

Triconex Emulator User's Guide



Triconex Emulator
Version 1.0.0

9700115-001
April 2010

Information in this document is subject to change without notice. Companies, names and data used in examples herein are fictitious unless otherwise noted. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Invensys Systems, Inc.

© 2004–2010 by Invensys Systems, Inc. All rights reserved.

Triconex, Tricon, Trident, TriStation 1131, TriStation MSW, and CEMPLE are trademarks of Invensys plc, its subsidiaries and affiliates. All other brands may be trademarks of their respective owners.

Contents

Preface	v
Summary of Sections	v
Related Documents	v
Product and Training Information	v
Technical Support	vi
We Welcome Your Comments	vii
Chapter 1 Getting Started	1
About the Triconex Emulator	2
Triconex Emulator Features	4
Network Communication	4
TSAA Protocol	4
Sequence of Events (SOE) Data	4
Modbus TCP Master/Slave Communication	5
Peer-to-Peer Communication	6
Process Alarm Function Blocks	6
MP and I/O Status Function Blocks	6
Disabling Points	7
Download All and Download Changes Commands	7
Installing the Emulator	8
System Requirements	8
Updating an Existing Emulator Installation	9
Uninstalling the Emulator	9
Specifying Emulator Project Options	10
Accessing the Emulator	13
Chapter 2 Configuring the Emulator	15
Configuring the Emulator's Settings	16
Saving the Emulator Configuration to a File	20
Loading the Emulator Configuration from a File	21
Configuring Peer-to-Peer Communication for Testing	22
Step 1: Verify the Node Number Configuration	22
Step 2: Configure the Communication Module	24
Step 3: Set the Peer-to-Peer Port Numbers	26
Step 4: Set the Peer-to-Peer Send/Receive Properties	26
Step 5: Add Send/Receive Function Blocks to the Application	27
Step 6: Select the CM to Get the Peer-to-Peer Configuration From	27
Chapter 3 Testing Applications on the Emulator	29
About Testing Applications on the Emulator	30
Testing SOE Communication	32
Testing Peer-to-Peer Communication	34
Testing Modbus TCP Communication	35
Downloading to the Emulator	36
Download All and Download Changes Command Differences	36

About the Download Changes Command	36
Downloading an Application to the Emulator	38
Changes to the Project Version After Downloading	39
Monitoring Scan and Execution Times	41
Monitoring a Program's Execution.	42
Monitoring Variables	43
Testing the Impact of MP System Status Changes	44
MP Status Parameter Default Values	46
Emulator and Controller MP Status Parameter Differences	47
Testing the Impact of I/O Status Changes	48
Interpreting the I/O Module Address and Description	50
I/O Module Status Parameter Default Values	51
Adding Annotation for Variables	53
Disabling Points	54
Disabling Points on the Emulator	54
Viewing Disabled Points on the Emulator	55
Re-enabling Disabled Points on the Emulator	55

Preface

The Triconex™ Emulator is software that allows you to emulate, execute, and test TriStation 1131™ applications without connecting to a Tricon or Trident controller.

Summary of Sections

- Chapter 1, *Getting Started* – Describes Emulator functionality and features, and provides instructions for updating an existing Emulator installation.
- Chapter 2, *Configuring the Emulator* – Explains how to configure Emulator settings.
- Chapter 3, *Testing Applications on the Emulator* – Explains how to test applications on the Emulator, including downloading applications, monitoring variables, forcing system status changes, and disabling points.

Related Documents

- *TriStation 1131 Developer's Guide*
- *TriStation 1131 Libraries Reference*
- *SOE Recorder User's Guide*
- *DDE Server User's Guide*
- *Enhanced Diagnostic Monitor User's Guide*
- *Communication Guide for Tricon v9-v10 Systems*
- *Communication Guide for Trident Systems*
- *Product Release Notices* for specific Tricon, Trident, and TriStation versions

Product and Training Information

To obtain information about Invensys products and in-house and on-site training, see the Invensys Web site or contact your regional customer center.

Web Site

<http://www.iom.invensys.com>

Technical Support

Customers in the U.S. and Canada can obtain technical support from the Invensys Global Customer Support (GCS) center at the numbers below. International customers should contact their regional support center.

Requests for support are prioritized as follows:

- Emergency requests are given the highest priority
- Requests from participants in the System Watch Agreement (SWA) and customers with purchase order or charge card authorization are given next priority
- All other requests are handled on a time-available basis

If you require emergency or immediate response and are not an SWA participant, you may incur a charge. Please have a purchase order or credit card available for billing.

Telephone

Toll-free number 866-746-6477, or
Toll number 508-549-2424 (outside U.S.)

Fax

Toll number 508-549-4999

Web Site

<http://support.ips.invensys.com/> (registration required)

E-mail

iom.support@invensys.com

We Welcome Your Comments

To help us improve future versions of Triconex documentation, we want to know about any corrections, clarifications, or further information you would find useful. When you contact us, please include the following information:

- The title and version of the guide you are referring to
- A brief description of the content you are referring to (for example, step-by-step instructions that are incorrect, information that requires clarification or more details, missing information that you would find helpful)
- Your suggestions for correcting or improving the documentation
- The version of the Triconex hardware or software you are using
- Your name, company name, job title, phone number, and e-mail address

Send e-mail to us at:

`triconextechpubs@invensys.com`

Please keep in mind that this e-mail address is only for documentation feedback. If you have a technical problem or question, please contact the Invensys Global Customer Support (GCS) Center. See *Technical Support* on page vi for contact information.

Or, you can write to us at:

Attn: Technical Publications — Triconex
Invensys
15345 Barranca Parkway
Irvine, CA 92618
USA

Thank you for your feedback.

Getting Started

About the Triconex Emulator	2
Triconex Emulator Features	4
Installing the Emulator	8
Specifying Emulator Project Options	10
Accessing the Emulator	13

About the Triconex Emulator

The Triconex Emulator is software that allows you to emulate and execute TriStation 1131 applications without connecting to a Tricon or Trident controller. Using the Emulator, you can test your applications in an offline environment, without exposing your online processes to potential application errors.

The latest version of the Emulator is installed with TriStation 1131; however, because the Emulator is a separate application from TriStation 1131, you can update it at any time without having to re-install TriStation 1131.

You access the Emulator via the Emulator Control Panel in TriStation 1131. Using the Emulator, you can emulate the following Triconex controller modules, communication modes, and TriStation 1131 application features:

- Tricon Communication Module (TCM)
- Trident Communication Module (CM)
- TSAA protocol communication, including support for DDE and SOE clients, and multicasting (DDE clients only)
- Sequence of Events (SOE) blocks, data generation, and response
- Modbus TCP Master and Slave communication
- Peer-to-Peer communication over UDP
- Process alarm function blocks
- System status function blocks for MP and I/O modules, including system state transitions
- Disabling of points
- Download Changes and Download All commands

You can also use third-party process simulation software with the Emulator (via external DDE servers, and/or Modbus Masters and Slaves) to provide a complete process control emulation.

