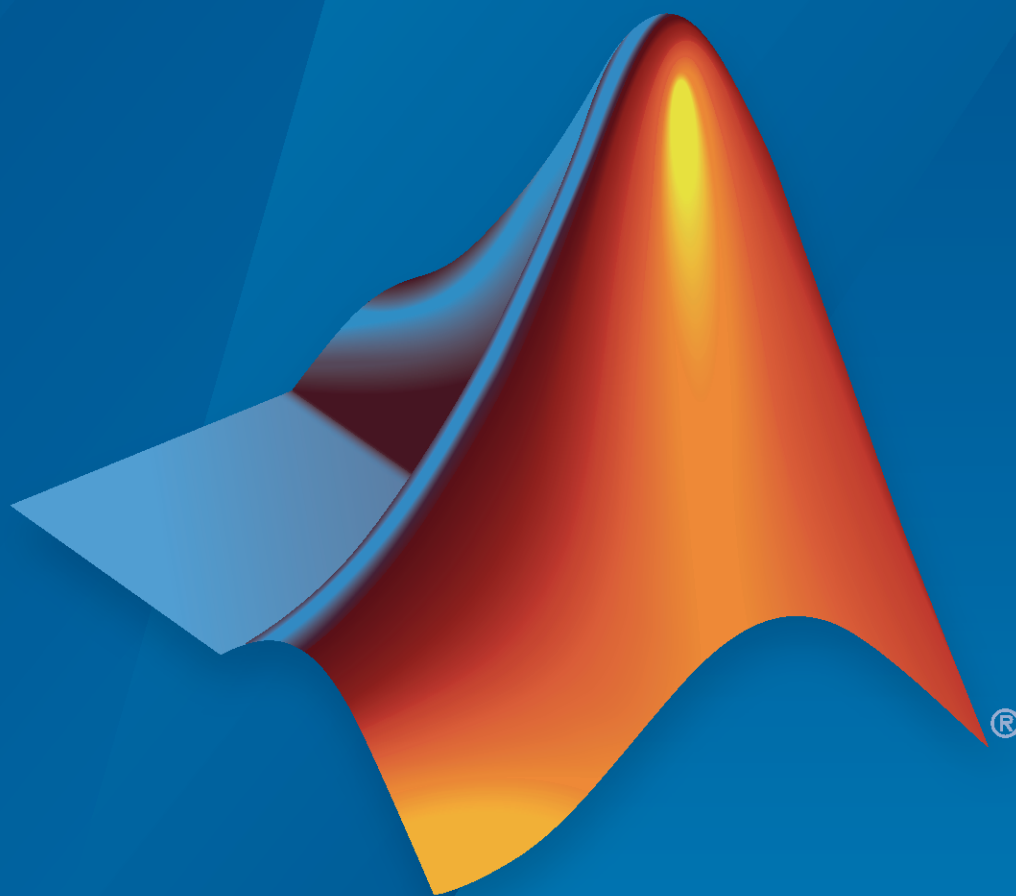


Simulink® Real-Time™

Getting Started Guide



MATLAB® & SIMULINK®

R2024b



How to Contact MathWorks



Latest news: www.mathworks.com

Sales and services: www.mathworks.com/sales_and_services

User community: www.mathworks.com/matlabcentral

Technical support: www.mathworks.com/support/contact_us



Phone: 508-647-7000



The MathWorks, Inc.
1 Apple Hill Drive
Natick, MA 01760-2098

Simulink® Real-Time™ Getting Started Guide

© COPYRIGHT 2000–2024 by The MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Patents

MathWorks products are protected by one or more U.S. patents. Please see www.mathworks.com/patents for more information.

Revision History

November 2000	First printing	New for Version 1 (Release 12)
June 2001	Online only	Revised for Version 1.2 (Release 12.1)
September 2001	Online only	Revised for Version 1.3 (Release 12.1+)
July 2002	Second printing	Revised for Version 2 (Release 13)
September 2003	Online only	Revised for Version 2.0.1 (Release 13SP1)
June 2004	Third printing	Revised for Version 2.5 (Release 14)
August 2004	Online only	Revised for Version 2.6 (Release 14+)
October 2004	Fourth printing	Revised for Version 2.6.1 (Release 14SP1)
November 2004	Online only	Revised for Version 2.7 (Release 14SP1+)
March 2005	Online only	Revised for Version 2.7.2 (Release 14SP2)
September 2005	Online only	Revised for Version 2.8 (Release 14SP3)
March 2006	Online only	Revised for Version 2.9 (Release 2006a)
May 2006	Fifth printing	Revised for Version 3.0 (Release 2006a+)
September 2006	Online only	Revised for Version 3.1 (Release 2006b)
March 2007	Online only	Revised for Version 3.2 (Release 2007a)
September 2007	Online only	Revised for Version 3.3 (Release 2007b)
March 2008	Online only	Revised for Version 3.4 (Release 2008a)
October 2008	Sixth printing	Revised for Version 4.0 (Release 2008b)
March 2009	Online only	Revised for Version 4.1 (Release 2009a)
September 2009	Online only	Revised for Version 4.2 (Release 2009b)
March 2010	Online only	Revised for Version 4.3 (Release 2010a)
September 2010	Seventh printing	Revised for Version 4.4 (Release 2010b)
April 2011	Online only	Revised for Version 5.0 (Release 2011a)
September 2011	Online only	Revised for Version 5.1 (Release 2011b)
March 2012	Online only	Revised for Version 5.2 (Release 2012a)
September 2012	Online only	Revised for Version 5.3 (Release 2012b)
March 2013	Online only	Revised for Version 5.4 (Release 2013a)
September 2013	Online only	Revised for Version 5.5 (Release 2013b)
March 2014	Online only	Revised for Version 6.0 (Release 2014a)
October 2014	Online only	Revised for Version 6.1 (Release 2014b)
March 2015	Online only	Revised for Version 6.2 (Release 2015a)
September 2015	Online only	Revised for Version 6.3 (Release 2015b)
March 2016	Online only	Revised for Version 6.4 (Release 2016a)
September 2016	Online only	Revised for Version 6.5 (Release 2016b)
March 2017	Online only	Revised for Version 6.6 (Release 2017a)
September 2017	Online only	Revised for Version 6.7 (Release 2017b)
March 2018	Online only	Revised for Version 6.8 (Release 2018a)
September 2018	Online only	Revised for Version 6.9 (Release 2018b)
March 2019	Online only	Revised for Version 6.10 (Release 2019a)
September 2019	Online only	Revised for Version 6.11 (Release 2019b)
March 2020	Online only	Revised for Version 6.12 (Release 2020a)
September 2020	Online only	Revised for Version 7.0 (Release 2020b)
March 2021	Online only	Revised for Version 7.1 (Release 2021a)
September 2021	Online only	Revised for Version 7.2 (Release 2021b)
March 2022	Online only	Revised for Version 8.0 (Release 2022a)
September 2022	Online only	Revised for Version 8.1 (Release 2022b)
March 2023	Online only	Revised for Version 8.2 (Release 2023a)
September 2023	Online only	Revised for Version 23.2 (R2023b)
March 2024	Online only	Revised for Version 24.1 (R2024a)
September 2024	Online only	Revised for Version 24.2 (R2024b)

1

Introduction

Simulink Real-Time Product Description	1-2
Simulink Real-Time Workflows and Licensing	1-3
License Requirements for Development and Test Roles	1-3
Deployment Approaches	1-3
Simulink Real-Time Operation Modes	1-4
Interfaces to the Real-Time Application	1-5
Configure and Control Target Computer	1-6
Simulink Real-Time Command-Line Interface in MATLAB	1-7
Simulink External Mode Interface	1-8
Visualize Signals in Simulink Real-Time Models	1-9
Target Computer Command Line	1-10

2

Installation and Configuration

Development Computer Requirements	2-2
Communication with Target Computer	2-2
Security for Development-to-Target Computer Communication	2-2
Enable Development Computer Communication (Windows)	2-3
Enable Development Computer Communication (Linux)	2-4
Verify Physical Connection of Ethernet to Target Computer	2-4
Configure Ethernet Interface IP Address and Subnet	2-4
Ensure Communication to Target Computer Is Not Blocked	2-5
Verify Development Computer Can Communicate with Target Computer	2-5
Install Development Computer Software	2-7
License Requirements	2-7
Location of Files on Development Computer	2-7
Target Computer Settings	2-9

Real-Time Application and Target Computer Modes	2-10
Build and Deploy a Real-Time Application	2-12
Deploy and Configure a Startup Real-Time Application	2-13

Basic Workflows

3

Real-Time Simulation and Testing	3-2
Create Simulink or Stateflow Model	3-2
Simulate Model as Non-Real-Time Application	3-2
Configure Development and Target Computers	3-2
Prepare Model for Real-Time Execution	3-3
Connect and Restart Target Computer	3-3
Build and Download Real-Time Application	3-3
Visualize Signals	3-3
Tune Parameters	3-4
Prepare Regression and Stress Tests	3-4

Tutorial and Examples

4

Set Up and Configure Simulink Real-Time	4-2
Configure Link Between Development and Target Computers	4-2
Create and Run Real-Time Application from Simulink Model	4-4
Build and Download Real-Time Application by Using Run on Target	4-8
Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands	4-9
Apply Simulink Real-Time Model Template to Create Real-Time Application	4-13
Configure and Control Real-Time Application by Using Simulink Real-Time Explorer	4-15
Execute Real-Time Application	4-15
Change Stop Time and Other Application Options	4-15
Simulate Simulink Model by Using MATLAB Language	4-17
Prepare Real-Time Application by Using MATLAB Language	4-19
Execute Real-Time Application with MATLAB Language	4-20
Application and Driver Scripts	4-23
General Examples	4-23

Driver Examples	4-23
Edit Scripts	4-24

Introduction

- “ Simulink Real-Time Product Description” on page 1-2
- “Simulink Real-Time Workflows and Licensing” on page 1-3
- “Configure and Control Target Computer” on page 1-6
- “Simulink Real-Time Command-Line Interface in MATLAB” on page 1-7
- “Simulink External Mode Interface” on page 1-8
- “Visualize Signals in Simulink Real-Time Models” on page 1-9
- “Target Computer Command Line” on page 1-10

Simulink Real-Time Product Description

Perform rapid control prototyping and hardware-in-the-loop testing

Simulink Real-Time and Speedgoat® take you from simulation to rapid control prototyping (RCP) and hardware-in-the-loop (HIL) testing in a single click. The products connect to electronic control units and physical systems with MATLAB® and Simulink.

You can create, control, and instrument real-time applications that run on Speedgoat real-time target computers directly from your Simulink model or with the MATLAB API and App Designer. You can simulate and test control designs and the dynamics of electric motors, electric vehicles and powertrains, wind turbines, power converters, battery management systems, robots and manipulators, autonomous systems, and other devices.

