Eventxx.En bit to 1.
3. Configure how you want the event to trigger the interrupt.



4. Configure the condition for an event to occur.
   - Set to **0** to indicate that all condition has to match the participating input for an event to occur.
   - Set to **1** to indicate that at least one of the participating inputs must match the configuration for an event to occur.

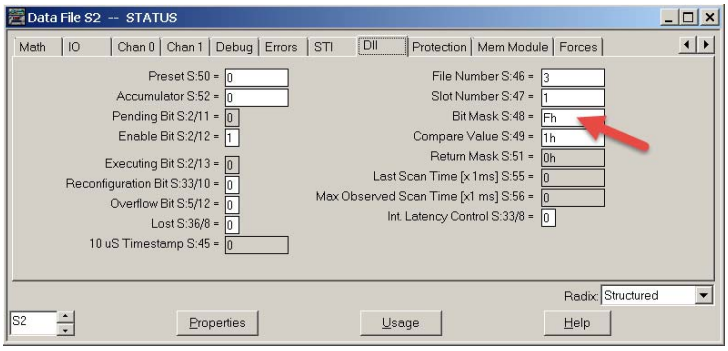


IMPORTANT The SLC 500 controller does not have this configuration. If you are using an SLC 500 controller, set this to '0' to mimic the original behavior.

5. Select the inputs that you want to participate for the event.

Local:1:EO.Event00.Pt00DataSelect	1
Local:1:EO.Event00.Pt01DataSelect	1
Local:1:EO.Event00.Pt02DataSelect	1
Local:1:EO.Event00.Pt03DataSelect	1
Local:1:EO.Event00.Pt04DataSelect	1
Local:1:EO.Event00.Pt05DataSelect	1
Local:1:EO.Event00.Pt06DataSelect	1
Local:1:EO.Event00.Pt07DataSelect	1
Local:1:EO.Event00.Pt08DataSelect	1
Local:1:EO.Event00.Pt09DataSelect	1
Local:1:EO.Event00.Pt10DataSelect	1
Local:1:EO.Event00.Pt11DataSelect	1
Local:1:EO.Event00.Pt12DataSelect	1
Local:1:EO.Event00.Pt13DataSelect	1
Local:1:EO.Event00.Pt14DataSelect	1
Local:1:EO.Event00.Pt15DataSelect	1

IMPORTANT During this step, enter the Boolean representation of the value in the 'Bit Mask S:48' text box of the SLC 500 controller.

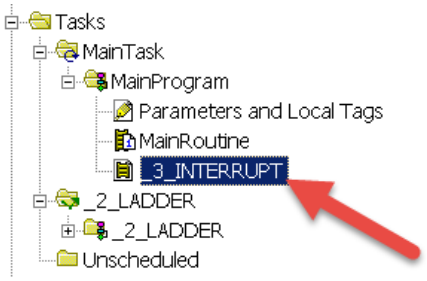


6. Configure the state in which the selected input must match to trigger the event.

Local:1:EO.Event00.Pt00DataValue	1
Local:1:EO.Event00.Pt01DataValue	0
Local:1:EO.Event00.Pt02DataValue	0
Local:1:EO.Event00.Pt03DataValue	0
Local:1:EO.Event00.Pt04DataValue	0
Local:1:EO.Event00.Pt05DataValue	0
Local:1:EO.Event00.Pt06DataValue	0
Local:1:EO.Event00.Pt07DataValue	0
Local:1:EO.Event00.Pt08DataValue	0
Local:1:EO.Event00.Pt09DataValue	0
Local:1:EO.Event00.Pt10DataValue	0
Local:1:EO.Event00.Pt11DataValue	0
Local:1:EO.Event00.Pt12DataValue	0
Local:1:EO.Event00.Pt13DataValue	0
Local:1:EO.Event00.Pt14DataValue	0
Local:1:EO.Event00.Pt15DataValue	0

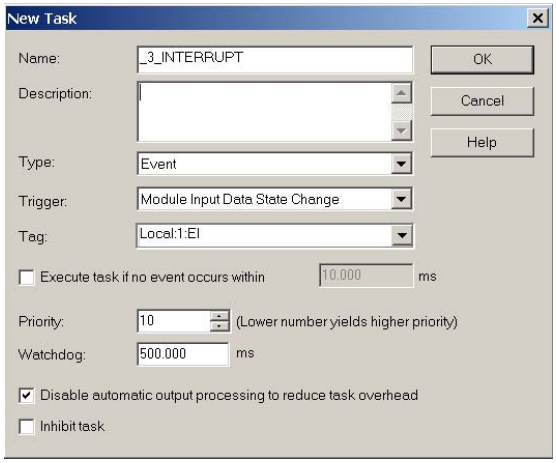
IMPORTANT During this step, enter the Boolean representation of the value in the 'Compare Value S:49' text box of the SLC 500 controller.

7. Create a new task for the interrupt routine.
   a. File Number S:46 indicates the program number of the interrupt routine in your SLC program. Identify the interrupt routine in your main task. Look for a routine name that carries the program number.