Depending on the protocol being used, external servers communicating with the Emulator can be running on the same PC as the Emulator, or on separate PCs, as long as they are on the same network.

The following figure shows how the Emulator works with TriStation 1131 and other Triconex and third-party elements in a simple system simulation.

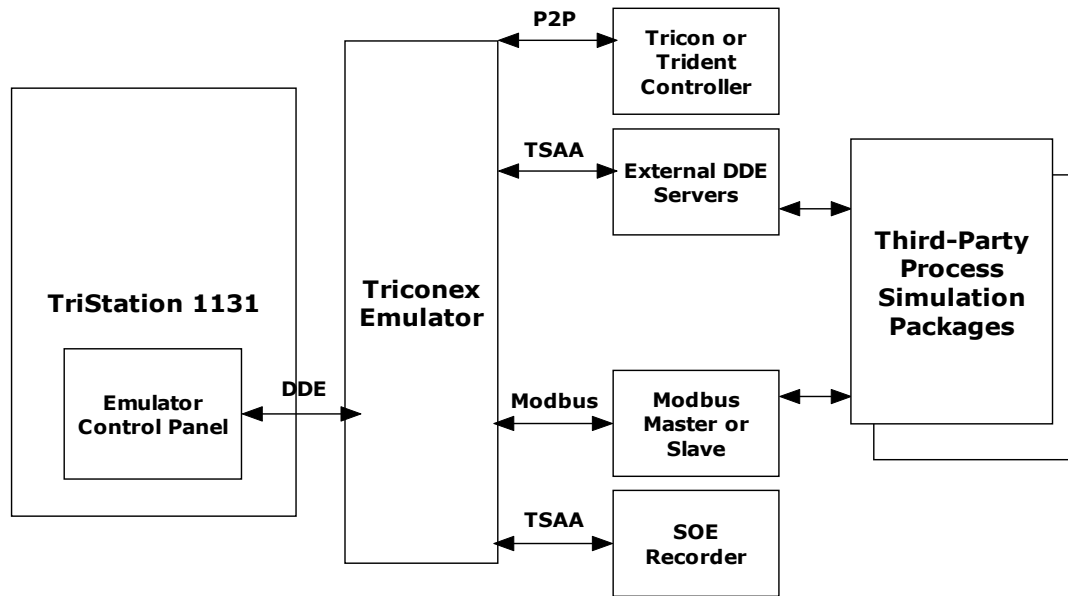


Figure 1 Overview of Triconex Emulator – Simple System Simulation

When simulating a safety or process control system that includes multiple Triconex controllers, as well as third-party DCS and control systems, multiple Emulator instances can be running at the same time, communicating with each other via the Peer-to-Peer (P2P) protocol. If you are using the DDE server built into the Emulator, the simulation application must be running on the same PC as the Emulator; if you are using an external DDE server (as shown above), the simulation application can run on a different PC.

The following figure shows how multiple Emulator instances can be used to simulate a more complex safety or process control system.

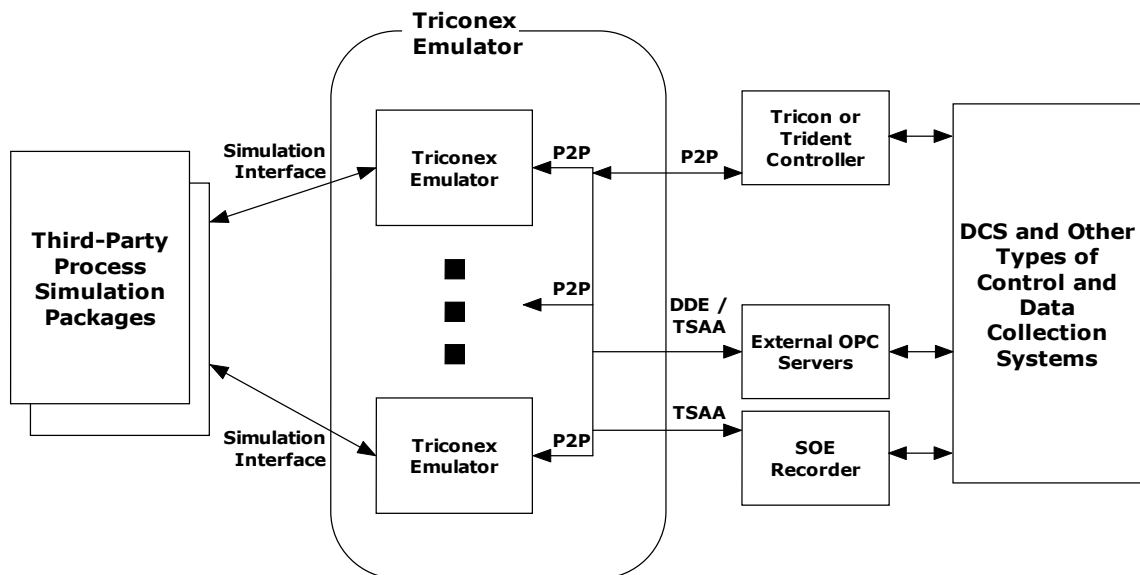


Figure 2 Overview of Triconex Emulator – Complex System Simulation

Triconex Emulator Features

The following sections describe Triconex Emulator features in more detail.

Network Communication

The Emulator supports redundant TSAA network communication, so that if the IP address you identify as the primary server goes offline, the Emulator will automatically switch over to the IP address you identify as the redundant server.

You can emulate multiple communication protocols (such as Modbus TCP and Peer-to-Peer) simultaneously in one application running on the Emulator, as long as separate ports are configured for each protocol. See [Configuring the Emulator's Settings](#) on page 16 for configuration information.

TSAA Protocol

The Emulator supports the TSAA protocol, allowing you to simulate communication between a Triconex controller and clients using TSAA (such as DDE Server and SOE Recorder). Only the following TSAA protocol commands are supported for emulation:

- TRICON_DATA_REQ
- WRITE_TRICON_DATA
- READ_TRICON_DATA
- SET_TRICON_CLOCK
- TRICON_GET_SYMBOL_REQ
- READ_TRICON_CLOCK
- ADJUST_TRICON_CLOCK
- TRICON_CPSTATUS_REQ
- TRICON_SOE_REQ

TSAA broadcast messaging and IP multicasting are also supported. If a TRICON_DATA_REQ message is not received from a client within two minutes, the TSAA connection will timeout, and TSAA broadcast messages will no longer be sent to that client. The timeout value does not apply to TSAA multicasting; multicast messages are sent until the Emulator is disconnected.