Simulink Real-Time Workflows and Licensing

After you create a Simulink model of your control system and simulation shows the system response, it is useful to simulate the model in real-time by using real-world I/O.

Simulink Real-Time lets you build a real-time application from your model, run the application on a target computer in real time, and interface real-world I/O to the application.

There are several workflows that you can use for deploying the real-time application, controlling the application, and visualizing the operation of the real-time application.

License Requirements for Development and Test Roles

Not all engineers or operators who are involved in developing and testing real-time applications need a software license for Simulink Real-Time and related software. The license requirements vary depending on the activities that the engineer or technician performs. A typical real-time application development and test organization could have these roles and license needs:

- Development engineer

To develop Simulink Real-Time models and build real-time applications from the models, a development engineer needs MATLAB, Simulink, Simulink Real-Time, MATLAB Coder™, Simulink Coder, Simulink Real-Time Support Package, and Speedgoat I/O Blockset. With these products installed on a development computer and a connection to a target computer, the development engineer can create real-time applications and App Designer instrument panels to interface with the real-time applications.

- Test engineer

To work with real-time applications provided by a development engineer, a test engineer needs MATLAB, Simulink Real-Time, Simulink Real-Time Support Package, and Speedgoat I/O Blockset. With these products installed on a development computer and a connection to a target computer, the test engineer can load and run the real-time application, tune parameters in the application, retrieve and parse data logs from the application execution, and view signals in the Simulation Data Inspector. Also, the test engineer can develop MATLAB test scripts and create App Designer instrumentation apps that connect to the target computer.

Deployment Approaches

Deploying a real-time application consists of building a real-time application on the development computer and installing it on the target computer. There are several ways to deploy a real-time application:

- Simulink Editor — From the **Real-Time** tab, click **Run on Target**. This selection provides a one-click operation for a model build, a real-time application load onto the target computer, and an application start on the target computer. You can use the UI for step-by-step build, load, connect model, and start operation by selecting from the buttons available from **Run on Target**.
- Simulink Real-Time Explorer — If you have completed the model build, you can use the Simulink Real-Time Explorer to load the real-time application onto the target computer and start the application.
- MATLAB commands — To deploy a real-time application to the target computer, in the MATLAB Command Window, type:

```
tg = slrealtime;  
% creates target object tg for the default target computer  
  
connect(tg);  
% connects Simulink Real-Time to the target computer  
  
slbuild('model_name');  
% builds the real-time application from the model  
  
load(tg,'model_name');  
% installs the real-time application and loads  
% it on the target computer  
  
start(tg);  
% starts the real-time application run  
  
% let the application run until stop time expires or . . .  
  
stop(tg);  
% stops the real-time application run
```

After you deploy the real-time application to the target computer, you can configure the target computer to run the application as the startup application when you start the target computer. This configuration is useful when you use the target computer for standalone mode operation instead of using the target computer for connected mode operation. For more information, see “Deploy and Configure a Startup Real-Time Application” on page 2-13.

Simulink Real-Time Operation Modes

Simulink Real-Time has operating modes and connection options for the development computer and the target computer.

When you start the target computer, it boots the real-time operating system (RTOS). The target computer always boots in standalone mode. In this mode, the target computer is operating independently of the development computer. If you previously deployed a real-time application to the target computer and configured the target computer to run the application on startup, the target computer runs the startup application. In standalone mode, you can control the target computer by using the command line interface from the target computer keyboard (console) or by using an SSH utility (for example, PuTTY) from the development computer.

After you start the target computer, on the development computer, you can connect to the target computer by using:

- Simulink Editor **Real-Time** tab
- Simulink Real-Time Explorer
- MATLAB commands to create a target object and connect the target object to the target computer

Any of these methods put the target computer into connected mode operation. In connected mode, you can interact with the target computer by deploying real-time applications to the target computer and executing other operations.

Interfaces to the Real-Time Application

To control the real-time application on the target computer, there are a number of available interfaces. Each interface provides different types of support for controlling and interacting with the real-time application.

Interface to Real-Time Application	More Information
Simulink Editor — The Real-Time tab lets you operate the basic and more advanced Simulink Real-Time workflows. Working from left-to-right on this tab, you can select and connect to a target computer, and then use the one-click operation of the Run on Target button to build, deploy, and run the real-time application. For more advanced workflows such as parameter tuning, the Signal Table selection in the Prepare section of this tab lets you interact with the real-time application when simulating in external mode.	“Build and Download Real-Time Application by Using Run on Target” on page 4-8
Simulink Real-Time Explorer — The explorer lets you load the real-time application onto the target computer and start the application. For more advanced workflows such as parameter tuning, on the Signals tab and Parameters tab, you can interact with the real-time application. The Simulink Real-Time Explorer provides a hierarchical display of application signals. You can select signals to stream to the Simulation Data Inspector for results visualization.	“Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands” on page 4-9
Use MATLAB commands — Simulink Real-Time provides a selection of MATLAB functions for deploying and controlling real-time applications. This command interface supports developing MATLAB scripts for frequently used operations and access to operations that are not available from the Simulink Editor Real-Time tab or from the Simulink Real-Time Explorer.	“Execute Real-Time Application with MATLAB Language” on page 4-20

Configure and Control Target Computer

The Simulink Real-Time environment has a modifiable interface to the Speedgoat target computer. You can use this interface from MATLAB or Simulink. Because of this open environment, there are several ways to interact with your real-time application from the MATLAB development computer and target computers.

This table compares the interfaces supported by the Simulink Real-Time product.

Interface	Environment Properties	Control	Signal Acquisition	Parameter Tuning	Runnable Outside MATLAB
Simulink Real-Time Explorer	Yes	Yes	Yes	Yes	Not applicable
"Simulink Real-Time Command-Line Interface in MATLAB" on page 1-7	Yes	Yes	Yes	Yes	Not applicable
"Simulink External Mode Interface" on page 1-8	Not applicable	Yes	Yes	Yes	Not applicable
"Visualize Signals in Simulink Real-Time Models" on page 1-9	Not applicable	Not applicable	Yes	Not applicable	Not applicable
"Target Computer Command Line" on page 1-10	Not applicable	Yes	Not applicable	Not applicable	Yes

See Also

More About

- Simulink Real-Time Explorer
- "Simulink Real-Time Command-Line Interface in MATLAB" on page 1-7
- "Simulink External Mode Interface" on page 1-8
- "Visualize Signals in Simulink Real-Time Models" on page 1-9
- "Target Computer Command Line" on page 1-10

Simulink Real-Time Command-Line Interface in MATLAB

You can interact with the Simulink Real-Time environment through the MATLAB command-line interface. On your development computer, enter Simulink Real-Time functions in the MATLAB Command Window. You can also write your own MATLAB scripts that use Simulink Real-Time functions for batch processing.

The Simulink Real-Time software has more than 50 MATLAB functions for controlling the real-time application from the MATLAB development computer. These functions define command-line operations that you can do in the Simulink Real-Time environment.

Complete your most common tasks by using the Simulink Real-Time UIs. Through the UIs, you have access to the Simulink Real-Time capabilities, but not their extended functionality. The command-line interface provides an interactive environment that you can extend.

The MATLAB command-line interface includes:

- **Environment** — Configure target computers and change the environment properties without using a graphical interface.
- **Control** — Restart the target computer, download a real-time application, start the real-time application, change stop time without regenerating code, and stop the real-time application.
- **Signal acquisition** — Transfer logged signal data to the MATLAB workspace by uploading data from the target computer to the Simulation Data Inspector on the development computer between runs. For more information, see `list`.
- **Parameter tuning** — Change tunable block parameters and tunable global parameters while the real-time application is running or change parameters in between runs.

See Also

More About

- “Set Up Target Computer Ethernet Connection”
- “Target Computer Update, Reboot, and Startup Application”
- “Execute Real-Time Application with MATLAB Language” on page 4-20
- “Tune Parameters by Using MATLAB Language”

Simulink External Mode Interface

To connect your Simulink model to your real-time application, use the procedure in “Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands” on page 4-9. The model becomes a user interface to the real-time application. By changing parameters in the Simulink blocks, you also change parameters in the real-time application.

The Simulink external mode interface includes:

- **Control** — Connect the Simulink block diagram to the real-time application and control real-time application start and stop.
- **Signal acquisition** — Establish a communication channel between your Simulink block diagram and your real-time application. The block diagram becomes a user interface to your real-time application. Simulink scopes can acquire signal data from the real-time application.
- **Parameter tuning** — Specify external mode and change parameters in the real-time application by changing parameters in the Block Parameters dialog boxes. When you change a value and click **OK**, the new value is downloaded to the target computer while the real-time application continues to run.

See Also

More About

- “Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands” on page 4-9
- “Stream Real-Time Signals by Using Simulink External Mode”
- “Tune Parameters by Using Simulink® External Mode”

Visualize Signals in Simulink Real-Time Models

When you use the external mode interface, you can view signal data from the simulation on Simulink Scope blocks in the model.

You can also use the Simulation Data Inspector to view signal data during the simulation for signals that you mark for logging in the Simulation Data Inspector.

To log signal data and view it in the Simulation Data Inspector after the simulation run, use the Simulink Real-Time File Log block.

See Also

File Log

More About

- “Real-Time Signal Logging and Streaming Basics”

Target Computer Command Line

You can interact with the Simulink Real-Time environment through the Speedgoat target computer command-line interface. On the target computer, enter commands at the command line by using the target computer keyboard or console. This interface is useful for working with standalone real-time applications that are not connected to the development computer.

The target computer command-line interface includes:

- Control — Start and stop the real-time application.
- Status — View the status of the real-time application.
- Status monitor screen — Swap target computer sessions and open the status monitor.

See Also

More About

- “Control Real-Time Application at Target Computer Command Line”
- “Target Computer Command-Line Interface”
- “Target Computer Status Monitor”

Installation and Configuration

- “Development Computer Requirements” on page 2-2
- “Enable Development Computer Communication (Windows)” on page 2-3
- “Enable Development Computer Communication (Linux)” on page 2-4
- “Install Development Computer Software” on page 2-7
- “Target Computer Settings” on page 2-9
- “Real-Time Application and Target Computer Modes” on page 2-10
- “Build and Deploy a Real-Time Application” on page 2-12
- “Deploy and Configure a Startup Real-Time Application” on page 2-13

Development Computer Requirements

To install and run Simulink Real-Time, MATLAB, Simulink, and other required and optional software, the development computer must be a 64-bit, PC-compatible system.

In the Simulink Real-Time software environment, the MATLAB development computer is usually your desktop computer. A notebook computer is also a viable development computer. For more information see Simulink Real-Time Product Requirements.

