Link for ModelSim® Release Notes

These release notes describe Version 1.4 of the Link for ModelSim. Topics include

The Link for ModelSim Release Notes also provide information about the earlier versions of the product, in case you are upgrading from an earlier version:

Contents

4 Link for ModelSim 1.2 Release Notes

1

Link for ModelSim 1.4 Release Notes

New Features

This section describes new features introduced in Link for ModelSim 1.4.

MATLAB Component Functions

MATLAB component functions let you simulate the behavior of VHDL entities in the MATLAB environment. A MATLAB component typically provides some functionality (such as a filter) that is not yet implemented in the VHDL code.

To use a MATLAB component function, you define a stub entity (providing port definitions only) in the VHDL model. The stub entity passes its input signals to the MATLAB component function. The MATLAB component processes this data and returns the results to the outputs of the stub entity.

The programming, interfacing, and scheduling conventions for MATLAB component functions are almost identical to those for MATLAB test bench functions. The input/output arguments for a MATLAB component function are the reverse of the port arguments for a MATLAB test bench function. That is, the MATLAB component function returns signal data to the outputs of the associated VHDL entity, and receives data from the inputs of the associated VHDL entity.

The `matlabcp` function is provided in support of MATLAB component functions. `matlabcp` starts the ModelSim client component of the Link for ModelSim, associates a specified instance of a VHDL entity created in ModelSim with a MATLAB function, and creates a process that schedules invocations of the specified MATLAB function.

See the “Coding a MATLAB Component Function” section of the Link for ModelSim documentation for information on programming conventions and a simple example function.

Major Bug Reports

To view major bug fixes made in R14SP3 for Link for ModelSim, use the Bug Reports interface on the MathWorks Web site.

Note If you are not already logged in to Access Login, when you link to the Bug Reports interface (see below), you will be prompted to log in or create an Access Login account.

After you are logged in, use this Bug Fixes link. You will see the bug report for Link for ModelSim. The report is sorted with fixed bugs listed first, and then open bugs.

If you are viewing these release notes in PDF form on the MathWorks Web site, you can refer to the HTML form of the release notes on the MathWorks Web site and use the link provided.

Known Software and Documentation Problems

To view important open bugs in R14SP3 for Link for ModelSim, use the Bug Reports interface on the MathWorks Web site.

Note If you are not already logged in to Access Login, when you link to the Bug Reports interface (see below), you will be prompted to log in or create an Access Login account.

After you are logged in, use this Open Bugs link. You will see the bug report for Link for ModelSim. The report is sorted with fixed bugs listed first, and then open bugs. You can select the **Status** column to list the open bugs first.

If you are viewing these release notes in PDF form on the MathWorks Web site, you can refer to the HTML form of the release notes on the MathWorks Web site and use the link provided.

2

Link for ModelSim 1.3.1 Release Notes

No significant new features have been introduced for Version 1.3.1 of the Link for ModelSim. With the exception of bug fixes, the product is essentially unchanged from Version 1.3.

3

Link for ModelSim 1.3 Release Notes

New Features

This section describes new features introduced in Link for ModelSim 1.3.

User-Defined Simulink and ModelSim Timing Relationship for Cosimulation

Overview

The Link for ModelSim 1.3 lets you define the timing relationship between Simulink and ModelSim during cosimulation. Using the new **Timescales** pane of the VHDL Cosimulation block, you can now overcome problems caused by differences in the representation of simulation time between ModelSim and Simulink.