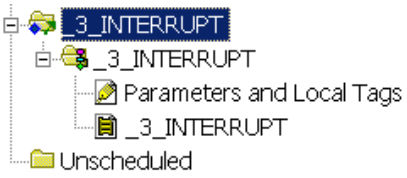


b. Create a new task with the following settings and click OK.

- From the Name text box, enter the name for the task.
- From the Description text box, enter a short description for the task.
- From the Type pull-down menu, select 'Event'.
- From the Trigger pull-down menu, select 'Module Input Data State Change'.
- From the Tag pull-down menu, select the 'EI' tag of the input module.



c. Drag and drop the interrupt routine into the newly created task and delete the PCE rung.



IMPORTANT You need to rewire the interrupt input device to the 5069-IB16F module in the same order as they are connected on the remote SLC I/O rack.

If these devices are also use in other locations within the program, you need to migrate them all and replace the SLC tag with new 5069-IB16F tags.

8. Test your application. Confirm that your program work as expected.

Interrupt Trigger by Counter

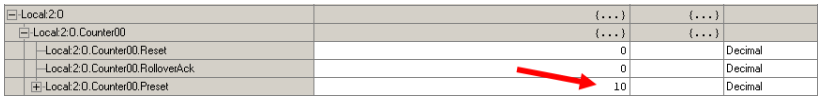
SLC 500 controller has a Direct Interrupt Input (DII) with built-in counter feature that is not available in CompactLogix 5380 controller. However, the new controller platform is able to perform similar task via Event Task with

high-speed input module. This section covers how to duplicate the DII with a counter task in a CompactLogix 5380 controller.

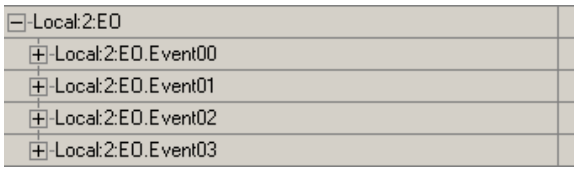
If the SLC configuration in Preset S:50 is greater than 1, it means that your SLC interrupt routine is triggered when the accumulator of the counter reaches the Preset value (S:50). The incremental condition is configured in Bit Mask S:48 and Compare Value S:49.

The following procedure shows how to configure the Compact 5000 I/O 5069-IB16F/IB6F-3W module as an input interrupt for your routine. Here you configure an event to trigger an interrupt.

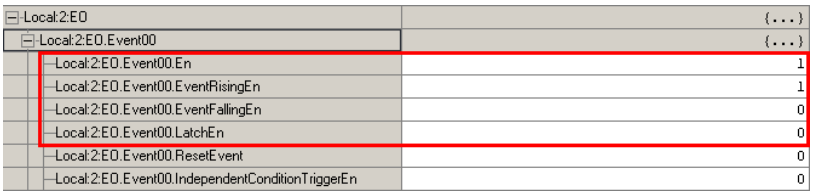
1. Configure the S:50 counter preset value in Local:x:O.Counterxx.Preset



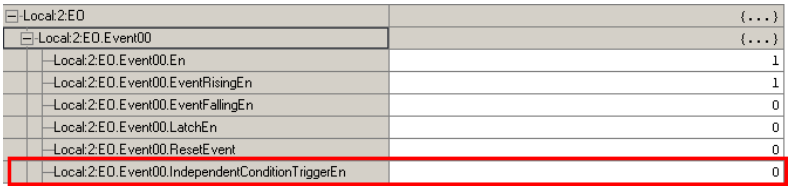
2. Select the event that you want to participate for the interrupt.
3. Expand the event and set Eventxx.En bit to 1.



4. Configure how you want the event to trigger the interrupt.



5. Configure the condition for an event to occur.
   - Set to **0** to indicate that all conditions have to match the participating inputs for an event to occur.
   - Set to **1** to indicate that at least one of the participating inputs must match the configuration for an event to occur.



The SLC 500 controller does not have this configuration. If you are using an SLC 500 controller, set this to '0' to mimic the original behavior.

6. Select the counters and inputs that will participate in the event.

Local:2:EO	{...}
Local:2:EO.Event00	{...}
Local:2:EO.Event00.En	1
Local:2:EO.Event00.EventRisingEn	1
Local:2:EO.Event00.EventFallingEn	0
Local:2:EO.Event00.LatchEn	0
Local:2:EO.Event00.ResetEvent	0
Local:2:EO.Event00.IndependentConditionTriggerEn	0
Local:2:EO.Event00.EventNumberAck	0
Local:2:EO.Event00.Counter00Select	0
Local:2:EO.Event00.Counter01Select	1
Local:2:EO.Event00.Counter02Select	0
Local:2:EO.Event00.Counter03Select	0
Local:2:EO.Event00.Pt04DataSelect	0
Local:2:EO.Event00.Pt05DataSelect	0
Local:2:EO.Event00.Pt06DataSelect	0
Local:2:EO.Event00.Pt07DataSelect	0
Local:2:EO.Event00.Pt08DataSelect	0
Local:2:EO.Event00.Pt09DataSelect	0
Local:2:EO.Event00.Pt10DataSelect	0
Local:2:EO.Event00.Pt11DataSelect	0
Local:2:EO.Event00.Pt12DataSelect	0
Local:2:EO.Event00.Pt13DataSelect	0
Local:2:EO.Event00.Pt14DataSelect	0
Local:2:EO.Event00.Pt15DataSelect	0