Sequence of Events (SOE) Data

For Triconex controllers, an *event* is the state change (transition) of a Boolean variable from True to False or from False to True. A Triconex controller can collect events and record the date, time, state, and variable name of each event in sequence of events (SOE) blocks that reside in the controller's memory.

All four SOE function blocks are supported for emulation:

- SOESTRT (SOE Start)
- SOECLR (SOE Clear)
- SOESTOP (SOE Stop)
- SOESTAT (SOE Statistics)

Events can be retrieved from the Emulator by using the Triconex SOE Recorder software.

You can use the Emulator to test the SOE blocks configured in your application, verifying that event collection is enabled and ensuring the correct retrieval of events by your SOE client. If you have SOE blocks configured in your application, SOE functionality will be available in the Emulator automatically after you download your application. No additional configuration in the Emulator is necessary. See [Testing SOE Communication](#) on page 32 for more information.

Modbus TCP Master/Slave Communication

Modbus is an industry-standard master/slave communication protocol that is traditionally used for energy management, transfer line control, pipeline monitoring, and other industrial processes.

Tricon Functionality

When the Emulator is acting as a Tricon controller with a TCM, it can operate as a Modbus master, slave, or both. A DCS typically acts as the master, while the Tricon acts as a slave. The master can also be an operator workstation or other device that is programmed to support Modbus devices.

The Tricon controller has network ports on the TCM that provide options for communication with Modbus devices. Each port can operate in a point-to-point configuration with a single Modbus device, or in a multi-point configuration with several Modbus devices.

Trident Functionality

The Trident controller has network ports on the CM and MP that provide options for communication with a Modbus devices. When the Emulator is acting as a Trident controller, each CM and MP port can operate in a point-to-point configuration with a single Modbus device. In addition, each CM port can operate in a multi-point configuration with several Modbus devices.

Using Modbus TCP with the Emulator

If you will be using Modbus TCP communication in your application, you can test it via the Emulator. However, only one Modbus configuration can be used by the Emulator.

If redundant communication modules are installed in a single logical slot, prior to downloading your application to the controller, you need to select the physical module (LEFT or RIGHT) with the Modbus configuration you want to use. This module selection is made in the Emulator Project Options dialog box; see [Specifying Emulator Project Options](#) on page 10 for more information.

See [Testing Modbus TCP Communication](#) on page 35 for more information.

For more information about using the Modbus protocol, see the *TriStation 1131 Developer's Guide*, the *TriStation 1131 Libraries Reference*, and the *Communication Guide* for your controller.

Peer-to-Peer Communication

Peer-to-Peer communication allows Triconex controllers to send and receive information from each other. Peer-to-Peer communication is performed through Send and Receive function blocks included in the application.

If you will be using Peer-to-Peer communication in your application, you can test it via the Emulator. However, only one Peer-to-Peer configuration can be used by the Emulator.

If redundant communication modules are installed in a single logical slot, prior to downloading your application to the controller, you need to select the physical module (LEFT or RIGHT) with the Peer-to-Peer configuration you want to use. This module selection is made in the Emulator Project Options dialog box; see *Specifying Emulator Project Options* on page 10 for more information.

See *Testing Peer-to-Peer Communication* on page 34 for more information.

Process Alarm Function Blocks

Support for process alarms was introduced in TriStation 1131 v4.7.0 via the Alarms library (ALARMS.lt2). The Emulator supports the following TriStation 1131 process alarm function blocks for emulation:

- ALARM_DEVIATION_DINT
- ALARM_DEVIATION_REAL
- ALARM_LEVEL_DINT
- ALARM_LEVEL_REAL
- ALARM_RATEOFCHANGE_DINT
- ALARM_RATEOFCHANGE_REAL
- ALARM_TRIP
- BYPASS_DINT
- BYPASS_REAL
- BYPASS_BOOL

For detailed information about these function blocks, please see the *TriStation 1131 Libraries Reference*.

MP and I/O Status Function Blocks

Tricon and Trident status function blocks allow you to view state transitions and status for Main Processor (MP) and I/O modules. The Emulator supports the following MP system status function blocks for emulation:

- TR_MP_STATUS (Tricon)
- SYS_MP_STATUS (Trident)

For instructions on changing the value of the output parameters you want to emulate for each MP system status function block, see *Testing the Impact of MP System Status Changes* on page 44.

The Emulator also supports the following I/O status function blocks for emulation:

- TR_SLOT_STATUS (Tricon)
- SYS_IO_STATUS (Trident)
- SYS_AI32_STATUS (Trident)
- SYS_AO04_STATUS (Trident)
- SYS_DI16_AI16_STATUS (Trident)
- SYS_DI32_STATUS (Trident)
- SYS_DO16_STATUS (Trident)
- SYS_HRDI32_STATUS (Trident)
- SYS_PI06_STATUS (Trident)
- SYS_RO32_STATUS (Trident)
- SYS_SDO16_STATUS (Trident)

For instructions on changing the value of the output parameters you want to emulate for each I/O module, see [Testing the Impact of I/O Status Changes](#) on page 48.

Note For detailed information about the MP and I/O status function blocks, please see the *TriStation 1131 Libraries Reference*.

Disabling Points

When a point is disabled, inputs from field instruments do not change the value of the point, and the application running on the controller (or the Emulator) will not write to the point. Disabling points is typically used when field instruments need to be replaced or repaired.

An application running on a controller should not contain disabled points unless there is a specific reason for disabling them, such as initial testing or maintenance. Testing your application on the Emulator allows you to freely disable points so that you can assess the impact of disabling those points in an application before downloading and running it on the controller.

For instructions on disabling points, see [Disabling Points](#) on page 54.

Download All and Download Changes Commands

You can use both the Download All and Download Changes commands to download your application to the Emulator. This allows you to test the impact of changes made using these commands before downloading your application to the controller.

The Emulator also supports adding or updating TriStation 1131 libraries when the project is in the Download Changes state, as well as the ability to make partial changes (add, update, or delete) to an existing library. This allows you to make changes to selected elements in a library without having to add, update, or delete the entire library.

Note Because Download Changes behavior on the Emulator differs from that on the controller, do not use the Emulator to test the impact of making library changes.

See the *TriStation 1131 Developer's Guide* for more information about updating and changing libraries.

See [Downloading to the Emulator](#) on page 36 for instructions on using these commands with the Emulator.

Installing the Emulator

When you install TriStation 1131, the Emulator is automatically installed. The version of the Emulator installed with TriStation 1131 is the most up-to-date version of the Emulator that was available at the time each version of TriStation 1131 was released. For example, Triconex Emulator v1.0.0 is automatically installed with TriStation v4.7.0.

