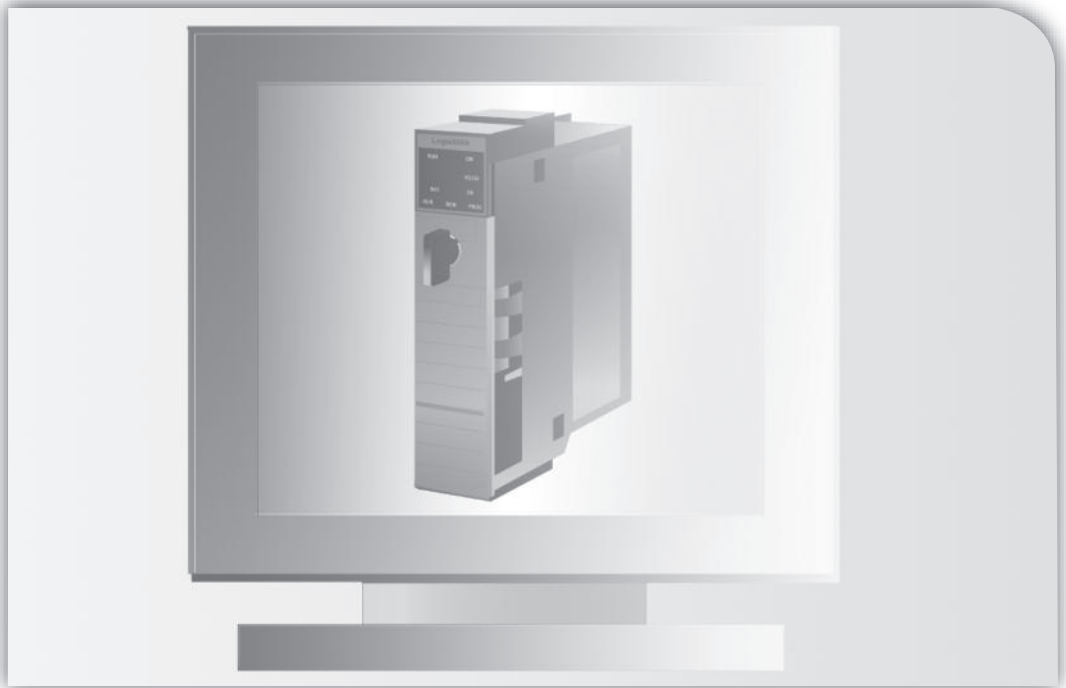


LISTEN.  
THINK.  
SOLVE.®

# *RSLogix™ Emulate 5000*



## **GETTING RESULTS GUIDE**

PUBLICATION LGEM5K-GR016G-EN-E – January 2013

Supersedes Publication LGEM5K-GR016F-EN-E

ALLEN-BRADLEY • ROCKWELL SOFTWARE

**Rockwell  
Automation**

**Contact Rockwell** Customer Support Telephone — 1.440.646.3434  
Online Support — <http://support.rockwellautomation.com/>

**Copyright Notice** © 2013 Rockwell Automation Technologies, Inc. All rights reserved. Printed in USA.

This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies, Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies, Inc. is strictly prohibited. Please refer to the license agreement for details.

**Trademark Notices** Allen-Bradley, Arena, ControlLogix, Data Highway Plus, DH+, Data Highway II, Datapac, 9 Emonitor, Enlab, Enlube, Enpac, Enshare, Entek, Entrx, Enwatch, eProcedure, FactoryTalk, GuardLogix, GuardPLC, Logix5000, Logix5550, MicroLogix, MobileView, MobileView Guard, MotorMonitor, PanelBuilder, PanelView, PhaseManager, PlantLink, PLC-2, PLC-3, PLC-5, Powermonitor, ProcessLogix, Propack Data, Rockwell, Rockwell Automation, Rockwell Software, RASAssetSecurity, RSBizWare, RSBizWare BatchCampaign, RSBizWare BatchHistorian, RSBizWare Coordinator, RSBizWare Historian, RSBizWare MaterialTrack, RSBizWare PlantMetrics, RSBizWare Scheduler, RSBizWare Tracker, RSEnergyMetrix, RSLinx, RSLogix, RSLoop Optimizer, RSNetWorx, RSNetWorx for ControlNet, RSNetWorx for DeviceNet, RS PMX, RSPower, RSSql, RSTune, RSVIEW, RSVIEW Administration Console, SIMAN, SLC, SoftLogix, VersaView, WINtelligent, and XM are trademarks of Rockwell Automation, Inc.

Any Rockwell Automation software or hardware not mentioned here is also a trademark, registered or otherwise, of Rockwell Automation, Inc

**Other Trademarks** ActiveX, Microsoft, Microsoft Access, SQL Server, Visual Basic, Visual C++, Visual SourceSafe, Windows, Windows ME, Windows NT, Windows 2000, Windows Server-, Windows 7, Windows XP, and Windows Vista are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Adobe, Acrobat, and Reader are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States and/or other countries.

ControlNet is a registered trademark of ControlNet International.

DeviceNet is a trademark of the Open DeviceNet Vendor Association, Inc. (ODVA)

OLE for Process Control (OPC) is a registered trademark of the OPC Foundation.

Oracle, SQL\*Net, and SQL\*Plus are registered trademarks of Oracle Corporation.

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

**Warranty** This product is warranted in accordance with the product license. The product's performance may be affected by This product is warranted in accordance with the product license. The product's performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product's implementation may vary among users.

This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at anytime without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

Version: 21.00

# Contents

<b>1 • Installing RSLogix™ Emulate 5000</b>	<b>1</b>
Process of Installing RSLogix Emulate 5000	1
Hardware and Software Requirements for RSLogix Emulate 5000	1
Installing the FactoryTalk® Activation Manager	3
Installing RSLinx Classic Lite	3
Installing RSLogix Emulate 5000	4
Considerations when uninstalling RSLogix Emulate 5000	6
Getting Assistance with RSLogix Emulate 5000	7
<b>2 • An Overview of RSLogix™ Emulate 5000</b>	<b>9</b>
Components of RSLogix Emulate 5000	9
Differences between the Emulator and a Physical Controller	9
About Floating Point Calculations and the Emulator	11
A Brief Outline of Using RSLogix Emulate 5000	11
<b>3 • Using the Chassis Monitor</b>	<b>13</b>
Starting the Chassis Monitor	13
Navigating in the Chassis Monitor	13
Creating an Emulation Module	15
Configuring the Emulator	18
Creating Simulated I/O	21
Configuring the Chassis Monitor	23
Using Emulator Modes	23
<b>4 • Connecting the Emulator to FactoryTalk® View ME or     FactoryTalk®View SE Projects through RSLinx® Enterprise</b>	<b>25</b>
Prerequisites	25
The RSLinx Enterprise Module in the Virtual Chassis	26
Configuring an RSLinx Enterprise Server for Your Emulator	27
<b>5 • Connecting the Emulator to RSLogix™ 5000</b>	<b>31</b>
Creating a Communications Driver for the Emulator	31
Setting a Logic Project to Use the Emulator	32
Configuring Communications in the Logic Project	33
Connecting to the Emulator from a Remote Computer	34



<b>6 • Connecting the Emulator to an RSView®32 Project through OLE</b>	<b>35</b>
Creating an OPC Topic for the Emulator . . . . .	35
Creating a Communications Node in an RSView32 Project . . . . .	35
Adding Tags from Your Logic Project to Your RSView32 Project . . . . .	37
<b>7 • Simulating I/O</b>	<b>39</b>
Configure Simulated I/O in the Logic Project . . . . .	39
Configuring Produced and Consumed Tags . . . . .	40
Using RSView32 or RSTestStand to Simulate I/O . . . . .	42
<b>8 • Programming Breakpoints and Tracepoints</b>	<b>43</b>
Programming Breakpoints . . . . .	43
Programming Tracepoints . . . . .	45
String Formats for Tracepoint and Breakpoint Instructions . . . . .	48
Using Trace Windows . . . . .	51
<b>9 • Specifying simulation options for multiple controllers</b>	<b>55</b>
Change the mode for multiple controllers . . . . .	55
Save the current state of multiple controllers . . . . .	56
Restore the previously saved state for multiple controllers . . . . .	56
<b>Index</b>	<b>59</b>

# 1

## Installing RSLogix™ Emulate 5000

You cannot install RSLogix™ Emulate 5000 on a computer that has a SoftLogix5800™ controller installed.

If you have a previous installation of RSLogix Emulate 5000 on your computer, you must uninstall RSLogix Emulate 5000 before installing or upgrading the software. See *Considerations when uninstalling RSLogix Emulate 5000* on page 6 for more information.



RSLogix Emulate 5000 supports projects that are developed with version 12 or higher of RSLogix™ 5000 software.

### Process of Installing RSLogix Emulate 5000

The process of installing RSLogix Emulate 5000 is as follows:

1. If you have a previously installed version of RSLogix Emulate 5000 on your computer, uninstall it. See *Considerations when uninstalling RSLogix Emulate 5000* on page 6 for more information.
2. Install prerequisite software for RSLogix Emulate 5000. See *Software Requirements* on page 2 for more information.
3. Install RSLogix Emulate 5000. See *Installing RSLogix Emulate 5000* on page 4 for more information.

### Hardware and Software Requirements for RSLogix Emulate 5000

#### Hardware Requirements

The computer on which you are installing RSLogix Emulate 5000 must meet the following minimum requirements.

For this component:	Use:
Personal computer	Any with an Intel® Pentium® 4 Processor running at 2.8 GHz or higher. Hyperthreaded and dual-core CPUs are supported.
RAM	1 GB or more
Monitor and video card	Any with 24-bit color VGA graphics device 1024 x 768, or greater resolution (DirectX 9 compatible graphics device with WDDM 1.0 or higher driver, for optimal resolution)
Hard drive	500 MB of available hard disk space plus an additional 64 MB per controller instance

## Software Requirements

The following software must be installed before installing RSLogix Emulate 5000.

For this software component:	Install:	Notes:
Operating system	<p>One of the following operating systems:</p> <p>32-bit Windows operating systems:</p> <ul style="list-style-type: none"> <li>• Windows 7 Professional with Service Pack 1</li> <li>• Windows 7 Home Premium with Service Pack 1</li> </ul> <p>64-bit Windows operating systems:</p> <ul style="list-style-type: none"> <li>• Windows Server 2008 R2 Standard Edition with Service Pack 1</li> <li>• Windows 7 Professional with Service Pack 1</li> <li>• Windows 7 Home Premium with Service Pack 1</li> </ul>	
Logix5000 programming software	RSLogix 5000, version 12.0 or higher	Required to program the emulator and observe its operation. To emulate a controller that has been released with a specific version of RSLogix 5000, the matching version of RSLogix 5000 is required. For example, if you want to emulate a v18 controller, you must use RSLogix 5000 v18.
Communication software	RSLinx® Classic, version 3.51.01 or higher	Required for communicating with the emulator.
Software activation	FactoryTalk® Activation version 3.51.00 or higher (supplied with the software)	Must be activated before you can use it. Refer to FactoryTalk Activation Manager help for more information on activation. It is optional if you are upgrading a previously activated version of RSLogix Emulate 5000.



## Installing the FactoryTalk® Activation Manager

Software activation is a process by which you identify that you have installed a legitimate copy of RSLogix Emulate 5000 on your computer.

From v20, FactoryTalk Activation Manager is required to activate the software. To install the FactoryTalk Activation Manager:

1. Log onto the computer on which you want to install the software as an administrator (or a user with administrative privileges).
2. Insert the RSLogix Emulate 5000 CD-ROM into your computer's CD-ROM drive.
3. The RSLogix Emulate 5000 installation program should start automatically. If it does not, use Windows Explorer to open your computer's CD-ROM drive and run the AUTORUN.EXE program.
4. In the installation program, click **Install FactoryTalk Activation Manager**.
5. If the InstallShield Wizard prompts you to install HASP USB dongle drivers, click **Continue** to install the drivers. Follow the on-screen instructions.
6. Continue following the instructions on the screen to complete the FactoryTalk Activation Manager installation.
7. When the FactoryTalk Activation Manager installation is complete, follow the onscreen instructions to open the Manage Activations tab of FactoryTalk Activation Manager. Once you have the Manage Activations tab open, click **Finish** on the InstallShield Wizard Completed dialog box. If you are prompted to restart your computer for the configuration changes made to FactoryTalk Activation Manager to take effect, click **No** to restart later.
8. In the Manage Activations tab, click Get New Activations and select the activation method you want to use.
9. Continue following the on-screen instructions to obtain your activations.
10. Click Exit on the window with the FactoryTalk Activation installation menu.
11. Restart your computer for the configuration changes to FactoryTalk Activation Manager to take effect.

