
VHDL VITAL™

Simulation Guide



Actel Corporation, Mountain View, CA 94043

© 2006 Actel Corporation. All rights reserved.

Printed in the United States of America

Part Number: 5579006-11

Release: November 2006

No part of this document may be copied or reproduced in any form or by any means without prior written consent of Actel.

Actel makes no warranties with respect to this documentation and disclaims any implied warranties of merchantability or fitness for a particular purpose. Information in this document is subject to change without notice. Actel assumes no responsibility for any errors that may appear in this document.

This document contains confidential proprietary information that is not to be disclosed to any unauthorized person without prior written consent of Actel Corporation.

Trademarks

Actel and the Actel logo are registered trademarks of Actel Corporation.

Adobe and Acrobat Reader are registered trademarks of Adobe Systems, Inc.

Cadence is a registered trademark of Cadence Design Systems, Inc.

Mentor Graphics is registered trademark of Mentor Graphics, Inc.

Synopsys is a registered trademark of Synopsys, Inc.

Verilog is a registered trademark of Open Verilog International.

Windows is a registered trademark and Windows NT is a trademark of Microsoft Corporation in the U.S. and other countries.

All other products or brand names mentioned are trademarks or registered trademarks of their respective holders.

Table of Contents

1	Introduction	5
	Document Organization	5
	Document Assumptions	5
	Document Assumptions	5
	Actel Manuals	6
	Online Help	6
2	Setup	7
	Software Requirements	7
3	Design Flow	11
	VHDL VITAL Design Flow Illustrated	11
	VHDL VITAL Design Flow Described	12
4	Generating Netlists	15
	Generating an EDIF Netlist	15
	Generating a Structural VHDL Netlist	15
5	Simulation with ModelSim	17
	Behavioral Simulation	17
	Structural Simulation	18
	Timing Simulation	20
6	Simulation with Cadence NC-VHDL	23
	Behavioral Simulation	23
	Structural Simulation	24
	Timing Simulation	25
A	Product Support	27
	Customer Service	27
	Actel Customer Technical Support Center	27
	Actel Technical Support	27
	Website	27
	Contacting the Customer Technical Support Center	28

Index 29

Introduction

This *VHDL Vital Simulation Guide* contains information about using the ModelSim and Cadence NC-VHDL to simulate designs for Actel devices. Refer to the *Designer User's Guide* for additional information about using the Designer software. Refer to the documentation included with your simulator for information about performing simulation.

Document Organization

The *VHDL Vital Simulation Guide* contains the following sections:

Chapter 1 - Setup contains information about setting up ModelSim and Cadence VHDL simulator.

Chapter 2 - Design Flow describes how to use the VHDL design flow to design an Actel device using the synthesis-tool software, and VHDL simulator software.

Chapter 3 - Generating Netlists contains information regarding how to generate a netlist using the synthesis-tool software.

Chapter 4 - Simulation with ModelSim contains information about simulating for Actel designs using the ModelSim simulator.

Chapter 5 - Simulation with Cadence NC-VHDL contains information about simulating for Actel designs using the Cadence NC-VHDL simulator.

Appendix A - Product Support provides information regarding contacting Actel for customer and technical support.

Document Assumptions

This document assumes the following:

1. You have installed the Designer Series software.
2. You have installed your VHDL VITAL simulator.
3. You are familiar with UNIX workstations and operating systems or with PCs and Windows operating environments.
4. You are familiar with FPGA architecture and FPGA design software.

Document Conventions

This document uses the following conventions:

Information input by the user follows this format:

keyboard input

The contents of a file follows this format:

file contents

This document uses the following variables:

- Actel FPGA family libraries are shown as <act_fam>. Substitute the desired Actel FPGA family variable with Fusion, IGLOO™/e, ProASIC3/E, act1, act2 (for ACT2 and 1200XL devices), ACT3, 3200DX, MX, SX, SX-A, eX, Axcelerator, A500k, and APA as needed. For example:

```
vcom -work <vhd_fam> <act_fam>.vhd
```

- Compiled VHDL libraries are shown as <vhd_fam>. Substitute <vhd_fam> for the desired VHDL family variable with Fusion, IGLOO/e, ProASIC3/E, act1, act2 (for ACT2 and 1200XL devices), ACT3, 3200DX, MX, SX, SX-A, eX, Axcelerator, A500k, and APA as needed. The VHDL language requires that the library names begin with an alpha character.

Online Help

Designer software comes with online help. Online help specific to each software tool is available in Libero, Designer, SmartGen, Silicon Expert, Silicon Explorer II, and Silicon Sculptor.