However, because the Emulator is a separate application, you can purchase and install an updated version at any time, without having to also update your TriStation 1131 installation (as long as the TriStation 1131 and Emulator versions are compatible).

Note Each installed version of TriStation 1131 can be associated with only one installation of the Emulator. If you have multiple versions of TriStation 1131 installed on your PC, you can install multiple versions of the Emulator as well, as long as the TriStation and Emulator versions are compatible.

The procedures in this section describe how to update an existing Emulator installation.

Topics include:

- System Requirements on page 8
- Updating an Existing Emulator Installation on page 9
- Uninstalling the Emulator on page 9

System Requirements

The following are the minimum system requirements for the Triconex Emulator:

- PC running Windows XP Professional, Windows Server 2003, or Windows Server 2008 (32-bit or 64-bit)
- 233 MHz processor with 128 MB RAM
- 125 MB free hard drive space
- CD-ROM drive
- Network adapter card
- SVGA monitor running at 1024 x 768 resolution with 64,000 colors

Updating an Existing Emulator Installation

This procedure explains how to update an existing Triconex Emulator installation. The setup program provided by Triconex installs all the components of the Emulator on your PC.

The Emulator must be installed on a local drive, on the same PC as TriStation 1131 4.7.0 or later. Do not install and run the Emulator on a network server.

Note If the Emulator setup cannot locate a compatible version of TriStation 1131 to associate with the Emulator, installation will be cancelled automatically. For example, TriStation 1131 v4.6 is not compatible with the Emulator v1.0.0; attempting to install the Emulator v1.0.0 on a PC where only TriStation 1131 v4.6 is installed will result in an error, causing the Emulator install program to automatically exit.

Procedure

- 1 Log on as an administrator or as a user with administrator privileges.
- 2 Close all open applications.
- 3 Insert the Triconex Emulator CD in the CD-ROM drive.

The installation should start automatically. If it does not, browse to locate the setup.exe file on the CD, and then double-click setup.exe to start the installation.
- 4 Follow the InstallShield Wizard instructions.
- 5 When prompted, select the version of TriStation 1131 you want to associate with this installation of the Emulator, and then click Next.

Only versions of TriStation 1131 (4.7.0 or later) currently installed on your PC can be selected. If you do not select a TriStation 1131 version, installation will be cancelled.
- 6 Click Install.

The Emulator must be installed in the default destination folder, which is: C:\Program Files\Triconex\TriStation 1131 <x.x.x>\Programs, where <x.x.x> is the version of TriStation 1131 you selected in step 5.
- 7 To complete the installation, click Finish.

You are not required to restart your PC before running the Emulator.

Uninstalling the Emulator

You cannot uninstall the Triconex Emulator separately from TriStation 1131. To uninstall the Emulator, you must also uninstall TriStation 1131. See the *TriStation 1131 Developer's Guide* for instructions on uninstalling TriStation 1131.

Specifying Emulator Project Options

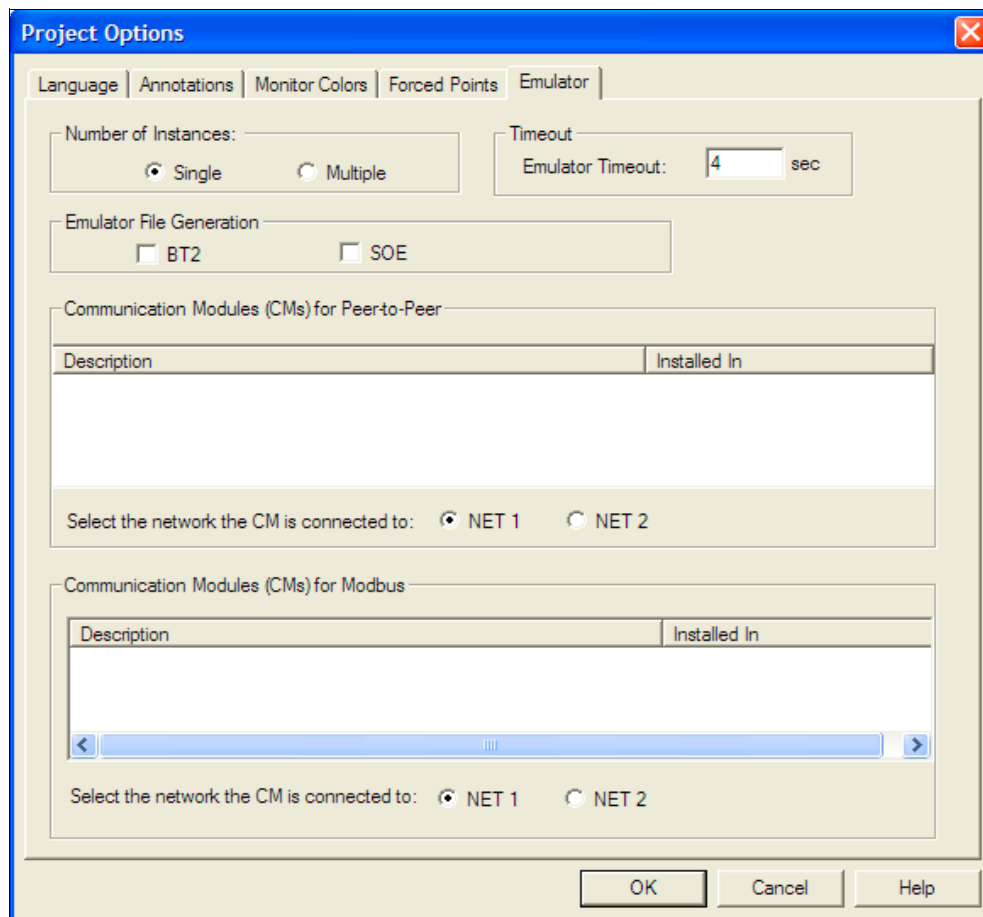
This procedure explains how to specify Emulator project options in TriStation 1131. You can change the default settings when you begin a new project or at any time during project development.

Emulator project options should be set prior to starting the Emulator. Options changed while the Emulator is running will not take effect until the Emulator is stopped and restarted.

Emulator project options are saved on a per-project basis, and do not need to be reconfigured each time you start the Emulator.

Procedure

- 1 Start TriStation 1131 v4.7.0 or later, and open the project with the programs you want to run on the Emulator.
- 2 On the Project menu, click Project Options, and then click the Emulator tab.



- 3 Specify these properties on the Emulator tab.