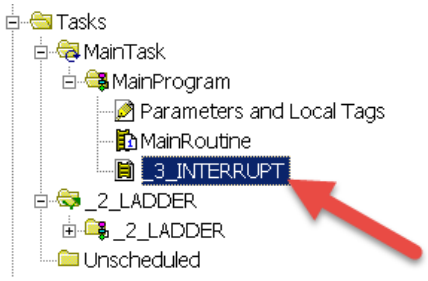
IMPORTANT During this step, enter the Boolean representation of the value in the 'Bit Mask S:48' text box of the SLC 500 controller. The above configuration indicates that counter 1, which corresponds to input terminal 1, is participating in the interrupt event.

7. Configure the state in which the selected counter/input must match to trigger the event.

Local:2:EO.Event00.Pt15DataSelect	0
Local:2:EO.Event00.Counter00Value	0
Local:2:EO.Event00.Counter01Value	1
Local:2:EO.Event00.Counter02Value	0
Local:2:EO.Event00.Counter03Value	0
Local:2:EO.Event00.Pt04DataValue	0
Local:2:EO.Event00.Pt05DataValue	0
Local:2:EO.Event00.Pt06DataValue	0
Local:2:EO.Event00.Pt07DataValue	0
Local:2:EO.Event00.Pt08DataValue	0
Local:2:EO.Event00.Pt09DataValue	0
Local:2:EO.Event00.Pt10DataValue	0
Local:2:EO.Event00.Pt11DataValue	0
Local:2:EO.Event00.Pt12DataValue	0
Local:2:EO.Event00.Pt13DataValue	0
Local:2:EO.Event00.Pt14DataValue	0
Local:2:EO.Event00.Pt15DataValue	0

A value '1' for counterxxValue denotes that the respective counter's done bit must be TRUE to trigger an event.

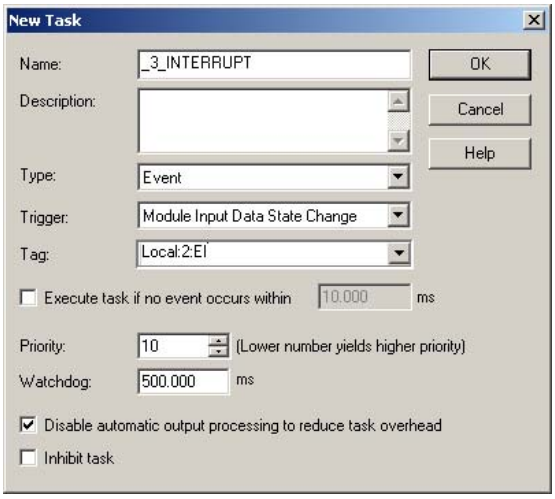
8. Create a new task for the interrupt routine.
   a. File Number S:46 indicates the program number of the interrupt routine in your SLC program. Identify the interrupt routine in your main task. There should be a routine name that carries the program number.



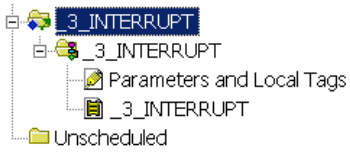
In the previous example, program 3 is the interrupt routine in the SLC project.

b. Create a new task with the following settings and click OK.

- From the Name text box, enter the name for the task.
- From the Description text box, enter a short description for the task.
- From the Type pull-down menu, select 'Event'.
- From the Trigger pull-down menu, select 'Module Input Data State Change'.
- From the Tag pull-down menu, select the 'EI' tag of the input module.



c. Drag and drop the interrupt routine into the newly created task and delete the PCE rung.



Here are some differences between the SLC 500 DII and CompactLogix 5380 interrupt:

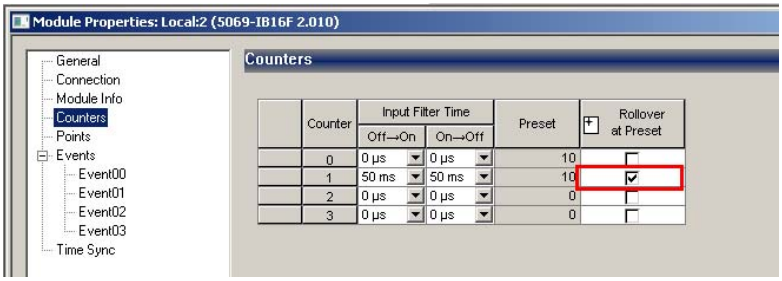
- You are not able to write to the count value (Local:X:I.CounterXX.Count) of CompactLogix 5380 controller while you can write to the DII's accumulator value (S:52) of SLC 500 controller.
- In a CompactLogix 5380 controller, you can configure the counter to roll over at preset. This allows the count value to be reset automatically to '0' when it reaches the preset value.
- A separate reset tag (Local:X:O.CounterXX.Reset) is available to reset the count to '0' upon setting it to TRUE.
- There is no way to reset the count value to a specific value. To do this, we must use another solution or module.