Setup

This chapter contains information on setting up the ModelSim or Cadence NC-VHDL simulator to simulate Actel designs.

This chapter includes software requirements, steps describing how to compile Actel FPGA libraries, and other setup information for the simulation tool you use.

Software Requirements

The information in this guide applies to the Actel Designer software release R1-2003 or later and IEEE-1076-compliant VHDL simulators. Additionally, this guide contains information about using ModelSim and Cadence NC-VHDL simulators.

For specific information about which versions this release supports, go to the technical support system on the Actel web site (<http://www.actel.com/custsup/search.html>) and enter the following in the Keyword box:

third party

Migration Libraries

In addition to the standard Actel libraries, Actel provides a set of migration libraries. These libraries contain cores supported in 3.1.1u1 and earlier versions of the Designer Series software and cores possibly needed to retarget designs from a different Actel family. Actel does not recommend using the migration libraries on new designs.

ModelSim

Since the installation path varies for each user and each installation, this document uses "\$ALSDIR" to indicate the location where the Actel Designer software is installed. If you are a Unix user, simply create an environment variable called ALSDIR and set its value to the installation path. If you are a Windows user, replace "\$ALSDIR" with the installation path in the commands.

Use the following procedure to compile libraries for the ModelSim simulators. Type UNIX commands at the UNIX prompt. Type Windows commands on the command line of the ModelSim Transcript window. The commands below are for Windows. To make the commands work for UNIX, use forward slashes instead of back slashes.

This procedure compiles an Actel VITAL library in the "\$ALSDIR\lib\vtl\95\mti" directory. You must compile the FPGA library models for the Actel VITAL libraries to work properly.

Note: If there is already an MTI directory in the \$ALSDIR\lib\vtl\95 directory, compiled libraries may be present, and you may not need to perform the following procedure.

1. Create a library called “mti” in the “\$ALSDIR\lib\vtl\95” directory.
2. Invoke the ModelSim simulator (Windows only).
3. Change to the “\$ALSDIR\lib\vtl\95\mti” directory. Enter the following command at the prompt:

```
cd $ALSDIR\lib\vtl\95\mti
```
4. Create a <vhd_fam> family library. Enter the following command at the prompt:

```
vlib <vhd_fam>
```
5. Map the Actel VITAL library to the <vhd_fam> directory. Enter the following command at the prompt:

```
vmap <vhd_fam> $ALSDIR\lib\vtl\95\mti\<vhd_fam>
```
6. Compile your VITAL libraries.

```
vcom -work <vhd_fam> ../<act_fam>.vhd
```

For example, to compile the 40MX library for your simulator, type the following command:

```
vcom -work a40mx ../40mx.vhd
```
7. (Optional) Compile the migration library. Only perform this step if you need to use the migration library. Type the following command at the prompt:

```
vcom -work <vhd_fam> ../<act_fam>_mig.vhd
```

Cadence NC-VHDL

Use the following procedure to compile libraries for the Cadence NC-VHDL simulator. This procedure compiles an Actel VITAL library in the “\$ALSDIR/lib/vtl/95/ncvhd1” directory. You must compile the FPGA library models for the Actel VITAL libraries to work properly.

1. Create a directory called “ncvhd1” in the “\$ALSDIR/lib/vtl/95” directory.
2. Change to the “\$ALSDIR/lib/vtl/95/ncvhd1” directory.
3. Create a directory named <vhd_fam>.
4. Map the library. Compile the models and create the “cds.lib” file as follows:

```
INCLUDE $CDS/tools/inca/files/cds.lib
```

```
DEFINE <vhd_fam> $ALSDIR/lib/vt1/95/ncvhd1/<vhd_fam>
```

5. Compile the library. Type the following command at the prompt:

```
ncvhd1 -v93 -work <vhd_fam> -messages $ALSDIR/lib/vt1/95/<act_fam>.vhd
```

For example, to compile the 40MX library for your simulator, type the following command:

```
ncvhd1 -v93 -work a40mx -messages $ALSDIR/lib/vt1/95/40mx.vhd
```

6. (Optional) Compile the migration library. Only perform this step if you are using the migration library. Type the following command at the prompt:

```
ncvhd1 -work <vhd_fam> -messages $ALSDIR/lib/vt1/95/ <act_fam>_mig.vhd
```


VHDL VITAL Design Flow Illustrated

Actel VHDL Design Flow

Version 5.1

The diagram illustrates the Actel VHDL Design Flow, organized into four main stages: Design Creation/Verification, Design Implementation, Programming, and System Verification.