Property	Action
Number of Instances	Select how many instances of the Emulator you want to be able to run at any one time. If this option is set to Multiple, more than one instance of the Emulator can be running at the same time. If this option is set to Single, only one instance of the Emulator can be running at any one time. Single Emulator instances are terminated when TriStation disconnects from the Emulator, but multiple Emulator instances must be terminated manually. The default is Single.
Emulator Timeout	Select the timeout value for the connection to the Emulator. During a Download All or Download Changes operation on the Emulator, the processing time required for applications with a very large number of variables (symbols) may exceed the default Emulator timeout value of 4 seconds. If you receive a “download aborted” message when downloading an application to the Emulator, disconnect from the Emulator, increase the Emulator timeout value, reconnect to the Emulator, and then try downloading again. Note: In most cases, you should not have to change this value. The default timeout value should be adequate for the majority of applications. The valid range is from 0 to 65,535 seconds. The default is 4 seconds.
Emulator File Generation	Select the files you want generated each time you download the application to the Emulator. • Select the BT2 check box if you want a backup project file (.BT2) generated (not recommended). • Select the SOE check box if you want an SOE definition file (.SOE) generated. If you want to be able to test your SOE configuration with the Emulator, you must select this option. If selected, the .BT2 file and/or .SOE file are generated in the same directory where the TriStation project is located. The default setting is for the files to not be generated. Note: These files will look just like the .BT2 and .SOE files generated by TriStation when you download an application to the controller. You will not be able to distinguish between files generated by a download to the Emulator, and files generated by a download to the controller. You may want to make a note of the date and time each time you download the application to the Emulator, so you can compare that timestamp to the date/time of the generated files.

Property	Action
Communication Modules (CMs) for Peer-to-Peer	Optional. Required only if you want to test Peer-to-Peer communication using the Emulator. Select the communication module you want to get Peer-to-Peer configuration information from and then select the network that the communication module you selected is connected to. The default is NET 1. Note: If no CM modules are listed, you have not configured any communication modules for Peer-to-Peer communication in your project. The Peer-to-Peer configuration from the selected communication module will be used by the Emulator so that you can test your Peer-to-Peer functionality using emulated nodes. Only one Peer-to-Peer configuration can be used by the Emulator. If redundant communication modules are installed in a single logical slot, you must select the physical communication module (LEFT or RIGHT) with the configuration you want to use. The default is LEFT. See Testing Peer-to-Peer Communication on page 34 for more information about using Peer-to-Peer communication with the Emulator.
Communication Modules (CMs) for Modbus	Optional. Required only if you want to test Modbus communication using the Emulator. Select the communication module you want to get Modbus configuration information from and then select the network that the CM you selected is connected to. The default is NET 1. Note: If no CM modules are listed, you have not configured any communication modules for Modbus communication in your project. The Modbus configuration from the selected communication module will be used by the Emulator so that you can test your Modbus functionality using emulated nodes. Only one Modbus configuration can be used by the Emulator. If redundant communication modules are installed in a single logical slot, you must select the physical communication module (LEFT or RIGHT) with the configuration you want to use. The default is LEFT. See Testing Modbus TCP Communication on page 35 for more information about using Modbus communication with the Emulator.

-
- 4 Click OK to apply your changes and save the settings.

Accessing the Emulator

The Triconex Emulator can be accessed only from within TriStation 1131 v4.7.0 or later. It does not run as a separate application outside of TriStation 1131.

If the Emulator **Number of Instances** option is set to Multiple, more than one instance of the Emulator can be running at the same time. Otherwise, only one instance of the Emulator can be running at any one time. See *Specifying Emulator Project Options* on page 10 for more information.

This procedure explains how to start the Emulator from within TriStation 1131.

CAUTION

If you have Triconex DDE Server or SOE Recorder running on the same PC as the Emulator, they must be configured to use different UDP port numbers (1500 for Emulator, 1501 for DDE Server, and 1509 for SOE Recorder). Prior to connecting to the Emulator, verify that you have configured the UDP ports correctly.

Please note that the Emulator will not be able to communicate with a DDE Server or SOE Recorder application running on the same PC as the Emulator. To test DDE and SOE communication with the Emulator, the DDE and SOE clients must be running on a separate PC.

Procedure

- 1 Start TriStation 1131 v4.7.0 or later, and open the project with the programs you want to run on the Emulator.
- 2 Open the Controller workspace, expand the Configuration tree, and double-click Emulator Panel.
- 3 Click Connect . The Emulator configuration dialog box opens in the background.
- 4 Go to *Chapter 2, Configuring the Emulator* for instructions on configuring Emulator settings.

Configuring the Emulator

Configuring the Emulator's Settings	16
Saving the Emulator Configuration to a File	20
Loading the Emulator Configuration from a File	21
Configuring Peer-to-Peer Communication for Testing	22

Configuring the Emulator's Settings

This procedure explains how to configure the Emulator's settings prior to downloading and testing your application. Each application you test on the Emulator will have its own set of configuration settings.

For instructions on connecting to the Emulator, see [Accessing the Emulator](#) on page 13.

CAUTION

If you have Triconex DDE Server or SOE Recorder running on the same PC as the Emulator, they must be configured to use different UDP port numbers (1500 for Emulator, 1501 for DDE Server, and 1509 for SOE Recorder). Prior to connecting to the Emulator, verify that you have configured the UDP ports correctly.

Please note that the Emulator will not be able to communicate with a DDE Server or SOE Recorder application running on the same PC as the Emulator. To test DDE and SOE communication with the Emulator, the DDE and SOE clients must be running on a separate PC.

Procedure

- 1 In the Windows taskbar, click the TRxEMCodeEmulator button.

The TRxEMCodeEmulator dialog box appears.

The dialog box name differs, depending on the controller type and the Emulator options settings:

- For Tricon applications, the dialog box is named TR1EMCodeEmulator.
- For Trident applications, the dialog box is named TRDEMCodeEmulator.
- For both Tricon and Trident applications, if the multiple Emulator instances option is enabled (see *Specifying Emulator Project Options* on page 10), the dialog box is named the same as the node name defined on the TriStation Communication screen (the default is TRINODEx, where x is the node number configured in the application).

2 Do one of the following:

- If you have previously configured the Emulator for use with this application, and you saved the Emulator settings to a file, go to *Loading the Emulator Configuration from a File* on page 21.
- If you have not configured the Emulator for use with this application before, or you do not want to use saved configuration settings, go to the next step.

3 Specify the following properties.

Note Only the Node Number and Server IP Address properties are required. All other properties are optional. You do not need to specify properties that do not apply to your desired testing scenario. For example, if you will not be using TSAA functionality, you do not need to configure the TSAA properties.

