Quickstart Guide to Using
NCVHDL®

Table of Contents

1. Introduction

This manual is specifically targeted for students taking the CS147 and CS247 Computer Architecture classes under Dr. Robert Chun. It serves as a tutorial for getting started on the Cadence Logical Verification tool kit.

Chapter1 deals with the brief introduction of UNIX commands. This would enable students to perform basic file management operations through the console (terminal). Also discussed in this chapter are the available text editor options under the Solaris environment.

Chapter 2 focuses on the method to initialize and start NCVHDL (core Cadence verification environment).

Chapter 3 provides detailed instructions for compiling, linking, and simulating design units written in VHDL using the Cadence tool kit.

2. Chapter 1

The Unix operating system is case sensitive. In general, commands in Unix are lower case. Commands can include three parts: a command, an option, and an argument.

A command is a word or group of characters that Unix recognizes as a request to perform a specific task.

An option, as the name implies, is not mandatory. Options give extra functions to a command; they are always preceded by a dash (-).

An argument is a variable supplied by the user. Often, it represents the name of a file or directory targeted for action.

For example, the Unix command **ls -F dir_name** uses the command **ls** to request a listing, the option **F** to specify a detailed listing, and to only list files and directories in the directory specified in the argument **dir_name**.
Note: The Unix commands are issued through the terminal or a console. Refer to section 3.3 for more information.

Following is list of common Unix commands under each category.

Help Commands

man
Displays information about specific commands. For example, **man cat** gives information about the command **cat**. Man (short for manual) pages describe commands and allowable options, and gives examples.

whatis
Displays a short description of a command's function. For a short description of the **cat** command, type **whatis cat**.

Directory operations

pwd
Prints the full name of the current directory (the physical path).

cd
This command is used to change the directory. The command **cd /** ***home*** changes the directory path to */home*. Typing **cd ..** moves up one directory (parent directory); typing **cd** or **cd ~** changes to the home directory */home/user_name*.

ls
It lists the contents of the current directory. Typing **ls -a** lists all files in the directory, including hidden or system files that begin with a period. Typing **ls –F** lists the files and directories in a folder/file view.

mkdir
This command creates directories. To make a directory named *test*, type **mkdir *test***

rmdir
It deletes directories. The command **rmdir *test*** removes the directory *test.* Only empty directories can be removed. To remove non-empty directories type **rmdir –r *dir_name.***

Manipulating Files

cp
This copies a file. For example, **cp *copy1 copy2*** creates a new file, called *copy2*, identical to *copy1*.

mv
It renames a file. The command **mv *old new*** gives the file: *old* the new name: *new.* This command also can be used to rename directories.

rm
This removes a file. For example, **rm *old*** deletes the file *old*.

Creating and Viewing Text Files

cat
This displays a text file. Typing **cat *file1*** will display the file *file1*. If the file is not present then an empty file with the same name is opened.

more
Displays a text file, pausing at the bottom of each page. Useful when viewing longer files. The command **more *file1*** displays the file *file1*; pressing the **<Return>** key shows the next line; pressing the **<Space Bar>** displays the next page; typing **b** displays the previous page.

3. Chapter 2

3.1. Desktop

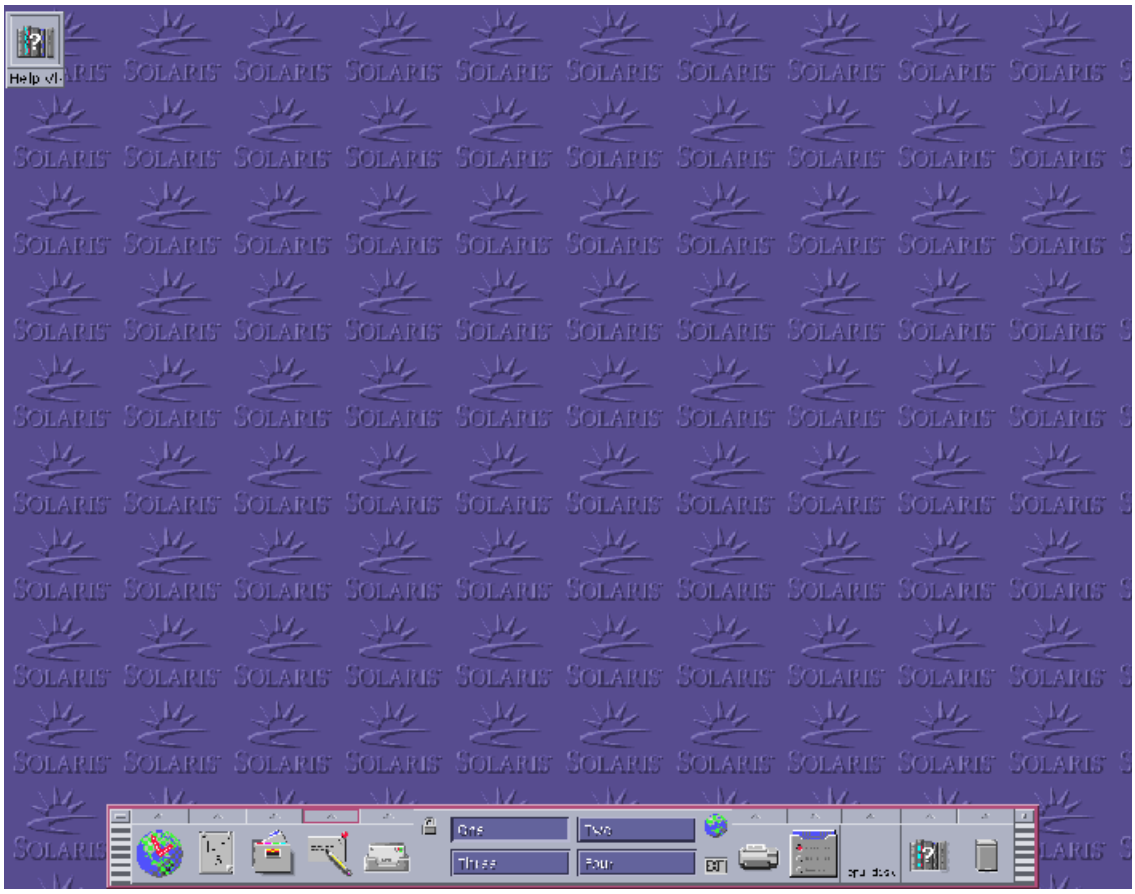

Fig 1: Start up interface on the Solaris environment.