## Installing RSLinx Classic Lite

If you do not already have version 3.51.01 or higher of RSLinx Classic or RSLinx Classic Lite installed on the computer on which you are installing the emulator, you need to install it. RSLinx Classic Lite (a stripped-down version of RSLinx Classic) is provided on the RSLogix Emulate 5000 installation CD-ROM.



If you have RSLinx Classic installed on your computer, but it is not version 3.51.01 or higher, installing RSLinx Classic Lite will upgrade your RSLinx Classic installation.

To install RSLinx Classic Lite:

1. Log onto the computer on which you want to install the emulator as an administrator (or a user with administrative privileges).
2. Insert the RSLogix Emulate 5000 CD-ROM into the computer's CD-ROM drive.
3. The setup program should start automatically. If it does not, open the CD-ROM drive with Windows Explorer and run the AUTORUN.EXE program on that drive.
4. The Setup program displays a menu of choices. Click **Install RSLinx Classic Lite**.
5. Follow the on-screen installation instructions.
6. RSLinx Classic must be configured to connect to the emulator. After you have installed the emulator, see *Creating a Communications Driver for the Emulator* on page 31.

## Installing RSLogix Emulate 5000



You cannot install RSLogix Emulate 5000 on a computer that has a SoftLogix5800 controller installed.

If you are reinstalling RSLogix Emulate 5000, you must uninstall it before proceeding with the installation. See *Considerations when uninstalling RSLogix Emulate 5000* on page 6 for more information.

To install and uninstall RSLogix Emulate 5000, you must logon to your computer as an Administrator.

Use the Windows Add/Remove Programs application (located in the Windows Control Panel) to remove the emulator.

To install RSLogix Emulate 5000:

1. Log onto the computer on which you want to install the emulator as an administrator (or a user with administrative privileges).
2. Insert the RSLogix Emulate 5000 CD-ROM into the computer's CD-ROM drive.
3. The setup program should start automatically. If it does not, open the CD-ROM drive with Windows Explorer and run the AUTORUN.EXE program on that drive.
4. The Setup program displays a menu of choices. Click **Install RSLogix Emulate 5000**.
5. The Setup program displays a welcome screen. Click **Next**.





6. The Setup program displays the License Agreement for the software. Read the agreement carefully. If you choose to adhere to the terms of the agreement, click **I accept the terms in the license agreement**, and then click **Next**. If you do not accept the terms of the license agreement, click **Cancel** to stop the Setup program.
7. The Setup program displays the **Customer Information** screen. In this screen, enter your name, the name of the company you work for, and your ten-digit software serial number (provided with the software) in the fields provided. Click **Next**.
8. The Setup program displays the Setup Type screen. On this screen, the software asks whether you want a complete install or a custom install. (Both installation methods result in the same software components being installed on your computer, but if you use a custom install you can choose where the software is installed.) By default, the software is installed in the C:\Program Files\Rockwell Software\RSLogix Emulate 5000 folder.

If you want to install the software in the default folder, click **Complete**, and then click **Next**.

If you want to change where the software will be installed, click **Custom**, and then click **Change**. You can then browse or create a new installation folder. Once you have selected your installation folder, click **OK**, and then click **Next**.

9. The Setup program displays the **Ready to Install** screen. Click **Install** to start the software installation.
10. The Setup program installs the software. After the software is installed, the RSLogix Emulate 5000 setup program displays a screen saying that its installation is complete.

If you want to display the release notes for the software, select the **I would like to view the Release Notes** check box. The Release Notes contain information concerning the software that has changed since the publication of this book.

If you want to create a desktop shortcut for the software, check **Create a shortcut on Desktop**. The shortcut will give you easy access to the emulator's Chassis Monitor application. (If you do not place the Chassis Monitor icon on your desktop, you will need to access the Chassis Monitor through the Start menu.)

11. Click **Finish** to end the RSLogix Emulate 5000 setup program. The setup program ends. If you chose to view the Release Notes in step 10, the file opens.

## Considerations when uninstalling RSLogix Emulate 5000



You **must** log into the computer as an administrator to uninstall RSLogix Emulate 5000.

If you get an error message when you attempt to uninstall RSLogix Emulate 5000, you will need to perform the following steps.

1. Remove all modules from the virtual backplane (except for the RSLinx Classic and RSLinx® Enterprise modules, which cannot be removed). To remove a module, right-click the module and then click **Remove**.



Rockwell Automation recommends removing all modules from the virtual backplane before uninstalling RSLogix Emulate 5000.

2. Stop the services for RSLinx Classic and RSLinx Enterprise (if installed on the computer). You can stop the RSLinx Classic service through the RSLinx Launch Control Panel (click **Start > Programs > Rockwell Software > RSLinx > RSLinx Classic Launch Control Panel**).

The procedure for stopping the RSLinx Enterprise service depends on the version of Windows that you are using. See the documentation for Windows for information about using the Services MMC snap-in (open the Windows help file and search for “services”).



## Getting Assistance with RSLogix Emulate 5000

### Online Help

You can access the online help for RSLogix Emulate 5000 through the Chassis Monitor. Open the Chassis Monitor and click **Help > Help Topics**.

### Rockwell Automation Technical Support

For information about all of the products available from Rockwell Software or for technical support, go to the following Web site:

**<http://www.rockwellautomation.com/rockwellsoftware/>**

If you cannot connect to the Internet or cannot find answers to your questions in this help file or on the Technical Support Web site, you can call Technical Support.

Phone: 440-646-3434 in North America

When you call you should be at your computer and be prepared to give the following information:

- the product version number
- the type of hardware you are using
- the exact wording of any messages that appeared on your screen
- a description of what happened and what you were doing when the problem occurred
- a description of how you tried to solve the problem

### Support Information on the World Wide Web

To obtain support information from Rockwell Automation's World Wide Web site, go to <http://www.rockwellautomation.com/support/>

•  
•  
• GETTING RESULTS WITH RSLOGIX™ EMULATE 5000  
•  
•

# 2

## An Overview of RSLogix™ Emulate 5000

RSLogix™ Emulate 5000 is a software product that emulates the behavior of Logix5000™ controllers. The emulator allows you to experiment with and debug your Logix5000 programs in a safe, controlled environment without investing in physical controllers and I/O modules. Additionally, RSLogix Emulate 5000 allows you to test your HMI applications without using a real controller.

The emulator can be programmed and accessed both on the computer where it is installed and remotely through RSLinx® Classic Gateway. For more information about configuring communications for the emulator, see Chapter 5, *Connecting the Emulator to RSLogix™ 5000*.



Logix5000 controllers include: ControlLogix®, CompactLogix™, FlexLogix™, SoftLogix™, and DriveLogix™ controllers. For more information about the Logix5000 family of controllers, see the Allen-Bradley Web site (<http://www.ab.com>) or contact your local Rockwell Automation sales office.

### Components of RSLogix Emulate 5000

There are two components of RSLogix Emulate 5000:

- The **Chassis Monitor**, which is a software application that allows you to configure emulation and simulated I/O modules
- **Emulation modules**, which are Windows services that simulate the behavior of Logix5000 controllers and certain I/O modules

Emulation modules run as Windows services, which are applications that run without interfaces of their own. The Chassis Monitor serves as an interface to the emulation modules, allowing you to create and configure instances of the modules.

### Differences between the Emulator and a Physical Controller

RSLogix Emulate 5000 simulates most of the behaviors of Logix5000 controllers. There are exceptions, and you should keep those exceptions in mind as you use the emulator. The following table shows the differences between Logix5000 controllers and RSLogix Emulate 5000.



---

Do not depend on the emulator to match your physical controller's performance or operation. Some instructions are interpreted differently in the emulator than in a physical controller, and the execution times for instructions and program files will be significantly different in an emulated controller than in a physical controller.

---

<b>Features:</b>	<b>Logix5000 Controllers</b>	<b>RSLogix Emulate 5000</b>
Breakpoints and tracepoints	Not supported	Supported <sup>a</sup>
Control real I/O	Supported	Not supported
Forcing	Supported	Supported
Interface to non-Rockwell Software HMI software	DDE/OPC (requires RSLinx Classic Professional, RSLinx Classic Single Node, RSLinx Classic OEM, or RSLinx Classic Gateway; RSLinx Classic Lite does <b>not</b> support DDE or OPC)	DDE/OPC (requires RSLinx Classic Professional, RSLinx Classic Single Node, RSLinx Classic OEM, or RSLinx Classic Gateway; RSLinx Classic Lite does <b>not</b> support DDE or OPC)
Programming languages	Ladder diagram, function block, sequential function chart, and structured text	Ladder diagram, function block, sequential function chart, and structured text
Messaging	Supported	Supported (local only)
Motion instructions	Supported	Supported (virtual axes only)
Jump to subroutines (JSR instructions)	Supported	Supported - however, parameters are passed to the subroutine in reverse order (the last parameter is passed first)
Jump to external routines (JXR instructions)	Supported <sup>b</sup>	Not supported
Communications through RS232 ports	Supported	Supported
Communications with RSView32/FactoryTalk View	Supported	Supported
Time scaling	Not supported	Supported
Single Scan Mode	Not supported	Supported
Network cards	Yes	Not supported
Task priority levels	16	3
Trending	Yes	Yes
Online editing of sequential function chart and structured text programs <sup>c</sup>	Yes	Yes

*a. Supported only in ladder diagram programming.*

*b. Supported only in SoftLogix5800™ controllers.*

*c. Supported only in version 13 and higher.*



## About Floating Point Calculations and the Emulator

There are instruction differences across programming languages as well as across Logix platforms. The following examples use the operation “ $i = n / m$ ” where all the operands are integers. Ladder DIV instructions perform an integer divide operation and store the immediate result in the destination. For “ $n = 800$  and “ $m = 1000$ ,” “ $i$ ” is equal to 0. The DIV function block instructions only does floating point operation. The function block instruction converts the inputs to reals (if necessary) and then converts the result from a real to the destination type is (if necessary). In this example, “ $n$ ” is converted to 800.0 and “ $m$ ” is converted to 1000.0. The result of the operation is 0.8. That result then gets converted to an integer where rounding rules apply and the final destination value is 1. This difference between ladder and function block instructions applies to all Logix platforms. Function block instructions only perform floating point operations.

## A Brief Outline of Using RSLogix Emulate 5000

Typically, RSLogix Emulate 5000 is used to check logic (for example, the logic behind HMI systems) before that logic is put into physical controllers. Here’s a brief outline of how you can implement RSLogix Emulate 5000 in your project development.

1. In RSLogix 5000, create a new project using the Emulator controller type.
2. Set up the project’s I/O configuration to match that of the Chassis Monitor, configuring I/O simulation modules or additional emulator modules to emulate your application.
3. Develop the logic program for your application, configuring it to run on the emulator. Use symbolic tags for your I/O points and alias them to simulated I/O modules or other emulators (using produced and consumed tags). You can also use tags in RSTestStand™ or RSView® to simulate I/O.
4. Download your program to the emulator and debug it using the tools in RSLogix 5000.
5. After verifying your program’s operation with the emulator, save a copy of your project for reference.
6. In RSLogix 5000, change the controller type of your project to use the physical controller that will run your application. Reconfigure your project’s I/O to use the modules for your application.
7. Re-alias the I/O points in your project to use the I/O modules for your application.
8. Remove any tracepoint and breakpoint instructions you may have used in the project (these will not work in a physical controller).
9. Verify your project and correct any errors. You can then download your project to your physical controller.

•  
•  
• GETTING RESULTS WITH RSLOGIX™ EMULATE 5000  
•  
•



# 3

## Using the Chassis Monitor

The Chassis Monitor is a software application used to configure RSLogix™ Emulate 5000 and I/O simulation modules.

### Starting the Chassis Monitor

There are two methods for starting the Chassis Monitor. You can launch the Chassis Monitor from the Windows Start menu, or you can click the Chassis Monitor icon on the Windows desktop (if you chose to place it there during installation)



Unlike the SoftLogix 5800™ controller, RSLogix Emulate 5000 stops and starts with the Chassis Monitor. RSLogix Emulate 5000 runs only while the Chassis Monitor is running.

### Starting the Chassis Monitor from the Start Menu

To start the Chassis Monitor, click **Start > Programs > Rockwell Software > RSLogix Emulate 5000 > RSLogix Emulate 5000**.

### Starting the Chassis Monitor from the Windows Desktop

If you chose to place an icon for the Chassis Monitor on the Windows Desktop, you can start the Chassis Monitor by double-clicking the icon. The icon looks like this:



### Navigating in the Chassis Monitor

The Chassis Monitor is where you can create and configure RSLogix Emulate 5000 emulation modules and simulated I/O modules. Modules reside in the Chassis Monitor's "slots," much as they would reside in a physical chassis.

Menus available at the top of the Chassis Monitor window and from right-clicking slots and modules allows you to configure your chassis with the modules you need.

