Xilinx® ISE Simulator (ISim) VHDL Test Bench Tutorial

Revision: February 27, 2010

215 E Main Suite D | Pullman, WA 99163
(509) 334 6306 Voice and Fax

Overview

This tutorial provides instruction for using the basic features of the Xilinx ISE simulator with the WebPACK environment. This tutorial uses VHDL test bench to simulate an example logic circuit.

More detailed tutorials for the Xilinx ISE tools can be found at http://www.xilinx.com/support/techsup/tutorials/.

Getting Started

You first need to install Xilinx ISE WebPACK on your PC or laptop. The latest version of the software is currently 11.1, which is what we use in this tutorial. It is available as a free download from www.xilinx.com.

This tutorial uses the project example1-VHDL, from another Digilent tutorial on the Xilinx ISE tools. This project is available as a free download from www.digilentinc.com.

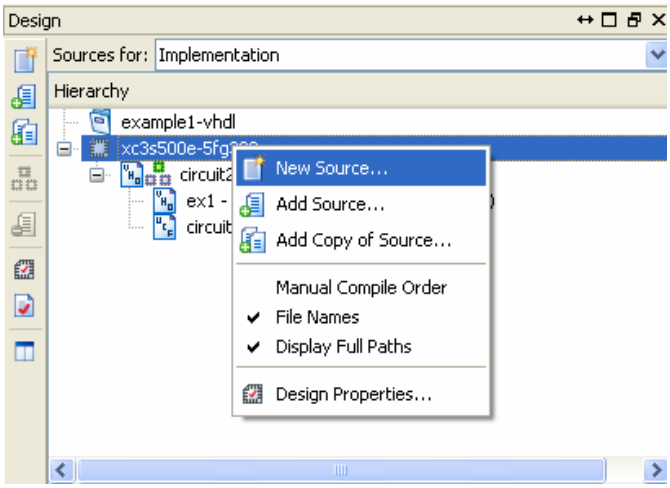
Starting Sample Project

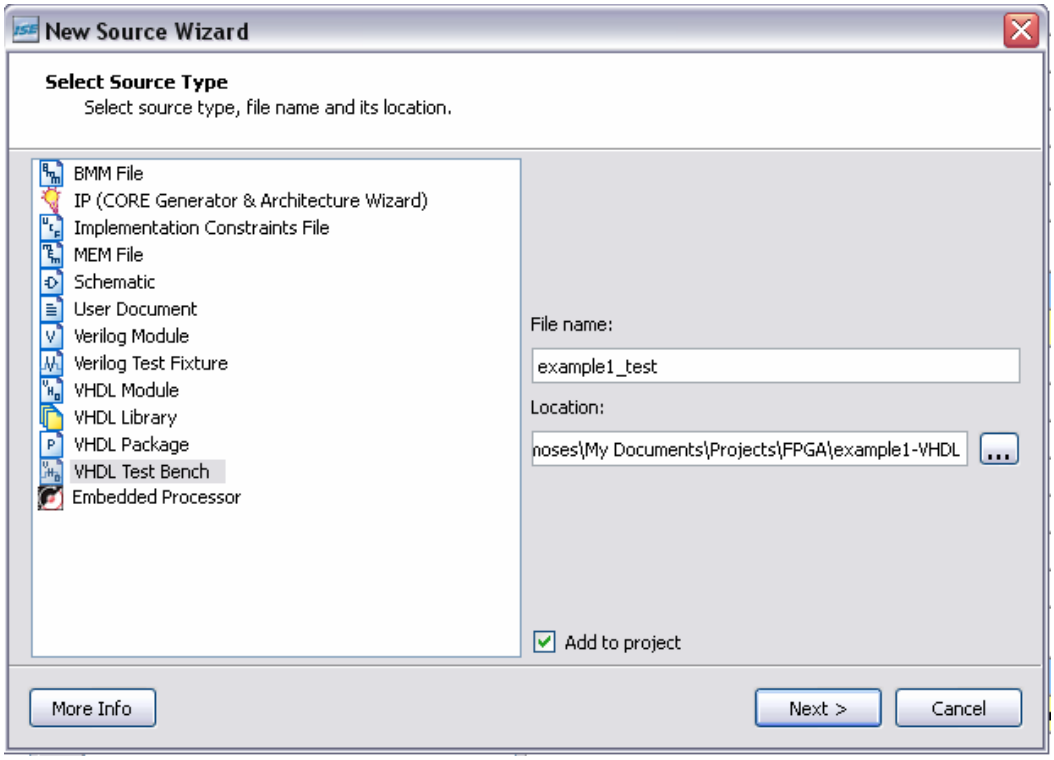
First, open Project Navigator by selecting Start > Programs > Xilinx ISE Design Suite 11 > ISE > Project Navigator. Once the application opens, specify an ISE project file to open by selecting File > Open Project and navigate to the appropriate directory to choose your project. In this tutorial, we use example1-VHDL.xise.