You have to rewire the interrupt input device to the 5069-IB16F module in the same order they are connected in the remote SLC rack.

If these devices are also use in other locations within the program, you need to migrate them all and replace the SLC tag with new 5069-IB16F tags.

9. Based on the behavioral differences and your interrupt task, you may need to change your program. For example:

a. If the interrupt task is to reset the accumulator value (S:52) to 0 and resume counting, you can simply enable the 'Rollover at Preset' in the 5069-IB16F module configuration.



b. If your original program uses the S word to configure the DII, you must replace the S word with corresponding tag.

SLC 500 DII S Word	CompactLogix 5380 Corresponding Tag Name
Bit Mask S:48/0 to /15	Local:X:E0.EventXX.Counter00Select to Local:X:E0.EventXX.Pt15DataSelect
Compare value S:49/0 to /15	Local:X:E0.EventXX.Counter00value to Local:X:E0.EventXX.Pt15value
Preset S.50	Local:X:0.CounterXX.Preset
Accumulator S:52	Local:X:I.CounterXX.Count

10. Test your application. Confirm that your program works as expected.

Create a Status File

Within the continuous task, the RSLogix Project Migrator automatically creates a subroutine that is named StatusFile. This StatusFile contains GSV instructions to retrieve the following controller information.

- The controller local date and time in human readable format
- Fault information about the controller provided by the FAULTLOG object
- Status for the Battery, bad or missing
- The physical hardware of the controller identified by the CONTROLLERDEVICE object
- Status for Mode switch in REMOTE
- Status for Forces enabled and present

There are special considerations for some data in the status file as shown in the following table.

This Status Data:	Is Handled This Way:
MCP status data	The PLC-5 processor can support from 1...16 main control programs. Each MCP uses 3 words of status data. Status words 80...127 contain this information.
STI status data	The Enhanced PLC-5 processor can also support a selectable timed interrupt. The processor status file contains the interrupt time interval and the number of the program file to execute. Status word 31 contains the program file number; status word 30 contains the interrupt time interval.
DII/PII status data	The PLC-5 and SLC 500 processors support an input interrupt. Status word 46 contains the number of the program file to execute. A Logix controller does not support this feature. If the import/export file contains PII status data, the PII program file is converted and placed as a routine in the Continuous program. The conversion process also places a PCE instruction in the converted routine to identify that the routine was used for a PII.
Indexed addressing	Status word 24 contains the current address index that is used for indexed addressing. A Logix controller does not use this index value. During the conversion, the process creates a tag for 524: 524 INT (Radix:=Decimal) := <value>

Program Conversion Errors (PCE) Messages

Introduction

Table 1 lists all of the messages that are generated with a PCE instruction. The text is appended to the rung comments that have the PCE instruction. The message text begins with asterisks (*) and the words “Generated by RSLogix Project Migrator”, and ends with asterisks.

Table 1 lists the message identifiers, descriptions, and when they are logged.

Table 1 - PCE Messages

ID	Text	When logged
101	The address references a counter’s Update Accum (UA) bit field. This is not supported in the Logix Designer application.	Each time a reference to a counter’s UA field is encountered (SLC only)
102	The address references a counter’s Overflow(OV) or Underflow(UN) field. This has been translated but the translation needs to be validated.	Each time a reference to a counter’s OV or UN field is encountered
103	**Warning**: Status files do not exist in the Logix Designer application. GSV instructions are used in the Logix Designer application to obtain controller information where applicable. This translation must be validated.	Each time a reference to the S file is encountered
105	The address references an indirect file number. It was not translated.	Each time an address reference with an indirect file number is encountered
107	The address reference may have an incorrect index. The translation needs to be validated.	Each time suitable index into the array could not be determined
108	The BTR, BTW or MSG instruction has been translated. However, the translation needs to be validated. These instructions have many parameters that cannot be directly translated and require review.	Each time a BTR, BTW or MSG instruction is translated
109	PLC-5 and SLC s use 0.01 second and 1 second timebases. the Logix Designer application uses a 0.001 second time base. The address references a ’s Accumulator (ACC) field. The translation needs to be validated.	Each time a reference to a ’s ACC field was encountered
110	PLC-5 and SLC s use 0.01 second and 1 second timebases. the Logix Designer application uses a 0.001 second time base. The address references a ’s Preset (PRE) field. The translation needs to be validated.	Each time a reference to a ’s PRE field was encountered
113	Follow the <FBC or DDT> instruction with MOV and FAL instruction on parallel branches to make sure the correct bits are being operated on.	Each FBC and DDT instruction
114	Although the PID instruction has been translated, the PID instruction has many parameters that do not translate directly to the Logix Designer application. The translation must be verified.	Each time a PID instruction is translated
115	16-bit parameters have been extended to 32-bit. Verify bit manipulation is correct.	Each time BSL, BSR, BTD instruction is translated
116	The structure of FOR/NXT/BRK statements has changed in the Logix architecture. In the PLC-5 processor, the FOR and NXT instruction enclosed a section of code that was to be iterated multiple times, while the BRK instruction provided a way to break out of the repeating code. In the RSLogix architecture, the FOR instruction calls a given routine a specific number of times, so a NXT instruction is not needed. The BRK instruction works in a similar fashion as in the PLC-5 processor. Because this architecture change is significant, you may need to restructure your logic.	Each time FOR/NXT/BRK instructions are encountered
117	AGA instruction not supported.	Each time a AGA instruction is found
119	CIR/COR not supported.	Each time a CIR or CIO instruction is found
120	Source and destination types differ	When source and destination types differ in a COP instruction
121	DFA instruction not supported	Each time a DFA instruction is found
122	ERI/ERO instruction not supported.	Each time a ERI or ERO instruction is found
123	IDI/IDO instruction not supported.	Each time a IDI or IDO instruction is found