The RSLinx™ Classic module represents a communication module for the chassis. By default, the RSLinx Classic module resides in slot 0, but you can specify an alternate slot position when you configure the Virtual Backplane driver in RSLinx Classic. For more information, see the RSLinx Classic online help.

•  
•  
• GETTING RESULTS WITH RSLOGIX™ EMULATE 5000  
•  
•

The following illustration shows a Chassis Monitor window with three emulator and two simulated I/O modules.

*This is a simulated I/O module.  
You can click on the module's terminal cover to open it.*

*This is an emulator module.*

*The red box surrounding this module indicates it is selected. You can right-click modules to display a configuration menu.*

*The RSLinx Classic module in slot 0 (default location). The RSLinx Classic module can reside in any available slot.*

*This button allows you to contract the Chassis Monitor into a smaller display. This smaller display hides slots 9 through 16.*

*These "slots" hold emulator or simulated I/O modules. Click **Slot > Create Module** to create a module.*

## Creating an Emulation Module



You can have up to 16 emulation modules in the Chassis Monitor at one time. However, your computer's performance may be impaired significantly if you run multiple emulation modules simultaneously.

To create an emulation module:

1. Click **Slot > Create Module**.
2. From the Module Type field, choose the type of module you want to create. Choose **Emulator RSLogix Emulate 5000 Controller**.
3. In the **Slot** field, enter the slot where you want the emulator to reside. The emulator can be in any slot (except the slot occupied by the RSLinx Classic module.).



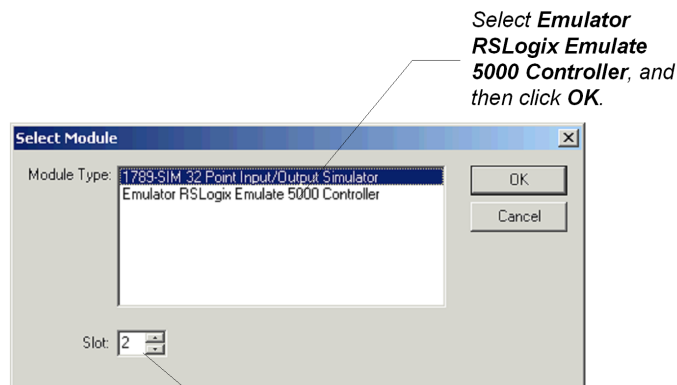
If you want the emulator controller to reside at slot 0, see the Application Notes section of the RSLogix Emulate 5000 Release Notes.



If you modify the project to use a physical Logix5000 controller, make sure your rack and slot configurations in your project match those in your control system. Otherwise, you will not be able to communicate properly with the physical controller.

4. Click **OK** to accept your changes and add the emulator to the virtual chassis.

You can click **Cancel** if you wish to cancel creating the emulator.

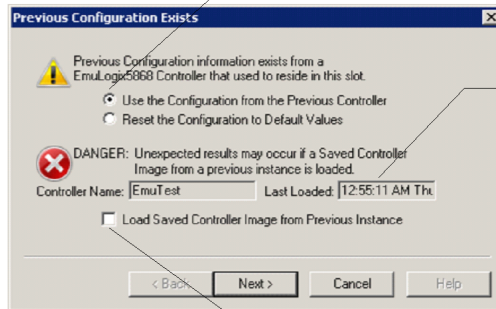




The only modules that function with the emulator and appear in the module type list are the **Emulator RSLogix Emulate 5000 Controller** and the **1789-SIM 32 Point Input/Output Simulator**.

5. If you have previously created a emulator module in the same slot and you did not delete the configuration for the module when you removed it, the software tells you there is previous configuration information for the module. The software asks whether you want to use the previous configuration, or reset the configuration to default values.
  - a. To use the previous configuration, click **Use the Configuration from the Previous Controller**. To reset the configuration, click **Reset the Configuration to Default Values**. (Even if you reset the configuration to its default values, the name of the emulator remains the same as the one that was deleted.)
  - b. If there was a program in the deleted emulator module, the software shows the controller name and the date the program was last downloaded into the controller. If you want to load this program into the emulator module you are creating, check the **Load Saved Controller Image from Previous Instance** check box.

*Choose whether you want to retain the previous controller's configuration or reset the configuration to default values.*



*If there is a previously loaded controller image, the name of that image and the time it was loaded into the previous controller is shown here.*

*If you want to load the previously loaded controller image into the new emulator module, check this box.*



Using a previously loaded controller image may result in unexpected operation. Make sure the program is correct before running your emulations.

- c. Click **Next** to continue.



6. A configuration wizard appears to allow you to enter module-specific properties. The first screen of the wizard allows you to enter general properties for the emulator.

<b>For this:</b>	<b>Select or type:</b>
Version	Select the version of RSLogix™ 5000 used to create the project you want to emulate. You can select version 12 or higher.
Memory size	The memory size for the emulator. Make this equal to the memory size of the controller you are emulating.
Periodic Save Interval	Specify whether you want to save the current controller information (program, data, and configuration information) periodically, and if so, specify how often (minutes). The default is enabled for 10 minutes.

7. Click **Next**. The software displays a window for setting additional features of the emulator.

<b>For this:</b>	<b>Select or type:</b>
Continuous Task Dwell Time	Specify the timeslice (0-1000 ms) made available for all other Windows applications. The default is 10 ms.  The dwell time is the time between the end of the continuous task and the start of the next execution of the continuous task.
CPU Affinity	If you are running the emulator on a computer that has more than one CPU, set the CPU that you want to run the emulator (the emulator can run on only one CPU).
Channel 0 Serial Port	If you want to emulate serial communications with a Logix5000 controller, set the computer serial port that you want to use to emulate Channel 0 of the Logix5000™ controller.

8. Click **Finish**. The Chassis Monitor places an emulator module in the slot you selected.



If you right-click the virtual chassis on the slot where you wish to create your module, and then choose **Create**, the slot number is automatically filled in for you.

## Configuring the Emulator

Configuring the emulator is done through the Chassis Monitor.

### Display the Emulator Configuration

To display the configuration for an emulator:

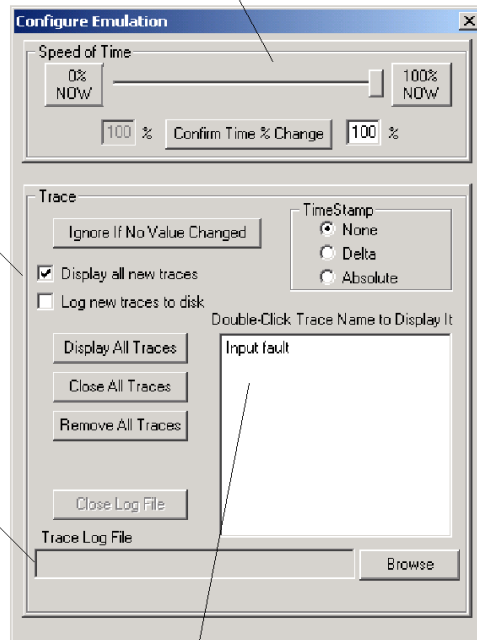
1. Open the Chassis Monitor.
2. Right-click the emulator module for which you want to display the configuration, then click **Configure Emulation**. The Configure Emulation window opens..

*These controls allow you to control the "speed of time".*

*These controls allow you to configure trace displays.*

*This control allows you to log traces to a file.*

*Active traces appear in this list.*



## Set the “Speed of Time”

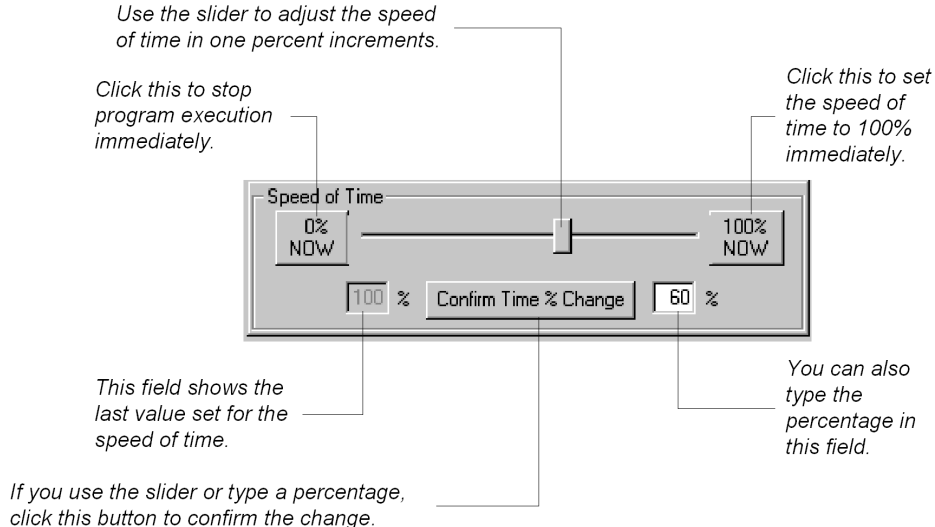
The “speed of time” setting allows you to slow down program execution in the emulator. When you slow the speed of time, timers slow down, and additional time is added to the program scan time, allowing you more time to observe program execution as it happens.

The speed of time is set as a percentage of “real time.” When the speed of time is set to 100%, program execution is in real time. Setting the speed of time to another value slows program execution by an amount proportional to the percentage you select. For example, setting the speed of time to 60 percent makes actions that normally take one second take around 1.67 seconds.

Changes to the speed of time setting take effect as soon as they are set.

To set the “speed of time”:

1. Open the Chassis Monitor.
2. Right-click the emulator module for which you want to display the configuration, then click **Configure Emulation**. The Configure Emulation window opens.
3. Set the speed of time using the controls displayed in the window. The following illustration shows how the controls work.



4. If you used the slider or entered a percentage manually, click the **Confirm Time % Change** button.

## Configure Trace Displays

When tracepoint and breakpoint instructions execute, the emulator opens a trace display. To set a trace, program a tracepoint or breakpoint instruction. See *Programming Breakpoints and Tracepoints* on page 43 for information about programming these instructions.

To configure how the emulator handles and displays these traces:

1. Open the Chassis Monitor.
2. Right-click the emulator module for which you want to display the configuration, then click **Configure Emulation**. The Configure Emulation window opens.
3. Refer to the following table for the settings.

This setting:	Does this:
Ignore If No Value Changed	This button toggles whether the trace displays should record every trace or only those traces where there are changes to the value being traced. If the button is indented, the trace displays will ignore those traces where the value does not change.
TimeStamp	<p>Set this to record how you want time recorded in traces. (If you have a time stamp mode selected and you change it while a trace window is displayed, subsequent traces use the new time stamp mode.)</p> <p><b>None</b> records no time information in the trace displays.</p> <p><b>Delta</b> records the difference between the current trace and the last trace for that particular trace window. The first trace for the window always shows 0 milliseconds.</p> <p><b>Absolute</b> records the time that the trace triggered. The time is in the number of milliseconds since the emulator was last changed to the Run mode.</p>
Display all new traces	Check this box if you want all new traces to display automatically in a trace window.
Log new traces to disk	Check this box if you want new traces to automatically log to disk.
Display All Traces	Click this button to display all of the current traces in their trace windows. (The current traces are listed by name in the Configure Emulation window.)
Close All Traces	Click this button to close all of the open trace windows.
Remove All Traces	Click this button to remove all of the current traces from the trace list.
Close Log File	Click this button to stop logging traces to a file.
Trace Log File	This field shows the currently set log file. To set the log file, click <b>Browse</b> . For more information, see <i>Log Traces to Disk</i> on page 21.





## Log Traces to Disk

You can configure the emulator to log traces to a text file. To do this:

1. Display the emulator configuration.
2. Check the **Log new traces to disk** check box.
3. Click the **Browse** button.
4. Select the text file you want to hold trace information. If you want to create a new file, type the name of the new file in the **File name** field.



You can log data to a file with any extension (or with no extension at all). However, to be able to view traces with a text editor (such as Windows Notepad), use a TXT file extension.

5. Click **Open**. If the file you selected already has data in it, the software asks if you want to append your traces to the end of the file, overwrite the data in the file with the traces, or cancel.
6. Execute your logic. When a tracepoint output instruction (TPT) executes, the trace information is logged to the file you selected.
7. When you are ready to view the traces stored in the log file, display the emulator configuration, then click **Close Log File**. You can then open the log file in a text editor (such as Windows Notepad).



If you try to open the log file without closing it in the emulator configuration, you will see an error message saying that the file cannot be accessed. You must close the log file in the emulator before it can be viewed by another application.

## Creating Simulated I/O

You can use a simulated I/O module (1789-SIM) to simulate discrete I/O with the emulator. (This is the only form of I/O you can simulate with the emulator.) This module resides in the Chassis Monitor and provides inputs and outputs that you can use in logic programs.