Once the project is open, add a VHDL test bench source file to your project. In this source file, you are able to define circuit inputs over time so the simulator knows how to drive the outputs.

To add the source file, right-click on the device in the Sources window and choose the New Source option. In the New Source wizard, choose VHDL test bench for the source type and enter a meaningful name for the file. We call ours “example1_test”.

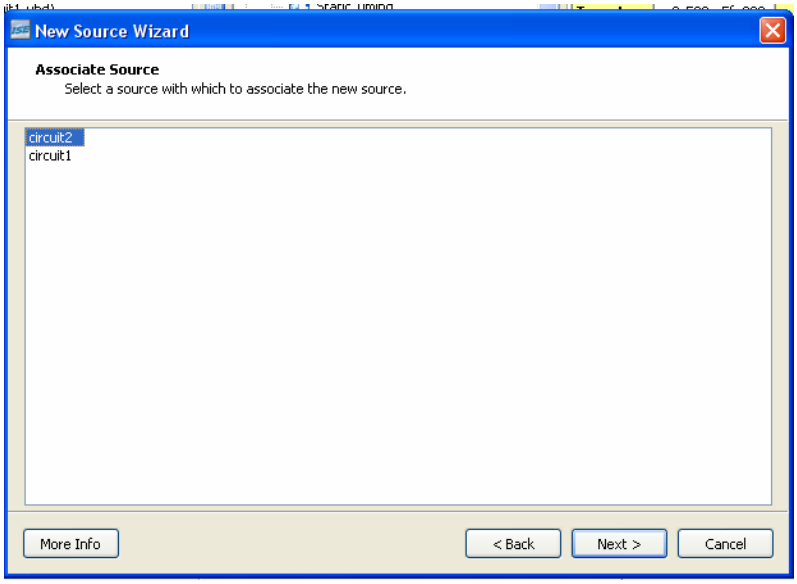
Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.



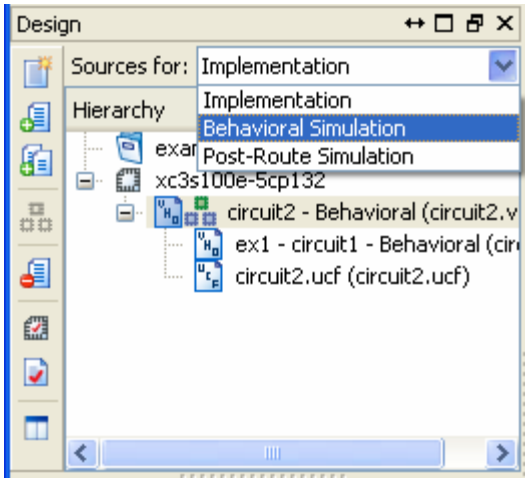


After clicking Next, the following dialog box asks to you select the source file you want to associate with the given test bench file. This dictates which source file you actually run the simulation on. In this tutorial, we run the simulation on the top-level module of the example1-VHDL design (circuit2.vhd).

Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.