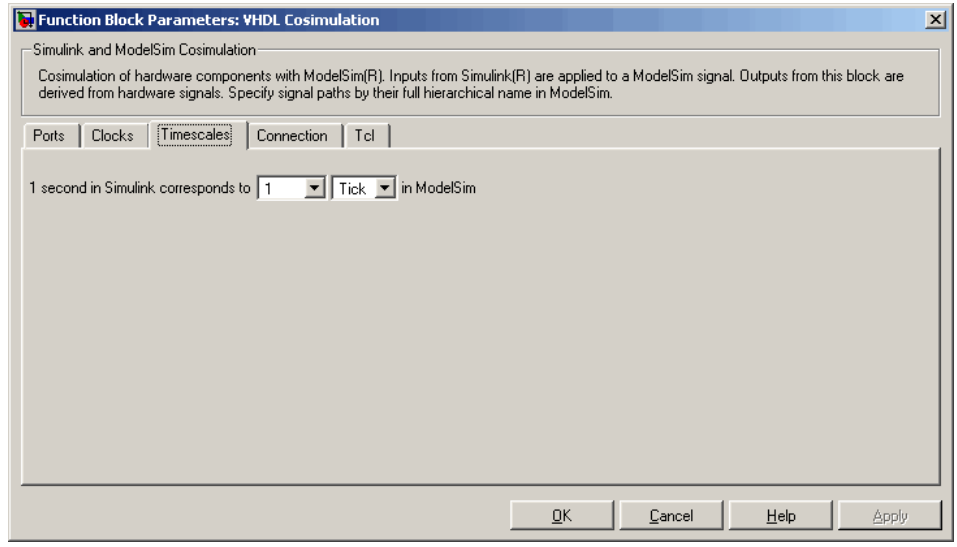
In ModelSim, the unit of simulation time is referred to as a *tick*. The duration of a tick is defined by the ModelSim *resolution limit*. The default resolution limit is 1 ns. In Simulink, simulation time is represented as a double-precision value scaled to seconds. This representation accommodates continuous models and discrete controllers.

In previous releases, the VHDL Cosimulation block supported only a fixed correspondence between simulation time in Simulink and ModelSim. In the older timing mode, one time step in Simulink corresponded to one tick in ModelSim. For example, if the total simulation time in Simulink were specified as 100 time steps, then the ModelSim VHDL simulation would run for exactly 100 ticks (i.e., `100 ns` at the default resolution limit).

New Timing Modes

The Link for ModelSim 1.3 continues to support the older timing model as a default. However, the new **Timescales** pane of the VHDL Cosimulation block lets you specify the relationship between timestep sizes in a Simulink/ModelSim cosimulation with much more control and flexibility.

The figure below shows the default settings of the **Timescales** pane.



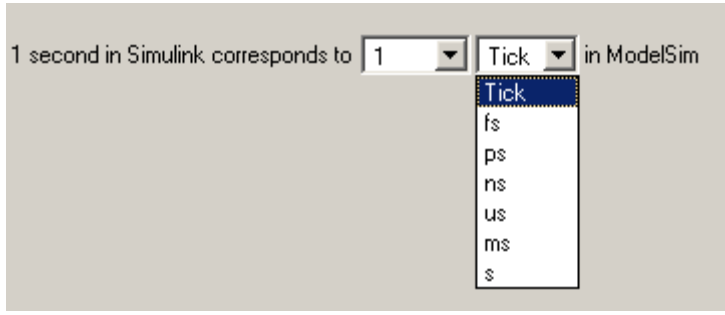
The **Timescales** pane specifies a correspondence between one second of Simulink time and some quantity of ModelSim time. This quantity of ModelSim time can be expressed in one of the following ways:

- In *relative* terms (i.e., as some number of ModelSim ticks). In this case, the cosimulation is said to operate in *relative timing mode*. In relative timing mode, *one second* in Simulink corresponds to *N ticks* in ModelSim, where `N` is a scale factor.

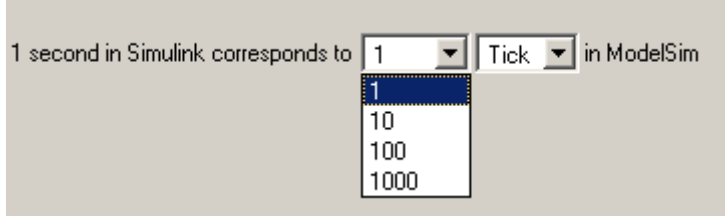
  Relative timing mode is the default.

- In *absolute* units (such as milliseconds or nanoseconds). In this case, the cosimulation is said to operate in *absolute timing mode*. In absolute timing mode, *one second* in Simulink corresponds to *(N * Tu) seconds* in ModelSim, where *Tu* is an absolute time unit (e.g., ms, ns, etc.) and *N* is a scale factor.