In RSLogix 5000, the simulated I/O module is called “1756-MODULE.” In RSLogix Emulate 5000, the simulated I/O module is called “1789-SIM.” For information about configuring a simulated I/O module in an RSLogix 5000 project, see *Configure Simulated I/O in the Logic Project* on page 39.

To create a SIM module in the Chassis Monitor:

1. In the Chassis Monitor, click **Slot > Create Module**. The software displays the Select Module window.
2. In the Select Module window, select the **1789-SIM 32 Point Input/Output Simulator** from the **Module Type** list. Set the **Slot** field to the slot number into which you want to place the SIM module. Click **OK**. The software displays the Select Device window.
3. The Select Device window displays the slot number for the module. Perform the following:
  - a. If you deleted a SIM module from this slot without clearing its configuration, the software displays a message saying that previous configuration information exists for the module. You can retain the previous configuration, or you can reset the configuration to default values. To retain the previous configuration, click **Use the Previous Configuration**. To reset the module to its default values, click **Reset the Configuration to Default Values**.
  - b. Click **Next**.
3. The software displays the General window. In this window, you can label the SIM module with a text string that will scroll across the face of the module to help you identify it while it is in the Chassis Monitor. Type the name for the module in the **Label for 1789-SIM Module Marquee** field. (If you do not type a name, the software uses the words “Simulator Module.”)

## Using Produced and Consumed Tags

Another method for simulating I/O is to use another emulator module to produce and consume tags. See *Configuring Produced and Consumed Tags* on page 40 for information about configuring these tags.



## Configuring the Chassis Monitor

You can configure the behavior and display of the Chassis Monitor. The following table shows how to configure the Chassis Monitor.

To set the Chassis Monitor to:	Click this menu option:
Hide slots 9-16 (if you are not using those slots or you simply want the Chassis Monitor to take up less space on-screen)	<b>View &gt; Hide Slots 9-16</b>
Hide the computer name	<b>View &gt; Compact</b>
Change the rate at which the module display updates	<b>Options &gt; Auto Refresh Rate &gt; (High, Normal, or Low)</b>
Keep the Chassis Monitor on top of other windows when it is displayed	<b>Options &gt; Always On Top</b>
Change one or more modules to Run mode	<b>All Modules &gt; Run</b>
Change one or more modules to Program mode	<b>All Modules &gt; Program</b>
Create a snapshot that saves the current state (including program and data information) for one or more controllers at one time	<b>All Modules &gt; Save Snapshot</b>
Restore controllers to a previously saved state (from snapshot)	<b>All Modules &gt; Restore</b>
When you delete a module, automatically check the Clear Chassis Monitor module configuration check box. (If this feature is not set, the software does not automatically check that box for you.)	<b>Options &gt; Clear Configuration on Remove</b>

## Using Emulator Modes

Like a physical ControlLogix controller, RSLogix Emulate 5000 has several modes of operation.

- **Remote** mode, where an application (such as RSLogix 5000) can send a command to the emulator to change its mode.
- **Program** mode, which allows the emulator to be programmed.
- **Run** mode, where the emulator runs its logic programs. This mode also allows the emulator to be programmed (with the same kinds of limits, such as changes to tags)

The emulator does not have a test mode (there is no physical I/O for the emulator, therefore there is no need to prevent outputs from functioning).

To set the mode for the emulator, right-click the emulator in the virtual chassis and choose the mode you want.

## Single Scan Mode

The emulator has one mode that is not available in a physical controller: **Single Scan**. In single scan mode, the emulator runs through its logic programs once and then stops, allowing you to see the results of running your program one scan at a time.

To set single scan mode for the emulator, right-click the emulator in the virtual chassis and select **Single Scan**.



You must set each emulator to Single Scan mode individually.

When the emulator has been set to single scan mode and is placed in Run mode (such as through RSLogix 5000 in online mode or by right-clicking the emulator module and selecting **Run**), the emulator executes its logic through one scan and then stops. When it stops, the emulator displays a dialog asking whether you want to run the emulator continuously or to single scan again. Click **Yes** to run the emulator continuously, or **No** to run another single scan.

# 4

## Connecting the Emulator to FactoryTalk® View ME or FactoryTalk®View SE Projects through RSLinx® Enterprise

You can use either RSLinx® Enterprise or RSLinx® Classic to connect RSLogix Emulate 5000 to your FactoryTalk® View ME or FactoryTalk® View SE projects. Both methods use OLE for Process Control (OPC) to communicate, however, it is easier to use RSLinx Enterprise to create the necessary connections. This chapter describes only how to use RSLinx Enterprise to produce those connections — for information about using RSLinx Classic to create those connections, see the FactoryTalk View ME or FactoryTalk View SE documentation.

Using RSLinx Enterprise to connect the emulator to an FactoryTalk View project requires installing RSLinx Enterprise on both the computer running the emulator and on the computer running FactoryTalk View®.

RSLinx Enterprise uses FactoryTalk to automatically connect tags for Rockwell Software applications. Once RSLinx Enterprise is installed and configured, you do not need to configure anything else to use tags from the emulator in FactoryTalk View projects. Once a tag is used in RSLogix Emulate 5000, the tag becomes available to FactoryTalk View ME or FactoryTalk View SE.



For information about connecting RSLogix Emulate 5000 to RSVIEW®32 projects, see Chapter 6, *Connecting the Emulator to an RSVIEW®32 Project through OLE* on page 35.

### Prerequisites

For FactoryTalk View SE projects, RSLinx Enterprise must be installed on both the computer running FactoryTalk View SE and the computer running RSLogix Emulate 5000, and both computers must be using the same FactoryTalk Directory. (If RSLogix Emulate 5000 and FactoryTalk View SE are installed on different computers, both computers must use the same Network Directory.)



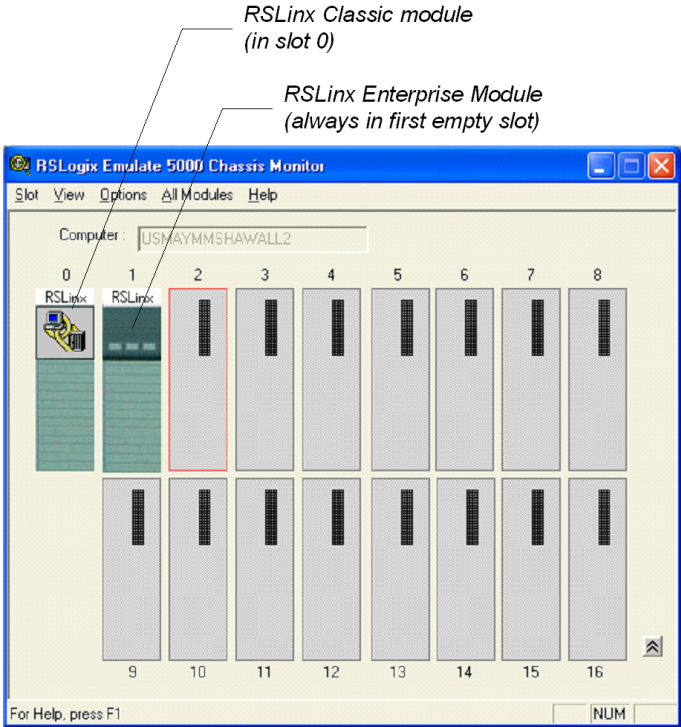
You can install RSLinx Enterprise and the FactoryTalk® Services Platform from your FactoryTalk View ME or FactoryTalk View SE installation disks.

For FactoryTalk View ME projects, RSLinx Enterprise, RSLogix Emulate 5000, and FactoryTalk View ME must be installed on the same computer. In this case, the FactoryTalk Directory used must be the Local Directory (FactoryTalk View ME does not support using a Network Directory).

### The RSLinx Enterprise Module in the Virtual Chassis

When RSLogix Emulate 5000 runs on a computer that is also running RSLinx Enterprise, an RSLinx Enterprise module appears in the emulator's virtual chassis. This module will occupy the first unoccupied slot in the chassis and cannot be removed (other than by uninstalling RSLinx Enterprise).

For example, if you install RSLinx Enterprise and then install RSLogix Emulate 5000 from scratch, the first time you open RSLogix Emulate 5000, slot 0 will be occupied by the RSLinx Classic module and slot 1 will be occupied by the RSLinx Enterprise module.



If you already have modules in your virtual chassis when you install RSLinx Enterprise, the RSLinx Enterprise module appears in the first empty slot.

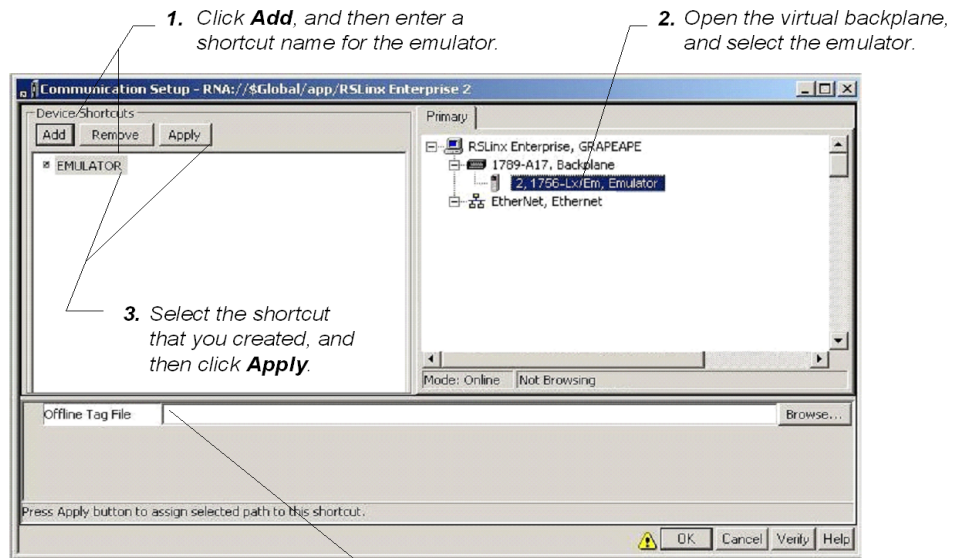


## Configuring an RSLinx Enterprise Server for Your Emulator

To configure an RSLinx Enterprise Server for your emulator:

1. Make sure the emulator is running.
2. Open your FactoryTalk View SE or FactoryTalk View ME project.
3. In the FactoryTalk® View Explorer tree, right-click the area of your project where you want to create the RSLinx Enterprise data server, and then click **Add New Server > Rockwell Automation Device Server (RSLinx Enterprise)**. (You can also create the server in the root of your FactoryTalk View project.)
4. Type a name for the RSLinx Enterprise server in the **Name** field. Optionally, you can type a description for the server in the **Description** field.
5. In the **Computer hosting the RSLinx Enterprise server** field, enter the name of the *computer running RSLinx Emulate 5000*. If this computer is the same as the computer running FactoryTalk View, enter either the name of the computer or **localhost**. (For FactoryTalk View ME systems, localhost is the only possible name.)
6. In the Explorer tree for your FactoryTalk View project, open the RSLinx Enterprise server and then open **Communication Setup**. The Communication Setup window for the server opens.
7. In the Communication Setup window, you need to create a “device shortcut” to the emulator. The shortcut allows you to identify which tags are coming from the emulator in the FactoryTalk View tag browser.
  - a. Click the **Add** button, and then type a name for the shortcut. This is the name that will appear in the FactoryTalk View tag browser.

b. Open the virtual backplane and select the emulator.



1. Click **Add**, and then enter a shortcut name for the emulator.

2. Open the virtual backplane, and select the emulator.

3. Select the shortcut that you created, and then click **Apply**.

4. You can also associate your offline project file with this shortcut in case you want to work with your FactoryTalk View project while the emulator is not running. (See the FactoryTalk View documentation for more information.)

c. Click **Apply**. You are prompted to apply your changes. Click **Yes**; this associates the shortcut with the emulator.

d. Click **OK**. This closes the Communication Setup window.

## Browsing for Tags from the Emulator in FactoryTalk View SE or FactoryTalk View ME

After you have configured the RSLinx Enterprise server for your FactoryTalk View SE or FactoryTalk View ME project, you can then browse for tags from the emulator whenever you need to do so in your project.