- Design Creation/Verification:** This stage involves the initial design and simulation. It starts with **SmartGen Core Generator** and **Schematic Entry**, both feeding into the **EDIF Netlist**. The **EDIF Netlist** is then processed by **Synthesis** (which also receives input from the **HDL Editor**) and **PALACE™ Physical Synthesis** (which receives input from the **EDIF Netlist** and **VITAL Libraries**). The output of **PALACE™ Physical Synthesis** is the **edn2vhdl** file. The **edn2vhdl** file is then used by the **VHDL Simulator** (which also receives input from the **User TestBench** and **Stimulus Generation**) to perform **Timing Simulation**. The **VHDL Simulator** outputs a **Prelayout VHDL Structural Netlist**.
- Design Implementation:** This stage involves the physical implementation of the design. The **EDIF Netlist** and **Prelayout VHDL Structural Netlist** are used by the **Compile** block (which also receives input from the **MultiView Navigator**) to perform **Optimization and DRC**. The output of **Compile** is the **Layout** block, which performs **Timing-Driven Place-and-Route**. The **Layout** block is then used by the **Program File Generation** block to generate the **Program File**. The **Program File** is then used by the **SmartTime** block (which performs **Static-Timing Analysis and Constraints Editor**) and the **SmartPower** block (which performs **Power Analysis**). The **SmartTime** block outputs a **Timing File**, which is then used by the **Back-Annotate** block (which performs **Back-Annotated Timing for Simulation**) to generate the **Postlayout VHDL Structural Netlist**. The **Postlayout VHDL Structural Netlist** is then used by the **Cross-Probing** block.
- Programming:** This stage involves the programming of the device. The **Program File** is used by the **ChainBuilder** (Flash Families), **FlashPro** (Flash Families), and **Silicon Sculptor*** (Antifuse/Flash Families) blocks.
- System Verification:** This stage involves the verification of the system. The **Postlayout VHDL Structural Netlist** is used by the **Silicon Explorer** (Antifuse Families) block.

At the bottom of the diagram, it is noted that **All Programming and Verification Software Requires Separate Hardware**.

Figure 2-1. Actel-VHDL Design Flow

VHDL VITAL Design Flow Described

The Actel VHDL VITAL design flow has four main steps:

1. Design Creation/Verification
2. Design Implementation
3. Programming
4. System Verification

The following sections detail these steps.

Design Creation/Verification

During design creation/verification, a design is captured in an RTL-level (behavioral) VHDL source file. After capturing the design, you can perform a behavioral simulation of the VHDL file to verify that the VHDL code is correct. The code is then synthesized into an Actel gate-level (structural) VHDL netlist. After synthesis, you can perform an optional pre-layout structural simulation of the design. Finally, an EDIF netlist is generated for use in Designer and a VHDL structural post-layout netlist is generated for timing simulation in a VHDL VITAL-compliant simulator.

VHDL Source Entry

Enter your VHDL design source using a text editor or a context-sensitive HDL editor. Your VHDL design source can contain RTL-level constructs, as well as instantiations of structural elements, such as SmartGen cores.

Behavioral Simulation

Perform a behavioral simulation of your design before synthesis. Behavioral simulation verifies the functionality of your VHDL code. Typically, you use zero delays and a standard VHDL test bench to drive simulation. Refer to the documentation included with your simulation tool for information about performing functional simulation.

Synthesis

After you have created your behavioral VHDL design source, you must synthesize it. Synthesis transforms the behavioral VHDL file into a gate-level netlist and optimizes the design for a target technology. The documentation included with your synthesis tool contains information about performing design synthesis.

1. The grey boxes in Figure 2-1 denote Actel-specific utilities/tools.

EDIF Netlist Generation

After you have created, synthesized, and verified your design, you must generate an Actel EDIF netlist for place-and-route in Designer.

This EDIF netlist is also used to generate a structural VHDL netlist for use in structural simulation.

Structural VHDL Netlist Generation

Generate a gate-level VHDL netlist from your EDIF netlist for use in post-synthesis pre-layout structural simulation by either exporting it from Designer or by using the Actel “edn2vhdl” program.

Structural Simulation

Perform a structural simulation before placing-and-routing. Structural simulation verifies the functionality of your post-synthesis pre-layout structural VHDL netlist. Unit delays included in the compiled Actel VITAL libraries are used. Refer to the documentation included with your simulation tool for information about performing structural simulation.

Design Implementation

During design implementation, you place-and-route a design using Designer. Additionally, you may perform timing analysis on a design in Designer with the Timer tool. After place-and-route, perform postlayout (timing) simulation with a VHDL VITAL-compliant simulator.

Place-and-Route

Use Designer to place-and-route your design. Refer to the *Designer User's Guide* for information about using Designer.