Settings	Property	Action
Communication	Node Number	Enter the node number of the controller you are emulating, and then click Set Node. The Emulator supports Tricon nodes 1-31 and Trident nodes 1-63. This MUST be the same as the node number you entered on the TriStation Communication screen (see the <i>TriStation 1131 Developer's Guide</i> for more information). The default is 1.
	Server IP Address	Enter the primary IP address you want to use for the Emulator, and then click Set IP. This MUST be the IP address of the PC the Emulator is running on. Do not enter the IP address of the actual controller node. An error message will appear if you enter an incorrect IP address. If you have more than one network card in the PC, or if your network card is multi-homed, you can use any of the IP addresses assigned to the PC. If you are running multiple instances of the Emulator on one PC, you must have multiple network cards installed, or your network card must be multi-homed (configured to use more than one IP address), so you can have unique IP addresses for each instance of the Emulator.

Settings	Property	Action
Peer-to-Peer (P2P)	P2P (UDP) Port Number	Enter the UDP port number for the Emulator on the Peer-to-Peer network, and then click Set Port. The port number must be unique. It cannot be the same as the TSAA Port Number, or as any other UDP port number being used on the PC. The port number must be the same as the port configured for this node in the Peer-to-Peer configuration on other controllers communicating with this node using Peer-to-Peer. The default is 1503.
TSAA	Support TSAA Protocol	Select this check box if you want the Emulator to act as TSAA server. When this check box is selected, any TSAA client can connect to the Emulator to send and receive TSAA messages. The default is selected (TSAA is enabled). For more information about using TSAA with the Emulator, see <i>TSAA Protocol</i> on page 4. For general information about using the TSAA protocol, see the <i>Communication Guide</i> for your controller.
	TSAA Port Number	Enter the UDP port to use for TSAA connections to the Emulator (such as those from the DDE Server or SOE Recorder), and then click Set Port. The default is 1500. The port number must be unique. It cannot be the same as the P2P Port Number, or as any other UDP port number being used on the PC. Can be the same as the TSAA Port Number configured in your application, but it is not required. However, the TSAA Port Number configured in your TSAA client must match the TSAA Port Number configured for the Emulator. Additionally, if two or more instances of the Emulator are running simultaneously on the same PC, each instance must have a unique port value. See the <i>TriStation 1131 Developer's Guide</i> for additional information about UDP base port numbers.
	Multicast Interval	Enter the update rate, in milliseconds, for TSAA IP multicasting, and then click Set Interval. Can be from 250 to 2,000. The default value is 250. Bin data messages will be sent to the multicast IP address at the specified update rate (for example, by default, bin data will be multicast every 1000ms).

Settings	Property	Action
	Multicast IP Address	Enter the IP address that TSAA multicasts should be sent to, and then click Configure. The valid range is 224.0.0.0 – 239.255.255.255. The default is 0.0.0.0 (IP multicasting is disabled). Bin data will be sent to the specified address, so that all members of the group will receive the data. Note: To use TSAA multicasting, your network hardware (such as routers) must also support multicasting. See your Network Administrator or IT department for more information.
	Redundant IP Address	Enter the backup IP address you want to use for the Emulator, and then click Set IP. The default is 0.0.0.0. If the IP address set in the Server IP Address field (the primary IP address) becomes unavailable, the Emulator will automatically switch to the redundant IP address for TSAA messaging. This MUST be an IP address available on the PC the Emulator is running on. You must have more than one network card in the PC in order to use a redundant IP address. Do not enter the IP address of the actual controller node. An error message will appear if you enter an incorrect IP address.

- 4 Once you have configured all the settings required for your testing scenario, you are ready to download the application to the Emulator. Do one of the following:
 - If you will NOT be testing Peer-to-Peer communication, go to [Chapter 3, Testing Applications on the Emulator](#).
 - If you will be testing Peer-to-Peer communication, go to [Configuring Peer-to-Peer Communication for Testing](#) on page 22 to verify that Peer-to-Peer communication has been configured correctly in your application *before* downloading the application to the Emulator.

The Emulator's settings are NOT saved automatically. Once you disconnect from the Emulator (or click End in the TRxEMCodeEmulator dialog box), all settings are cleared and reset to the default values the next time you connect to the Emulator. To save these settings to a file for future use, see [Saving the Emulator Configuration to a File](#) on page 20.

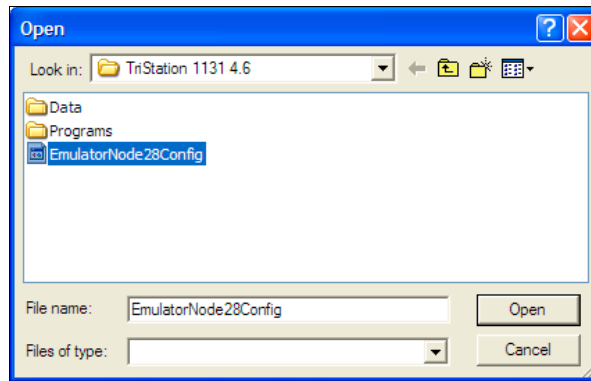
Saving the Emulator Configuration to a File

Once you have configured the Emulator's settings as necessary for testing your application, you should save the configuration to a file for future use. This ensures future testing uses the same settings, and saves you time in configuring the Emulator each time you start an instance of the Emulator for testing.

Note Changes you have made to the MP and I/O module status function block parameters will also be saved as part of the Emulator configuration file, as long as you have downloaded an application to the Emulator using those changes. See [Testing the Impact of MP System Status Changes](#) on page 44 and [Testing the Impact of I/O Status Changes](#) on page 48 for more information.

Procedure

- 1 In the TRxEMCodeEmulator dialog box, under Set Configuration File, click Browse. The Open dialog box appears.



- 2 Browse to the location where you want to save the Emulator configuration file.
- 3 In the File Name field, enter a name for the Emulator configuration file. You can use any naming convention that you like.
- 4 Click Open. The Open dialog box closes and you are returned to the TRxEMCodeEmulator dialog box.
- 5 Click Save Configuration. The configuration information you entered for the Emulator is saved to the file you created.

Loading the Emulator Configuration from a File

If you have previously configured and saved the Emulator's settings as necessary for testing your application, you can load the configuration from a file.

Procedure

- 1 In the TRxEMCodeEmulator dialog box, under Set Configuration File, click Load Configuration.
The Open dialog box appears.
- 2 Browse to the location where you previously saved the Emulator configuration file.
- 3 Select the configuration file and then click Open.
The Open dialog box closes and you are returned to the TRxEMCodeEmulator dialog box.
- 4 Click Load Configuration.
- 5 When asked to confirm that you want to load configuration information from the selected file, click OK.
The configuration information from the selected file is copied into the Emulator configuration dialog box.
- 6 Verify the Emulator settings loaded from the file are correct.
- 7 Once you have verified all the settings required for your testing scenario are correct, you are ready to download the application to the Emulator. Do one of the following:
 - If you will NOT be testing Peer-to-Peer communication, go to [Chapter 3, Testing Applications on the Emulator](#).
 - If you will be testing Peer-to-Peer communication, go to [Configuring Peer-to-Peer Communication for Testing](#) on page 22 to verify that Peer-to-Peer communication has been configured correctly in your application *before* downloading the application to the Emulator.