Table 1 - PCE Messages (Continued)

ID	Text	When logged
124	IIN/IOT instruction not supported.	Each time a IIN or IOT instruction is found
128	SFC routines aren't translated.	Each time a SFR or EOT instruction is found
129	Online edit instructions are not supported.	Each time a SDS, SIZ or SRZ instruction is found
130	User Interrupt instructions not supported.	Each time a UID, UIE or UIF instruction is found
131	DDV instruction not supported.	Each time a DDV instruction is found
132	High Speed Counter instructions not supported.	Each time a HSC/HSD/HSE/ SL or RHC/RAC/TDF instruction is found
133	I/O Interrupt Enable/Disable instructions not supported.	Each time a IID or IIE instruction is found
134	IIM/IOM instruction not supported.	Each time a IIM or IOM instruction is found
135	INT instruction not supported.	Each time a INT instruction is found
136	REF instruction not supported.	Each time a REF instruction (in SLC) is found
137	RPI instruction not supported.	Each time a RPI instruction is found
138	Selectable Timed Interrupt instructions not supported.	Each time a STD/STE or STS instruction is found
139	SUS instruction not supported.	Each time a SUS instruction is found
141	RMP instruction not supported.	Each time a RMP instruction is found
142	RPC instruction not supported.	Each time a RPC instruction is found
143	SVC instruction not supported.	Each time a SVC instruction is found
144	SWP instruction not supported.	Each time a SWP instruction is found
145	SQC instruction not supported.	Each time a SQC instruction is found
146	INV instruction not supported.	Each time a INV instruction is found
147	DCD/ENC instruction not supported.	Each time a DCD or ENC instruction is found
148	The CEM, DEM, or EEM instruction has been translated. However, the translation needs to be validated. These instructions have many parameters that cannot be directly translated and require review.	Each time a CEM, DEM or EEM instruction is found
149	Modbus messaging is not supported in the Logix Designer application.	If MSG instruction is configured for Modbus
150	MSG instruction and associated MESSAGE tag need to be manually verified.	Each time a MSG instruction is found
151	**WARNING**: Status files do not exist in the Logix Designer application. However this status file value is handled through the StatusFile routine.	S file type indexes that can be directly translated to functionality in the Logix Designer application
152	the Logix Designer application has a different fault handling mechanism than the PLC-5/SLC. This fault routine will not be called.	Start of identified legacy processor fault routine
153	This PII/DII routine is not used by the Logix Designer application.	Start of identified legacy processor PII/DII routine

I/O Modules

Supported I/O Modules

The majority of SLC I/O discrete, analog, and specialty modules are compatible with the 1747-AENTR adapter.

Table 2 provides a list of supported modules.