3.2. Editor

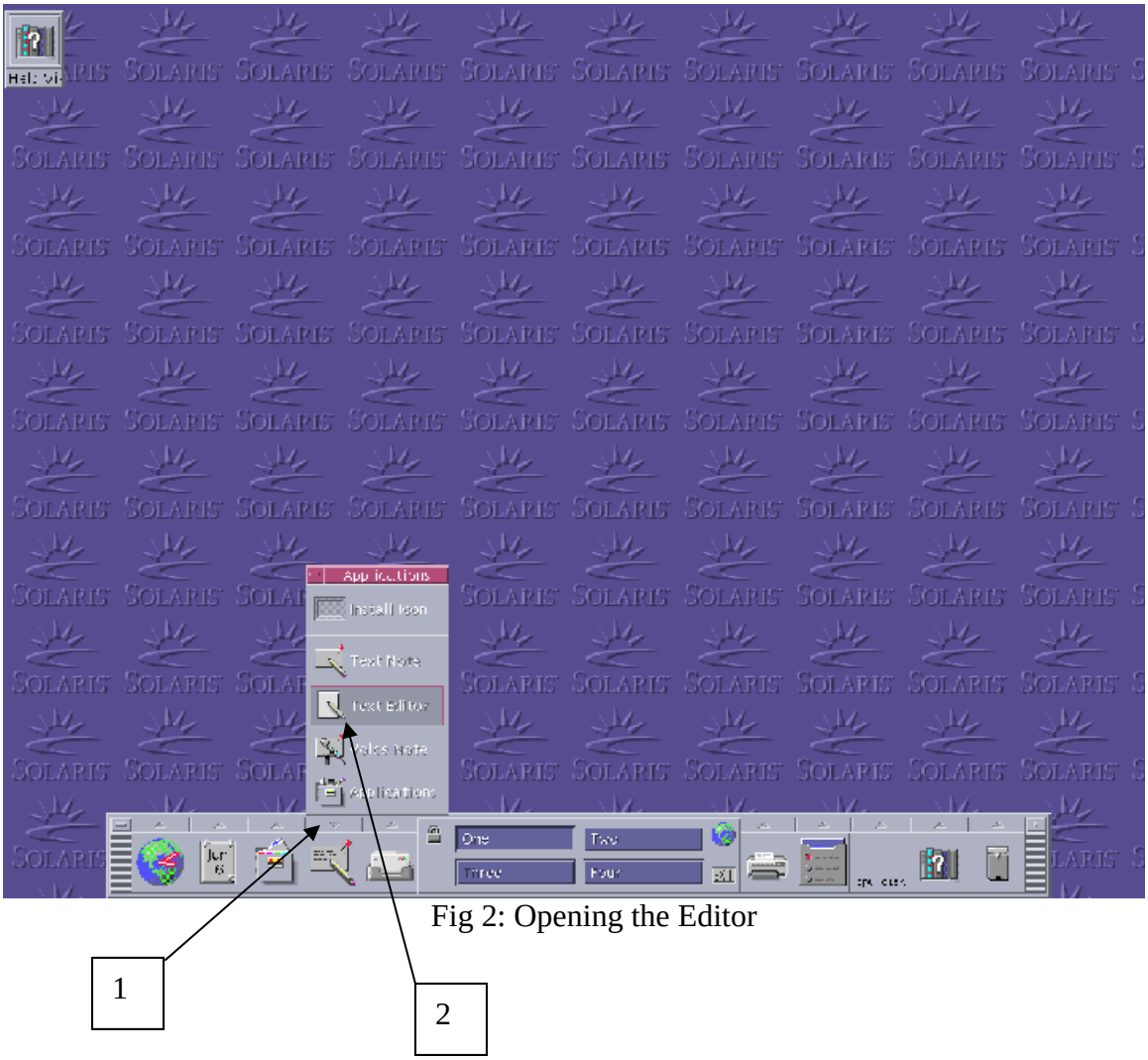

Fig 2: Opening the Editor

1. Click on the Editor tab in the menu.

2. Select the ‘Text Editor’ option to open a standard editor. The editor is much like the Notepad in Microsoft Windows®.

Tip:
Use this standard editor for keying in the design units in VHDL. Do not forget to save the files in ‘.vhd’ format in the desired location.

3.3. Console

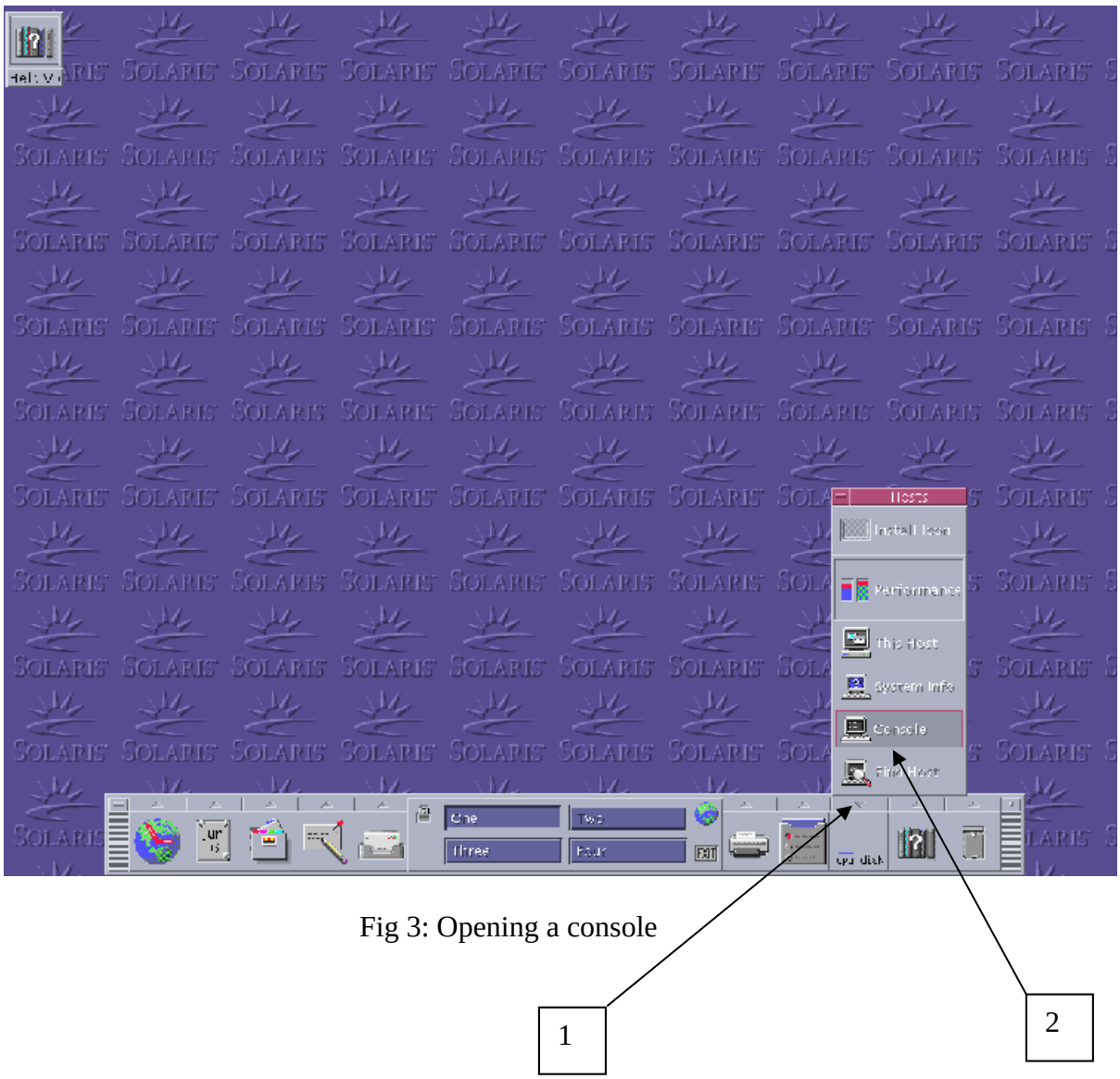

Fig 3: Opening a console

1. To open a terminal or console, click on the tab above the performance monitor menu option.

2. Then select the ‘Console’ option.

3.4. Start up script

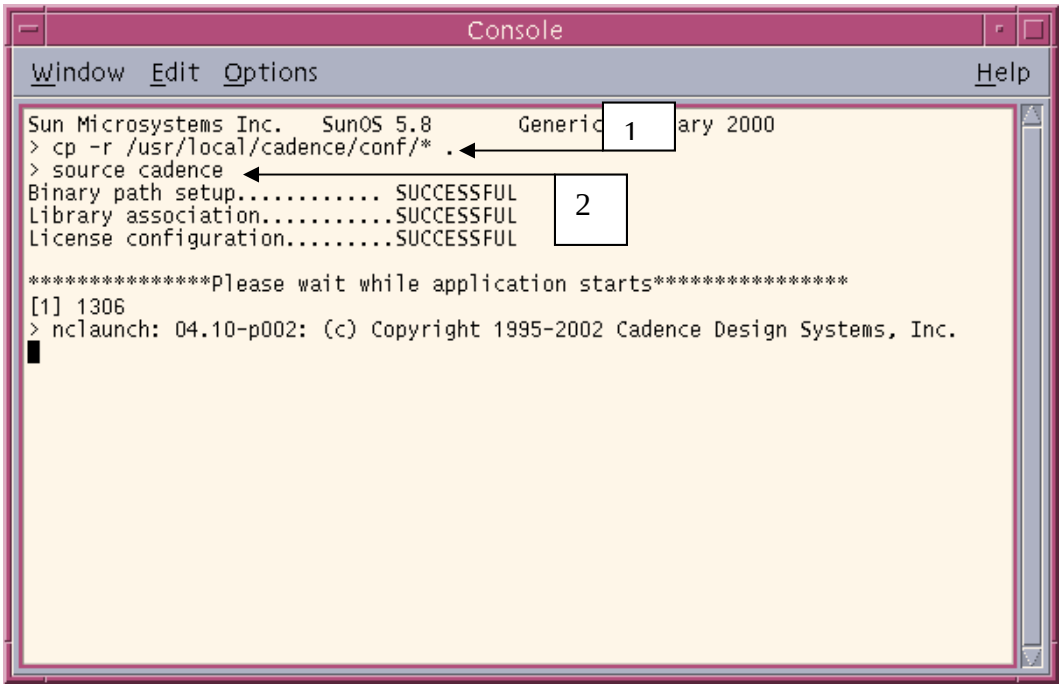

Fig 4: Running the script

1. If running the Cadence tool kit for the first time, copy the start up script and ADDER directory, present in the location /usr/local/cadence/conf to the home directory.

2. Run the script by typing in 'source cadence'. Please wait a few moments for the cadence environment (Fig 5) to start up. Then select the 'Multiple Step' option.