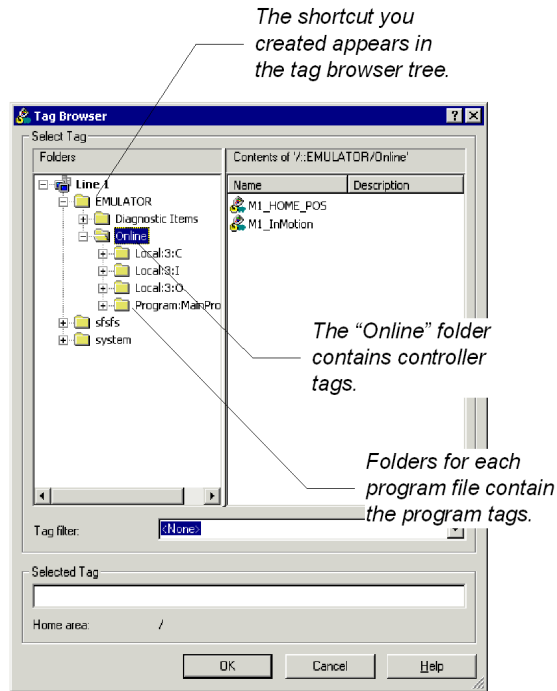
When you use the FactoryTalk View tag browser, you will see a folder corresponding to the shortcut you created in the RSLinx Enterprise server Communication Setup window. The folders inside that shortcut folder contain tags from the emulator.





The Online folder contains controller tags, and the folders for each program file contain program tags.

For more information about browsing for tags in an FactoryTalk View project, see the documentation for FactoryTalk View.



•  
•  
• GETTING RESULTS WITH RSLOGIX™ EMULATE 5000  
•  
•

# 5

## Connecting the Emulator to RSLogix™ 5000

Connecting RSLogix™ Emulate 5000 to RSLogix™ 5000 requires making some changes to your logic project.

1. In RSLinx® Classic (or RSLinx Classic Lite), create a driver and specify a slot number for the emulator.
2. In RSLogix 5000, set your logic project to use the emulator.
3. Configure communications in the logic project to use the emulator driver.
4. Add the simulated I/O to your logic project.
5. Program breakpoint and tracepoint instructions.

### Creating a Communications Driver for the Emulator

In RSLinx Classic, you need to create a communications driver for RSLogix Emulate 5000. This driver will allow other applications to communicate with the emulator.

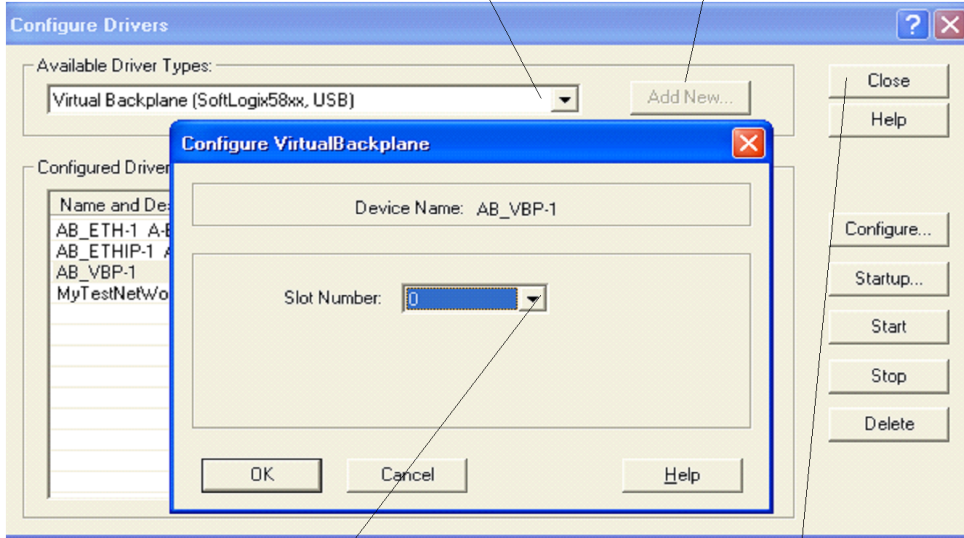
To create the communication driver:

1. In RSLinx Classic, click **Communications > Configure Drivers**. The Configure Drivers window appears.
2. Select the **Virtual Backplane (SoftLogix 58xx, USB)** driver from the **Available Driver Types** list.
3. Click **Add New**. The Add New RSLinx Driver window appears. This window allows you to name the driver. You can keep the default name, or type a name of up to 15 characters. Click **OK**. The Configure Virtual Backplane window appears.
4. In the Slot number field, specify the slot number where you want the emulator to reside. The default is slot 0. When complete, click **OK**.

5. The new driver appears in the **Configured Drivers** list. Click **Close**.

1. Select the **Virtual Backplane (SoftLogix 58xx, USB)** communications driver.

2. Click **Add New**.



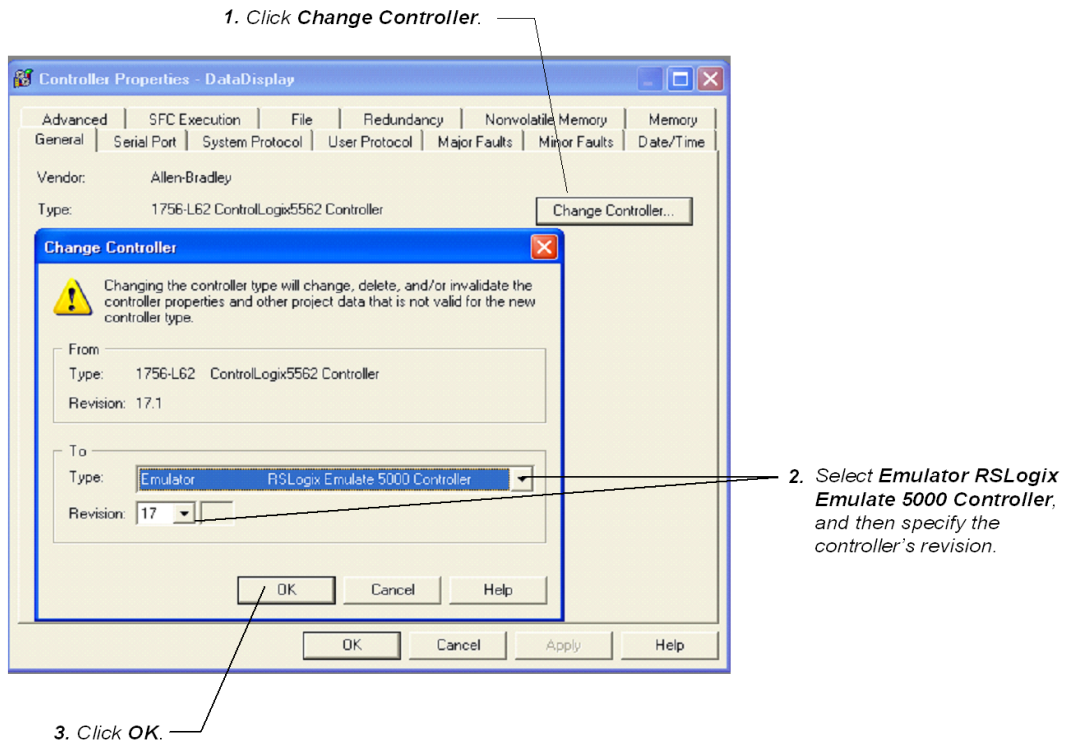
3. Specify the slot number.

4. Click **Close**.

## Setting a Logic Project to Use the Emulator

Before you can emulate a logic project, you must configure the project's controller type to use the emulator. To do this:

1. In RSLogix 5000, open the project that you want to emulate.
2. Click **Edit > Controller Properties**. The Controller Properties window opens.
3. On the **General** tab of the Controller Properties window, set the **Slot** field to correspond with the slot in the Chassis Monitor containing the emulator you want to use to emulate the project.
4. Click **Change Controller**. The Change Controller Type window opens.



5. From the **To** list, select **Emulator RSLogix Emulate 5000 Controller**, and then specify the controller's revision.
6. Click **OK**.

## Configuring Communications in the Logic Project

Before you can download to or go online with the emulator, you must configure communications in your logic project. To do this:

1. In RSLogix 5000, open the project you want to emulate.
2. Click **Communications > Who Active**. The Who Active window opens.
3. The Who Active window displays a tree of the available network nodes using the communication drivers you have configured in RSLinx Classic. In the tree, click the emulator you want to use for this project.
4. Click **Set Project Path**. This sets the communication path to use the emulator.

## Connecting to the Emulator from a Remote Computer

You can connect remote computers to a computer running RSLinux Emulate 5000. This allows you to use programs such as RSLinux 5000, RSTestStand™, and FactoryTalk® Batch from a computer other than the one running the emulator. For example, if you connect a remote computer running RSLinux 5000 to the computer running the emulator, you can upload, download, and edit online just as if you were running RSLinux 5000 locally.

To make this type of connection, you need to have RSLinux Classic Gateway software installed on the computer running the emulator. You can then use RSLinux Classic on the remote computers to connect to the computer running the emulator.

You will need to configure the RSLinux Classic software on the remote computer to use the **Remote Devices via Linux Gateway** driver.

For more information about using RSLinux Classic Gateway and RSLinux Classic, see the documentation for those products.



Connections to RSView®32, FactoryTalk® View ME, and FactoryTalk® View SE are configured differently. For information about connecting the emulator to these products, see Chapter 4, *Connecting the Emulator to FactoryTalk® View ME or FactoryTalk® View SE Projects through RSLinux® Enterprise* or Chapter 6, *Connecting the Emulator to an RSView®32 Project through OLE*.

# 6

## Connecting the Emulator to an RSView®32 Project through OLE

Connecting RSLogix™ Emulate 5000 to an RSView®32 project requires creating an OLE for Process Control (OPC) topic for communicating between RSLinx™ Classic and RSView32.



For information about connecting the emulator to FactoryTalk® View SE or FactoryTalk® View ME projects, see Chapter 4, *Connecting the Emulator to FactoryTalk® View ME or FactoryTalk® View SE Projects through RSLinx® Enterprise* on page 25.

### Creating an OPC Topic for the Emulator



You must use one of the following products to create the OPC topic: RSLinx Classic Single Node, RSLinx Classic OEM, or RSLinx Classic Gateway. RSLinx Classic Lite does **not** support OPC communication. For more information about using RSLinx Classic, see the RSLinx Classic documentation.

To create an OPC topic for the emulator:

1. Open the Chassis Monitor, and make sure your emulator is running.
2. In RSLinx Classic, open an RSWho window (click **Communications > RSWho**). The emulator should be visible in the RSWho window. If it is not, make sure there is a communication driver for the emulator. See *Creating a Communications Driver for the Emulator* on page 31 for more information.
3. In the RSWho window, right-click the emulator, then click **Configure New DDE/OPC Topic**. This creates a new OPC topic, and displays the DDE/OPC Topic Configuration window with a new topic for the emulator.
4. Click **Done**. This closes the DDE/OPC Topic Configuration window and saves the topic for your emulator.

### Creating a Communications Node in an RSView32 Project

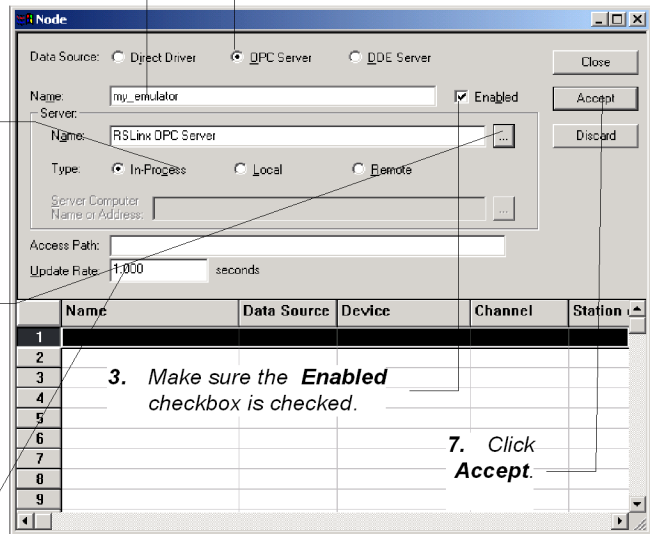
To create a communications node in RSView32 that uses the OPC topic for the emulator:

1. Open your RSView32 project.
2. In your project's **System** folder, open the **Node** editor. In RSView32, a node represents a communications device, usually a programmable controller or another Windows application. In the Node editor, we will create a node that represents the emulator.

- Set up the Node editor window as shown in the following table.

For this field or control:	Set or enter:
Data Source	OPC Server
Name	Enter a name for the node (representing the emulator). This can be the name of the emulator, if you like.
Enabled	Make sure this check box is checked.
Server	Click the button next to the Server Name field. If the emulator is running on the same computer as RSView32, select <b>RSLinx OPC Server</b> . Otherwise, select <b>RSLinx Remote OPC Server</b> .
Type	If the emulator is running on the same computer as RSView32, select <b>In-Process</b> . Otherwise, select <b>Remote</b> and enter the name of the computer running the emulator in the <b>Server Computer Name or Address</b> field.
Access Path	Leave this blank.
Update Rate	This defaults to 1.000 seconds (one update per second). If you need a faster or slower update rate, enter the update frequency.