The **Timescales** pane contains two lists that let you select the timing mode or time unit and the scale factor. The list on the right specifies the timing mode or the time unit (see the figure below). To choose relative mode, select `Tick`. To choose absolute mode, select one of the available time units (`fs`, `ps`, `ns`, `us`, `ms`, or `s`).

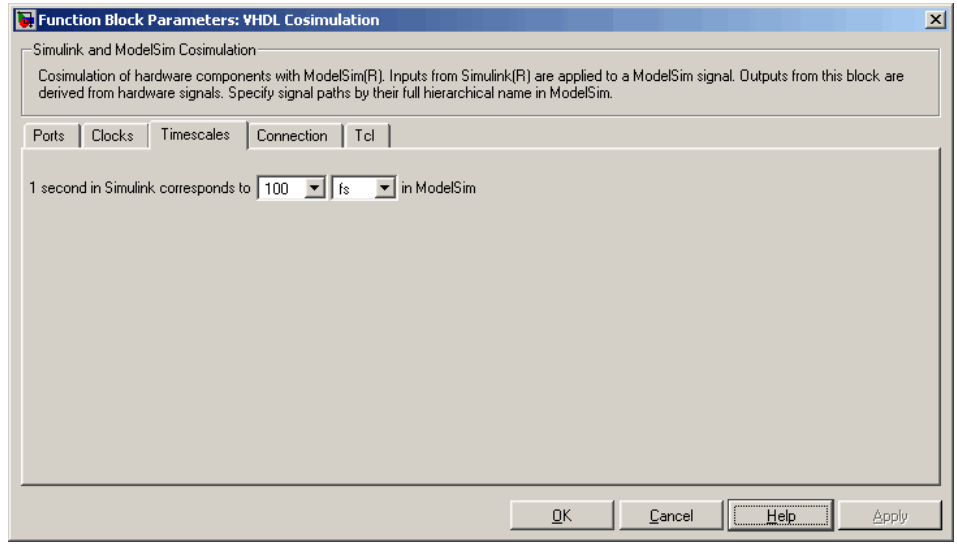


The list on the left specifies the scale factor applied to the time unit (see the figure below).



The default **Timescales** settings (see above) specify relative mode with a scale factor of 1. This default Simulink / ModelSim timing relationship is the same as the relationship defined in previous releases. The default ensures backward compatibility for existing models.

In the figure below, the **Timescales** parameters are configured for absolute mode. An absolute time unit (`fs`) and a scale factor of `100` are selected. During cosimulation, one second in Simulink corresponds to `10 fs` in ModelSim.



Representation of Simulation Time in the Link for ModelSim documentation gives a detailed description of the **Timescales** pane and the supported timing modes, with cosimulation examples.

ModelSim 6.0 Supported

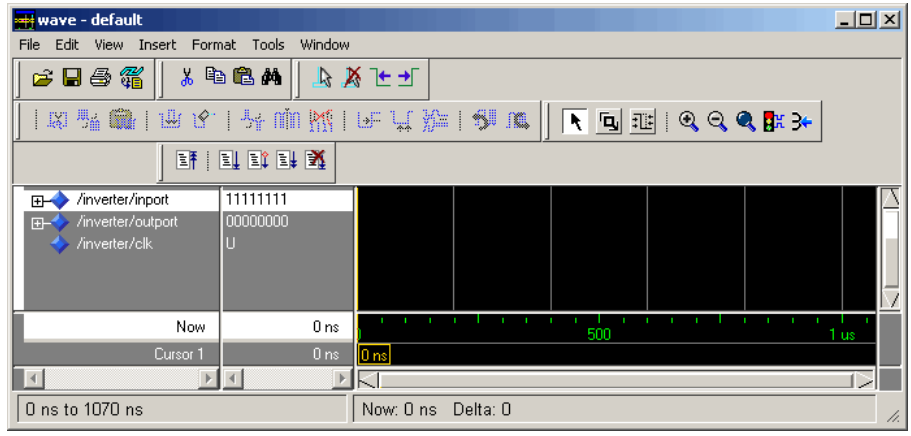
The Link for ModelSim now supports ModelSim Version 6.0.