A Note About Loading MP and I/O Module Status Parameter Information

If you saved changes to the MP and I/O module status function block parameters in the configuration file, during the loading process you may see an error message stating that the MP and I/O module parameter information could not be loaded.

This error message appears because the MP and I/O module parameter information can be loaded only if an application has already been downloaded to the Emulator.

Once you have downloaded an application to the Emulator, use this same procedure to load the configuration file again, so that the MP and I/O module parameter information will be loaded into the application.

See [Testing the Impact of MP System Status Changes](#) on page 44 and [Testing the Impact of I/O Status Changes](#) on page 48 for more information.

Configuring Peer-to-Peer Communication for Testing

Peer-to-Peer communication allows Triconex controllers to send and receive information from each other. If you will be using Peer-to-Peer communication in your Tricon or Trident v2.x application, you can test it via the Emulator. The Emulator does not support Peer-to-Peer communication in Trident v1.x applications.

Note For detailed information about configuring and using Peer-to-Peer in an application, see the *TriStation 1131 Developer's Guide*.

The following six steps describe how to configure Peer-to-Peer communication in your application so it can be tested on the Emulator:

- Step 1: Verify the Node Number Configuration on page 22
- Step 2: Configure the Communication Module on page 24
- Step 3: Set the Peer-to-Peer Port Numbers on page 26
- Step 4: Set the Peer-to-Peer Send/Receive Properties on page 26
- Step 5: Add Send/Receive Function Blocks to the Application on page 27
- Step 6: Select the CM to Get the Peer-to-Peer Configuration From on page 27

You should complete all of these steps *before* downloading your application to the Emulator.

Step 1: Verify the Node Number Configuration

Verify that the node number configured for the Emulator is the same as the node number configured on the TriStation Communication screen.

Peer-to-Peer communication on the Emulator will not work if these node numbers do not match.

Procedure

- 1 In TriStation 1131, expand the Controller tree, and double-click Configuration.
- 2 On the Configuration tree, click TriStation Communication.

TriStation Communication

Select Connections
 TriStation can communicate with the controller over a network or serial port connection. A network connection requires an NCM or TCM module. A serial connection requires an EICM or TCM module. Which connections will you be using?

☒ Network Connection ☐ Serial Connection

Network Connection Setup

Node Number: IP Address:

Node Name:

TriStation UDP Port Number:

Node Number: The number specified by the switches on the communication module.
 IP Address: The internet protocol address of the module (e.g. 192.168.1.1).
 Node Name: Any alphanumeric name up to 20 characters.

TriStation PC

Serial Port: The port on the PC that will be connected to the controller.
 Baud Rate:

Default Connection
 When you connect to the controller, which connection would you like to use as the default?

☒ Network Connection ☐ Serial Connection

- 3 Locate the number entered in the Node Number field. For the TCM, this is the number represented by the Network Node Address setting on the MP front panel.
 You may want to write this number down, so you can compare it to the Emulator's node number in the next step.
- 4 Verify the node number entered on the Emulator configuration screen matches the node number entered on the TriStation Communication screen.

See [Configuring the Emulator's Settings](#) on page 16 for instructions on accessing the Emulator configuration screen.

Step 2: Configure the Communication Module

Verify that the communication module in your project is configured for Peer-to-Peer communication.

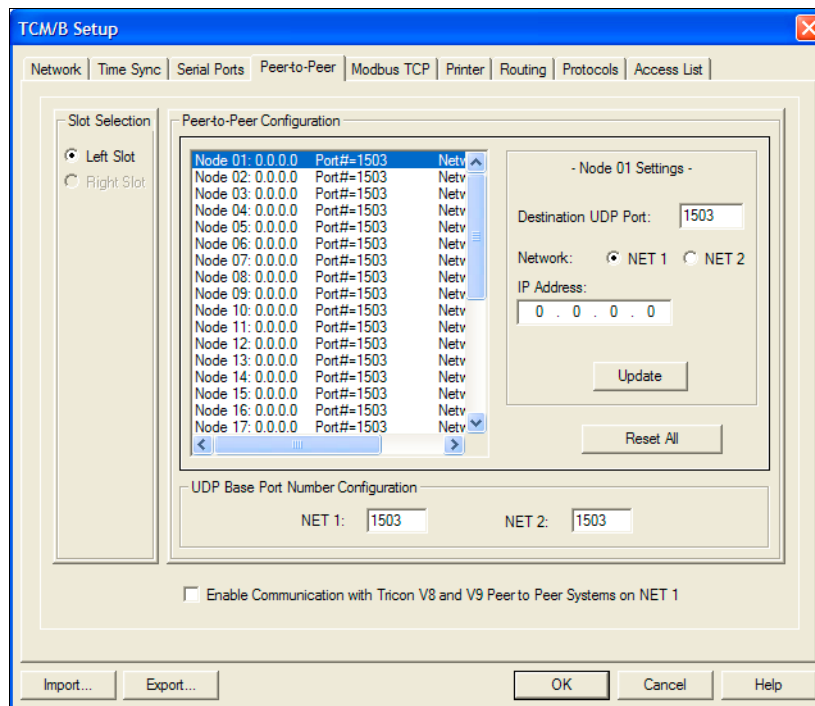
Note Peer-to-Peer communication with the Trident v1.x CM is not supported by the Emulator.

This procedure explains how to configure the IP addresses for controllers communicating on a Peer-to-Peer network through a Tricon TCM or a Trident v2.x CM.

Note Because NET 2 is reserved for OPC communication, the Peer-to-Peer protocol is not available on NET 2 for model 4353 and 4354 TCMs. Peer-to-Peer communication can be configured on NET 1 only for these TCMs.

Procedure

- 1 In TriStation 1131, expand the Controller tree, double-click Configuration, and expand Hardware Allocation.
- 2 Double-click the slot where the TCM or CM module is installed, and then click Setup. The TCM Setup (Tricon) or CM Setup (Trident) dialog box appears.
- 3 Click the Peer-to-Peer tab.



- 4 Under Slot Selection, select the module you want to configure Peer-to-Peer ports for.
- 5 Select a node (controller), and specify these properties.

Property	Action
Destination UDP Port	Enter the UDP port number for each controller to be communicated with on the Peer-to-Peer network. This must be the same number that the controller uses as its UDP Base Port Number. The default is 1503.
Network	Click the network port that the TCM or CM uses to communicate with the selected Peer-to-Peer node. The default is NET 1. NET 2 is not available for model 4353 and 4354 TCMs.
IP Address	Enter the IP address for the controller.