Timing Analysis

Use the Timer tool in Designer to perform static-timing analysis on your design. Refer to the *Timer User's Guide* for information about using Timer.

Timing Simulation

Perform a timing simulation on your design after place-and-route. Timing simulation verifies that the design meets your timing constraints.

Programming

Program a device with programming software and hardware from Actel or a supported third-party programming system. Refer to the *Designer User's Guide* and the *Activator and APS Programming System Installation and User's Guide* or *Silicon Sculptor User's Guide* for information about programming an Actel device.

System Verification

You can perform system verification on a programmed device using the Actel Silicon Explorer diagnostic tool. Refer to the *Activator and APS Programming System Installation and User's Guide* or *Silicon Explorer Quick Start* for information about using the Silicon Explorer.

Generating Netlists

This chapter describes the procedures for generating EDIF and structural VHDL netlists.

Generating an EDIF Netlist

After capturing your schematic or synthesizing your design, generate an EDIF netlist from your schematic capture or synthesis tool. Use the EDIF netlist for place-and-route in Designer. Refer to the documentation included with your schematic capture or synthesis tool for information about generating an EDIF netlist.

Generating a Structural VHDL Netlist

You can generate a structural VHDL netlist using Designer or the “edn2vhdl” program. Use the structural VHDL netlist for structural and timing simulation.

To generate a structural netlist using Designer,

1. **Invoke Designer.**
2. **Import the EDIF netlist.** Choose the Import Source File command from the File menu. This displays the Import Source dialog box. Type the full path name of your EDIF netlist or use the Browse button to select your design. Click OK. In the EDIF Import Options dialog box, specify Generic as the EDIF flavor.
3. **Export a structural VHDL netlist.** From the File menu, choose Export, then Netlist. This displays the Export Netlist Files dialog box. In the Export Netlist Files dialog box, specify the File name and set the Save As type to VHDL. Click OK.

To generate a structural netlist using “edn2vhdl,”

1. **Change to the directory that contains the VHDL design files.**
2. **Type the following command at the UNIX or DOS prompt:**

```
edn2vhdl FAM:{<act_fam>} [ EDNIN:<Edif_File1>[+<Edif_File2...>] ] [ VHDOUT:<Vhdl_File>
]<design_name>
```

The “EDNIN” option specifies the EDIF input file(s). You can specify multiple files with the “+” delimiter between file names. The default EDIF input file is <design_name>.edn. The “VHDOUT” option specifies the VHDL output file names. The default VHDL output file is <design_name>.vhd.

Simulation with ModelSim

This chapter describes steps to perform Functional (Behavioral and Structural) and Timing simulation for Actel devices using the ModelSim simulator.

The procedures shown are for PC. The same setup procedures work similarly for UNIX. Use forward slashes in place of back slashes. For PC, type commands into the MTI window. For UNIX, type commands into a UNIX window.

Behavioral Simulation

Use the following procedure to perform a behavioral simulation of a design. Refer to the documentation included with your simulation tool for additional information about performing behavioral simulation.

1. **Invoke your ModelSim simulator. (PC only)**
2. **Change directory to your project directory.** This directory must include your VHDL design files and testbench. Type:

```
cd <project_dir>
```

3. **Map to the Actel Library.** If any Actel cores are instantiated in your VHDL source, type the following command to map them to the compiled Actel VITAL library.

```
vmap <vhd_fam> $ALSDIR\lib\vt1\95\mti\<vhd_fam>
```

To reference the Actel family library in your VHDL design files, add the following lines to your VHDL design files:

```
library <vhd_fam>;
use <vhd_fam>.components.all;
```

4. **Create a “work” directory.** Type:
5. **Map to the “work” directory.** Type the following command:

```
vlib work
```

```
vmap work .\work
```

6. **Perform a behavioral simulation of your design.** To perform a behavioral simulation using your V-System or ModelSim simulator, compile your VHDL design and testbench files and run a simulation. For hierarchical designs, compile the lower-level design blocks before the higher-level design blocks.

The following commands demonstrate how to compile VHDL design and testbench files:

```
vcom -93 <behavioral>.vhd
vcom -93 <test_bench>.vhd
```

To simulate the design, type:

```
vsim <configuration_name>
```

For example:

```
vsim test_adder_behave
```

The entity-architecture pair specified by the configuration named test_adder_behave in the testbench will be simulated.

If your design contains a PLL core, use a 1ps resolution:

```
vsim -t ps <configuration_name>
```

For example:

```
vsim -t ps test_adder_behave
```

Structural Simulation

Use the following procedure to perform structural simulation.

1. **Generate a structural VHDL netlist.**

If you are using Synopsys Design Compiler, generate a structural VHDL netlist using this tool.