Smart Copy of Signal Names from ModelSim Wave Window

You can now copy HDL signal names (including the full HDL signal path) from the ModelSim **wave** window and paste them directly into the **Full HDL Name** field of the **Ports** or **Clocks** pane of the VHDL Cosimulation block. This convenience can save you time and errors when cosimulating an HDL design that includes long or complex signal pathnames.

To copy and paste a signal name:

1. Activate ModelSim. Select the desired signal from the signal list in the ModelSim **wave** window. In the figure below, the signal `/inverter/inport` is selected.



2 Right-click on the selected signal. Then select **Copy** from the context menu.

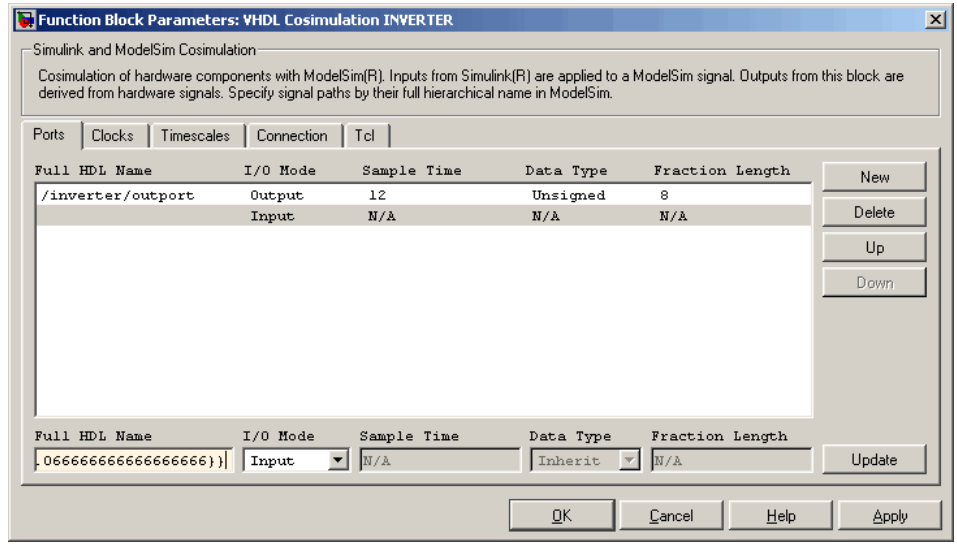
3 Activate Simulink. Then open the block parameters dialog for the desired VHDL Cosimulation block in your model.

4 Activate the appropriate (**Ports** or **Clocks**) pane of the VHDL Cosimulation block.

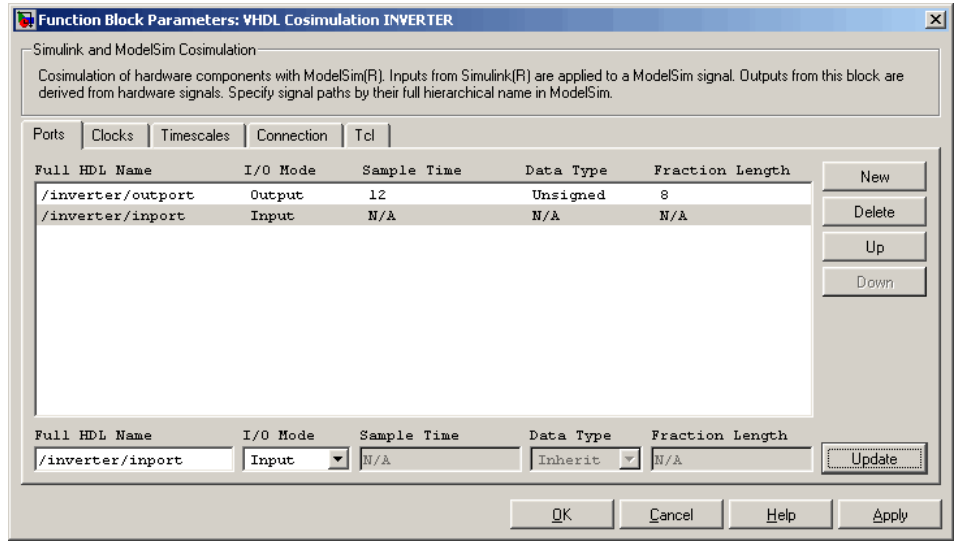
5 Select the desired signal entry from the signal list, or click the **New** button to create a new entry.