Fig 5: Cadence Logic Verification environment

4. Chapter 3

4.1. Design units

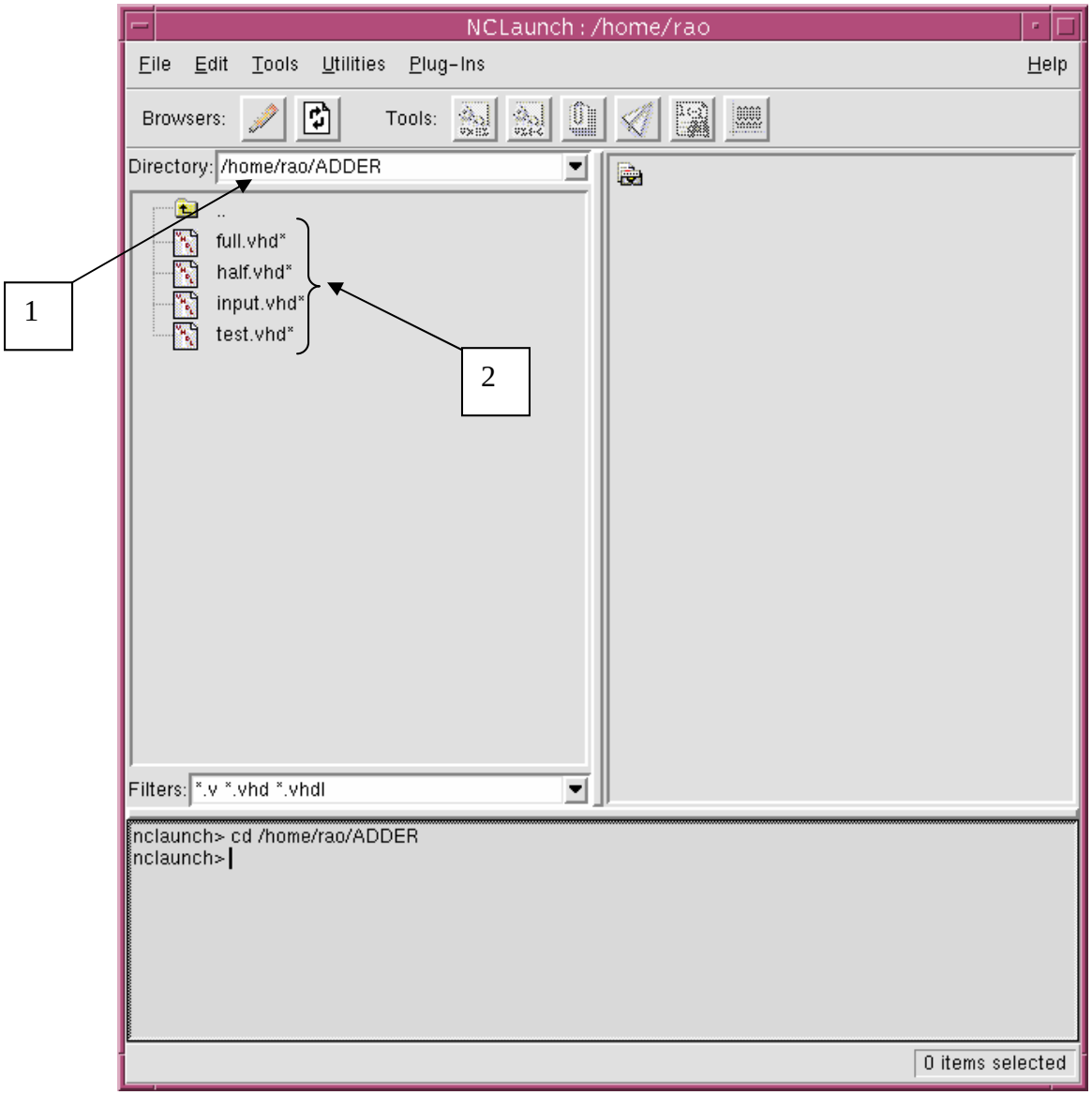

Fig 6: Design units

1. By default the start up script opens the cadence tool kit in the home directory '/home/*user_name*'. In order to run the sample ADDER design, change the location to '/home/rao/ADDER'.

2. The ADDER directory contains 4 design units as shown.

> Tip:
> To compile your own design units, select the respective location (directory) where your '.vhd' files are saved.

4.2. Design Directory

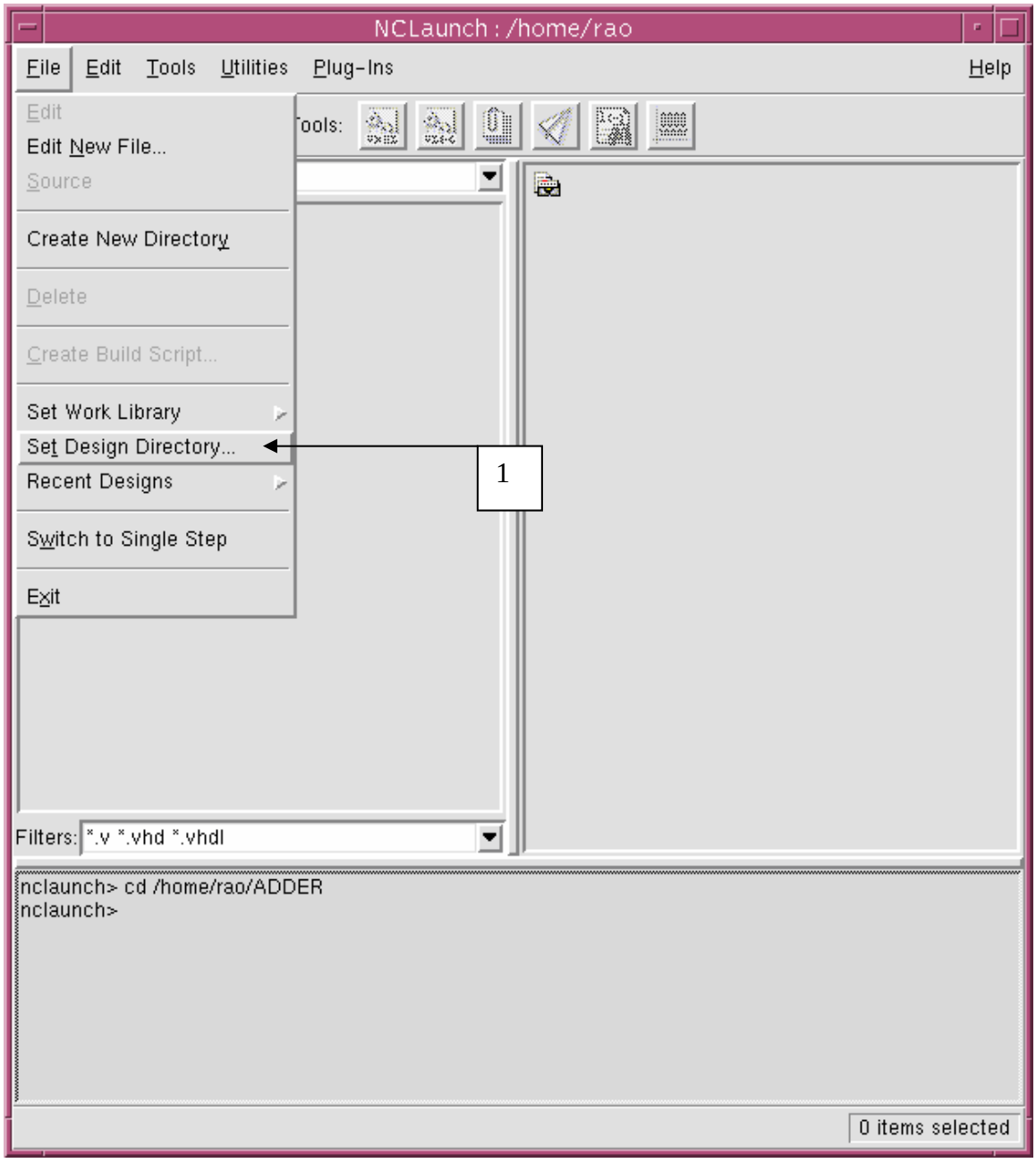

Fig 7: Design directory

1. The first step involved in the process of compiling the design units is to associate it with libraries. This is done by selecting ‘File’-> ‘Set Design Directory’ option.