If you are using other synthesis tools, generate a gate-level VHDL from your EDIF netlist by either exporting it from Designer or by using the edn2vhd1 program.

To generate a structural netlist using “edn2vhd1,”

2. **Change to the directory that contains the VHDL design files.**
3. **Type the following command at the UNIX or DOS prompt:**

```
edn2vhd1 FAM:{<act_fam>} [ EDNIN:<Edif_File1>[+<Edif_File2...>] ][ VHDOUT:<Vhdl_File>
]<design_name>
```

The “EDNIN” option specifies the EDIF input file(s). You can specify multiple files with the “+” delimiter between file names. The default EDIF input file is <design_name>.edn. The

“VHDOUT” option specifies the VHDL output file names. The default VHDL output file is <design_name>.vhd.

To generate a netlist using Designer Series software,

4. **Export a structural VHDL netlist.** From the File menu, choose Export, then Netlist. This displays the Export Netlist Files dialog box. In the Export Netlist Files dialog box, specify the File name and set the Save As type to VHDL. Click OK.

Note: The VHDL generated by both Designer and the edn2vhd1 program will use std_logic for all ports. The bus ports will be in the same bit order as they appear in the EDIF netlist.

5. **Map to the Actel VITAL library.** Run the following command to map the compiled Actel VITAL library.

```
vmap <vhd_fam> $ALSDIR\lib\vtl\95\mti\<vhd_fam>
```

6. **Compile the structural netlist.** Compile your VHDL design and testbench files. The following commands demonstrate how to compile VHDL design and testbench files:

```
vcom -just e -93 <structural>.vhd
vcom -just a -93 <structural>.vhd
vcom <test_bench>.vhd
```

Note: First, the application compiles the entities. Then, it compiles the architectures, as required for VHDL netlists written by some tools.

7. **Run the structural simulation.** To simulate your design, type:

```
vsim <configuration_name>
```

For example:

```
vsim test_adder_structure
```

The entity-architecture pair specified by the configuration named test_adder_structure in the testbench will be simulated.

If your design contains a PLL core, use a 1ps resolution:

```
vsim -t ps <configuration_name>
```

For example:

```
vsim -t ps test_adder_structure
```

Timing Simulation

Use the following procedure to perform timing simulation.

1. **Place-and-route your design in Designer.** Refer to the *Designer User's Guide* for information about using Designer.
2. **Extract timing information for your design.** From the File menu, click Export. Then, click Timing Files. Choose SDF and click Save (or click Back Annotate). The Back Annotate dialog box is displayed. Create a <design_name>.sdf file by specifying SDF as the CAE type. Click OK.
3. **Compile the structural netlist.** To perform a timing simulation using your V-System or ModelSim simulator, compile your VHDL design and testbench files, if they have not already been compiled for a structural simulation, and run a simulation. The following commands demonstrate how to compile VHDL design and testbench files:

```
vcom -just e -93 <structural>.vhd
vcom -just a -93 <structural>.vhd
vcom <test_bench>.vhd
```

Note: Performing the previous steps compiles the entities first and then the architectures, as required for VHDL netlists written by some tools.

4. **Run the back-annotation simulation using the timing information in the SDF file.** Type:

```
vsim -sdf[max|typ|min] /<region>=<design name>.sdf -c
<configuration_name>
```

The <region> option specifies the region (or path) to an instance in a design where back annotation begins. You can use it to specify a particular FPGA instance in a larger system design or testbench that you wish to back annotate. For example:

```
vsim -sdfmax /uut=addder.sdf -c test_adder_structural
```

In this example, the entity “adder” has been instantiated as instance “uut” in the testbench. The entity-architecture pair specified by the configuration named “test_adder_structural” in the testbench will be simulated using the maximum delays specified in the SDF file.

If your design contains a PLL core, use a 1ps resolution:

```
vsim -t ps -sdf[max|typ|min] /<region>=<design name>.sdf -c  
<configuration_name>
```

For example:

```
vsim -t ps -sdfmax /uut=addder.sdf -c test_adder_structural
```


Simulation with Cadence NC-VHDL

This chapter describes steps to perform Functional (Behavioral and Structural) and Timing simulation for Actel devices using the Cadence NC-VHDL simulator.

Behavioral Simulation

After you have coded the VHDL descriptions of the logic blocks, test and debug the design using the NC-VHDL simulator. Use the following procedure to perform behavioral simulation.