6 Select the **Full HDL Name** field.

7 Right-click and select **Paste** from the context menu to paste the signal name into the **Full HDL Name** field. At this point, the signal name is in a special clipboard format (shown below).



8 Click **Update**. The Link for ModelSim translates the signal name into its final format (in this example, `/inverter/inport`) and updates the signal list.



9 If required, configure other parameters of the signal.

10 Click **Apply** when you have finished entering signal data.

Upgrading from a Previous Release

VHDL Cosimulation Output Port Sample Times and Clock Periods Must Be Specified Explicitly

In previous releases, you could assign the default value `-1` as

- The sample time for VHDL Cosimulation block output ports
- The clock period for time for VHDL Cosimulation block clocks

When this default was assigned, Simulink set the port sample time or the clock period equal to the fastest sample time used in the block.

The VHDL Cosimulation block no longer supports use of `-1` as a block output port sample time or clock period. You must explicitly specify sample times for all VHDL Cosimulation block output ports and clock periods, or accept default values. Default values are

- `1` for output port sample times
- `2` for clock periods

VHDL Cosimulation blocks in existing models should be modified to specify explicit output port sample times and clock periods. Use of the value `-1` will cause an error at simulation time.

VHDL Source and Sink Blocks Removed

The VHDL Source and VHDL Sink blocks have been removed from the Link for ModelSim block library. These blocks were simply VHDL Cosimulation blocks that were preconfigured with only output ports (Source block) or only input ports (Sink block).

Existing models that use VHDL Source and VHDL Sink blocks will continue to operate correctly, using the Simulink block forwarding mechanism. However, we recommend that you change existing models to use VHDL Cosimulation blocks rather than VHDL Source and VHDL Sink blocks.

setupmodelsim Command Renamed to configuremodelsim

The `setupmodelsim` command has been renamed to `configuremodelsim`. The two commands are functionally identical.

For backward compatibility, the `setupmodelsim` command continues to work in this release. However, we recommend that you replace `setupmodelsim` in your scripts, using `configuremodelsim` instead.

4

Link for ModelSim 1.2 Release Notes

New Features

This section describes new features introduced in Link for ModelSim 1.2.

VHDL Cosimulation Block Enhancements

We have made major enhancements and revisions to the functionality and the appearance of the VHDL Cosimulation block. This release note summarizes these changes. For a full description of the VHDL Cosimulation block, see the VHDL Cosimulation block reference in the online Link for ModelSim documentation.

Per-Port Sample Time Specification for Outputs Supported

You can now specify an independent sample time for each output port on a VHDL Cosimulation block. Using the **Ports** pane of the VHDL Cosimulation block parameters dialog (see “Ports Pane” on page 4-3) you can specify an explicit sample time, or specify a default (`-1`). In the default case, Simulink sets the sample time to the fastest sample time used in the block.

Per-Port Data Type Specification for Outputs Supported

You can now force fixed point data types on individual output ports of a VHDL Cosimulation block, using the **Ports** pane of the VHDL Cosimulation block parameters dialog (see “Ports Pane” on page 4-3). By default, Simulink determines the data type by back-propagation or by querying ModelSim. Alternatively, you can assign an explicit data type (with optional fraction length) using the **Data Type** and **Fraction length** fields.