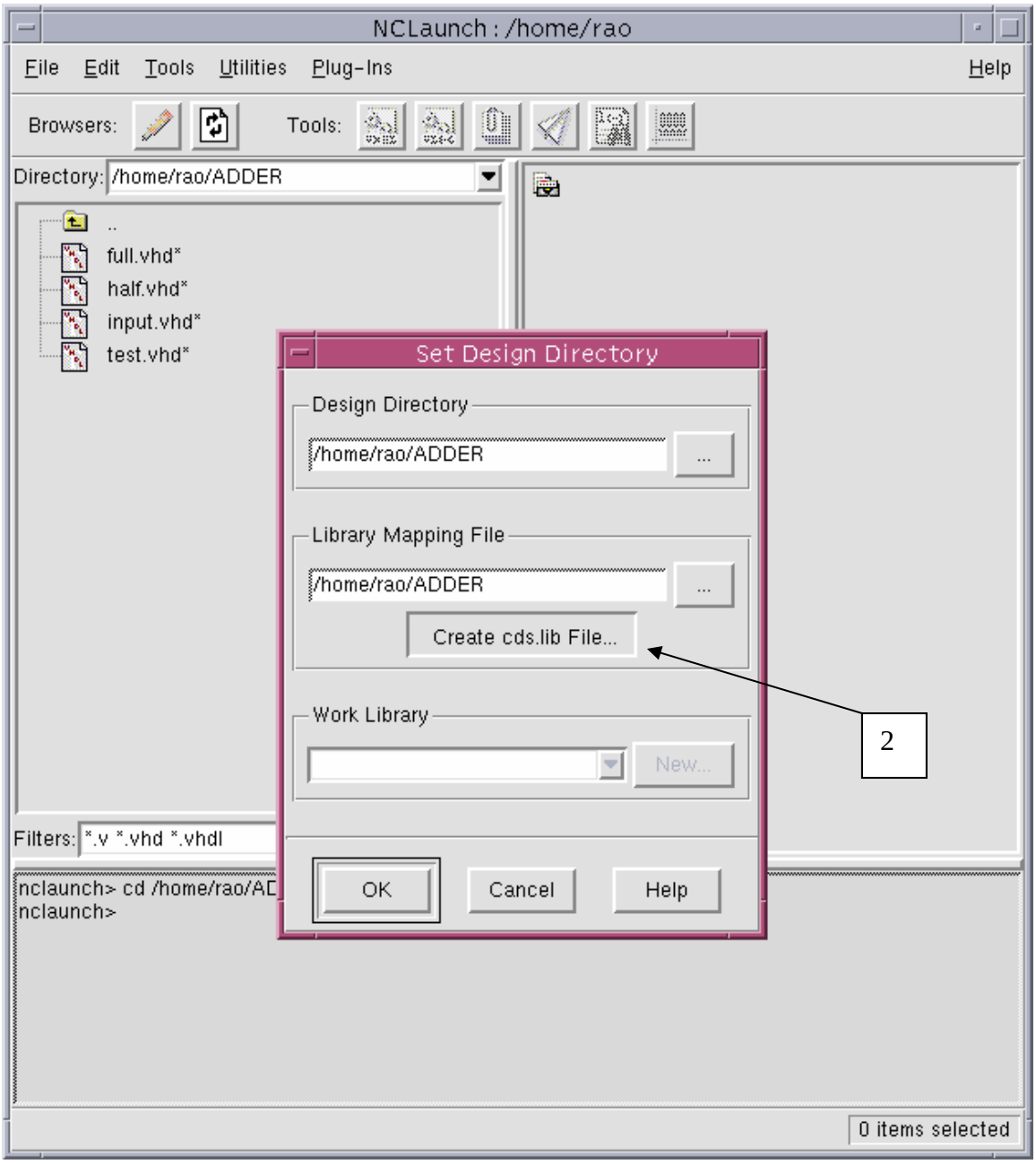

Fig 8: Design Directory window

2. Click on the 'Create cds.lib File' button in the set design directory window. Make sure the design directory and library mapping file are created in the location (directory) where the design units are stored.