1. Create a work directory and a cds.lib file in the project directory. At the UNIX prompt, type:

```
mkdir work
```

2. Using a text editor, create a cds.lib file and enter the following lines:

```
INCLUDE $CDS/tools/inca/files/cds.lib
```

3. Map to the Actel VITAL library. If any Actel cores are instantiated in your VHDL source, add the following lines to your cds.lib file to map them to the compiled Actel VITAL library.

```
DEFINE <vhd_fam> $ALSDIR/lib/vt1/95/ncvhd1/<vhd_fam>
DEFINE WORK ./work
```

Add the following lines to your VHDL design files to reference the Actel Family library in your VHDL design files:

```
library <vhd_fam>;
use <vhd_fam>.components.all;
```

4. Compile the VHDL design and testbench files. Type:

```
ncvhd1 -work work -messages <behavioral>.vhd
ncvhd1 -work work -messages <test_bench>.vhd
```

5. Elaborate the design. Type the following commands at the prompt:

```
ncelab -work work -messages <configuration_name>
```

In the above, <configuration_name> is the name of the configuration in the testbench.

For example

```
ncelab -work work -messages test_add_behave
```

The entity-architecture pair specified by the configuration named `test_adder_behave` in the testbench will be elaborated.

6. **Simulate the design.** Type the following command at the prompt:

```
ncsim -batch -run <configuration_name>
```

In the above, `<configuration_name>` is the name of the configuration in the testbench.

For example:

```
ncsim -batch -run test_add_behave
```

Structural Simulation

Use the following procedure to perform structural simulation.

1. **Map to the Actel VITAL library.** Create a `cds.lib` file in the project directory as follows:

```
INCLUDE $CDS/tools/inca/files/cds.lib  
DEFINE <vhd_fam> $ALSDIR/lib/vt1/95/ncvhd1/<vhd_fam>  
DEFINE WORK ./work
```

2. **Compile the structural netlist.** Compile your VHDL design and testbench files. If you have not already generated a structural netlist, go to [“Generating a Structural VHDL Netlist” on page 15](#). The following commands demonstrate how to compile VHDL design and testbench files:

```
ncvhd1 -work work -messages <structural>.vhd  
ncvhd1 -work work -messages <test_bench>.vhd
```

3. **Elaborate the design.** Type:

```
ncelab -work work -messages <configuration_name>
```

In the above, `<configuration_name>` is the name of your configuration that binds the testbench entity and architecture.

For example:

```
ncelab -work work -messages test_adder_structure
```

In the example, “`test_adder_structure`” is the name of configuration for the testbench.

4. **Simulate the design.** Type following at the prompt:

```
ncsim -batch -run <configuration_name>
```

In the above, <configuration_name> is name of the testbench configuration.

For example:

```
ncsim -batch -run test_adder_structure
```

Here the name of the testbench configuration is “test_adder_structure.”

Timing Simulation

Use the following procedure to perform timing simulation.

1. **Place-and-route your design in Designer.** Refer to the *Designer User's Guide* for information about using Designer.
2. **Extract timing information for your design.** From the File menu, click Export. Then, click Timing Files. Choose SDF and click Save (or click Back Annotate). The Back Annotate dialog box is displayed. Create a <design_name>.sdf file by specifying SDF as the CAE type. Click OK.
3. **Map Actel VITAL library.** If not already done for structural simulation, create a cds.lib file in the project directory using the following command:

```
INCLUDE $CDS/tools/inca/files/cds.lib
DEFINE <vhd_fam> $ALSDIR/lib/vtl/95/ncvhd1/<vhd_fam>
DEFINE WORK ./work
```

4. **Compile the structural netlist.** Compile your VHDL design and testbench files. If you have not already generated a structural netlist, go to [“Generating a Structural VHDL Netlist” on page 15](#). The following commands demonstrate how to compile VHDL design and testbench files:

```
ncvhd1 -work work -messages <structural>.vhd
ncvhd1 -work work -messages <test_bench>.vhd
```

5. **Compile the SDF file.** Type:

```
ncsdfc <design_name>.sdf
```

6. **Write an SDF command file.** Using a text editor, create the “<design_name>.sdf_cmd” file. Include the following lines:

```
COMPILED_SDF_FILE = "<design_name>.sdf.x"
```

```
SCOPE = :<UUT> ,
```

```
MTM_CONTROL = "MINIMUM/TYPICAL/MAXIMUM";
```

In the above, <design_name> is the name of the top-level entity, <UUT> is the instance of the top-level entity in the testbench, and <configuration_name> is the name of your configuration that binds the testbench entity and architecture.

7. Elaborate the design. Type:

```
ncelab -work work -messages -sdf_cmd_file <design_name>.sdf_cmd  
<configuration_name>
```

For example:

```
ncelab -work work -messages -sdf_cmd_file adder.sdf_cmd  
test_adder_structure
```

In the above example, “adder” is the name of the top-level entity, “test_adder_structure” is the name of the configuration for the testbench.