Communication with Target Computer

For communicating with the Speedgoat target computer, the development computer requires one available Ethernet interface (PCI or USB) connected to a network. Configure the development computer Ethernet port to use Internet Protocol Version 4 (TCP/IPv4) only. Specify a nonroutable static IP address of the form 192.168.x.x.

Security for Development-to-Target Computer Communication

Ensure that access to your development computer and the target computer is secure.

Provide a secure communication channel between your development computer and the real-time application on the target computer by applying the security measures listed in this table.

Communication Protocol	Security Measure
XCP on TCP/IP, TCP/IP	Run your development computer and target computer within a trusted private network or virtual private network (VPN).

See Also

More About

- “Enable Development Computer Communication (Windows)” on page 2-3
- “Install Development Computer Software” on page 2-7
- “Target Computer Settings” on page 2-9

Enable Development Computer Communication (Windows)

To communicate from your MATLAB development computer (Windows) to your Speedgoat target computer by using an Ethernet connection, perform a one-time configuration of your MATLAB development computer (Windows) Ethernet port to use Internet Protocol Version 4 (TCP/IPv4) only. Specify a nonroutable static IP address of the form 192.168.x.x.

- 1 On your development computer, from the **Control Panel**, navigate to **Network and Sharing Center**.
- 2 Click the **Local Area Connection** for the Ethernet adapter that you are using. It is often listed as Unidentified network.
- 3 Click the **Properties** button.
- 4 In the **Networking** pane, clear all check boxes.
- 5 Select the **Internet Protocol Version 4 (TCP/IPv4)** check box.
- 6 Click the **Properties** button.
- 7 To specify a static IP address of the form 192.168.x.x, select the **Use the following IP address** check box.
- 8 Configure the IP address boxes:

 IP address — 192.168.7.2

 Subnet mask — 255.255.255.0

 Default gateway — leave blank
- 9 Click **OK**.

Tip Do not set the development computer IP address to the gateway IP address 192.168.7.1 (notice the .1 at the end). Using the gateway IP address as the address of your development computer is not supported for Simulink Real-Time communications with your target computer.

See Also

More About

- “Development Computer Requirements” on page 2-2
- “Install Development Computer Software” on page 2-7
- “Troubleshoot Communication Failure Through Firewall (Windows)”

Enable Development Computer Communication (Linux)

In this section...

- “Verify Physical Connection of Ethernet to Target Computer” on page 2-4
- “Configure Ethernet Interface IP Address and Subnet” on page 2-4
- “Ensure Communication to Target Computer Is Not Blocked” on page 2-5
- “Verify Development Computer Can Communicate with Target Computer” on page 2-5

To communicate with the Speedgoat target computer by using Ethernet, perform a one-time configuration of your MATLAB development computer (Linux) Ethernet port to use a nonroutable static IP address of the form 192.168.x.x. The configuration process includes:

- 1 Verify the physical connection of Ethernet interface to the target computer.
- 2 Configure the Ethernet interface IP address and subnet by using an automated network manager service or configure it manually.
- 3 Ensure that communication to the target computer is not blocked by a firewall or rules.
- 4 Verify that the development computer can communicate with the target computer by using the ping command.

Verify Physical Connection of Ethernet to Target Computer

There are various schemes by which wired and wireless network interfaces are assigned names. These names are the underlying system labels such as `eth0` or `wlx800e1319c734`. The first step is to ensure you find the correct Ethernet interface. This step is dependent on the Linux distribution that is installed on your development computer. Often, you can list the available interfaces by using the `ls` command:

```
ls /sys/class/net  
  
enp6s0 lo vpn0 wlan0 eth0
```

In this example, `eth0` is the physical Ethernet interface to which the Ethernet cable to the target computer is connected. Verify that the cable for this interface on the development computer is connected to the corresponding interface on the target computer.

Configure Ethernet Interface IP Address and Subnet

Configure the Ethernet interface on the development computer to use an appropriate subnet either manually or by using a network manager service.

If configuring the Ethernet interface by using management daemon software, there are some considerations that depend on the Linux distribution. Based on the distribution installed on the development computer, there are automated methods or services that can manage the network interfaces. For example, you could use a management daemon software, such as NetworkManager (NM) (network-manager and associated packages) or Wicd (wicd and associated packages). These managers include UIs and commands. To ensure a static IP address is set and ensure that the development computer IP and target computer IP are in the same subnet, refer to the manager documentation manuals.

In all the distributions, if the interface is not managed by an automated service, you can configure the interface manually.

If configuring the interface manually, you can do the majority of network setup by editing the `interfaces` configuration file at `/etc/network/interfaces`. In this file, you can assign your network card an IP address or use `dhcp`, set up routing information, set default routes, and set other interface features. Be sure to add interfaces that are started at boot time to the `'auto'` line.

To manually modify the `interfaces` configuration file and set up an interface named `eth0` for a development computer that uses IP address `192.168.7.2` and a default gateway `192.168.7.254`, add these lines to the file `/etc/network/interfaces`:

```
auto eth0
iface eth0 inet static
address 192.168.7.2/24
gateway 192.168.7.254
```

Tip Do not set the development computer IP address to the gateway IP address `192.168.7.1` (notice the `.1` at the end). Using the gateway IP address as the address of your development computer is not supported for Simulink Real-Time communications with your target computer.

Ensure Communication to Target Computer Is Not Blocked

By default, most Linux distributions do not have firewalls that block Ethernet communications to the target computer.

Ensure there is no firewall, such as `iptables` or `ufw`, that is configured to block the connection to the target computer. For example, the default Debian Linux installation comes with the program `iptables` that is configured to allow `all` traffic.

If your organization uses a firewall, add a rule to ensure network traffic to the target from the host is unaffected. For more information, see `nftables`.

Verify Development Computer Can Communicate with Target Computer

Verify that the development computer can communicate with the target computer by using the `ping` command. For a target computer with IP address `192.168.7.5`, in the MATLAB Command Window, type:

```
system('ping 192.168.7.5')
```

```
Pinging 192.168.7.5 with 32 bytes of data:
Reply from 192.168.7.5: bytes=32 time<1ms TTL=255
Reply from 192.168.7.5: bytes=32 time<1ms TTL=255
Reply from 192.168.7.5: bytes=32 time<1ms TTL=255
Reply from 192.168.7.5: bytes=32 time<1ms TTL=255
```

```
Ping statistics for 192.168.7.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

ans =

0

See Also

More About

- “Development Computer Requirements” on page 2-2
- “Install Development Computer Software” on page 2-7
- “Troubleshoot Communication Failure Through Firewall (Windows)”

Install Development Computer Software

You install the Simulink Real-Time software only on the MATLAB development computer. The development computer downloads the real-time application to the target computer at run time. The Simulink Real-Time software is distributed on a DVD or as a file that you download from the Internet.

Simulink Real-Time supports development workflows and test engineering workflows:

- **Development workflow** — To model real-time execution, build the real-time applications, download the application to a target computer, and run the applications, you set up your development computer with a development installation. This installation includes MATLAB, Simulink, Simulink Coder, MATLAB Coder, Simulink Real-Time, and Simulink Real-Time Target support package, Speedgoat I/O Blockset, and Stateflow® (recommended).
- **Test engineering workflow** — To just download and run real-time applications that you receive from a developer, you set up your development computer with a test installation. This installation includes MATLAB, Simulink Real-Time, and Speedgoat I/O Blockset.

For information about installing target computer software, see “Speedgoat Target Computers and Speedgoat Support”.

License Requirements

Before you install the Simulink Real-Time product, you must have a valid File Installation Key and License file. The File Installation Key identifies the products that you purchased from MathWorks®. The License file activates the installation.

If you have not received either of these items, go to the License Center at the MathWorks website. For a list of required products, go to Product Requirements & Platform Availability for Simulink Real-Time.

After you install Simulink Real-Time, if you are setting up a development installation, you must install the Simulink Real-Time Target Support package. During support package installation, you accept the license terms for the BlackBerry® QNX® Neutrino® development tools and run-time components as part of the target computer. These tools generate a real-time application that runs on the target computer.

Using Simulink Real-Time and BlackBerry QNX Neutrino, you can develop Simulink Real-Time applications that you run on a Speedgoat target machine (target computer). You can run these real-time applications by using the target computer in connected mode. In connected mode, the development and target computers are connected via Simulink Real-Time software and an Ethernet cable. You also can run these real-time applications by using the target computer in standalone mode. In standalone mode, the development and target computers are not connected via Simulink Real-Time software.

For more information about the connected mode and the standalone mode, see “Real-Time Application and Target Computer Modes” on page 2-10.

Location of Files on Development Computer

Select a working folder outside the MATLAB root. Simulink Real-Time generates files and folders in this location:

- MATLAB working folder — Simulink cache files (.slxc), dynamic link libraries (for example, .mex64), and Simulink Real-Time applications (.mldatax).
- Simulink Coder Build folder — The Simulink Coder C code files (.c, .cpp, .h, .hpp) are in a subfolder called *modelname_slrealtime_rtw*.

See Also

slbuild

More About

- “Development Computer Requirements” on page 2-2
- “Enable Development Computer Communication (Windows)” on page 2-3

Target Computer Settings

To run a Simulink Real-Time model on a Speedgoat target computer, configure the target settings to match the capabilities of the target computer.

- 1 Select the Simulink Real-Time template from the Simulink start page and create the exampleSlrtApp model. In the MATLAB Command Window, type:


```
simulink
```
- 2 In the Simulink Editor, open the Simulink Real-Time Explorer. In the **Real-Time** tab, from the targets list, click **SLRT Explorer**.
- 3 From the **Target Computers** list, select the target computer node.
 - a To add a node representing another target computer, in the **Targets** pane, click the **Add target** button.
 - b To remove a node representing a target computer, select the node and click the **Remove target** button.
 - c To connect to a target computer, toggle the **Disconnected** indicator to **Connected**.
- 4 If not previously configured, select the **Target Configuration** tab and set the target computer IP address for the target computer definition by using the **Change IP Address** button. For example, Speedgoat target machines are preconfigured to IP address 192.168.7.5 and subnet mask 255.255.255.0.
- 5 If your target machine has been used with a previous version of MATLAB and you have not yet migrated the target machine to release R2020b or later, you will see this message:

```
Please enter the following at the MATLAB command prompt
>> speedgoat.migrateTarget
before continuing with the next step.
```

Repeat this procedure as required for each target computer.

See Also

Simulink Real-Time Explorer

More About

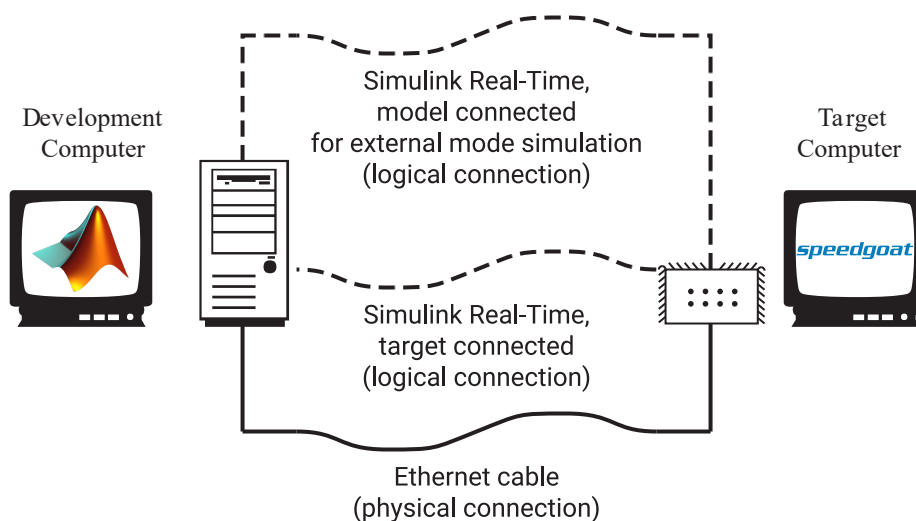
- “Real-Time Application and Target Computer Modes” on page 2-10

Real-Time Application and Target Computer Modes

At startup, the Speedgoat target machine (target computer) boots the QNX Neutrino RTOS. After the boot process is complete, you can deploy a real-time application to the target computer and run the application in one of these target computer modes:

- **Connected mode** — To put the target computer into connected mode, toggle the Disconnected / Connected selection on the Simulink Editor **Real-Time** tab. When the target computer is connected, the **Real-Time** tab buttons, Simulink Real-Time Explorer, and the Target Computer Manager provide interfaces to the target computer and real-time application. In the figure, the connected mode is represented by the physical connection between the development computer and target computer plus the logical connection between Simulink Real-Time software and the target computer.
- **External mode** — To run the real-time application into external mode, deploy the application to the target computer in the connected mode, and then on the Simulink Editor **Real-Time** tab select **Run on Target > Connect Model**. When the target computer is connected and the application is connected to the model, the model provides an interface to interact with the application. In the figure, the external mode is represented by the physical connection between the development computer and target computer plus the logical connection between Simulink Real-Time software and the target computer plus the logical connection between the model on the development computer and the real-time application on the target computer.
- **Standalone mode** — To operate the target computer in the standalone mode, control the real-time application by using the command-line interface from the target computer keyboard (console) or by using the command-line interface through an SSH utility (for example PuTTY) on the development computer. In the figure, the standalone mode is represented by the physical connection between the development computer and the target computer. You use this connection if controlling the target computer remotely instead of using the target computer keyboard.

Note Do not use the model interface and Simulink Real-Time Explorer at the same time to control the real-time application.



Physical Connections and Logical Connections	Simulink Real-Time Modes
Simulink Real-Time establishes a logical connection to target computer and a logical connection between the model on the development computer and the real-time application on the target computer. Simulink Real-Time uses the Ethernet cable for physical connection	The model on the development computer provides an interface to control the real-time application in model connected mode on the target computer in external mode simulation.
Simulink Real-Time establishes a logical connection to target computer and uses the Ethernet cable for physical connection.	The development computer and target computer operate in target connected mode . You can control the target computer by using the Simulink Real-Time Explorer or MATLAB commands.
An Ethernet cable provides physical connection between development computer and target computer.	The target computer operates in standalone mode . You can control the target computer by using the target computer keyboard (console) or by using PuTTY to send commands from a computer that does not have MATLAB installed.
There is no physical or logical connection between development computer and target computers.	The target computer operates in standalone mode . You can control the target computer by using the target computer keyboard (console).

See Also

More About

- “Create and Run Real-Time Application from Simulink Model” on page 4-4
- “Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands” on page 4-9
- “Control Real-Time Application at Target Computer Command Line”
- “Execute Target Computer RTOS Commands at Target Computer Command Line”
- “Target Computer Status Monitor”

Build and Deploy a Real-Time Application

After you build a real-time application, use Simulink Real-Time Explorer to deploy the application to a target computer and run the application in standalone mode on the target computer. For more information, see “Deploy and Configure a Startup Real-Time Application” on page 2-13.

- 1 Open an example Simulink model, for example, `slrt_ex_osc_rt`. In the Command Window, type:

```
openExample('slrealtime/SlrtCreateAndRunRealTimeAppFromSimulinkModelExample', ...  
    'supportingFile','slrt_ex_osc_rt.slx')
```
- 2 Build the model. In the Simulink Editor, on the **Real-Time** tab, click **Run on Target > Build Application**.
- 3 Open Simulink Real-Time Explorer. In the Simulink Editor, on the **Real-Time** tab, click **Prepare > SLRT Explorer**.
- 4 From the target computers list, select a target computer and connect to it.
- 5 To deploy the real-time application, click **Load Application**, use the **File Selector** to select the application, and click **Load**.

Simulink Real-Time loads the real-time application on the target computer. You can start the application by using Simulink Real-Time Explorer or by using the target computer command line.

See Also

Simulink Real-Time Explorer | `load` | `install`

Deploy and Configure a Startup Real-Time Application

After you build a real-time application `slrt_ex_osc_rt`, you can use Simulink Real-Time Explorer to deploy the application to a target computer and run the application in standalone mode on the target computer. For more information, see “Build and Deploy a Real-Time Application” on page 2-12.

- 1 Open Simulink Real-Time Explorer. In the Simulink Editor, on the **Real-Time** tab, click **Prepare > SLRT Explorer**.
- 2 From the target computers list, select a target computer and connect to it.
- 3 To deploy the real-time application `slrt_ex_osc_rt`, click **Load Application**, use the **File Selector** to select the `slrt_ex_osc_rt` application, and click **Load**.
- 4 To configure the real-time application to load and run each time the target computer starts, in the **Target Configuration** tab, select the **Startup App** checkbox for the application.
- 5 To restart the target computer and run the start-up application, click the **Reboot** button.

To remove the start-up application configuration, de-select the application **Startup App** checkbox.

See Also

Simulink Real-Time Explorer | `load` | `install` | `getStartupApp` | `setStartupApp` | `clearStartupApp`

Basic Workflows

Real-Time Simulation and Testing

In this section...

“Create Simulink or Stateflow Model” on page 3-2
“Simulate Model as Non-Real-Time Application” on page 3-2
“Configure Development and Target Computers” on page 3-2
“Prepare Model for Real-Time Execution” on page 3-3
“Connect and Restart Target Computer” on page 3-3
“Build and Download Real-Time Application” on page 3-3
“Visualize Signals” on page 3-3
“Tune Parameters” on page 3-4
“Prepare Regression and Stress Tests” on page 3-4

This workflow starts with real-time simulation and testing then leads from model creation through regression and stress testing on the Speedgoat target computer.

Create Simulink or Stateflow Model

- 1 In the Simulink Editor, on the **Simulation** tab, from **New**, select **Model**.
- 2 From the Simulink start page, select a model template or click **Blank Model**.
- 3 Create block diagrams in Simulink by dragging blocks to your model. If possible, set model and block parameters to use a fixed-step solver and specify a sample time compatible with the real-time requirements of your model.

Simulate Model as Non-Real-Time Application

Simulink uses a computed time vector to step the model. After computing the outputs for a given time value, Simulink immediately repeats the computation for the next time value until it reaches the stop time.

Because the computed time vector is not connected to a physical clock, the outputs are calculated as fast as your computer can run. The elapsed time of the simulation can differ significantly from the elapsed time of the real system.

You can log simulation results for later comparison.

Configure Development and Target Computers

- 1 Configure the communication method between the development and target computers.
- 2 Configure the MATLAB development computer and Speedgoat target computers by using:
 - Simulink Real-Time Explorer
 - “Simulink Real-Time Command-Line Interface in MATLAB” on page 1-7

Prepare Model for Real-Time Execution

- 1 Set the model Configuration Parameters to values compatible with real-time execution:
 - a On the **Apps** tab, click **Simulink Real-Time**. This selection sets the **System target file** model parameter to `speedgoat.tlc` (or another Simulink Real-Time STF) and sets the **Type** model parameter to **Fixed-step**.
 - b On the **Real-Time** tab, click **Hardware Settings**. In **Configuration Parameters > Solver > Fixed-step size**, select a step size that is compatible with the real-time requirements of your model.
 - c On the **Real-Time** tab, click **Hardware Settings**. Select options for **Configuration Parameters > Code Generation > Simulink Real-Time Options**. These settings set the initial values for real-time application options. If needed, you can change these options for the real-time application after building the application.
- 2 Add Simulink Real-Time I/O blocks representing your I/O boards to the model.
- 3 To visualize the simulation results, mark signals in the model for data logging with the Simulation Data Inspector. See “View Simulation Data in Simulation Data Inspector”.

Connect and Restart Target Computer

- 1 Connect the development computer to the Speedgoat target machine. Connect the IO Modules in the target computer to the physical hardware where applicable.
- 2 Restart the target computer with the Simulink Real-Time real-time RTOS by using:
 - Target computer restart button.
 - “Simulink Real-Time Command-Line Interface in MATLAB” on page 1-7

Build and Download Real-Time Application

- 1 On the **Real-Time** tab, toggle the **Toggle target computer connection** to **Connected** and click **Run on Target**. For more information about building and downloading the real-time application, see:
 - Simulink Real-Time Explorer
 - “Simulink Real-Time Command-Line Interface in MATLAB” on page 1-7
 - “Simulink External Mode Interface” on page 1-8
 - “Target Computer Command Line” on page 1-10
- 2 You also can build, deploy, and run the real-time applications by using in a step-by-step approach. For more information, see “Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands” on page 4-9.

Visualize Signals

- 1 To view signal data while the real-time application runs, mark signals for logging with the Simulation Data Inspector. Use the marked signals to acquire and display signal data from the real-time application. You can group hierarchical signals for streaming to the Simulation Data Inspector by using Simulink Real-Time Explorer.

- 2 To log signal data on the target computer and view the data after the real-time application runs, insert File Log blocks and connect these to the signals for logging. After the run, the data is loaded into Simulation Data Inspector for viewing.
- 3 To view signal data in a MATLAB instrument panel application, create an App Designer instrument panel application and connect the application to the real-time application by using a `Instrument` object. You insert graphical instruments into the instrument panels for data display.
- 4 Visualize signals by using:
 - Simulink Real-Time Explorer
 - “Simulink Real-Time Command-Line Interface in MATLAB” on page 1-7
 - “Simulink External Mode Interface” on page 1-8
 - “Visualize Signals in Simulink Real-Time Models” on page 1-9
 - “Target Computer Command Line” on page 1-10

Tune Parameters

- 1 Tune observable parameters such as time delays, input and output amplitudes, and input and output frequencies.

Note Simulink Real-Time does not support parameters of multiword data types.

- 2 Tune parameters by using:
 - Signal Table selection from Real-Time tab in Simulink Editor
 - Simulink Real-Time Explorer
 - “Simulink Real-Time Command-Line Interface in MATLAB” on page 1-7
 - “Simulink External Mode Interface” on page 1-8
 - “Target Computer Command Line” on page 1-10

Prepare Regression and Stress Tests

Write MATLAB scripts that perform parameter sweep and extreme-value testing in a repeatable manner, accumulating results as known good data.

See Also

Related Examples

- “Real-Time Signal Logging and Parameter Tuning”

More About

- “Configure and Control Target Computer” on page 1-6

Tutorial and Examples

This tutorial describes Simulink Real-Time functionality by using a simple Simulink model without I/O blocks. You can try these examples without I/O modules on the target computer. When you are familiar with the setup, build, and target execution process, you can try some of the more advanced Simulink Real-Time examples.

- “Set Up and Configure Simulink Real-Time” on page 4-2
- “Create and Run Real-Time Application from Simulink Model” on page 4-4
- “Build and Download Real-Time Application by Using Run on Target” on page 4-8
- “Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands” on page 4-9
- “Apply Simulink Real-Time Model Template to Create Real-Time Application” on page 4-13
- “Configure and Control Real-Time Application by Using Simulink Real-Time Explorer” on page 4-15
- “Simulate Simulink Model by Using MATLAB Language” on page 4-17
- “Prepare Real-Time Application by Using MATLAB Language” on page 4-19
- “Execute Real-Time Application with MATLAB Language” on page 4-20
- “Application and Driver Scripts” on page 4-23

Set Up and Configure Simulink Real-Time

For this tutorial, you must have this configuration:

- Simulink Real-Time software installed on the MATLAB development computer
- Simulink Real-Time Target Support package installed on the development computer
- Speedgoat target machine (target computer) connected to the development computer

Configure Link Between Development and Target Computers

To run a Simulink Real-Time model on a target computer, you must connect the development and target computers in a network.

The target network setting values are representative only. Consult your network administrator for actual values. For more information about network configuration, see “Target Computer Settings” on page 2-9.

- 1 Select the Simulink Real-Time template from the Simulink start page and create the `exampleSlrtApp` model. In the MATLAB Command Window, type:

`simulink`
- 2 In the Simulink Editor, open the Simulink Real-Time Explorer. In the **Real-Time** tab, from the targets list, click **SLRT Explorer**.
- 3 From the **Target Computers** list, select the target computer node.
 - a To add a node representing another target computer, in the **Targets** pane, click the **Add target** button.
 - b To remove a node representing a target computer, select the node and click the **Remove target** button.
- 4 If not previously configured, select the **Target Configuration** tab and set the target computer IP address for the target computer definition by using the **Change IP Address** button. For example, Speedgoat target machines are preconfigured to address `192.168.7.5`.
- 5 Check that you have started the target computer and wait until the target computer boots the RTOS.
- 6 To connect Simulink Real-Time to the target computer, click **Disconnected** to toggle to **Connected**.

Note If Simulink Real-Time reports a connection issue, contact MathWorks support. For more information, see the MathWorks Support website MathWorks Help Center website.

See Also

Targets

More About

- “Create and Run Real-Time Application from Simulink Model” on page 4-4

- “Troubleshoot Missing Real-Time Tab”

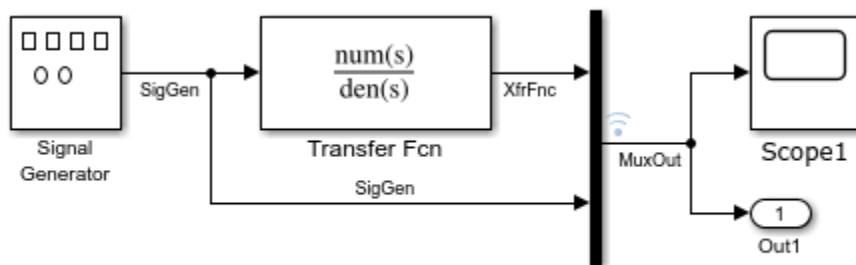
Create and Run Real-Time Application from Simulink Model

This tutorial begins with a non-real-time Simulink® model of a damped oscillator, `slrt_ex_osc_nrt`.

Open the Non-Real-Time Model

To open the model, in the MATLAB® Command Window, type:

```
model_nrt = 'slrt_ex_osc_nrt';
open_system(model_nrt);
```

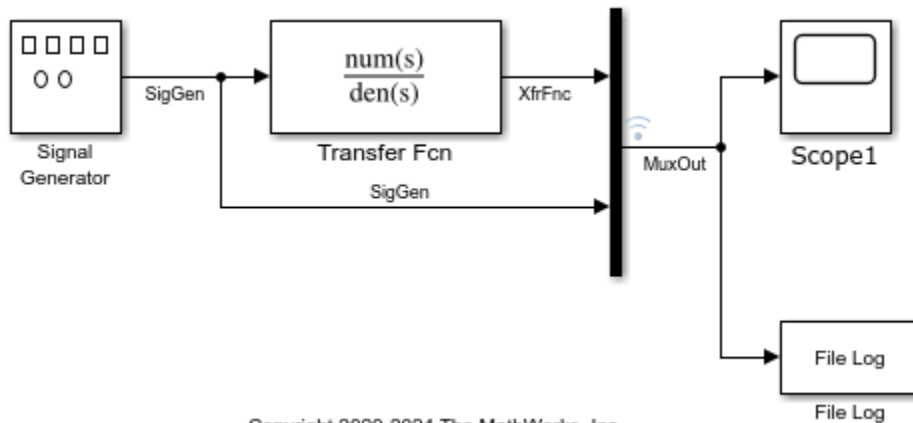


Copyright 2020-2024 The MathWorks, Inc.

To complete the tutorial, you must already know how to create, configure, and simulate such a model. As you go through the tutorial, you transform `slrt_ex_osc_nrt` into a Simulink Real-Time™ model configured to build as a real-time application. You then build, download, and execute the real-time application on your target computer.

The final model is named `slrt_ex_osc_rt`. To open the model, in the MATLAB Command Window, type:

```
model_rt = 'slrt_ex_osc_rt';
open_system(model_rt);
```



Copyright 2020-2024 The MathWorks, Inc.

Transform Simulink Model to Real-Time Application

Add blocks and set configuration parameters for running a Simulink model as a real-time application.

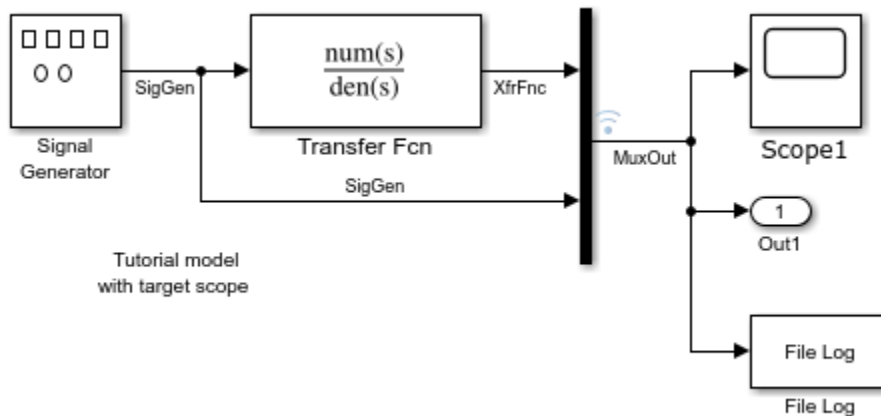
To run a Simulink model as a real-time application under Simulink Real-Time, add and configure a real-time File Log block and set configuration parameters for code generation and target execution.

Add Simulink Real-Time File Log Block

Simulink Real-Time supports a real-time File Log block. This tutorial uses the File Log block, which logs signal data for display in the Simulation Data Inspector after the simulation run.

- 1 In the MATLAB Command Window, type `slrt_ex_osc_nrt`. MATLAB loads the oscillator model and displays the Simulink block diagram.
- 2 In Simulink Editor, from the **Apps** tab, click **Simulink Real-Time**.
- 3 From the **Real-Time** tab **Prepare** section, click **Library Browser**.
- 4 In the Simulink left pane, browse to and double-click the node **Simulink Real-Time**.
- 5 From the list of I/O block categories, click node **Logging**.
- 6 Click and drag the File Log block to the Simulink block diagram. Simulink adds a new real-time File Log block to the model.
- 7 Connect the Mux output to the File Log block input.
- 8 On the **Simulation** tab, from the **Save** button, click **Save As**. Enter a file name. For example, enter `slrt_ex_osc_ucf`, and then click **OK**. To open the unconfigured model `slrt_ex_osc_ucf`, in the MATLAB Command Window, type:

```
model_ucf = 'slrt_ex_osc_ucf';
open_system(model_ucf);
```



Copyright 2020-2024 The MathWorks, Inc.

Set File Log Block Parameters

File Log block parameters define the decimation for logged signals. After the simulation run, the signals that you log by using this block are available in the Simulation Data Inspector. For this tutorial, configure a File Log block.

- 1 In the MATLAB Command Window, type `slrt_ex_osc_ucf`. MATLAB loads the oscillator model and displays the Simulink block diagram.

- 2 Double-click the File Log block.
- 3 In the **Decimation** text box, type 1. This value means that data is collected at each sample interval.
- 4 Click OK, and then from the **Simulation** tab, click **Save**.

Set Configuration Parameters

The example model, `slrt_ex_osc_ucf`, is a non-real-time model of a damped oscillator. You enter the simulation and real-time run parameters in the Configuration Parameters dialog box. These parameters provide information to Simulink Coder™ on how to build a real-time application from the Simulink model.

After you open a Simulink model and start the target computer, you can enter the simulation parameters.

- 1 In the MATLAB Command Window, type `slrt_ex_osc_ucf`.
- 2 In Simulink Editor, from the **Apps** tab, click **Simulink Real-Time**.
- 3 From the **Real-Time** tab, click **Hardware Settings**.
- 4 In the Configuration Parameters dialog box, click the **Solver** node.
- 5 In the **Solver selection** section, from the **Type** list, select **Fixed-step**.
- 6 From the **Solver** list, select a solver. For example, select the general-purpose solver `ode4` (Runge-Kutta).
- 7 Under **Solver details**, in the **Fixed-step size (fundamental sample time)** box, enter the sample time for the real-time application. For example, enter 0.00025 seconds (250 microseconds). After creating the real-time application, you can change this value. If you find that a value overloads the CPU on the target computer, try a larger **Fixed-step size** value, such as 0.0002 seconds. The sample times of the model blocks can be multiples only of **Fixed-step size**. If you enter 'auto' in **Fixed-step size**, the fundamental sample time is calculated from the sample times of the model blocks.
- 8 Click **OK**.
- 9 From the **Apps** tab, select **Simulink Real-Time**. This selection configures the system target file and other model settings for Simulink Real-Time.
- 10 In the **Simulation** tab, from the **Save** button, click **Save As**.
- 11 Enter a file name. For example, enter `slrt_ex_osc_rt`.

Start Target Computer and Display Status Monitor

Boot the RTOS on the target computer and display the status monitor.

Starting the target computer boots the RTOS. The target computer waits for the Simulink Real-Time software to download and start a real-time application from the development computer or to start a real-time application that was previously downloaded.

After you have configured the Simulink Real-Time product by using the Simulink Real-Time Explorer, you can start the target computer and connect the development computer to the target computer. Before building the real-time application, you start the target computer. After the build process is complete, you download the real-time application to the target computer.

The status monitor application on the target computer displays status of the real-time application and other target computer status information. The target computer display supports multiple sessions.

You can choose to display the status monitor (default, session 1) or display the target computer command line interface (session 2). To display the target computer status monitor:

- 1 Start the target computer.
- 2 The target computer displays session 1 (default) and the target computer status monitor.

To view the status monitor from the development computer, you can use PuTTY to open an SSH client and start the status monitor application `/usr/target/bin/statusmonitor` on the target computer.

For more information, see “Target Computer Status Monitor”.

Close Models