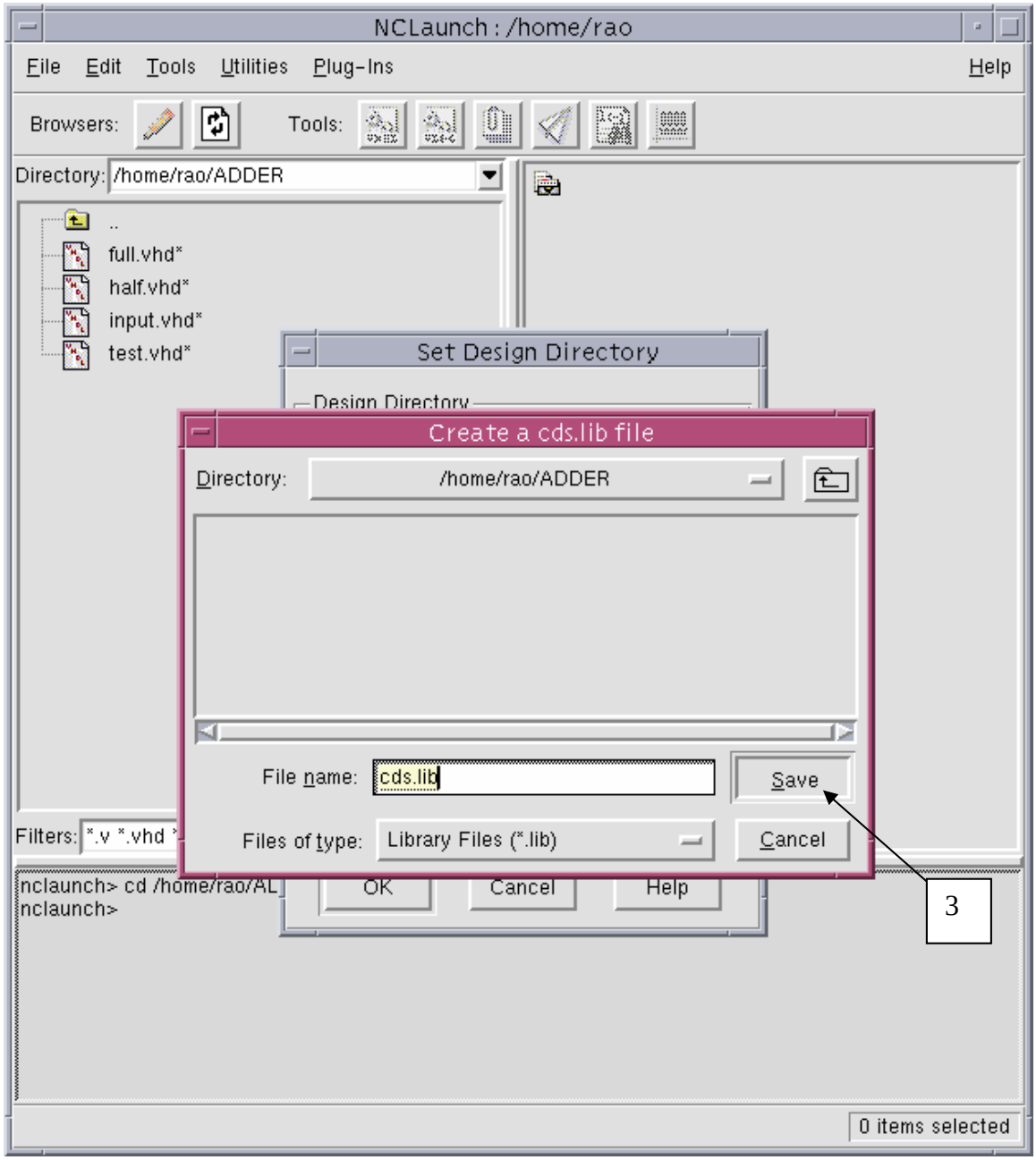

Fig 9: To create the library mapping

3. With the file name set as *cds.lib*, click on the 'Save' button.

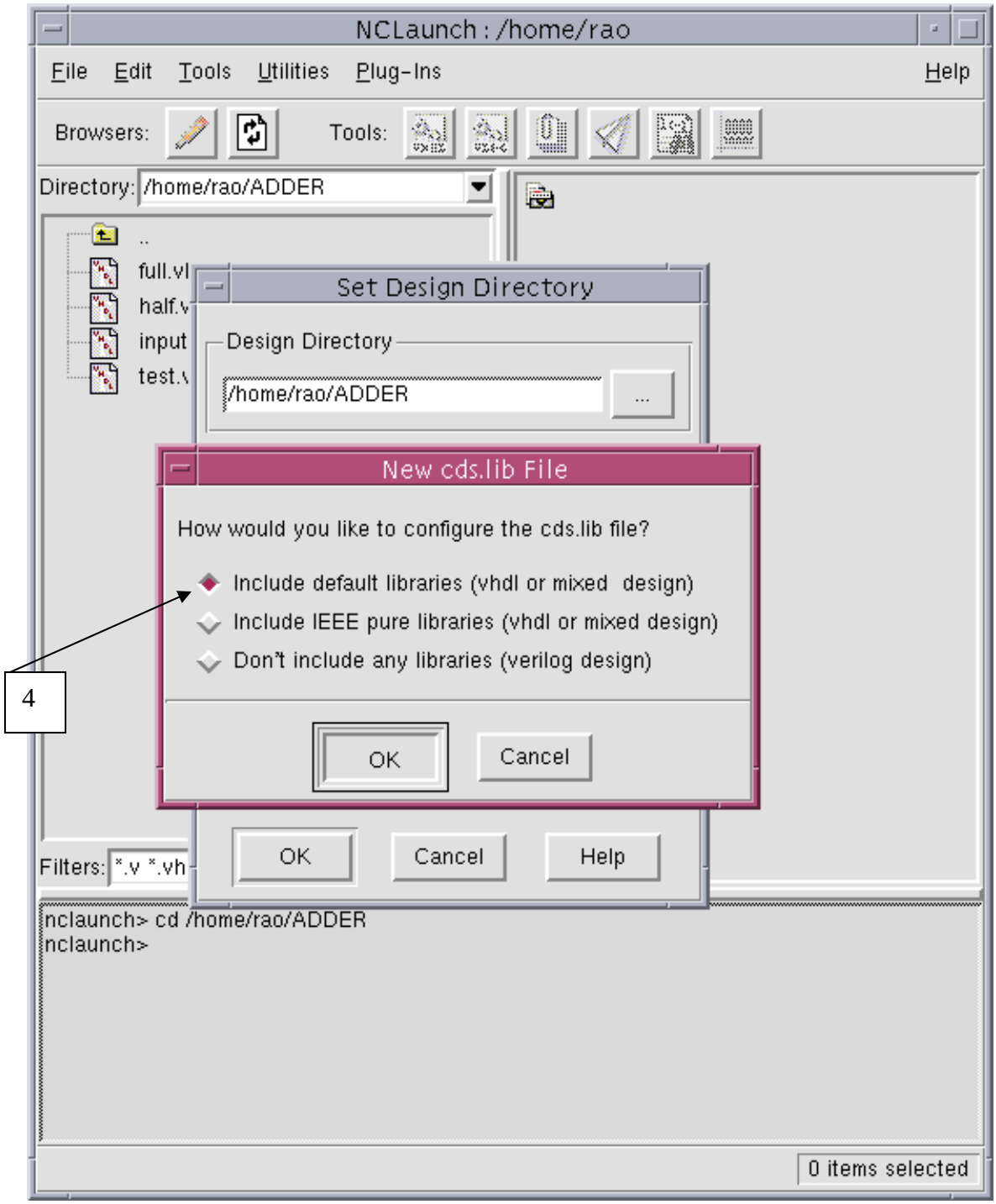

Fig 10: Include libraries

4. In the cds.lib file window, select the 'default library option' and then click ok.

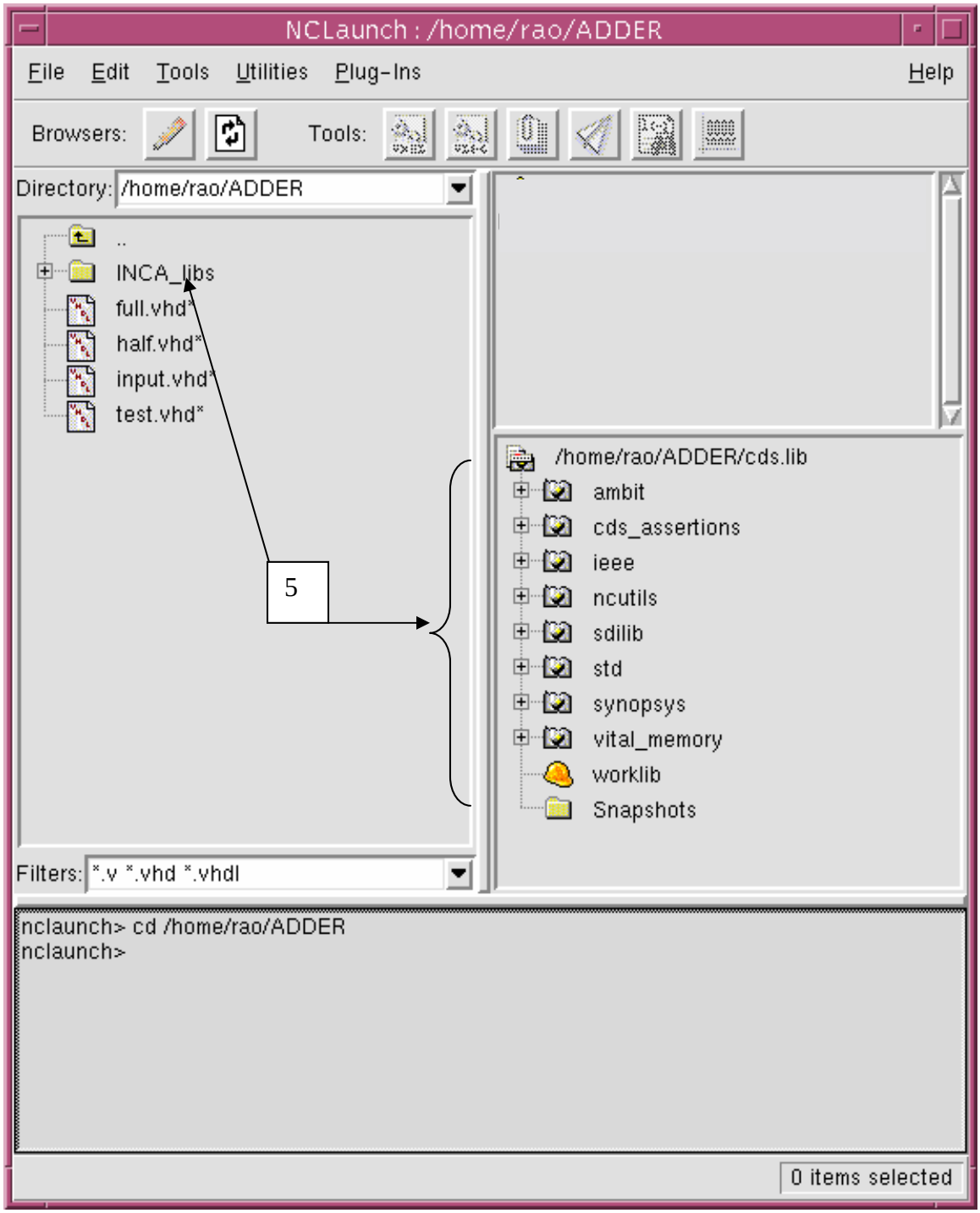

Fig 11: Final configuration