- Enter a name for the node.
- Click **OPC Server**.
- If the emulator is running on the same computer as RSView32, make sure **In-Process** is selected. Otherwise, select **Remote** and enter the name of the computer running the emulator in the **Server Computer Name or Address** field.
- Click this button. If the emulator is running on the same computer as RSView32, select **RSLinx OPC Server**. Otherwise, select **RSLinx Remote OPC Server**.
- Set the **Update Rate** to the frequency at which you want data to update your RSView32 project.



- Click **Accept**. This saves the node configuration.



## Adding Tags from Your Logic Project to Your RSView32 Project

Once you have created an OPC node for the emulator, you can add tags from your logic project to your RSView32 tag database. To do this:

1. Open your RSView32 project.
2. In your project's **System** folder, open the **Tag Database** editor.

3. *Select the data type for the tag.*

The image shows two overlapping windows from the RSView32 software. The top window is the 'Tag Database' editor, and the bottom window is the 'OPC Address Browser' dialog. Numbered callouts (1-7) point to specific elements in both windows to guide the user through the process of adding a tag.

**1. Click **Device**.** (Points to the 'Device' radio button in the 'Data Source' section of the Tag Database editor.)

**2. Click this button and select the OPC node you created for the emulator.** (Points to the button next to the 'Node Name' field in the Tag Database editor.)

**3. Select the data type for the tag.** (Points to the 'Type' dropdown menu in the Tag Database editor, which is set to 'Analog').

**4. Click this button to display the OPC Address Browser.** (Points to the button next to the 'Node Name' field in the Tag Database editor.)

**5. Browse for the address you want to use with the tag in this tree.** (Points to the tree view in the OPC Address Browser dialog.)

**6. Select the address.** (Points to the 'FlowIntoTank' address selected in the OPC Address Browser dialog.)

**7. Click **OK**.** (Points to the 'OK' button in the OPC Address Browser dialog.)

3. In the **Data Source** section of the Tag Database editor, click **Device**.
4. Click the button next to the Node Name field. This displays the Node Browser. Select the OPC node you created for the emulator and click **OK**.

5. In the **Tag** section of the Tag Database editor, select the type of tag you want to create from the **Type** list.
6. In the **Data Source** section of the Tag Database editor, click the button next to the **Address** field. This displays the OPC Address Browser.
7. In the OPC Address Browser, select the address you want to use with the tag and click **OK**.
8. Finish creating the tag by filling in the fields in the Tag Database editor.

<b>For this field or control:</b>	<b>Set or enter:</b>
Name	The name of the tag you want to create.
Security	If you have set up security for your RSView32 project, set the security class associated with the tag.
Description	Type a description of the tag (optional).
Minimum	The minimum value for the tag (as used in RSView32).
Maximum	The maximum value for the tag (as used in RSView32).
Scale	A scaling value for the tag.
Offset	A value to add to the tag.
Units	The units for the tag.
Data Type	Select the data type for the tag.

9. Click **Accept**. The tag is stored in the tag database. You can now use the tag in your RSView32 project.

# 7

## Simulating I/O

There are two methods of simulating I/O with RSLogix™ Emulate 5000. You can use a simulated I/O module to simulate discrete I/O, or you can use another emulation module to produce and consume tags of any type. You can use both methods simultaneously.



Additionally, you can use other software products such as RSView®32 and RSTestStand™ to simulate I/O.

### Configure Simulated I/O in the Logic Project

To be able to address simulated I/O in your logic project, you must create simulated I/O modules in the project. These modules are created just like any other I/O modules.



In RSLogix™ 5000, the simulated I/O module is called “1756-MODULE.” In RSLogix Emulate 5000, the simulated I/O module is called “1789-SIM.” For information about inserting a 1789-SIM module into the RSLogix Emulate 5000 Chassis Monitor, see *Creating Simulated I/O* on page 21.

To create a simulated I/O module in your RSLogix 5000 project:

1. In RSLogix 5000's Controller Organizer, right-click the **I/O Configuration** folder, and then click **New Module**. The software displays the Select Module window.
2. Open the **Other** folder. Select **1756-MODULE** from the modules list, then click **OK**.
3. The software displays the New Module window. Enter the following parameters for the module:

For this parameter:	Enter:
Name	A name for the module
Description (optional)	A description of the module and its function in the system
Comm Format	Set this to reflect the type of I/O you are simulating.
Slot	The slot in the virtual chassis that contains the module.

4. In the New Module window, specify the connection parameters for the module as shown in the table on the following page.



For read/write connections:		For listen only connections:	
Input Assembly Instance	1	Input Assembly Instance	1
Input Size	2	Input Size	2
Output Assembly Instance	2	Output Assembly Instance	3
Output Size	1	Output Size	1
Configuration Assembly Instance	16	Configuration Assembly Instance	16
Configuration Size	0	Configuration Size	0

- Click **OK**. The software displays another Module Properties window.
- Specify the Requested Packet Interval (RPI).



You **must** enter at least 50.0 ms for the RPI. The connection will fail if the RPI is less than 50 ms. Because this module uses the generic module profile, the default RPI is 5.0 ms. You must change this value.

- Click **OK**.

## Configuring Produced and Consumed Tags

Another method for simulating I/O for your emulated project is to produce and consume tags from another emulation module. (See the documentation for your Logix 5000 controller for background information about produced and consumed tags.)

### Configuring a Produced Tag

To configure a produced tag in the emulator:

- Open the logic project for the emulator in RSLogix 5000. You must be offline to perform this procedure.
- In the logic project, open the **Controller Tags** folder. (Produced tags must be scoped to the controller.)



3. Right-click the tag you want the emulator to produce, then click **Edit Tag Properties**. This displays the Tag Properties window.



A produced tag must be one of the following data types:

- DINT
- REAL
- an array of DINTs or REALs
- user-defined

A produced tag must also be less than 500 bytes.

4. In the Tag Properties window, select the **Produce** tag type.
5. Click **Connection**. The Produced Tag Connection dialog appears. Perform the following:
  - a. In the **Max. Consumers** field, set the number of consumers equal to the number of emulators that will consume the tag.
  - b. Click **OK**.
6. Click **OK**.

## Configuring a Consumed Tag

Configuring an emulator to consume a tag is a two-step process. First, you must make the emulator that is producing the tag an I/O module of the emulator that will consume the tag. Second, you must create a consumed tag to accept the data from the producer.

### MAKING THE PRODUCER AN I/O MODULE OF THE CONSUMER

To make the producing emulator an I/O module of the consuming emulator:

1. Open the logic project for the consuming emulator in RSLogix 5000. You must be offline to perform this procedure.
2. In the logic project, right-click the **I/O Configuration** folder, then click **New Module**. This displays the Select Module window.
3. In the Select Module window, open the Controllers folder. Select the **Emulator** module type (RSLogix Emulate 5000 Controller).
4. Click **OK**. This displays the Select Major Revision window.
5. In the Select Major Revision window, select the slot revision of the producing emulator.
6. Click **OK**. This displays the New Module window.

7. In the New Module window, type the name of the producing emulator in the **Name** field. Set the slot number for the producing emulator in the **Slot** field.
8. Click **OK**. This displays the Module Properties window.
9. On the Module Properties window, configure as necessary.
10. Click **OK**. The producing emulator is now an I/O module of the consuming emulator.

### CREATING A CONSUMED TAG

To create a tag that consumes data from the producing emulator:

1. In the logic project for the consuming emulator, create a tag (scoped either to the controller or to a program). You must be offline to perform this procedure.
2. Right-click the tag you want the emulator to consume, then click **Edit Tag Properties**. This displays the Tag Properties window.



A consumed tag must be of the same data type (including array size) as the produced tag it uses. If it is not, the emulator consuming the tag will experience an I/O fault.

3. In the Tag Properties window, select the **Consumed** tag type.
4. Click **Connection**. The Consumed Tag Connection dialog appears. Perform the following:
  - a. Select the producing emulator from the **Producer** drop-down list box.
  - b. Type the name of the tag that the producer is producing in the **Remote Data** field.
  - c. In the **RPI** field, set the Requested Packet Interval (RPI) to the amount of time (in milliseconds) you want the consuming emulator to wait between data requests for this tag. This is the minimum rate at which the emulator will consume data.
  - d. Click **OK**.
5. In the Tag Properties window, enter the data type for the consuming tag in the **Data Type** field. This must be the same data type (including array sizes) as the produced tag it consumes.
6. Click **OK**.

### Using RSView32 or RSTestStand to Simulate I/O

You can also simulate I/O by using other software packages, such as RSView32 or RSTestStand. RSView32 is an HMI design and runtime environment, and RSTestStand is a test stand simulation environment.

# 8

## Programming Breakpoints and Tracepoints

Breakpoint and tracepoint instructions allow you to monitor the state of your logic when it is in conditions that you determine. Breakpoints stop program emulation when a rung is true. Tracepoints log data you select when a rung is true.

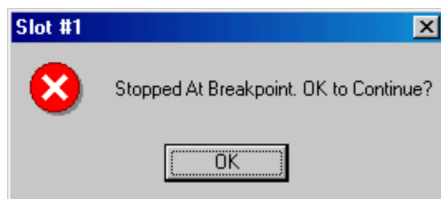


Breakpoint and tracepoint instructions are supported only in ladder diagram programs running in the emulator. These instructions are not supported by physical Logix5000™ controllers.

### Programming Breakpoints

Breakpoints are programmed with the Breakpoint output instruction (BPT). When the inputs on a rung containing a BPT instruction are true, the BPT instruction stops program execution. The software displays a window indicating that the breakpoint triggered and the values that triggered it.

When a breakpoint triggers, the emulator displays a window informing you that a breakpoint occurred. The title bar of the window shows the slot containing the emulator that encountered the breakpoint.



When you click **OK**, the emulator resumes program execution. If the conditions that triggered the breakpoint have not changed, the breakpoint will trigger again.

In addition, the emulator opens a trace window for the breakpoint. Information about the breakpoint and the values it traces are displayed in the trace window.



When a breakpoint triggers, you will not be able to edit your project until you permit execution to continue. You can go online with the emulator to observe the state of your project, but you will not be able to edit it. If you try to accept a rung edit while a breakpoint is triggered, you will see a dialog box saying the controller is not in the correct mode.

## Programming the BPT instruction

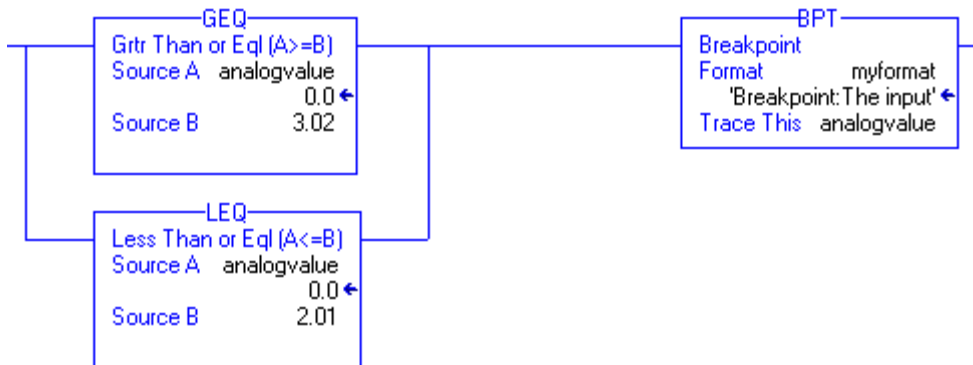
The Breakpoint (BPT) instruction has two kinds of parameters:

This parameter:	Contains:	Requires:
Format	A string that sets the formatting for the text that appears in the trace window for the breakpoint.	A STRING tag. See <i>String Formats for Tracepoint and Breakpoint Instructions</i> on page 48.
Trace This:	The tag that has a value you want to display in the trace window.	An INT, BOOL, SINT, DINT or REAL tag. You <b>cannot</b> trace tag structures (for example, entire timers or counters). However, you can trace individual structure members.

You can display a large number of tag values with the BPT instruction, however, the formatting string can contain only 82 characters. Because two characters are required in the formatting string for each tag you want in the breakpoint, the maximum number of tags you could trace with a single BPT instruction is 41. However, you will probably want to include spaces and other formatting to separate tag data in your traces, so the resulting number of tag values that can be effectively displayed by one BPT instruction is significantly lower than 41.

### EXAMPLE:

The following rung shows a breakpoint that stops program execution when an analog value is greater than 3.02 or less than 2.01.