Table 2 - I/O Modules Supported by the 1747-AENTR Adapter

Catalog Number	Type	Catalog Number	Type
1746-IA4	AC Digital Input Module	1746-OG16	DC Digital Output Module
1746-IA8	AC Digital Input Module	1746-OV8	DC Digital Output Module
1746-IA16	AC Digital Input Module	1746-OV16	DC Digital Output Module
1746-IB8	DC Digital Input Module	1746-OV32	DC Output Module
1746-IB16	DC Digital Input Module	1746-OVP16	DC Digital Output Module
1746-IB32	DC Input Module	1746-OW4	AC/DC Relay Output Module
1746-IC16	DC Digital Input Module	1746-OW8	AC/DC Relay Output Module
1746-IG16	DC Digital Input Module	1746-OW16	AC/DC Relay Output Module
1746-IH16	DC Digital Input Module	1746-OX8	AC/DC Relay Output Module
1746-IM4	AC Digital Input Module	1746-FIO4I	Analog Combination Module
1746-IM8	AC Digital Input Module	1746-FIO4V	Analog Combination Module
1746-IM16	AC Digital Input Module	1746-INT4	Thermocouple Isolated Input Module
1746-IN16	AC/DC Digital Input Module	1746-NI4	Analog Input Module
1746-IO4	Digital Combination Module	1746-NI8[1]	Analog Input Module
1746-IO8	Digital Combination Module	1746-NIO4I	Analog Combination Module
1746-IO12	Digital Combination Module	1746-NIO4V	Analog Combination Module
1746-IO12DC	Digital Combination Module	1746-NO4I	Analog Output Module
1746-ITB16	DC Digital Input Module	1746-NO4V	Analog Output Module
1746-ITV16	DC Digital Input Module	1746-NR4	RTD/Resistance Input Module
1746-IV8	DC Digital Input Module	1746-NT4	Thermocouple/mV Input Module
1746-IV16	DC Digital Input Module	1746-NI16V[1]	Analog Input Module
1746-IV32	DC Digital Input Module	1746-NI16I[1]	Analog Input Module
1746-OA8	AC Digital Output Module	1746-NR8[1]	RTD/Resistance Input Module
1746-OA16	AC Digital Output Module	1746-NT8	Thermocouple/mV Input Module
1746-OAP12	AC Digital Output Module	1746-NO8I[1]	Analog Output Module
1746-OB6EI	DC Digital Output Module	1746-NO8V[1]	Analog Output Module
1746-OB8	DC Digital Output Module	1746-HSTP1	Stepper Controller Module

Table 2 - I/O Modules Supported by the 1747-AENTR Adapter (Continued)

Catalog Number	Type	Catalog Number	Type
1746-OB16	DC Digital Output Module	1746-HSCE(2)	High-Speed Counter Module
1746-OB16E	DC Digital Output Module	1746-HSCE2(1)	Multi-Channel High-Speed Counter Module
1746-OB32	DC Digital Output Module	1746-QS(2)	Synchronized Axes Control Module
1746-OB32E	DC Digital Output Module	1746-OBP16	DC Digital Output Module
1746-OBP8	DC Digital Output Module	1746-BAS/B(1)(2)	BASIC Module
		1746-BAS-T(1)(2)	BASIC Module

(1) This module supports multiple operating classes. The 1747-AENTR adapter automatically promotes multiclass modules to the highest supported class. The 1746-NI16I, 1746-NI16V, 1746-NR8, 1746-NI8, 1746-NO8I, 1746-NO8V modules are promoted from class 1 to class 3. The 1746-BAS/B, 1746-BAS-T, and 1746-HSCE2 modules are promoted from class 1 to class 4.
(2) This specialty module uses Advanced module connection. Refer to the user manual specific to the I/O module to learn about its connection configuration requirements.

Unsupported I/O Modules

Table 3 lists unsupported modules. You can also have third-party SLC I/O modules in your system. Contact the third-party supplier to determine their product's compatibility. In general, third-party modules can be supported as long as the modules use fewer than 250 integer words and do not use G-files. However, a new EDS file needs to be developed to make the third-party module compatible with the 1747-AENTR adapter. If these modules are used in the current system, other plans must be made to incorporate the modules into a Logix system.

Consult the SLC I/O EtherNet/IP Adapter Module user manual, publication 1747-UM076 for the latest list of supported and unsupported modules.

Table 3 - I/O Modules NOT Supported by the 1747-AENTR Adapter

Catalog Number	Type
1746-HSRV	Servo Control Module
1746-BTM(1)	Barrel Temperature Module
1747-DCM1(1)	Direct Communication Modules
1747-DCM2(1)	
1747-DCM3(1)	
1747-DCM4(1)	
1747-KE/A(1)	DH-485/RS-232C Interface Modules
1747-KE/B(1)	
1747-KFC15	ControlNet to RS-232C Interface Module
1747-SDN/D(1)	DeviceNet Scanner module
1747-SCNR(1)	ControlNet Scanner module
1747-SN	Remote I/O Scanner Module
1747-BSN	Back-Up Remote I/O Scanner Module
1746-QV	Open Loop Velocity Control Module

Table 3 - I/O Modules NOT Supported by the 1747-AENTR Adapter (Continued)

Catalog Number	Type
1746-BLM	Blow Molding Module
1746-MPM	Mold Pressure Module
1203-SM1[1]	SCANport™ Module (Class 4 operation)

(1) The module is not supported by the 1747-AENTR but appears in the the Logix Designer application pick list (Select Module Type dialog box).

IMPORTANT Any SLC I/O module that is not included in the lists of supported and unsupported modules is **not** supported by the 1747-AENTR, as of the date of writing of this user manual.

Consult the manufacturer of any third-party module to determine if the module is supported by the 1747-AENTR adapter.

Notes:

Performance Expectations

Introduction

The data provided here was gathered by using real world testing and is provided as a guide to help you determine your expected results. The system consisted of a CompactLogix 5370 L3 controller and a rack of SLC I/O modules of different sizes and types. No two systems are alike and your system may be different than shown here. Use this data to help manage your performance expectations.