```
bdclose(model_nrt);  
bdclose(model_rt);  
bdclose(model_ucf);
```

See Also

Target | Mux | load

More About

- “System Configuration”
- “Set Up and Configure Simulink Real-Time” on page 4-2
- “Apply Simulink Real-Time Model Template to Create Real-Time Application”
- “Build and Download Real-Time Application by Using Run on Target” on page 4-8
- “Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands” on page 4-9
- “Troubleshoot Missing Real-Time Tab”

Build and Download Real-Time Application by Using Run on Target

The example model is a real-time model of a damped oscillator, `slrt_ex_osc_rt`. To open the model, in the MATLAB Command Window, type:

```
openExample('slrealtime/SlrtCreateAndRunRealTimeAppFromSimulinkModelExample', ...  
    'supportingFile','slrt_ex_osc_rt.slx')
```

To generate C code, compile, link, and download the real-time application to the target computer, use the Simulink Real-Time build process available from the Simulink Editor **Real-Time** tab.

After you enter changes in the Configuration Parameters dialog box, you can build the real-time application. For information about real-time application options, see “Simulink Real-Time Options Pane”.

- 1 Open the `slrt_ex_osc_rt` model.

MATLAB loads the oscillator model and displays the Simulink block diagram.

- 2 In the Simulink Editor, from the targets list on the **Real-Time** tab, select the target computer on which to run the real-time application.
- 3 Click **Run on Target**.

On the development computer, in the **Diagnostics Viewer**, you can view the build status. On the **Debug** tab, click **Diagnostics**. If the software detects an error during build and download, see “Troubleshooting in Simulink Real-Time”.

If you have a monitor connected to your target computer, the status monitor screen displays information about the application name. For more information, see “Target Computer Status Monitor”. You can also observe the event status of a `Target` object by installing a listener for object events.

See Also

Target | Mux | load

More About

- “System Configuration”
- “Set Up and Configure Simulink Real-Time” on page 4-2
- “Apply Simulink Real-Time Model Template to Create Real-Time Application”
- “Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands” on page 4-9
- “Troubleshoot Missing Real-Time Tab”

Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands

Control of the real-time application with Simulink is limited to connecting a Simulink model to a real-time application through external mode, and then starting the real-time application. Using Simulink external mode is one method to tune parameters.

The model connects to the target computer that you have selected in the Simulink Editor on the **Real-Time** tab.

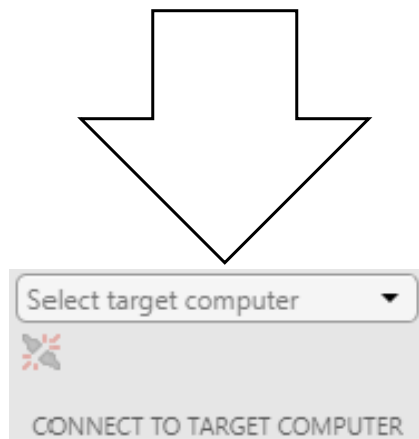
Note To avoid target computer communication issues, do not use Simulink external mode while Simulink Real-Time Explorer is running. Use only one interface.

After you build and download a real-time application to the target computer, you can run the real-time application. This procedure uses the Simulink model `slrt_ex_osc_rt`. To open the model, in the MATLAB Command Window, type:



```
openExample('slrealtime/SlrtCreateAndRunRealTimeAppFromSimulinkModelExample', ...
    'supportingFile','slrt_ex_osc_rt.slx')
```

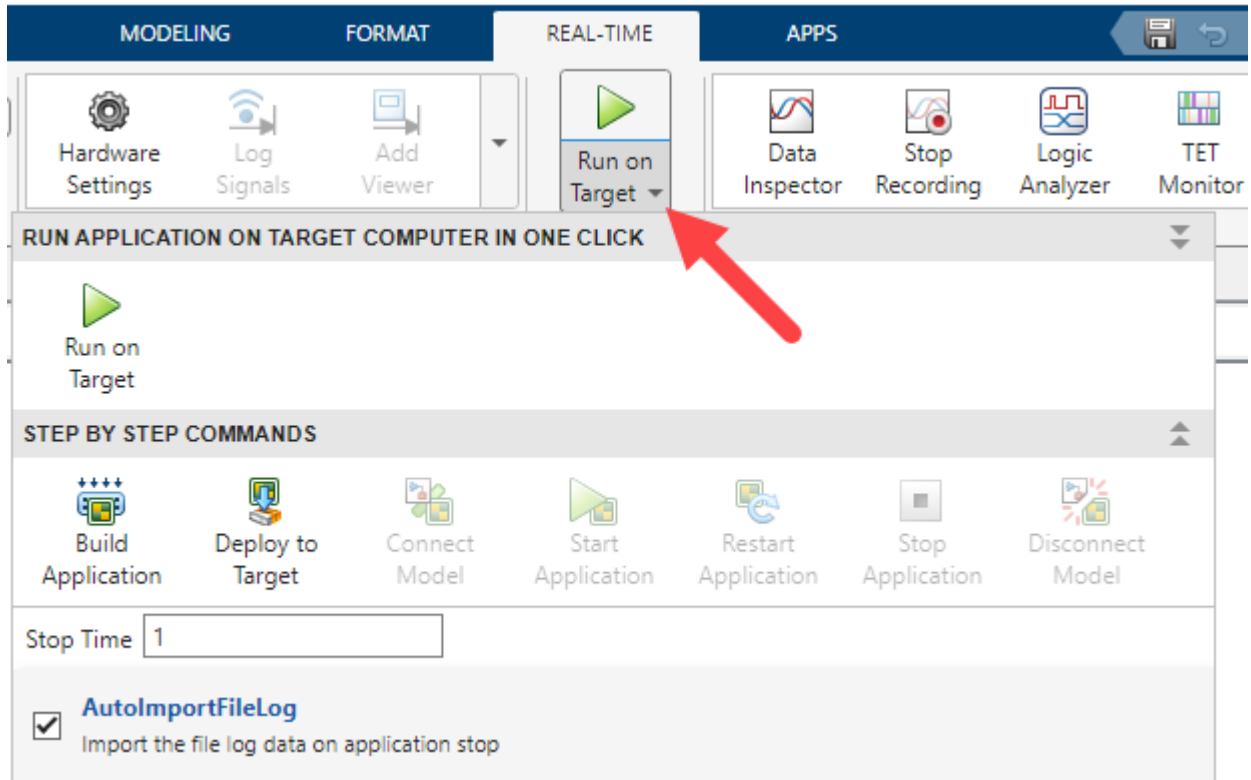
For more information about building applications, see “Build and Download Real-Time Application by Using Run on Target” on page 4-8.

You must already have specified the required target computer environment. Specifically, in the Simulink Editor, from the target computers list on the **Real-Time** tab, you must select the target computer to which you want to connect.



Select a target
to activate
Simulink Real-Time

- 1 In the Simulink Editor, connect to the target computer. In the **Real-Time** tab, click **Toggle target computer connection** . The connection status toggles to the connected state .
- 2 Click the **Run on Target** pull down.



- 3 To start the **Step by Step Commands**, select **Build Application**. This build operation produces the same real-time application MLDATX files as are produced by using the `slbuild` function.



- 4 After the model build completes, from the **Run on Target** pull down, select **Deploy to Target**.



- 5 After the real-time application deploys to the target, from the **Run on Target** pull down, select **Connect Model**.

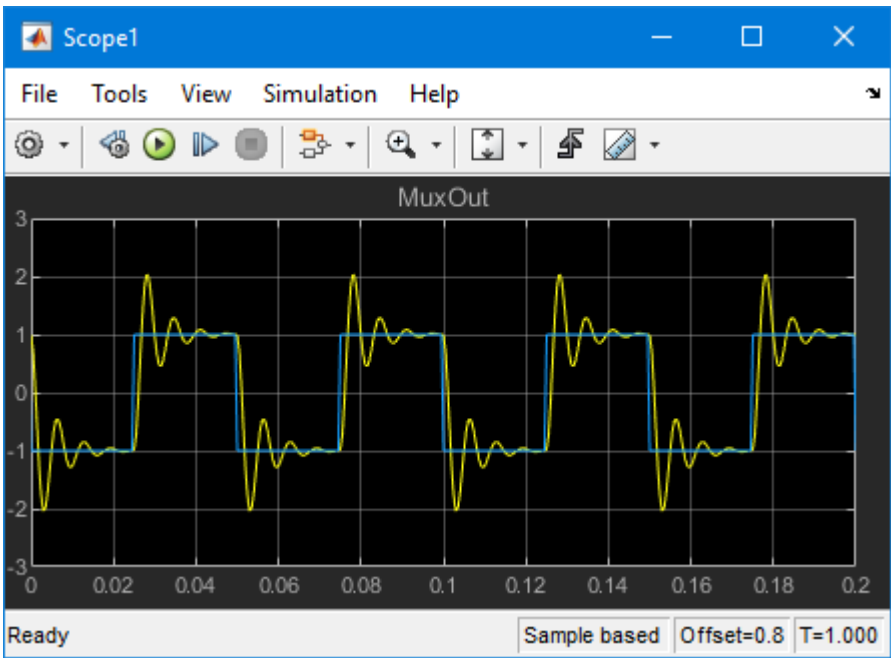


After you connect the model to the real-time application, the status bar displays External to indicate that the model is ready for external mode simulation. In this mode, the model serves as the interface to the application. This connection permits tuning parameters in the real-time application without rebuilding the application. This connection also permits observing signal activity in the real-time application by using, for example, Simulink Scope blocks in the model.

- 6 From the **Run on Target** pull down, select **Start Application**.






The current Simulink model parameters are downloaded from the development computer to the real-time application. The real-time application begins running and the Simulink Scope block displays signal data.




- 7 To stop the execution, from the **Run on Target** pull down, select **Stop Application**.



When the real-time application is deployed to the target computer, the step-by-step commands provide additional workflow options.

Commands	Step-by-Step Operations
	In the Step by Step Commands , after you deploy the real-time application to the target computer, you can connect the model to the real-time application. The Connect Model operation enables you to use the model as an interface to the real-time application for external mode simulation.
	In the Step by Step Commands , after you connect the model to the real-time application on the target computer, you can disconnect the model from the real-time application. The Disconnect Model operation enables you to keep the real-time simulation running on the target computer and reconnect to the model at a later time.
	<p>In the Step by Step Commands, after you deploy the real-time application to the target computer and connect the model to the real-time application, you can start the real-time application. The Start Application operation enables you to start the real-time application for external mode simulation.</p> <p>Before you start the application, you can select the AutoImportFileLog option from the Run on Target button. This checkbox corresponds to the AutoImportFileLog option of the start function.</p>

Commands	Step-by-Step Operations
	When you are connected in external mode, to restart the real-time application from time $t=0$, click Restart Application . This button restarts the application and maintains the external mode connection.

See Also

Target | Mux | load

More About

- “System Configuration”
- “Set Up and Configure Simulink Real-Time” on page 4-2
- “Apply Simulink Real-Time Model Template to Create Real-Time Application”
- “Build and Download Real-Time Application by Using Run on Target” on page 4-8
- “Troubleshoot Missing Real-Time Tab”

Apply Simulink Real-Time Model Template to Create Real-Time Application

This example shows how to use the Simulink® Real-Time™ template to create a Simulink® model. Starting from the model template provides a new model that has configuration parameters set up for building a real-time application.

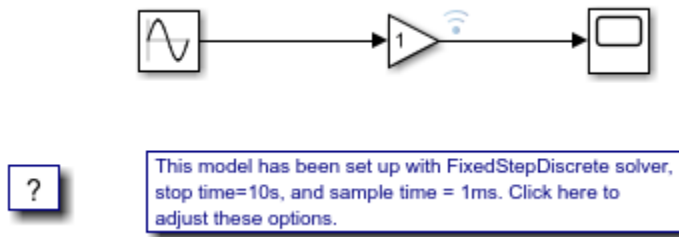
To see the Simulink Real-Time commands for each operation in this example, view the example code.

Create Simulink Model from Template

To create a Simulink model from the Simulink start page, in the MATLAB® Command Window, type:

```
simulink
```

Select the Simulink Real-Time template from the start page and create the `exampleSlrealtimeApp` model. Or, in the Command Window, use the `Simulink.createFromTemplate` command. See code for this script for full syntax.



Blocks, Connections, and Data Logging in the Model

The Simulink Real-Time model template contains a Gain block that connects a Signal Generator to a Scope block. The Gain block output is marked for logging with the Simulation Data Inspector (SDI).

Create Target Object and Connect

Create a Target object for the default target computer and connect to the target computer.

Simulate Real-Time Application and View Logged Data

Build the real-time application, run it on the target computer, and view the logged data:

1. Make sure that the development computer has a connection to the target computer.
2. Build the model and download the real-time application to the target computer. On the **Real-Time** tab, click **Run on Target**. Or, use the `slbuild` command and the `load` command.
3. Run the real-time application and log data by using the **Run on Target** button.
4. Open the Simulation Data Inspector by double-clicking the Simulation Data Inspector icon on the Gain block output signal or by using the `Simulink.sdi.view` command.

More Information

- “Create and Run Real-Time Application from Simulink Model” on page 4-4
- “Configure and Control Real-Time Application by Using Simulink Real-Time Explorer” on page 4-15
- Simulation Data Inspector

Configure and Control Real-Time Application by Using Simulink Real-Time Explorer

You can configure and control a real-time application by using Simulink Real-Time Explorer. This tutorial uses Simulink Real-Time Explorer to control the `slrt_ex_osc_rt` real-time application, which is a real-time model of a damped oscillator. To open the example model, in the MATLAB Command Window, type:

```
openExample('slrealtime/SlrtCreateAndRunRealTimeAppFromSimulinkModelExample', ...
    'supportingFile','slrt_ex_osc_rt.slx')
```

Execute Real-Time Application

This procedure assumes that you have built the real-time application `slrt_ex_osc_rt` from the model and are ready to load on the target computer. For more information, see “Execute Real-Time Application in Simulink External Mode by Using Step-by-Step Commands” on page 4-9.

- 1 In the Simulink Editor, open Simulink Real-Time Explorer. In the **Real-Time** tab, from the **Prepare** section, click **SLRT Explorer**.
- 2 In the **Targets Tree** panel, click the target computer icon for which you have downloaded the real-time application.
- 3 To connect to the target computer, click the **Disconnected** label. The label toggles to **Connected** and the target computer icon indicates connected status.
- 4 To select and load a real-time application on the target computer, click **Load Application**.
- 5 Click **File Selector** to select the application. In the dialog box that opens, select the file and click **Load**.
- 6 To view the contents of the real-time application on the target computer, click the **Signals** tab or **Parameters** tab.
- 7 To run the application for 10 seconds, change the **Stop time** value to 10.
- 8 To view application signals, click **Data Inspector**. The Simulation Data Inspector opens.
- 9 To run the real-time application, click **Start**.

In Explorer, clicking the **Start** button is equivalent to executing this command for target object `tg`:

```
start(tg, 'ReloadOnStop', true)
```

- 10 The Simulation Data Inspector displays the run data for 10 seconds.
- 11 To observe the event status of the **Target** object, install a listener for object events.

Alternatively, you can use functions in the MATLAB Command Window to start and stop the real-time application. For more information, see the functions for the **Target** object.

Change Stop Time and Other Application Options

To control the operation of the real-time application, you can change the stop time and other real-time application options by using MATLAB commands.

The real-time application `slrt_ex_osc_rt` is already built from the model and installed on the target computer. For more information, see “Build and Download Real-Time Application by Using Run on Target” on page 4-8.

- 1 Open Simulink Real-Time Explorer. In the **Real-Time** tab, from the **Prepare** section, click **SLRT Explorer**. The explorer shows information for the selected target computer and loaded real-time application.
- 2 To change the application stop time option, enter `Inf` in the **Stop time** field.
- 3 To change other application options, create an **Application** object. The other options are accessible as object properties. To create an **Application** object for the `slrt_ex_osc_rt` application, in the MATLAB Command Window, type:

```
my_app = slrealtime.Application('slrt_ex_osc_rt');
```

- 4 View application options by getting the application Options property values:

```
my_app.Options.get

ans =

    struct with fields:

        fileLogMaxRuns: 1
        loglevel: "info"
        pollingThreshold: 1.0000e-04
        stoptime: Inf
```

- 5 Change the application option value for file log maximum runs:

```
my_app.Options.set("fileLogMaxRuns",4);
```

- 6 Save application options to a MATLAB variable. Apply options from the variable to the real-time application by using the load function:

```
my_options = my_app.Options.get;
my_app.Options.set(my_options);
```

Alternatively, you can use functions in the MATLAB Command Window to change the real-time application stop time, get parameter values, and set parameter values. For more information, see the functions for the **Target** object.

See Also

More About

- “Blocks Whose Outputs Depend on Inherited Sample Time”
- “Troubleshoot Missing Real-Time Tab”

Simulate Simulink Model by Using MATLAB Language

To observe the non-real-time behavior of the model, simulate the Simulink model.

This procedure uses the Simulink model `slrt_ex_osc_nrt`. To open the example model, in the MATLAB Command Window, type:

```
openExample('slrealtime/SlrtCreateAndRunRealTimeAppFromSimulinkModelExample', ...  
    'supportingFile','slrt_ex_osc_nrt.slx')
```

Before you run the simulation, load this model. From the **Modeling** tab, click **Model Settings**. Enable **Data Import/Export > Time** variable `tout` and **Data Import/Export > Output** variable `yout`.

- 1 In the MATLAB Command Window, type:

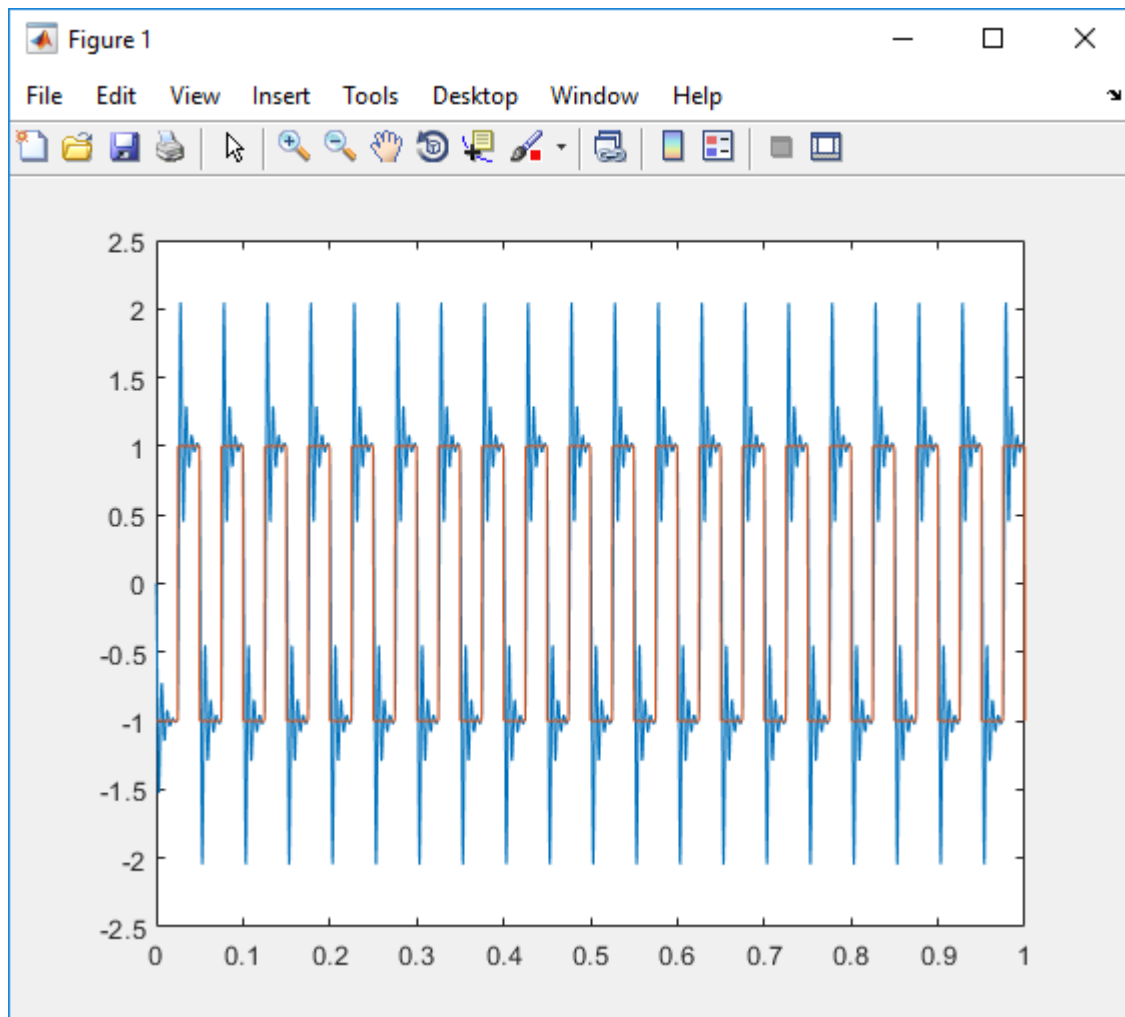
```
output=sim('slrt_ex_osc_nrt','SimulationMode','normal');
```

The signal from the signal generator is logged to memory through the Output block.

- 2 After Simulink finishes the simulation, type:

```
plot(output.get('tout'), output.get('yout'))
```

MATLAB opens a plot window and displays the output response.



When you change the model code generation target to `slrealtime.tlc` or another Simulink Real-Time STF, build the model, and run the real-time application, the signal data for the MuxOut signal is sent to the Simulation Data Inspector. You can export the signal data from the Simulation Data Inspector. For more information, see “Inspect and Compare Data Programmatically”.

See Also

Outport

More About

- “Configure and Control Real-Time Application by Using Simulink Real-Time Explorer” on page 4-15
- “Prepare Real-Time Application by Using MATLAB Language” on page 4-19

Prepare Real-Time Application by Using MATLAB Language

You can configure your real-time application by using the MATLAB language, and then build and download it to the target computer. This example uses model `slrt_ex_osc_rt`. To open the example model, in the MATLAB Command Window, type:

```
openExample('slrealtime/SlrtCreateAndRunRealTimeAppFromSimulinkModelExample', ...
    'supportingFile','slrt_ex_osc_rt.slx')
```

- 1 Create a Target object that represents the target computer and connect Simulink Real-Time Explorer to the target computer:

```
tg = slrealtime('TargetPC1');
connect(tg);
```

- 2 Open the model:

```
model_name = 'slrt_ex_osc_rt';
openExample(model_name);
```

- 3 Set the start and stop time parameters:

```
set_param(model_name,'StartTime','0');
set_param(model_name,'StopTime','10');
```

- 4 Build the model:

```
slbuild(model_name);
```

- 5 *(Optional)* Generate the ASAP2 file with required version using the Generate Calibration Files tool and customize the ASAP2 file as needed. For more information about generating and customizing the ASAP2 file, see “Generate ASAP2 and CDF Calibration Files”.

- 6 *(Optional)* Use `updateASAP2` function to pack the ASAP2 file to the application object (.mldatx):

```
updateASAP2(app_object,updated_a2l);
```

- 7 Load the real-time application onto the target computer:

```
load(tg,model_name);
```

See Also

Target | load

More About

- “Execute Real-Time Application with MATLAB Language” on page 4-20

Execute Real-Time Application with MATLAB Language

Run the real-time application by using generated code to observe the real-time behavior of the model. This procedure uses the Simulink model `slrt_ex_osc_rt`. To open the example model, in the MATLAB Command Window, type:

```
openExample('slrealtime/SlrtCreateAndRunRealTimeAppFromSimulinkModelExample', ...  
    'supportingFile','slrt_ex_osc_rt.slx')
```

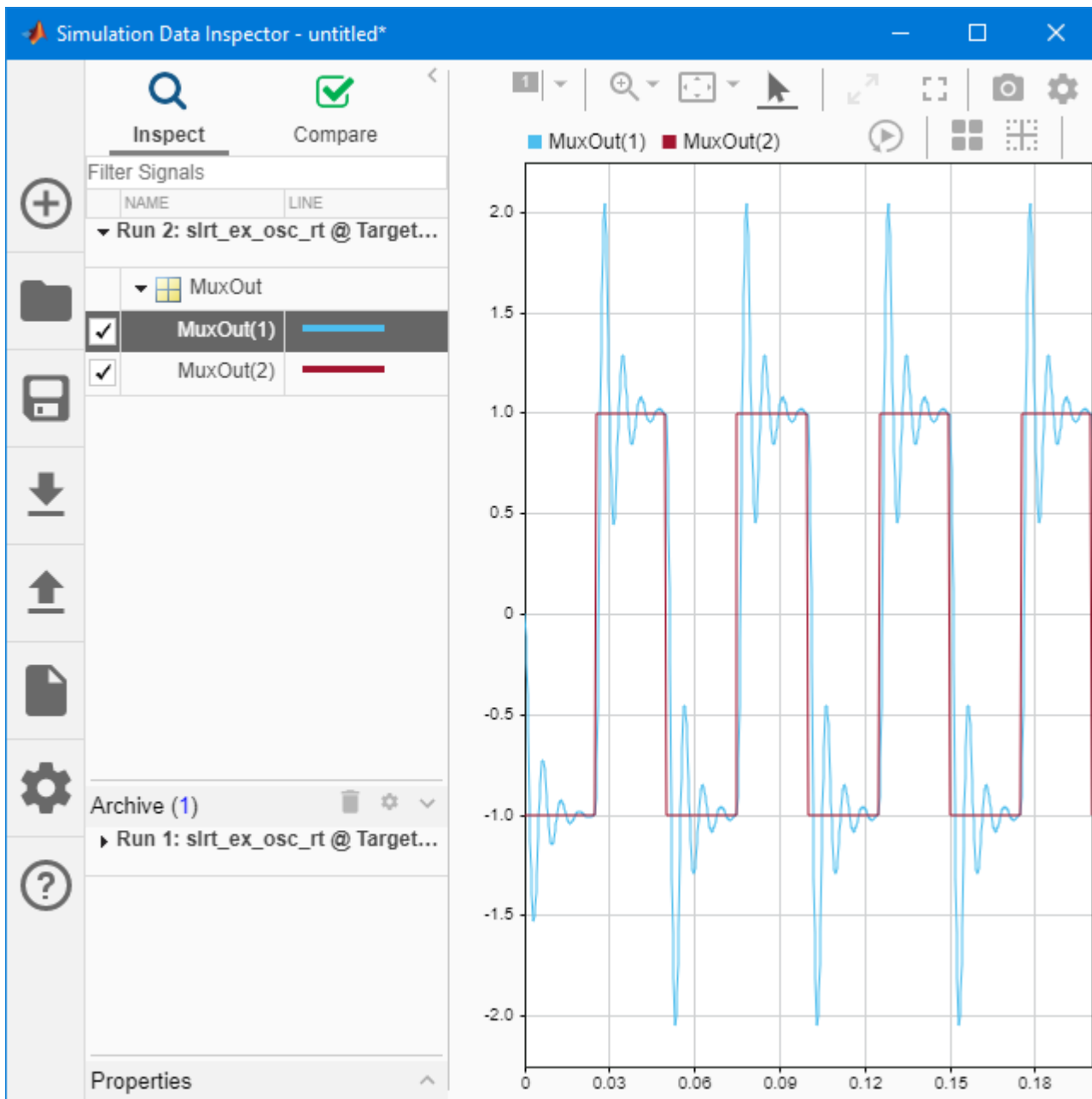
You must have already completed the steps in “Prepare Real-Time Application by Using MATLAB Language” on page 4-19. In the model, select the output signal and mark the signal for data logging with the Simulation Data Inspector.

This example uses functions in the MATLAB Command Window to start and stop the real-time application. For more information, see **Target** object functions.

- 1 Start the target computer.
- 2 The target computer displays session 1 (default) and the target computer status monitor.
- 3 Open the Simulation Data Inspector. In the MATLAB Command Window, type:

`Simulink.sdi.view`
- 4 Start the real-time application. In the MATLAB Command Window, type:

`start(tg)`
- 5 Observe that the real-time application starts running on the target computer. The status monitor displays information as the real-time application executes. The Simulation Data Inspector displays signal data.



6 Stop the real-time application. In the Command Window, type:

```
stop(tg)
```

7 To observe the event status of the Target object, install a listener for object events.

To view the status monitor from the development computer, you can use PuTTY to open an SSH client and start the status monitor application `/usr/target/bin/statusmonitor` on the target computer.

For more information, see "Target Computer Status Monitor".

See Also

Target | start | stop

More About

- “Prepare Real-Time Application by Using MATLAB Language” on page 4-19
- “Blocks Whose Outputs Depend on Inherited Sample Time”

Application and Driver Scripts

These examples show the features of Simulink Real-Time. Use these scripts as a starting point for writing your scripts for creating and testing real-time applications.

These examples include general applications examples and driver examples. You can access the general application and driver examples through MATLAB Help. In the help window, click **Real-Time Simulation and Testing > Simulink Real-Time > Examples**.

General Examples

The general application examples include models and scripts.

Description	File Name
Real-time parameter tuning and data logging	"Real-Time Signal Logging and Parameter Tuning"
Log signal data with the Simulation Data Inspector to a file on the development computer	"Real-Time Signal Logging and Parameter Tuning"
Create a custom client application to interface with the target computer	"Add App Designer App to Inverted Pendulum Model"
Concurrent execution of a model by using the Simulink Real-Time profiling tool	"Concurrent Execution on Simulink Real-Time"

Driver Examples

These driver examples include models and scripts showing the use of driver blocks in a Simulink Real-Time environment. To run some of these examples, you must install specific hardware.

- Analog and digital I/O
- ARINC 429
- Asynchronous events
- Audio
- CAN (CAN_MESSAGE data types)
- Counters, timers, pulse width modulators (PWM)
- Digital signal processing
- Encoders
- EtherCAT®
- FPGA
- J1939
- MIL-STD-1553
- Precision Time Protocol (PTP)
- Ethernet
- RS-232
- Shared/reflective memory

- UDP

Edit Scripts

To locate and edit an example such as “Apply Simulink Real-Time Model Template to Create Real-Time Application”:

- 1 Open the example scripts folder. In the MATLAB Command Window, type:

```
cd (fullfile(matlabroot, 'examples', 'slrealtime', 'main'))
```

- 2 Use the descriptive names of the example scripts (similar to the example title) to find the script.
- 3 Open the script. For example, in the MATLAB Command Window, type:

```
openExample('slrealtime/ApplySlrealtimeModelTemplateToCreateRtAppExample')
```