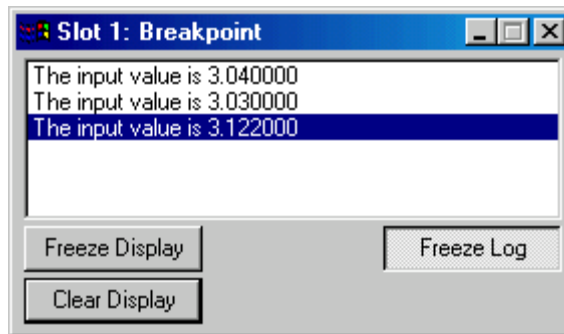




We want to display the breakpoint information in a particular format. That format is contained in the Format string (myformat). In this case, the format string contains the following text:

```
Breakpoint:The input value is %f
```

When the breakpoint triggers, the breakpoint trace window shows the characters before the colon (“Breakpoint”) in the title bar of the trace window. The other characters make up the traces. In this example, %f represents the first (and in this case, the only) tag to be traced (“analogvalue”). (See *String Formats for Tracepoint and Breakpoint Instructions* on page 48 for more information about formatting strings. The resulting traces appear as shown in the following illustration.



## Programming Tracepoints

Tracepoints are programmed with the tracepoint output instruction (TPT). When the inputs on a rung containing a TPT instruction are true, the TPT instruction writes a trace entry to a trace display or log file. (To configure how traces are displayed or logged, see *Configure Trace Displays* on page 20.)

## Programming the TPT Instruction

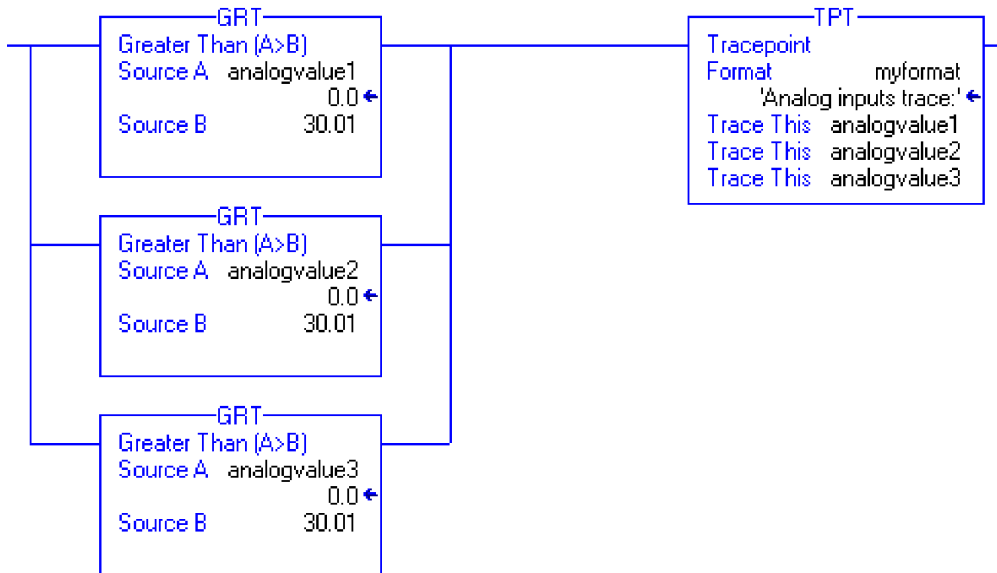
The tracepoint (TPT) instruction has two kinds of parameters:

<b>This parameter:</b>	<b>Contains:</b>	<b>Requires:</b>
Format	A string that sets the formatting for the trace reports (both on-screen and logged to disk).	A STRING tag. See <i>String Formats for Tracepoint and Breakpoint Instructions</i> on page 48.
Trace This:	The tag you want to trace.	An INT, BOOL, SINT, DINT or REAL tag. You <b>cannot</b> trace tag structures (for example, entire timers or counters). However, you can trace individual structure members.

You can trace a large number of tags with the TPT instruction, however, the formatting string can contain only 82 characters. Because two characters are required in the formatting string for each tag you want to trace, the maximum number of tags you could trace with a single TPT instruction is 41. However, you will probably want to include spaces and other formatting to separate tag data in your traces, so the resulting number of tags that can be effectively traced by one TPT instruction is significantly lower than 41.

**EXAMPLE:**

The following rung triggers a trace of three analog values when any one of them exceeds a given value (30.01).

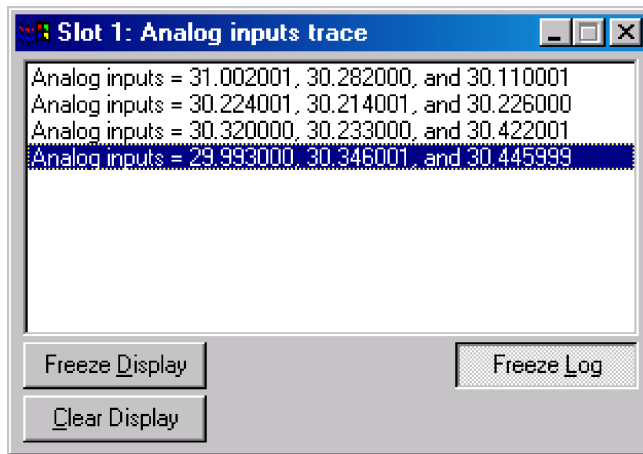


We want to display the tracepoint information in a particular format. That format is contained in the Format string (myformat). In this case, the format string contains the following text:

```
Analog inputs trace:Analog inputs = %f, %f, and %f
```

When the tracepoint triggers, the characters before the colon (“Analog inputs trace”) appear in the title bar of the trace window. The other characters make up the traces. In this example, the “%f” characters represent the tags to be traced (“analogvalue1,” “analogvalue2,” and “analogvalue3”). (See *String Formats for Tracepoint and Breakpoint Instructions* on page 48 for more information about formatting strings.

The resulting traces appear as shown in the following illustration.



When this trace is logged to disk, the characters before the colon appear in the traces. This indicates which tracepoint caused which trace entry. The following is an example of a trace entry. “Analog inputs trace:” is the heading text from the tracepoint’s format string.

```
Analog inputs trace:Analog inputs = 31.00201, 30.282000, and 30.110001
```

## String Formats for Tracepoint and Breakpoint Instructions

The Format string in the tracepoint and breakpoint instructions allows you to control how the traced tags appear in the traces or breakpoint windows. The format of the string is as follows:

```
heading: (text)%(type)
```

where *heading* is a text string identifying the tracepoint or breakpoint, *text* is a string describing the tag (or any other text you choose), and *%(type)* indicates the format of the tag. You need one type indicator for each tag you are tracing with the tracepoint or breakpoint instruction.

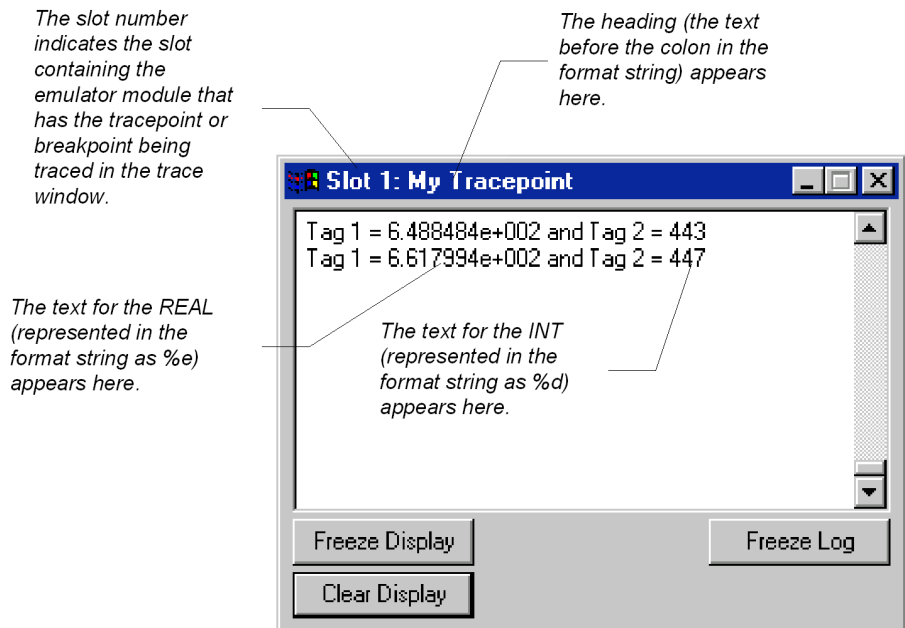
For example, you could format a tracepoint string like this:

```
My tracepoint:Tag 1 = %e and Tag 2 = %d
```

The `%e` formats the first traced tag as double-precision float with an exponent, and `%d` formats the second traced tag as a signed decimal integer. See the following table for more information.

In this case, you would have a tracepoint instruction that has two Trace This parameters (one for a REAL and one for an INT, although the value of any tag can be formatted with any flag).

The resulting tracepoint window that would appear when the tracepoint is triggered would look like:



## Types for Formatting Tag Values

The following table shows the available types for formatting tag values. Any type can be used for formatting any tag. (The types are similar to those used by the C++ print function.)

<b>This type:</b>	<b>Formats the tag value as:</b>	<b>Example:</b>
%d	Signed decimal integer. REAL values are truncated.	45.3202 is rendered as 45.
%o	Unsigned octal integer. Non-octal values are converted to octal. REAL values are truncated.	45.3202 is rendered as 55.
%x	Unsigned hexadecimal integer with letters rendered in lowercase. Non-hex values are converted to hex. REAL values are truncated.	45.3202 is rendered as 2d.
%X	Unsigned hexadecimal integer with letters rendered in uppercase. Non-hex values are converted to hex. REAL values are truncated.	45.3202 is rendered as 2D.
%e	Signed floating point value with exponent. The exponent delimiter (e) is rendered in lowercase.	45.320200 is rendered as 4.532020e+001.
%E	Signed floating point value with exponent. The exponent delimiter (E) is rendered in uppercase.	45.320200 is rendered as 4.532020E+001.
%f	Signed floating point value without exponent.	45.320200 is rendered as 45.320200.
%g	Signed floating point value in “%e” or “%f” format, whichever results in a more compact display. Trailing zeros are truncated, and the decimal point appears only if one or more digits follow it. The exponent delimiter (e) is rendered in lowercase (when it is appropriate).	45.320200 is rendered as 45.3202. 433222556.34322 is rendered as 4.3323e+008.
%G	Signed floating point value in “%e” or “%f” format, whichever results in a more compact display. Trailing zeros are truncated, and the decimal point appears only if one or more digits follow it. The exponent delimiter (E) is rendered in uppercase (when it is appropriate).	45.320200 is rendered as 45.3202. 433222556.34322 is rendered as 4.3323E+008.



## Using Trace Windows

### Opening and Closing Trace Windows

By default, tracepoint and breakpoint windows display automatically when a tracepoint or breakpoint instruction first executes. (You can configure the emulator to not display these windows automatically. See *Configure Trace Displays* on page 20 for more information about configuring the behavior of trace displays.)

If you close a tracepoint or breakpoint window, it is still available to you. You can access active traces through the Configure Emulation window (right-click the emulator, then click **Configure Emulation**). The Configure Emulation window allows you to perform the following operations on trace windows:

<b>This setting:</b>	<b>Does this:</b>
Ignore If No Value Changed	This button toggles whether the trace displays should record every trace or only those traces where there are changes to the value being traced. If the button is indented, the trace displays will ignore those traces where the value does not change.
TimeStamp	Set this to record how you want time recorded in traces. (If you have a time stamp mode selected and you change it while a trace window is displays, subsequent traces use the new time stamp mode.) <b>None</b> records no time information in the trace displays. <b>Delta</b> records the difference between the current trace and the last trace for that particular trace window. The first trace for the window always shows 0 milliseconds. <b>Absolute</b> records the time that the trace triggered. The time is in the number of milliseconds since the emulator was last changed to the Run mode.
Display all new traces	Check this box if you want all new traces to display automatically in a trace window.
Log new traces to disk	Check this box if you want new traces to automatically log to disk.
Display All Traces	Click this button to display all of the current traces in their trace windows. (The current traces are listed by name in the Configure Emulation window.)
Close All Traces	Click this button to close all of the open trace windows.
Remove All Traces	Click this button to remove all of the current traces from the trace list.
Close Log File	Click this button to stop logging traces to a file.
Trace Log File	This field shows the currently set log file. To set the log file, click <b>Browse</b> . For more information, see <i>Log Traces to Disk</i> on page 21.