5. After associating the libraries with the design units, the verification environment gets populated as shown.

4.3. Compile

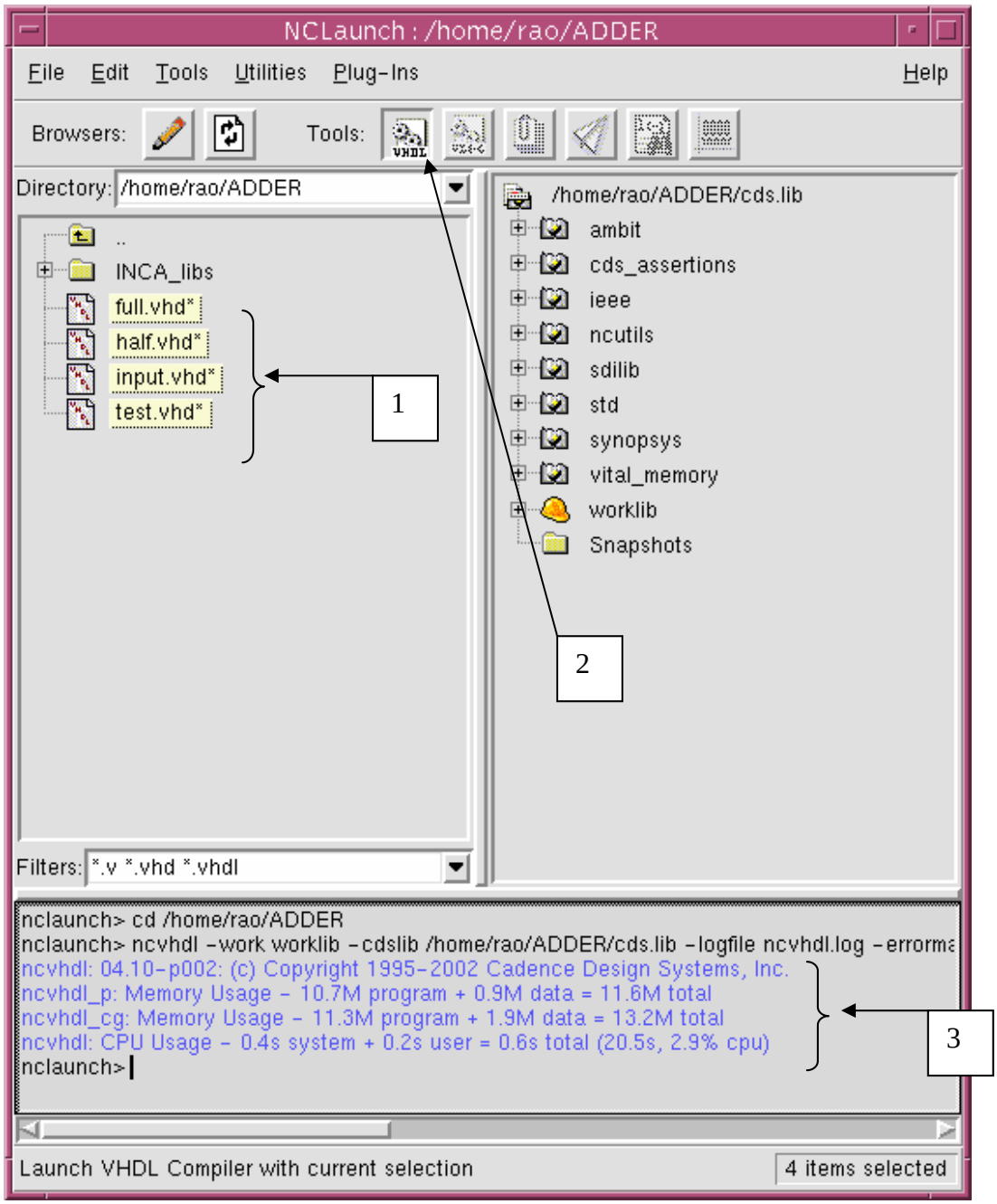

Fig 12: Compiling the design units

1. Select the design units in the correct order from the lowest design units to the top level units (hold won the CTRL keyboard button while selecting). In the adder design, select half.vhd, full.vhd, input.vhd, and test.vhd in that order.