- Testing goal was to keep both the 1769-L36ERM % CPU and the 1747-AENTR % CPU under approximately 70%.
- Each user I/O module configuration is unique in terms of number and types of modules within the chassis. Test results vary based on exact chassis configuration.
- The data was gathered with all modules contained in a single SLC 500 chassis.
- RPIs of the various modules within the chassis do not need to be equal. You can adjust individual module RPIs to meet your application needs. The RPIs used in these tests were to extremely tax the system. Using RPIs of 2 and 4 ms is not considered typical.

Table 4 - Performance - For Reference Only

Modules in chassis, tested to max size chassis 13 slots **1 - 1747-AENTR module** **12 - I/O modules**	**1 - 1746-IB16** **1 - 1746-OB16** **(best case throughput)**	**1 - 1746-BAS** **2 - 1746-NT4** **2 - 1746-IB16** **2 - 1746-OB16**	**1 - 1746-BAS** **2 - 1746-NT4** **5 - 1746-IB16** **4 - 1746-OB16**
All modules configured for same RPI	2 ms	2 ms	4 ms
Total number of INPUT bytes transferred across backplane	12	212	244
Total number of OUTPUT bytes transferred across backplane	4	184	192
L36ERM % Ethernet CPU, I/O Comms Utilization (Actual)	20%	71.5%	60.8%
1747-AENTR % CPU	23%	68%	66%
1747-AENTR backplane scan time Max/Avg	2 ms/2 ms	5 ms/5 ms	5 ms/5 ms
Typical discrete throughput with minimal Logix program scan (less than 1 ms)	7...11 ms	11...19 ms	12...19 ms
Typical discrete throughput with 15 ms Logix program scan	7...39 ms	12...47 ms	12...50 ms
Typical discrete throughput with 30 ms Logix program scan	9...70 ms	12...76 ms	11...85 ms

While the Logix controller might not scan the I/O in the 1747-AENTR chassis as quickly as the SLC scans the I/O as local I/O, you may still see an overall performance improvement after the conversion to Logix because you are likely to see a significant program scan time decrease.

IMPORTANT Logix Designer application version 21 and later, and 1747-AENTR module firmware revision 2.001 and later, support configurations of multiple physical 1746 chassis connected by cables and scanned by a single 1747-AENTR module. Multi-chassis support lets you use up to 3 physical chassis, with a maximum of 30 I/O slots, scanned by a single 1747-AENTR module.

The 1747-AENTR firmware revisions prior to revision 2.001 support only a single physical chassis and 13 I/O slots maximum.

When the 1747-AENTR module is used in a multi-chassis system you can expect slower performance than the data presented in Table 4. The backplane scan time can increase by as much as 35% especially in systems with a large number of Input and Output bytes transferred across the backplane. The minimum RPI is also affected by the number of I/O modules within the chassis. For a 30-I/O-module chassis, the RPI must be at least 11 mS.

IMPORTANT Performance is affected by the Logix controller % CPU, 1747-AENTR % CPU, number of bytes In/Out transferred across the 1746 backplane, 1746 backplane scan time and Ethernet bandwidth. Your results may vary from the data in the table. The data in the table is meant for reference only.

Advanced Modules

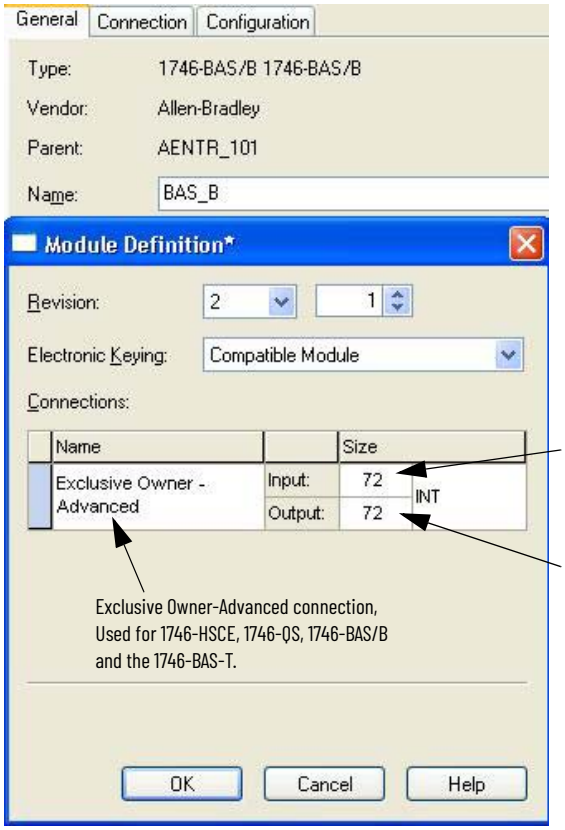
Use Advanced Modules in a Logix System

The SLC 500 EtherNet/IP Adapter user manual, publication 1747-UM076, has a section titled 'Add Specialty I/O Modules Using Advanced Connection' that covers using these modules. If your system has any of the advanced modules, you must review that section and plan accordingly.

Follow these steps to incorporate an advanced module in a Logix system.