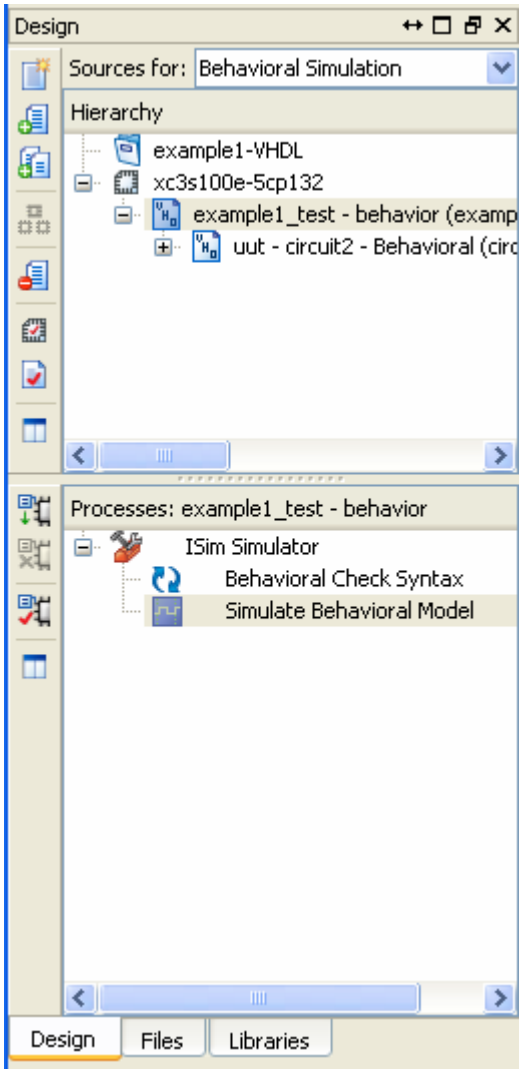


Complete the new source file creation by clicking Next and Finish. To view and edit the VHDL test bench, you first need to change the selected option in the sources drop-down menu from Implementation to Behavioral Simulation as follows:



Once this option is selected, the sources panel changes slightly so that example1_test.vhd is the first source file under the device. The options under the processes panel change so that the only option is the ISE Simulator.

Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.



VHDL Test Bench

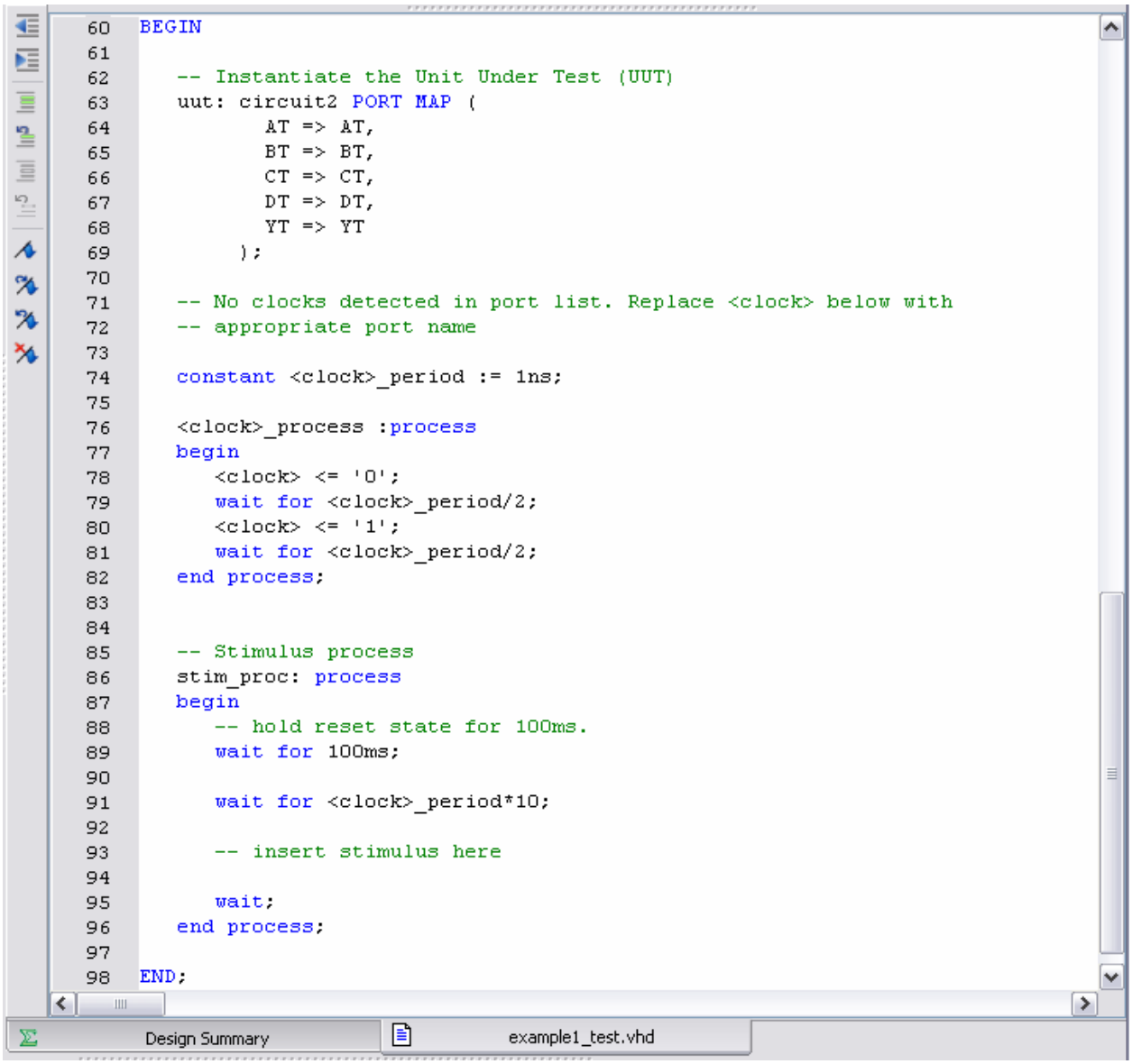
Open the VHDL test bench in the HDL editor by double-clicking it in the sources window. Like a standard VHDL source file, the Xilinx tools automatically generate lines of VHDL code in the file to get you started with circuit input definition. This generated code includes:

- library definitions
- an entity statement
- an architecture statement with begin and end statements included
- a comment block template for documentation

Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.

Due to the richness of the VHDL language, there are many different ways to define circuit inputs inside of a VHDL test bench module. In this tutorial, we present a basic example that can be used as a template for more complex approaches.

Scroll down to the end of the test bench file to see the “begin” and “end” statements of the module.



Most of the generated code in this section of the file is more complex than necessary for our example. We completely remove the first process (<clock>_process.) The second process statement, however, is of use to us.

Before we add any process code, we change the name and value of the constant <clock>_period. A constant, in VHDL, is an object class of a specified type whose value does not change. Simulating a digital circuit involves driving inputs at a certain value for a specified

Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.

time before the inputs are driven differently. Specifying time with constants is an easy and efficient way of keeping track of time between driving inputs. The constant must be defined before the first BEGIN statement (unlike the previous screen shot) in order for the simulation to run.

We name the constant period and set it to 10 ns as follows:

```
constant period := 10 ns;
```

We can now provide stimulus for our circuit inside the second process statement. First, note that all inputs of the circuit have also been initialized to '0' as signals, which indicates that if the inputs are not driven, they remain at the '0' state. Driving an input to a '1' or '0' state inside of the process statement is a simple assignment statement in VHDL:

AT<='1';

This statement drives the input AT to the '1' state where it remains until it is driven otherwise. Next, we add the statement "wait for period;" which tells the simulator to run its current stimulus for a period of time before continuing onto the next statement.

We add a few more lines of statements to show more circuit functionality in the simulator.

Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.