- 6 Click Update to save the new settings for the selected node.

If you forget to click Update before selecting another node, your settings will be lost.

- 7 Repeat steps 4 through 6 for each node to be included in the network.

Note If you need to start over for any reason, click Reset All to reset *all* node settings for the selected slot to their default, unconfigured state.

- 8 Once all nodes have been configured, specify these properties (applicable to all nodes on the Peer-to-Peer network).

Property	Action
UDP Base Port Number	For each network port, enter the UDP base port number for the controller. Enter 0 to disable Peer-to-Peer over UDP/IP on the network. The default is 1503 on both NET 1 and NET 2 (meaning Peer-to-Peer is enabled on both ports). UDP port numbers do not need to be unique, as long as the IP addresses for each port is unique. See the <i>TriStation 1131 Developer's Guide</i> for reserved values. Available only for NET 1 on model 4353 and 4354 TCMs.
Enable Communication with Tricon V8 and V9 Peer-to-Peer Systems on NET 1 (TCM Only)	Select the check box to enable communication with Tricon version 8 and 9 systems on the NET 1 port. The default is cleared. Available only for a module installed in the left slot.
Enable Communication with Tricon V8, Tricon V9, and Trident v1 Peer-to-Peer Systems (Trident 2.x CM Only)	Select the port—NET 1 or NET 2—on which you want to enable Peer-to-Peer communication with Tricon version 8 and 9 systems, and/or Trident version 1 systems. The default is None. Available only for a module installed in the left slot.

- 9 Click OK to save your changes.

Step 3: Set the Peer-to-Peer Port Numbers

Verify that the Peer-to-Peer (P2P) port number configured for the Emulator is the same as the UDP Base Port Number configured for the network (NET 1 or NET 2) the communication module in your project is connected to.

For example, if the TCM in your Tricon project is connected to NET 1, the Emulator Peer-to-Peer port should be the same as the UDP Base Port Number configured for NET 1.

See [Step 2: Configure the Communication Module](#) on page 24 for instructions on setting the UDP Base Port Number.

Note Be sure to select the same network for the Select CM to Download Peer-to-Peer Configuration From option prior to downloading your application (see [Specifying Emulator Project Options](#) on page 10). If you configure the Peer-to-Peer port for NET 1, but select NET 2 for the communication module, Peer-to-Peer will not work.

Step 4: Set the Peer-to-Peer Send/Receive Properties

Verify that the following properties on the Peer-to-Peer Configuration screen in the TriStation 1131 Implementation tree are set to a value other than zero (0):

- **Maximum Number of Peer-to-Peer Sends**

The Maximum Number of Peer-to-Peer Sends property specifies the maximum number of Send function blocks for each TriStation application. To conserve memory and minimize scan time, set this number as low as possible. However, if you plan to add Send function blocks in the future, specify the anticipated maximum to avoid having to perform a Download All later.

- **Maximum Number of Peer-to-Peer Receives**

The Maximum Number of Peer-to-Peer Receives property specifies the maximum number of Receive function blocks for each TriStation application. To conserve memory and minimize scan time, set this number as low as possible. However, if you plan to add Receive function blocks in the future, specify the anticipated maximum to avoid having to perform a Download All later.

CAUTION

The default value for these properties is zero (0). If Maximum Number of Peer-to-Peer Receives and Maximum Number of Peer-to-Peer Sends are both set to zero, Peer-to-Peer communication is NOT enabled in your application, and you will not see any Peer-to-Peer messages sent, even if Peer-to-Peer communication is configured on the TCM or CM.

Allocating Peer-to-Peer Memory

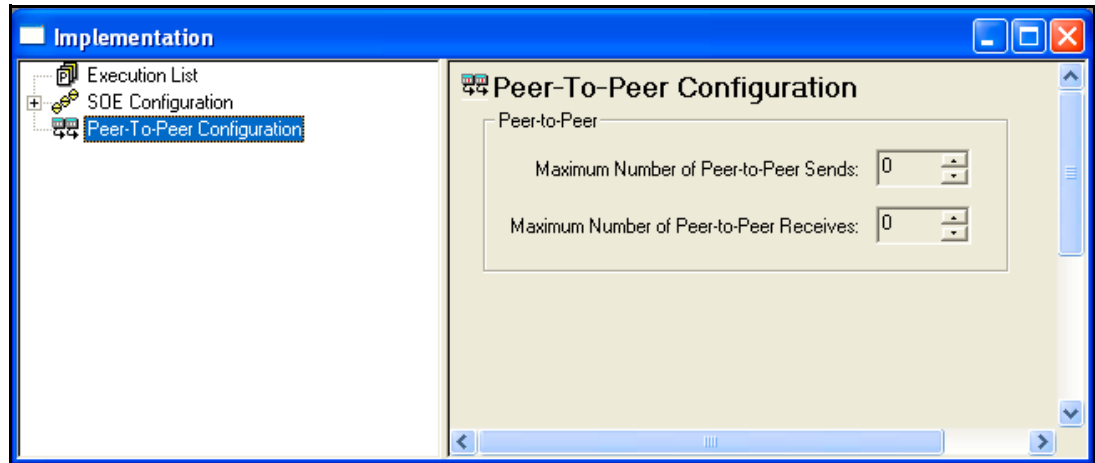
This procedure explains how to allocate memory for Peer-to-Peer functions, which is based on the maximum number of Send and Receive numbers you specify. To save memory and minimize scan time, you should use the lowest possible numbers.

The maximum number does not have to be the same for Sends and Receives. For example, a TriStation application might need to send messages to three applications, but need to receive messages from only one application.

A change in Peer-to-Peer allocation requires a Download All.

Procedure

- 1 In TriStation 1131, expand the Application tree, double-click Implementation, and then click Peer-to-Peer Configuration.



- 2 Set the Peer-to-Peer properties by clicking the up and down arrows.

Note If you change these settings for an application running on the controller or the Emulator, you must build the application and perform a Download All.

Step 5: Add Send/Receive Function Blocks to the Application

Verify that you have added at least one pair of Send and Receive function blocks to your application, and have configured them to use the Peer-to-Peer nodes you configured in your project:

- See the *TriStation 1131 Libraries Reference* for detailed information about available Send and Receive function blocks.
- See the *TriStation 1131 Developer's Guide* for more information about configuring Peer-to-Peer communication in your application.

Step 6: Select the CM to Get the Peer-to-Peer Configuration From

Verify that you have selected the communication module with the Peer-to-Peer configuration you want to use when you perform a Download All or Download Changes operation on the Emulator.

This selection is made in the Emulator Project Options dialog box; it must be set **before** starting the Emulator. See *Specifying Emulator Project Options* on page 10.

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/788072143134006122>