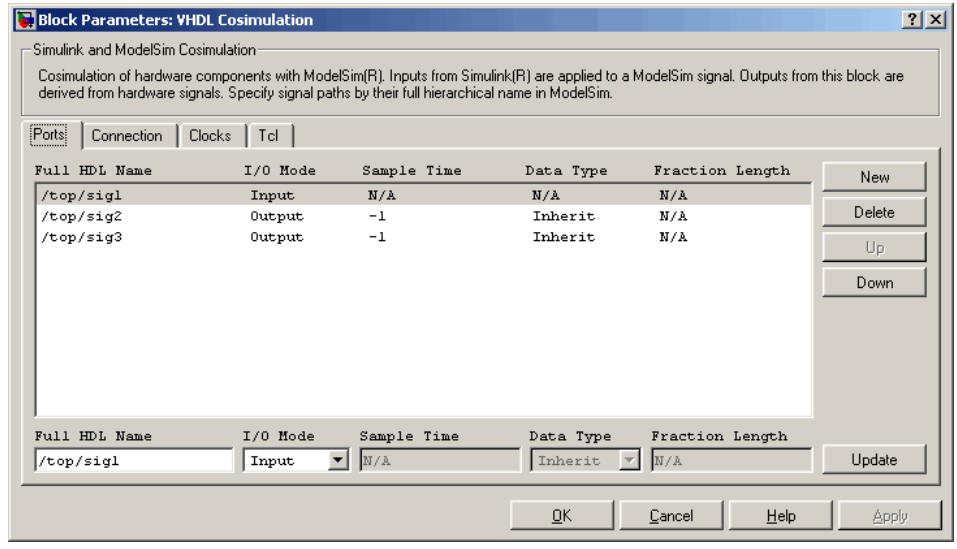
Specification of Independent Clock Sample Times Supported

Using the **Clocks** pane of the VHDL Cosimulation block parameters dialog (see “Clocks Pane” on page 4-5). you can now specify period of each clock in the model explicitly, or specify `-1` to use a default value supplied by Simulink. In the default case, Simulink sets the clock period to the fastest sample time used in the block.

Improved and Revised VHDL Cosimulation Block Parameters Dialog Box

The sections below illustrate and summarize the improvements that have been made to the VHDL Cosimulation block GUI.

Ports Pane. The figure below shows the revised layout of the **Ports** pane of the VHDL Cosimulation Block Parameters dialog box.



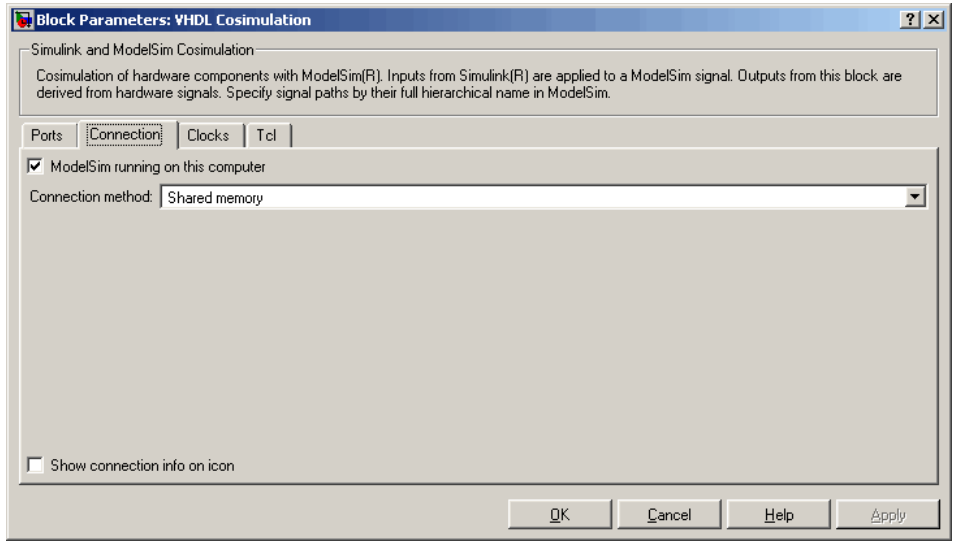
The **Ports** pane now displays a scrolling list of VHDL signals corresponding to ports on the VHDL Cosimulation block. The buttons to the right of the list let you add, delete, or reposition signals in the list. To set the properties of a signal, select the desired signal from the list and enter values into the property fields below the list.