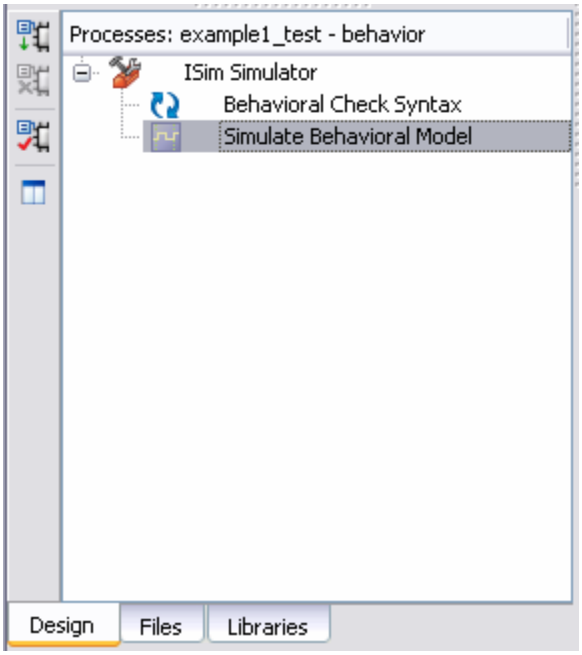
The final VHDL code of interest for the test bench is as follows:

```
59
60      constant PERIOD : time := 10 ns;
61
62  BEGIN
63
64      -- Instantiate the Unit Under Test (UUT)
65      uut: circuit2 PORT MAP (
66             AT => AT,
67             BT => BT,
68             CT => CT,
69             DT => DT,
70             YT => YT
71           );
72
73      -- No clocks detected in port list. Replace <clock> below with
74      -- appropriate port name
75
76      -- Stimulus process
77      stim_proc: process
78      begin
79         wait for PERIOD;
80         AT<= '1';
81
82         wait for PERIOD;
83         BT<='1';
84
85         wait for PERIOD;
86         AT<='0';
87         CT<='1';
88
89         wait for PERIOD;
90         DT<='1';
91
92         wait for PERIOD;
93      end process;
94
95  END;
96
```

Design Summary | example1_test.vhd

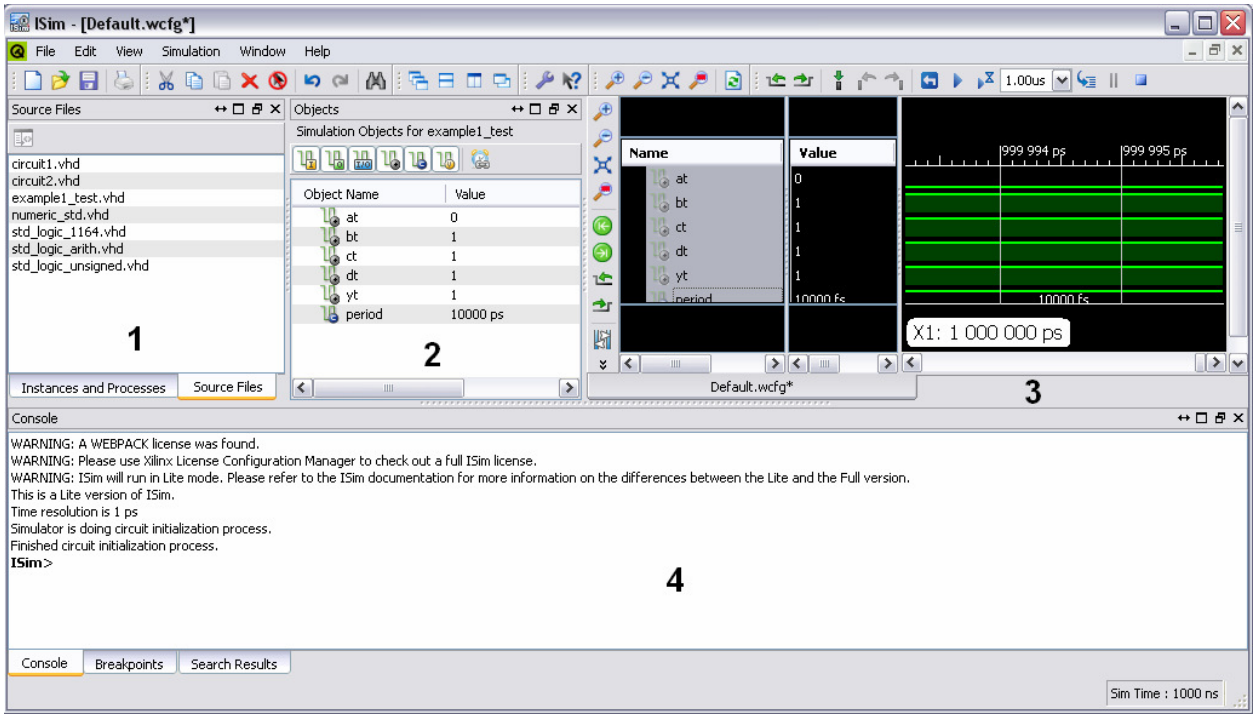
Now, save the test bench code and select it in the Sources window. Go to the Processes window, expand the ISim Simulator (sic), and double-click Simulate Behavioral Model.

Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.



ISE Simulator

Running the Simulate Behavioral Model process causes the ISim window to appear.



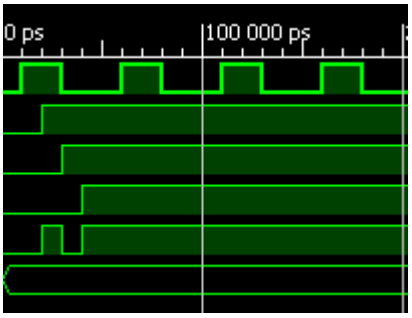
Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.

Some features of this window include:

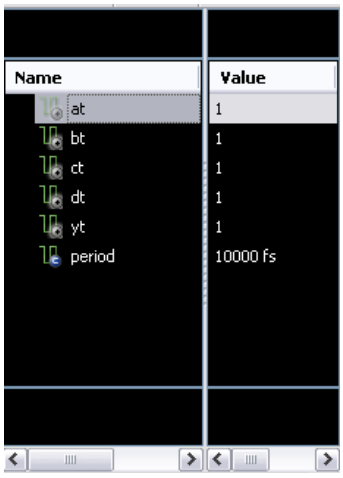
1. a Source Files panel where source files to be viewed can be selected
2. an Objects panel where different signals can be added to the simulation
3. a simulation panel where the state of signals can be observed
4. a Console panel

We first use the Zoom to Full View tool to see the full view of the simulation, which is located to the right of the magnifying glasses on the simulation panel toolbar.

This displays the useful part of the simulation. Use the magnifying glass with the plus sign to zoom in further, as follows:

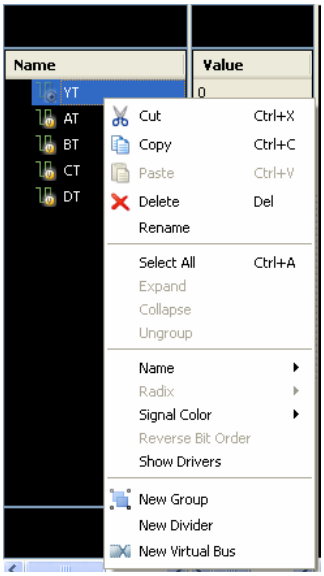


On the left side of the simulation panel there are columns labeled Name and Value:



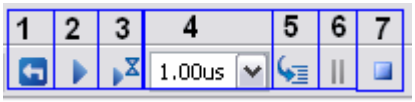
For a given item on these columns, you can right-click and choose options to delete, rename, or change the color of the signal color.

Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.



You may also use the scroll bars to observe the simulation at different times as well as observe more signals if you have a larger design.

The simulation control option on the top right side of the ISim toolbar contains the following features:



1. Restart simulation by stopping it and setting time back to 0.
2. Run simulation until all events are executed.
3. Run simulation for a specified amount of time indicated by the Value box.
4. Amount of time and unit simulation is to run for.
5. Run simulation for one executable HDL instruction at a time.
6. Pause simulation.
7. Stop simulation.

Changing Stimulus

If you have different cases of stimulus that you wish to try out in the simulator, simply close ISim, edit the VHDL test bench in ISE's text editor, and rerun the Simulate Behavioral Model process to open ISim again.

Copyright Digilent, Inc. All rights reserved.
Other product and company names mentioned may be trademarks of their respective owners.