2. Click on the VHDL compile icon on the menu.

3. The status indicates successful completion of the compilation process.

4.4. Elaboration

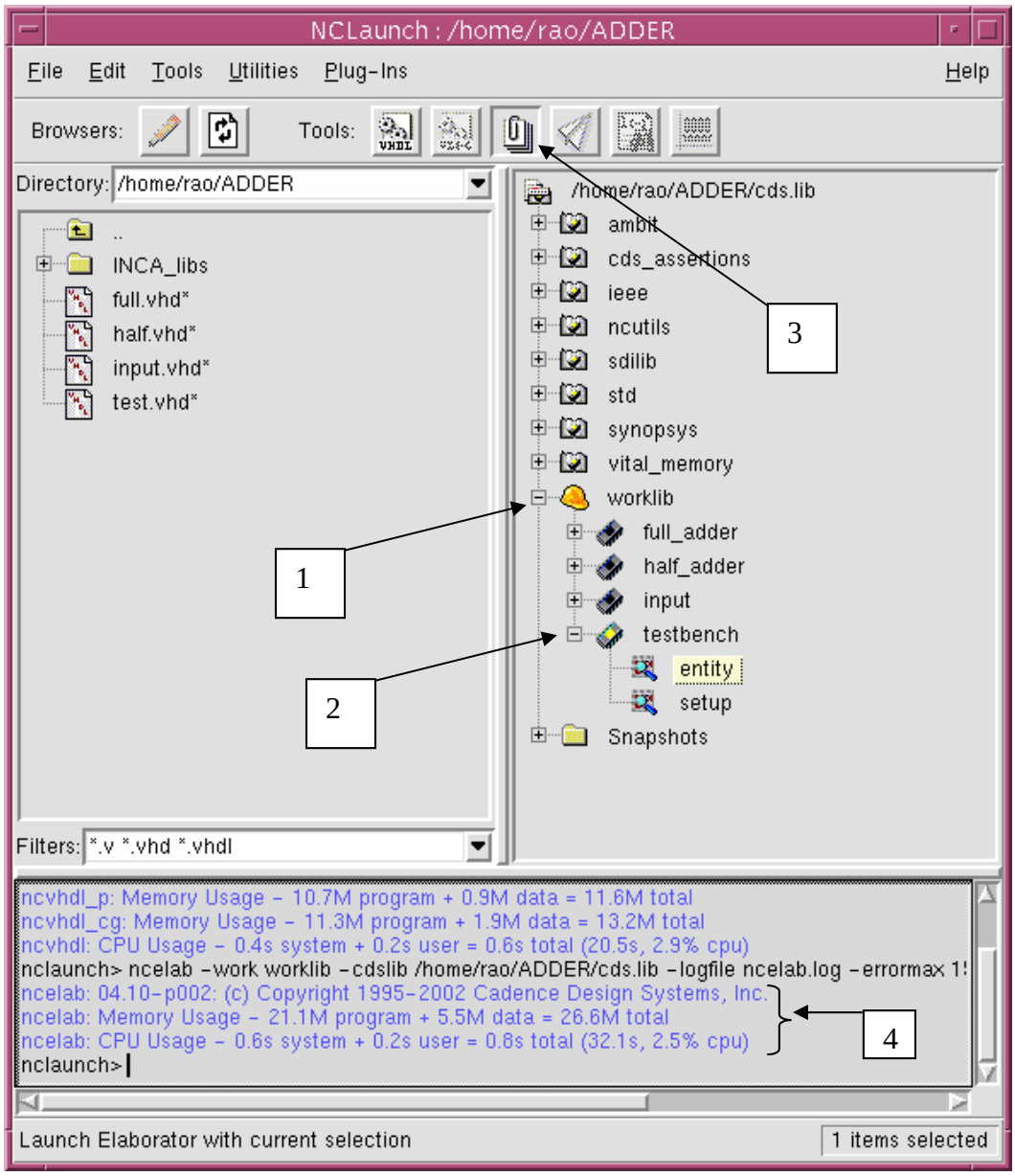

Fig 13: Elaboration phase

1. Expand ‘worklib’ (click on the + next to it).

2. Expand the top level design unit (usually the testbench), and select its entity.

3. Click on the Elaborator icon on the menu.

4. The status indicates that the elaboration phase was completed successfully.

4.5. Simulation

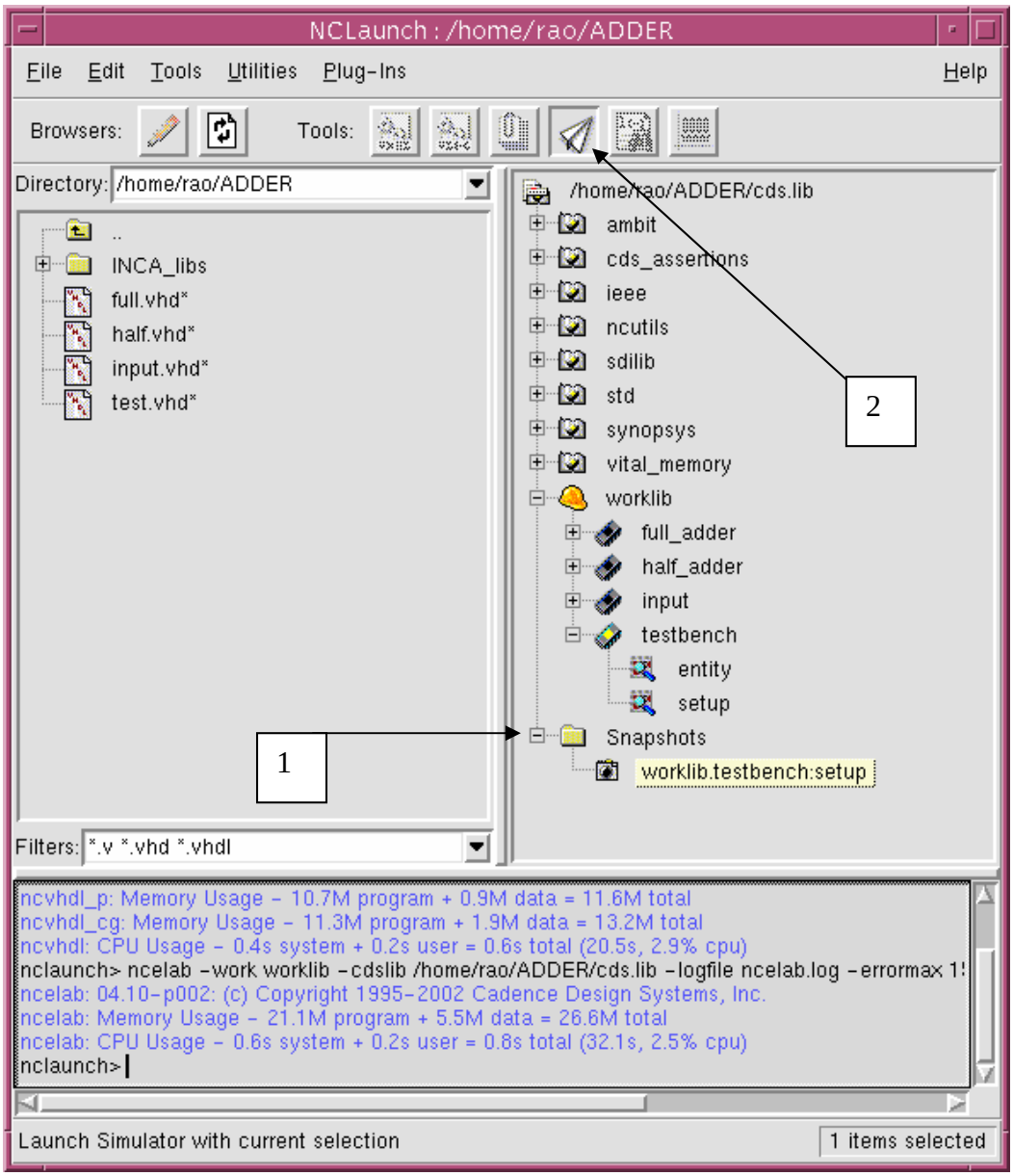

Fig 14: Enter the simulation environment

1. Expand the Snapshots directory and select the testbench.

   Note: The snapshot stores the system or design details and would be made use of during the simulation phase.