8. Simulate the design. Type the following at the prompt:

```
ncsim -batch -run <configuration_name>
```

In the above, <configuration_name> is the name of the testbench configuration.

For example:

```
ncsim -batch -run test_adder_structure
```

Here the name of the testbench configuration is test_adder_structure.

Product Support

Actel backs its products with various support services including Customer Service, a Customer Technical Support Center, a web site, an FTP site, electronic mail, and worldwide sales offices. This appendix contains information about contacting Actel and using these support services.

Customer Service

Contact Customer Service for non-technical product support, such as product pricing, product upgrades, update information, order status, and authorization.

From Northeast and North Central U.S.A., call **650.318.4480**

From Southeast and Southwest U.S.A., call **650.318.4480**

From South Central U.S.A., call **650.318.4434**

From Northwest U.S.A., call **650.318.4434**

From Canada, call **650.318.4480**

From Europe, call **650.318.4252** or **+44 (0)1276.401500**

From Japan, call **650.318.4743**

From the rest of the world, call **650.318.4743**

Fax, from anywhere in the world **650.318.8044**

Actel Customer Technical Support Center

Actel staffs its Customer Technical Support Center with highly skilled engineers who can help answer your hardware, software, and design questions. The Customer Technical Support Center spends a great deal of time creating application notes and answers to FAQs. So, before you contact us, please visit our online resources. It is very likely we have already answered your questions.

Actel Technical Support

Visit the [Actel Customer Support website \(www.actelcom/.custsup/search.html\)](http://www.actelcom/.custsup/search.html) for more information and support. Many answers available on the searchable web resource include diagrams, illustrations, and links to other resources on the Actel web site.

Website

You can browse a variety of technical and non-technical information on Actel's [home page](http://www.actel.com), at www.actel.com.

Contacting the Customer Technical Support Center

Highly skilled engineers staff the Technical Support Center from 7:00 A.M. to 6:00 P.M., Pacific Time, Monday through Friday. Several ways of contacting the Center follow:

Email

You can communicate your technical questions to our email address and receive answers back by email, fax, or phone. Also, if you have design problems, you can email your design files to receive assistance. We constantly monitor the email account throughout the day. When sending your request to us, please be sure to include your full name, company name, and your contact information for efficient processing of your request.

The technical support email address is tech@actel.com.

Phone

Our Technical Support Center answers all calls. The center retrieves information, such as your name, company name, phone number and your question, and then issues a case number. The Center then forwards the information to a queue where the first available application engineer receives the data and returns your call. The phone hours are from 7:00 A.M. to 6:00 P.M., Pacific Time, Monday through Friday. The Technical Support numbers are:

650.318.4460

800.262.1060

Customers needing assistance outside the US time zones can either contact technical support via email (tech@actel.com) or contact a local sales office. [Sales office listings](#) can be found at www.actel.com/contact/offices/index.html.

Index

A

- Actel
 - FPGA Libraries 15
 - web site 50
 - web-based technical support 49
- Actel Manuals 7
- Actel Project Setup 14, 16
 - Project Manager 14
 - SpeedWave 15
- Actel VITAL Libraries
 - Creating a swave Directory 11
- Analyzing
 - Test Bench 37, 38, 39
 - VHDL Source File 37, 38, 39
- Assumptions 5

B

- Back Annotate 39
- Behavioral Simulation 37
 - Cadence Leapfrog Simulator 31
 - Model Technology V-System Simulator 25
 - SpeedWave 20, 37

C

- Capturing a Design
 - VHDL-Based 20
- Configuration 37, 38, 39
- Contacting Actel
 - customer service 49
 - electronic mail 50
 - telephone 50
 - toll-free 49
 - web-based technical support 49
- Conventions 6
 - Document 6--??
- Creating

- swave Directory 11
- Customer service 49

D

- Delay Mode 39
- Design Creation/Verification 20
 - Behavioral Simulation 20
 - EDIF Netlist Generation 21
 - Structural Simulation 21
 - VHDL Source Entry 20
- Design Flow
 - Design Creation/Verification 20
 - Design Implementation 21
 - Programming 22
 - Schematic-Based 22
- Design Implementation 21
 - Place-and-Route 21
 - Timing Analysis 21
 - Timing Simulation 21
- Design Layout 21
 - Model Technology V-System Simulator 28
- Designer
 - EDIF Option 23
 - GENERIC Option 23
 - Place-and-Route 21
 - Software Installation Directory 5
 - Structural Netlist Generation 23
 - Timer Tool 21
 - Timing Analysis 21
 - VHDL Option 21, 23
 - VIEWLOGIC Option 21
- Device
 - Programming 22
 - Verification 22
- Document Assumptions 5
- Document Conventions 6, 6--??