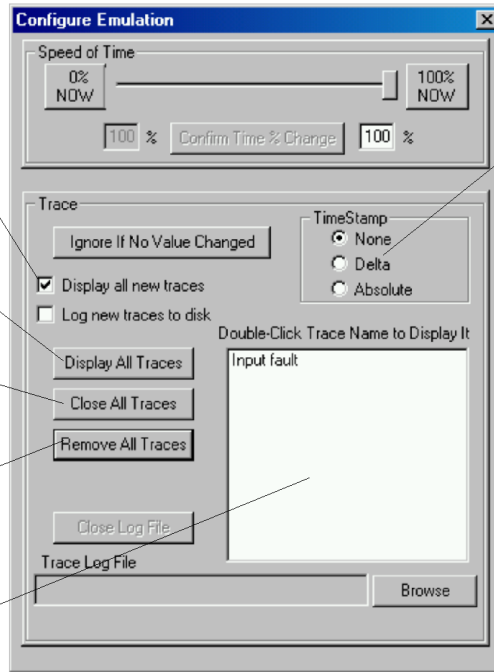
Check **Display all new traces** to automatically display trace windows when new tracepoints or breakpoints execute.

Click **Display All Traces** to open all of the active traces.

Clicking **Close All Traces** closes all open trace windows.

Click **Remove All Traces** to remove all active traces.

Active traces appear in this list – double-click a trace to display it.



Use these buttons to select a timestamp mode for the trace.

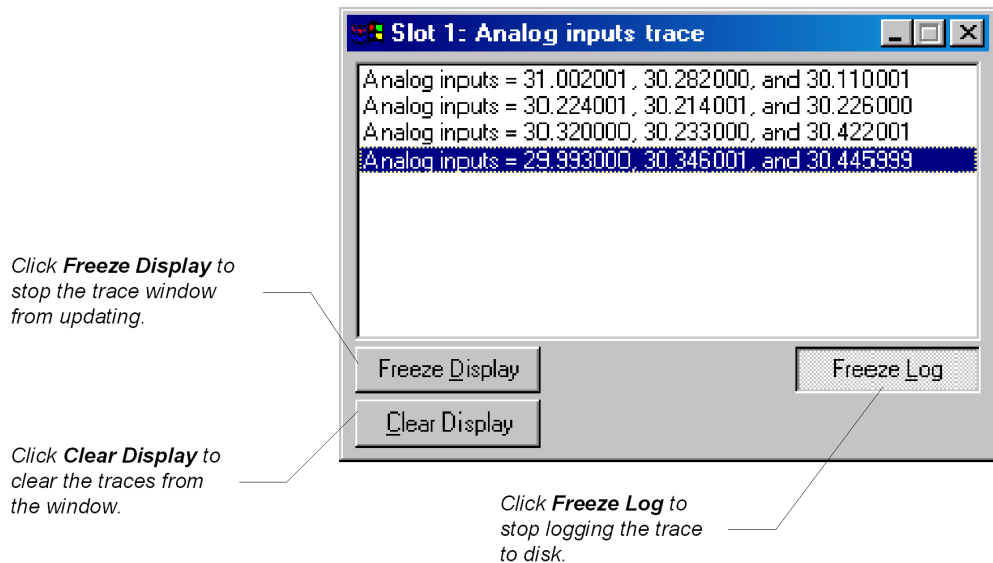




## Trace Window Controls

Trace windows have several controls that allow you to control their behavior.

This button:	Does this:
Freeze Display	Stops the trace window from updating. Use this when you want to read values from a rapidly updating trace.
Clear Display	Clears traces from the trace window.
Freeze Log	When you are logging a trace to disk, this button stops and starts logging. When the button is “pressed,” logging stops. If you are not logging the trace to disk, this button has no effect.



- 
- 
- GETTING RESULTS WITH RSLOGIX™ EMULATE 5000
- 
-

# 9

## Specifying simulation options for multiple controllers

In some cases, it may be beneficial to emulate multiple controllers in RSLogix™ Emulate 5000 at the same time to fully represent your application. With RSLogix Emulate 5000, you can:

- Change the mode for multiple controllers
- Save the current state of multiple controllers
- Restore the previously saved state for multiple controllers

### Change the mode for multiple controllers

If you have multiple RSLogix Emulate 5000 controllers within a single Chassis Monitor, you can change the mode of one or more controllers at one time.

#### Change one or more controllers to Run mode

Perform the following steps:

1. Click **All Modules > Run**. The Run Mode dialog appears.



By default, all slots that contain RSLogix 5000 Emulator controllers in the Chassis Monitor are shown and selected.

2. Deselect the slots that contain controllers that you don't want to place in Run mode.
3. Click **OK**. All controllers in the specified slots transition to Run mode.

#### Change one or more controllers to Program mode

Perform the following steps:

1. Click **All Modules > Program**. The Program Mode dialog appears.



By default, all slots that contain RSLogix 5000 Emulator controllers in the Chassis Monitor are shown and selected.

2. Deselect the slots that contain controllers that you don't want to place in Program mode.
3. Click **OK**. All controllers in the specified slots transition to Program mode.

## Save the current state of multiple controllers

When you have multiple RSLogix Emulate 5000 controllers within a single Chassis Monitor, you can create a snapshot and save the current state (including program and data information) for one or more controllers at one time. Perform the following steps:

1. Click **All Modules > Save Snapshot**. The Save Snapshot dialog appears.



By default, all slots that contain RSLogix 5000 Emulator controllers in the Chassis Monitor are shown and selected.

2. Deselect the slots that contain controllers that you don't want to include in a snapshot.



Any controllers that are configured with the Periodic Save feature enabled will be grayed out and cannot be selected.

3. In the **Directory** field, specify a complete directory path where you would like the snapshot to be saved. If necessary, click the browse icon to browse for an alternate directory path.

4. In the **Root name for files** field, type the filename that will contain the snapshot information.



The file will use the following naming convention: name\_Slotx.acd (where name is the name you provide, and x is the numeric slot number. For example, Sim1\_Slot 3. acd. The following characters are not supported: V?:\*?\"<>|.

5. Click **OK**. All controllers in the specified slots transition to Run mode.



If any of the selected controllers are in Program mode, you will be prompted to halt the processor (set the speed of time to 0%) prior to the save. After the save operation is completed, the controller will be returned to Run mode.

## Restore the previously saved state for multiple controllers

After saving a snapshot of the state of multiple RSLogix Emulate 5000 controllers within a single Chassis Monitor, you can later return those controllers to that previously saved state. Perform the following steps:

1. Click **All Modules > Restore**. The Restore from Snapshot dialog appears.



By default, slots represented by all of the files in the Available File Groups field are shown and selected. If a file for a particular slot exists, but that slot no longer contains a controller, the selection will be grayed out.



2. In the **Directory** field, specify a complete directory path to where the snapshot files are saved and available for restoration. If necessary, click the browse icon to browse for an alternate directory path.
3. In the **Available File Groups** field, select a single snapshot file.



The slots represented by the selected snapshot file are shown and selected.

4. Deselect the slots that contain controllers that you don't want to return to a previous state from the selected snapshot file.
5. Click **OK**. All controllers in the specified slots transition to Run mode.

The selected file group(s) are copied to the persistent file list used at start-up. RSLogix Emulate 5000 will then shutdown, and then restart. All of selected files are automatically loaded into the appropriate controllers.

- 
- 
- GETTING RESULTS WITH RSLOGIX™ EMULATE 5000
- 
-

# Index

## Numerics

- 1756-MODULE ■ 21, 39
- 1789-SIM module ■ 21, 39

## B

BPT. See *breakpoints*.

breakpoints

- example ■ 44
- format string ■ 44
- programming ■ 43, 44
- string formats for ■ 48
- traced tags ■ 44
- types for the format string ■ 50
- when triggered ■ 43

## C

calculations

- floating point ■ 11

Channel 0 serial port ■ 17

Chassis Monitor

- configuring ■ 23
- navigating in ■ 13
- starting ■ 13

clear display ■ 53

communications

- driver ■ 31

node

- configuring in RSView32 ■ 35
- to FactoryTalk View ME or FactoryTalk View SE ■ 25
- to remote computer ■ 34
- to RSLogix 5000 ■ 31
- to RSView32 ■ 35

comparison of Emulate 5000 and Logix5000 controllers ■ 9

Configuration Size ■ 40

Configure Emulation window ■ 18

configuring an emulator ■ 18

consumed tags ■ 22

continuous task dwell time ■ 17

CPU affinity ■ 17

## D

dwell time ■ 17

## E

emulation module

- creating ■ 15

emulation modules ■ 9

emulator

- configuring ■ 18

## F

FactoryTalk Activation Manager

- installing ■ 3

FactoryTalk Batch

- remote communications to ■ 34

FactoryTalk Directory

- configuring for use with FactoryTalk View SE or FactoryTalk View ME ■ 25

FactoryTalk View ME

- browsing for emulator tags ■ 28
- connecting the emulator to ■ 25

FactoryTalk View SE

- browsing for emulator tags ■ 28
- connecting the emulator to ■ 25

floating point ■ 11

folder

- default installation ■ 5

freeze display ■ 53

freeze log ■ 53

## H

- hardware requirements ■ 1
- help
  - accessing ■ 7

## I

- I/O
  - Configuration Assembly Instance ■ 40
  - Input Assembly Instance ■ 40
  - Input Size ■ 40
  - Output Assembly Instance ■ 40
  - Output Size ■ 40
  - simulating ■ 39
- ignore if no value changed ■ 20, 51
- installation
  - default folder ■ 5
  - procedure ■ 4
  - process ■ 1
  - required software and hardware ■ 1
- installing ■ 3

## J

- JSR instruction ■ 10
- JXR instruction ■ 10

## L

- license agreement ■ 5
- limitations of the software ■ 9
- log
  - freezing and unfreezing ■ 53
- log traces to disk ■ 21
- Logix5000 controllers
  - compared to Emulate 5000 ■ 9

## M

- memory size ■ 17
- mode
  - program ■ 23
  - remote ■ 23
  - run ■ 23
  - single scan ■ 24

## modes

- using ■ 23

## module

- 1756-MODULE ■ 21, 39
- 1789-SIM ■ 21, 39
- creating emulator ■ 15
- RSLinx Enterprise ■ 26
- multiple controllers
  - changing the mode ■ 55
  - restoring previously saved state ■ 56
  - saving the current state ■ 56

## N

### node

- configuring in RSView32 ■ 35

## O

- OLE for Process Control. See *OPC*.
- OPC ■ 25, 35
  - configuring topic for emulator ■ 35
- operating system
  - version required ■ 2
- overview ■ 9
  - of using Emulate 5000 ■ 11

## P

- periodic save interval ■ 17
- produced tags ■ 22
- program mode ■ 23

## R

- Release notes ■ 5
- remote mode ■ 23
- RSLinx Classic ■ 3
  - creating communication driver for emulator
    - 31
  - using with RSLinx Classic Gateway ■ 34
  - version required ■ 2
- RSLinx Classic Gateway ■ 34
- RSLinx Enterprise ■ 25
  - configuring server in FactoryTalk View SE
    - or FactoryTalk View ME ■ 27
  - module in virtual chassis ■ 26



- RSLogix 5000
    - configuring communications for the emulator ■ 33
    - connecting to the emulator ■ 31
    - remote communications to ■ 34
    - setting project to use the emulator ■ 32
    - version required ■ 2
  - RSTestStand
    - remote communications to ■ 34
    - using to simulate I/O ■ 39, 42
  - RSView32
    - communications node ■ 35
    - configuring tag ■ 37
    - connecting the emulator to ■ 35
    - using to simulate I/O ■ 39, 42
  - run mode ■ 23
- S**
- serial number ■ 5
  - simulated I/O
    - configuring in a logic project ■ 39
    - creating ■ 21
  - simulating I/O ■ 39, 55, 56
  - single scan mode ■ 24
  - software requirements ■ 2
  - specifying simulation options
    - multiple controllers ■ 55
  - speed of time ■ 19
- T**
- tags
    - browsing in FactoryTalk View SE or FactoryTalk View ME ■ 28
    - configuring in RSView32 ■ 37
    - consumed ■ 22, 40
      - configuring ■ 41
    - produced ■ 22, 40
      - configuring ■ 40
  - technical support ■ 7
  - time
    - dwell ■ 17
    - speed of ■ 19
  - timeslice ■ 17
  - timestamp ■ 20, 51
  - TPT. *See tracepoints.*
  - trace
    - configuring display ■ 20
    - logging to disk ■ 21
    - windows
      - controls on ■ 53
  - trace log file ■ 20, 51
  - tracepoints
    - configuring display ■ 20
    - example ■ 47
    - format string ■ 46
    - programming ■ 46
    - string formats for ■ 48
    - traced tags ■ 46
    - types for the format string ■ 50
- U**
- uninstallation ■ 4
  - uninstalling RSLogix Emulate 5000
    - considerations ■ 6
  - user name ■ 5
  - using Emulate 5000 ■ 11
- V**
- version required for installation
    - operating system ■ 2
    - Windows ■ 2
- W**
- Web site
    - technical support ■ 7
  - Windows
    - version required ■ 2

•  
•  
• GETTING RESULTS WITH RSLOGIX™ EMULATE 5000  
•  
•