The **Ports** pane supports the following properties and capabilities:

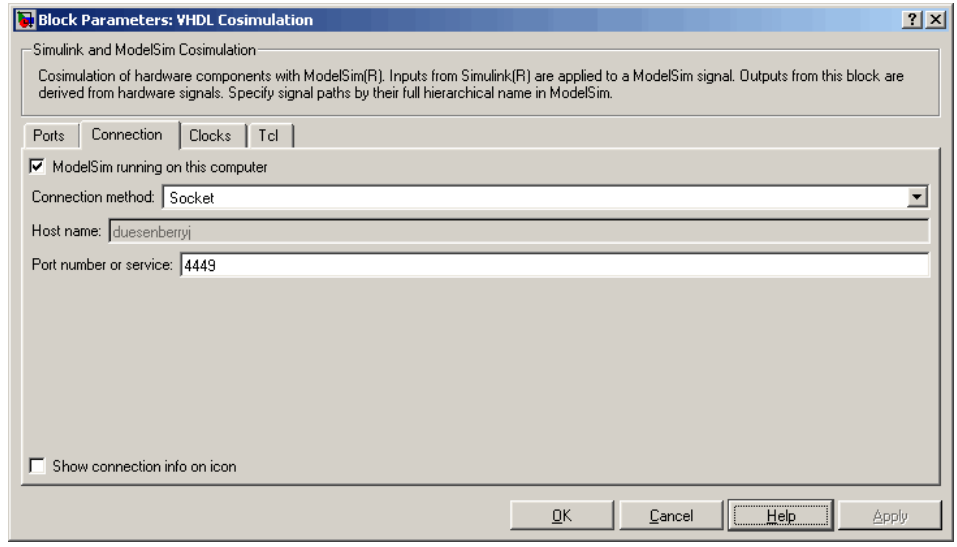
- The **Full HDL Name** field lets you enter the VHDL pathname for a signal anywhere in the hierarchy of the VHDL model.
- The **I/O Mode** menu lets you select whether a signal is associated with an input or output port.
- The **Sample Time** field lets you specify a sample time, per port, for outputs.

- The **Data Type** and **Fraction length** fields let you specify a fixed point data type for individual output ports of a VHDL Cosimulation block.

Connection Pane. The figure below shows the default layout of the **Connection** pane (formerly labelled as the **Comm** pane) of the VHDL Cosimulation Block Parameters dialog box.

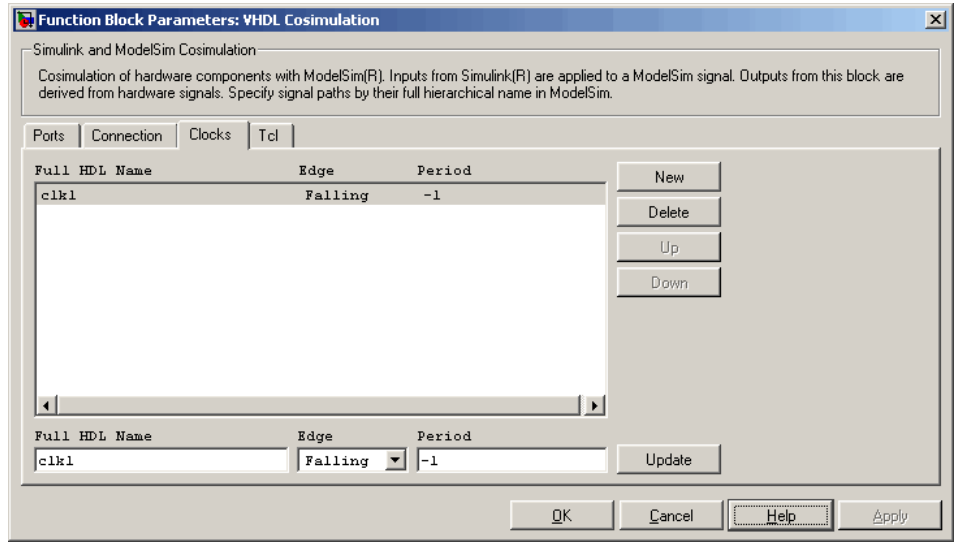


By default, as shown above, the block is configured for shared memory communication. If you select TCP/IP socket mode communication, the pane displays additional properties, as shown in the figure below.