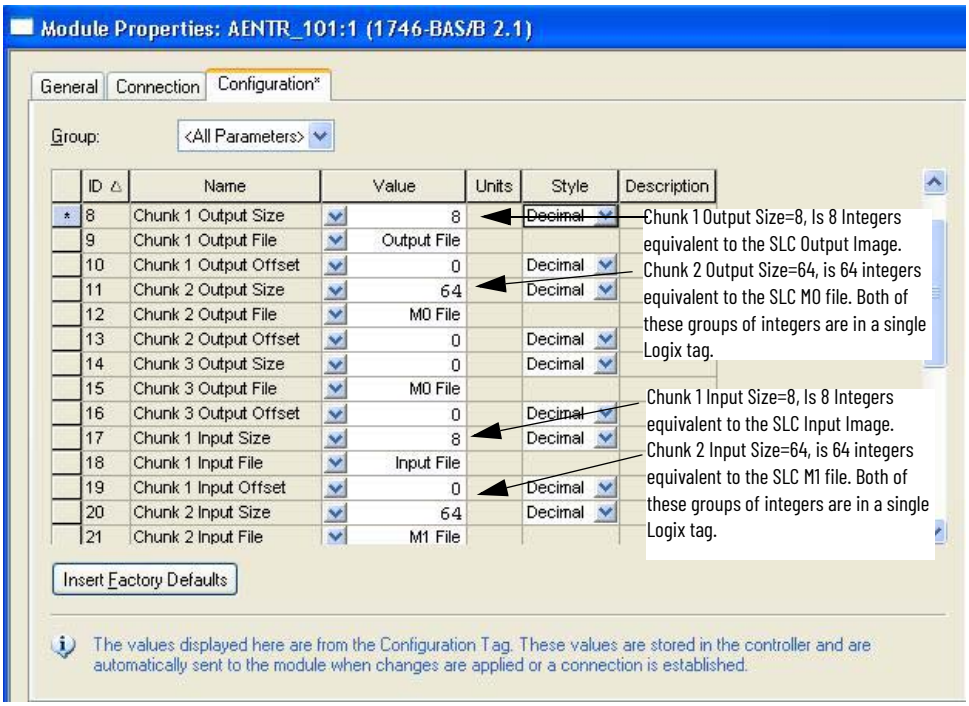
1. On the General tab, configure the total size and type of data (SINT, INT, DINT, REAL) that exists in the SLC environment for your advanced module.

   The values placed here determine the type and size of tag created in the Logix environment.



2. On the Configuration tab, you are directing the Logix controller where and how to place the data from the Advanced module into the Logix tag.

   Remember the advanced module is actually supplying Input data, Output data, M1 data and M0 data. This tab configures how to distribute the Advanced module data into the Logix tag.



This Configuration dialog box maps Input data, Output data, M0, and M1 data from the SLC environment and places that data in a single Logix Input and Output tag.

This is the Logix tag created with the Advanced module. The tag is a simple array (no descriptive tag names) of the size specified on the Module Definition dialog box from the Configuration tab. All of the data to/from the SLC I/O module show up in this tag based on the mapping specified on the Configuration tab.

AENTR_101:1:I	{...}		AB:1746_BASB_504289DA:I:0
AENTR_101:1:I.ConnectionFaulted	0	Decimal	BOOL
AENTR_101:1:I.Data	{...}	Decimal	INT[72]
AENTR_101:1:I.Data[0]	0	Decimal	INT
AENTR_101:1:I.Data[1]	0	Decimal	INT
AENTR_101:1:I.Data[2]	0	Decimal	INT
AENTR_101:1:I.Data[3]	0	Decimal	INT

A 72 Integer tag is created to hold the Input data and M1 data. This tag is a simple array with no descriptive tags.

AENTR_101:1:O	{...}		AB:1746_BASB_7F28A5EE:O:0
AENTR_101:1:O.Data	{...}	Decimal	INT[72]
AENTR_101:1:O.Data[0]	0	Decimal	INT
AENTR_101:1:O.Data[1]	0	Decimal	INT
AENTR_101:1:O.Data[2]	0	Decimal	INT
AENTR_101:1:O.Data[3]	0	Decimal	INT

A 72 Integer tag is created to hold the Output data and M0 data. This tag is a simple array with no descriptive tags.

Numerics

A

B

C

D

E

F

I

L

M

N

O

P

R

S

T

U

V

Rockwell Automation Support

Use these resources to access support information.

Technical Support Center	Find help with how-to videos, FAQs, chat, user forums, and product notification updates.	rok.auto/support
Knowledgebase	Access Knowledgebase articles.	rok.auto/knowledgebase
Local Technical Support Phone Numbers	Locate the telephone number for your country.	rok.auto/phonesupport
Literature Library	Find installation instructions, manuals, brochures, and technical data publications.	rok.auto/literature
Product Compatibility and Download Center (PCDC)	Download firmware, associated files (such as AOP, EDS, and DTM), and access product release notes.	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, Fax: (1) 414.382.4444
EUROPE/MIDDLE EAST/AFRICA: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640
ASIA PACIFIC: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846

Publication 5069-AP001A-EN-P - August 2020
Supersedes Publication 1769-AP001C-EN-P - March 2019