E

- EDIF Netlist Generation 23
 - Synthesis-Based 21
- EDIF Option 23
- edn2vhdl 23, 26
- Electronic mail 50

F

- FPGA Libraries 15

G

- Generating
 - EDIF Netlist 21, 23
 - Structural Netlist 23
- Generating an EDIF netlist
 - Model Technology V-System Simulator 26
 - Viewlogic Vantage Speedwave Simulator 41
- GENERIC Option 23

I

- Importing Timing Information 39

L

- Libraries
 - Actel FPGA 15
 - Project 15

N

- Netlist Generation
 - EDIF 21

O

- Online Help 7

P

- Place-and-Route 21
- Postsynthesis Simulation 21
- Primary Directory 14
- Product Support 49–51
- Product support
 - customer service 49
 - electronic mail 50
 - oll-free line 49
 - technical support 49
 - web site 50
- Programming a Device 22
- Project Manager
 - Project Setup 14
 - Selecting an Actel FPGA Library 15
 - Setting the Primary Directory 14
- Project Setup 16
 - Project Manager 14
 - SpeedWave 15

S

- Schematic-Based Design Flow 22
- Scope Options 39
- Selecting an Actel FPGA Library 15
- Selecting Scope Options 39
- Setting the Primary Directory 14
- Setting Up
 - an Actel Project in Project Manager 14
 - an Actel Project in SpeedWave 15
- Setup Procedure
 - Cadence Leapfrog Simulator (VITAL 95) 11
- Setup Procedures 16
 - Creating a Project Library 15
 - Project Setup 16
 - Setting Up an Actel Project 14
 - User Setup 12

Simulation

- Behavioral 20
 - SpeedWave
 - Behavioral Simulation 37
 - Configuration 37, 38, 39
 - Postsynthesis 21
 - SpeedWave 20, 21
 - Structural 21, 37
 - Synthesis-Based 20, 21
 - Synthesis-Based Simulation
 - SpeedWave 37
 - Test Bench 37, 38, 39
 - Timing 21
- SpeedWave
- Behavioral Simulation 20
 - Creating a Project Library 15
 - Importing Timing Information 39
 - Postsynthesis Simulation 21
 - Scope Options 39
 - Structural Simulation 21, 37
 - Timing Simulation 21
- Static-Timing Analysis 21
- Structural Netlist Generation
- Designer 23
 - EDIF Option 23
 - edn2vhdl 23, 26
 - GENERIC Option 23
- Structural Simulation 21, 37
- Cadence Leapfrog Simulator 32
 - Model Technology V-System Simulator 26
 - SpeedWave 21, 37
 - Viewlogic Vantage Speedwave Simulator 41
- swave Directory 11
- System Verification 22
- Silicon Explorer 22

T

- Test Bench 37, 38, 39
 - Modifying 21
- Timer, Static-Timing Analysis 21
- Timing Analysis 21
- Timing Information 39
- Timing Simulation 21
 - SpeedWave 21
- Toll-free line 49

U

- Unit Delays 20
- User Setup 12

V

- Verilog Design Flow
 - Programming 22
- VHDL
 - Option 23
- VHDL Option 21
- VHDL Source Entry 20
- VHDL Source File
 - Analyzing 37, 38, 39
- VHDL Synthesis-Based Design Flow
 - Design Creation/Verification 20
 - Design Implementation 21
- VIEWLOGIC Option 21

W

- Web-based technical support 49

***For more information about Actel's products, visit our website at
<http://www.actel.com>***

Actel Corporation • 2061 Stierlin Court • Mountain View, CA 94043 USA
Customer Service: 650.318.1010 • Customer Applications Center: 800.262.1060

Actel Europe Ltd. • River Court, Meadows Business Park • Station Approach, Blackwater • Camberley, Surrey GU17 9AB
United Kingdom • Phone +44 (0) 1276 609 300 • Fax +44 (0) 1276 607 540

Actel Japan • EXOS Ebisu Bldg. 4F • 1-24-14 Ebisu Shibuya-ku • Tokyo 150 • Japan
Phone +81.03.3445.7671 • Fax +81.03.3445.7668 • www.jp.actel.com

Actel Hong Kong • Suite 2114, Two Pacific Place • 88 Queensway, Admiralty Hong Kong
Phone +852 2185 6460 • Fax +852 2185 6488 • www.actel.com.cn

5579006-11/11.06