When the new **Show connection info on icon** option is selected, information about the selected communication method and (if applicable) communication options is displayed on the VHDL Cosimulation block icon in the Simulink model.

Clocks Pane. The figure below shows the default layout of the **Clocks** pane of the VHDL Cosimulation Block Parameters dialog box.

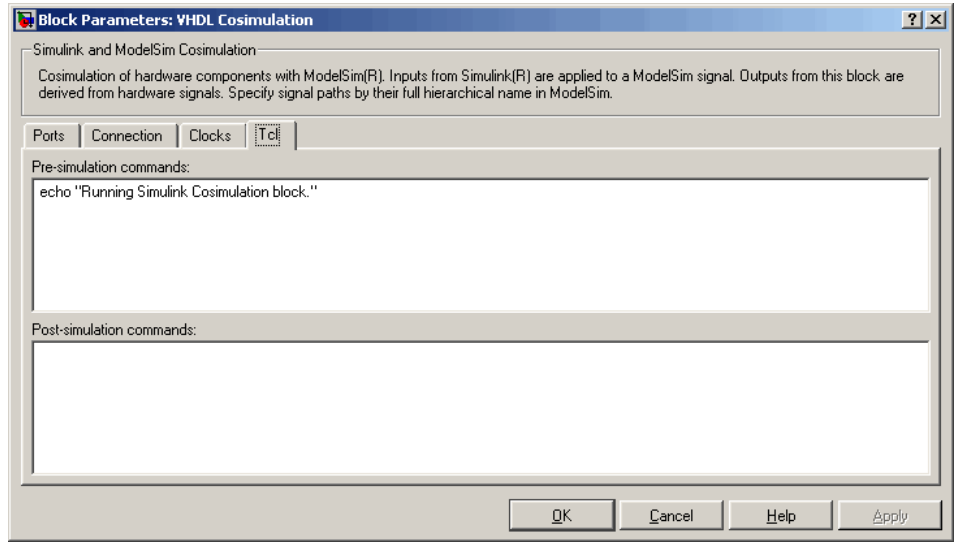


The **Clocks** pane now displays a scrolling list of VHDL clock signals The buttons to the right of the list let you add, delete, or reposition clock signals in the list. To set the properties of a clock signal, select the desired signal from the list and enter values into the property fields below the list.

The **Clocks** pane supports the following properties and capabilities:

- The **Full HDL Name** field lets you enter the VHDL pathname for a clock signal.
- The **Edge** menu lets you specify either a rising-edge clock or a falling-edge clock.
- The **Period** field lets you specify the clock period explicitly, or specify `-1` to use a default value supplied by Simulink.

Tcl Pane. The figure below shows the revised layout of the **Tcl** pane of the VHDL Cosimulation Block Parameters dialog box.



You can now specify Tcl commands in the text boxes in one line per command format, or enter multiple commands per line by appending each command with a semicolon (;), the standard Tcl concatenation operator.

Support for MATLAB/ModelSim Sessions Between Platforms of Differing Byte Ordering

You can now run MATLAB/ModelSim sessions in TCP/IP socket mode between platforms having different byte ordering.

In previous releases, Link for ModelSim required that when MATLAB/ModelSim sessions were run in TCP/IP socket mode, all connected systems must support the same byte ordering (e.g., little-endian or big-endian). This restriction has been removed.

The following table illustrates the currently supported MATLAB / ModelSim connections.

MATLAB / ModelSim Platforms	PC	Linux	Solaris
PC	Yes	Yes	Yes (*new in Link for ModelSim v. 1.2*)
Linux		Yes	Yes (*new in Link for ModelSim v. 1.2*)
Solaris			Yes