2. Then click on the Simulation icon on the menu to get the simulation environment loaded (Fig 15).

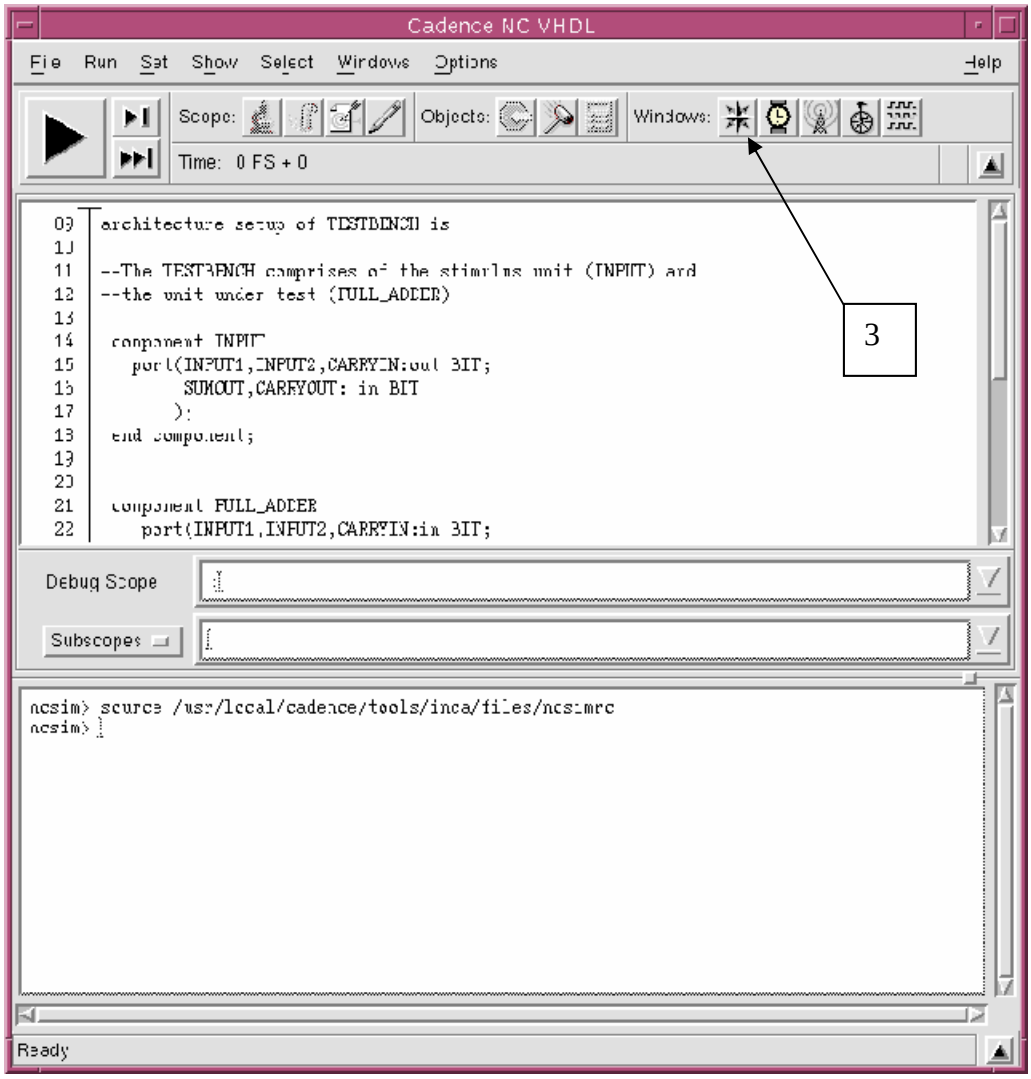

Fig 15: Simulation environment

3. Click on the Navigator icon in the menu to open the Navigator window (Fig 16).

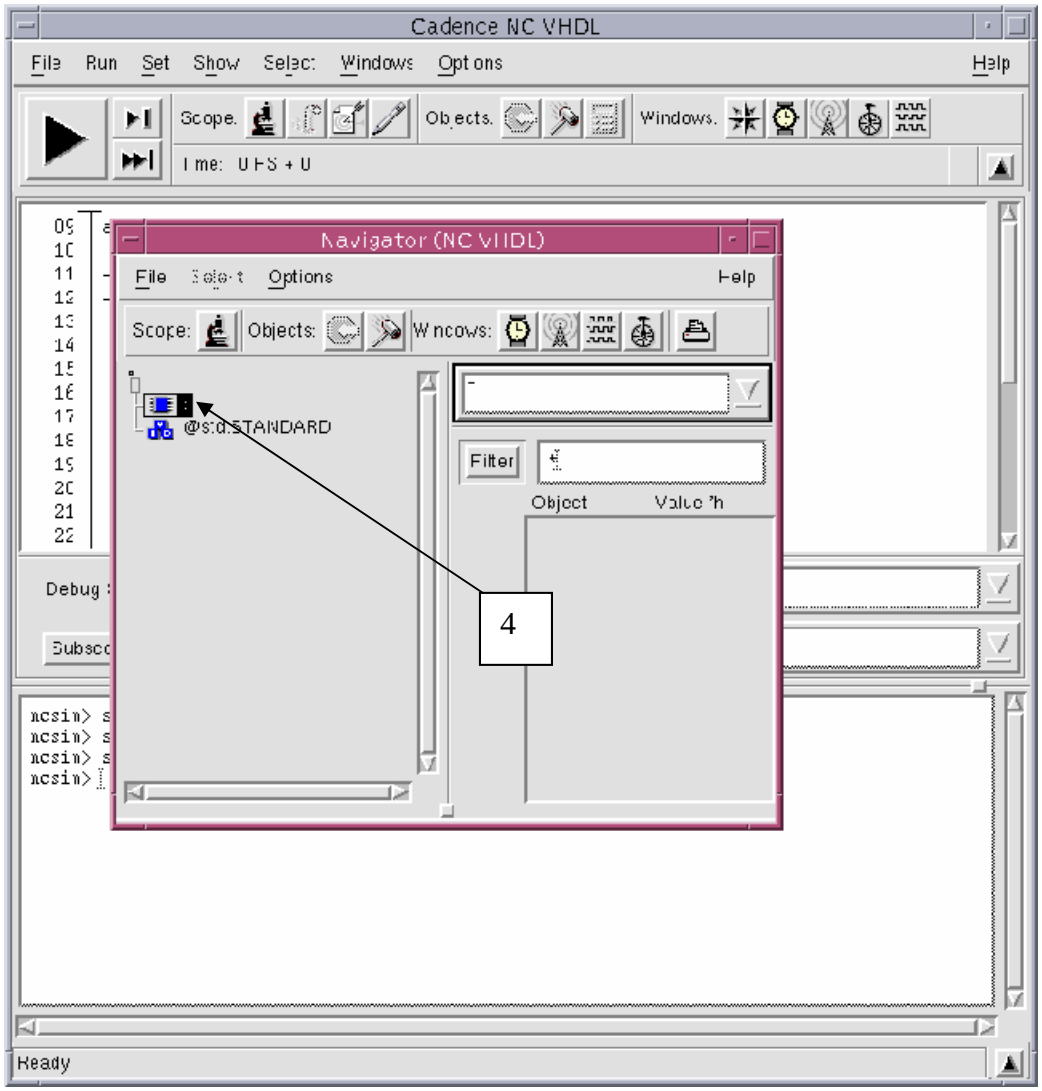

Fig 16: Navigator window

4. Select the top level entity, as shown.

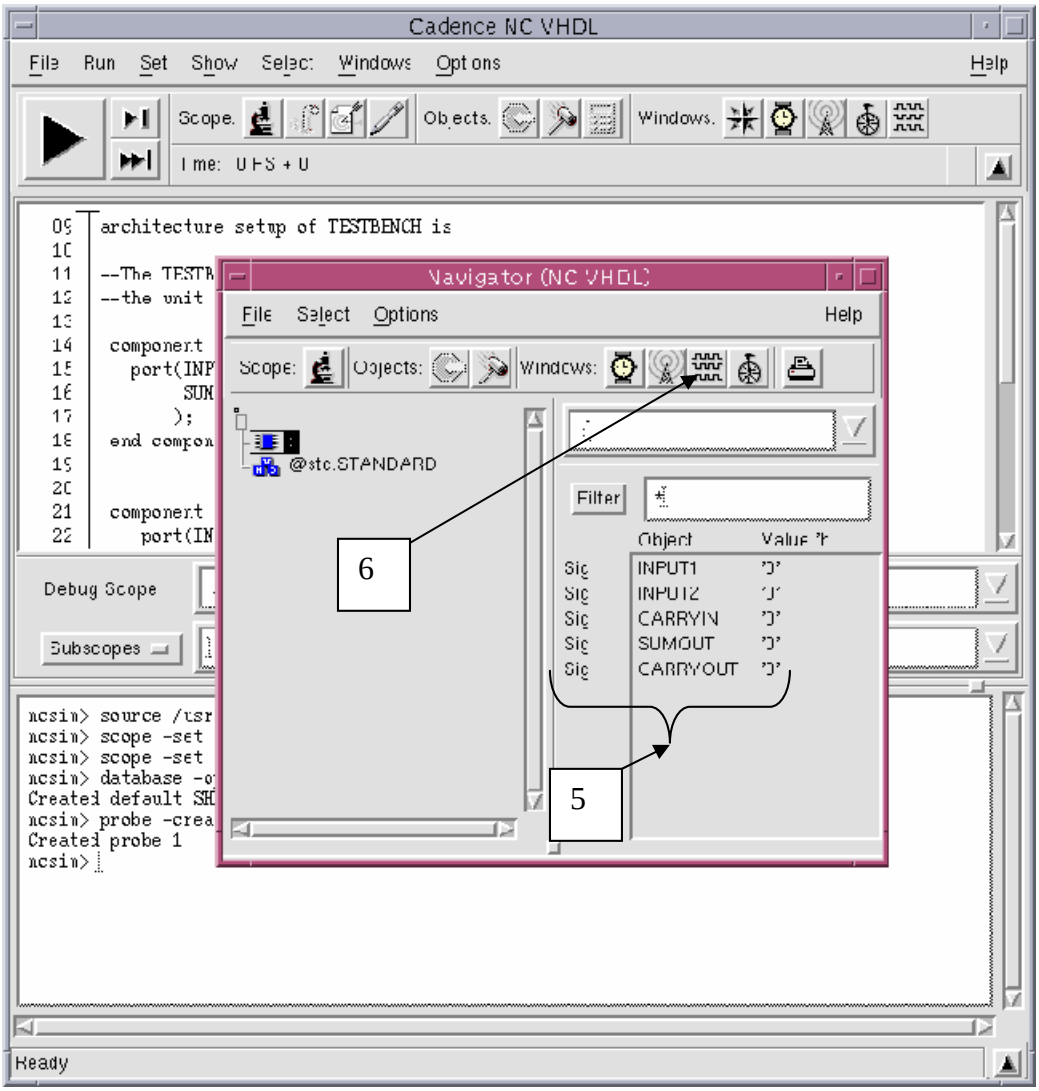

Fig 17: Object watch list

5. Upon selecting the top level entity, the signals available and their initial values are transferred to the signal (object) watch list.

6. Then click on the ‘waveform’ icon on the menu. This brings up the waveform window as shown in Fig 18.

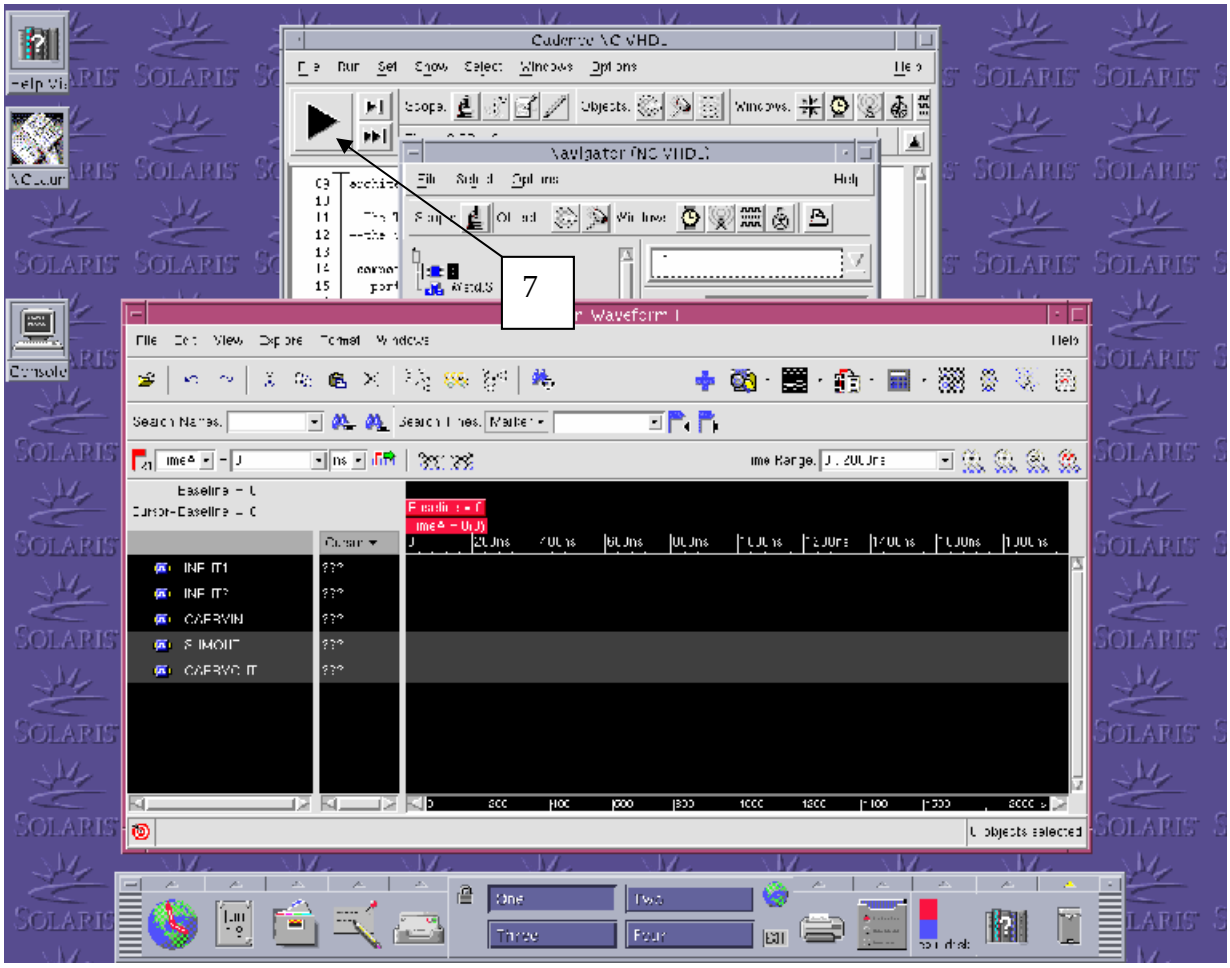

Fig 18: Waveform window

7. Click the ‘Run’ button on the first simulation window opened, as shown. This generates the waveform as see in Fig 19.

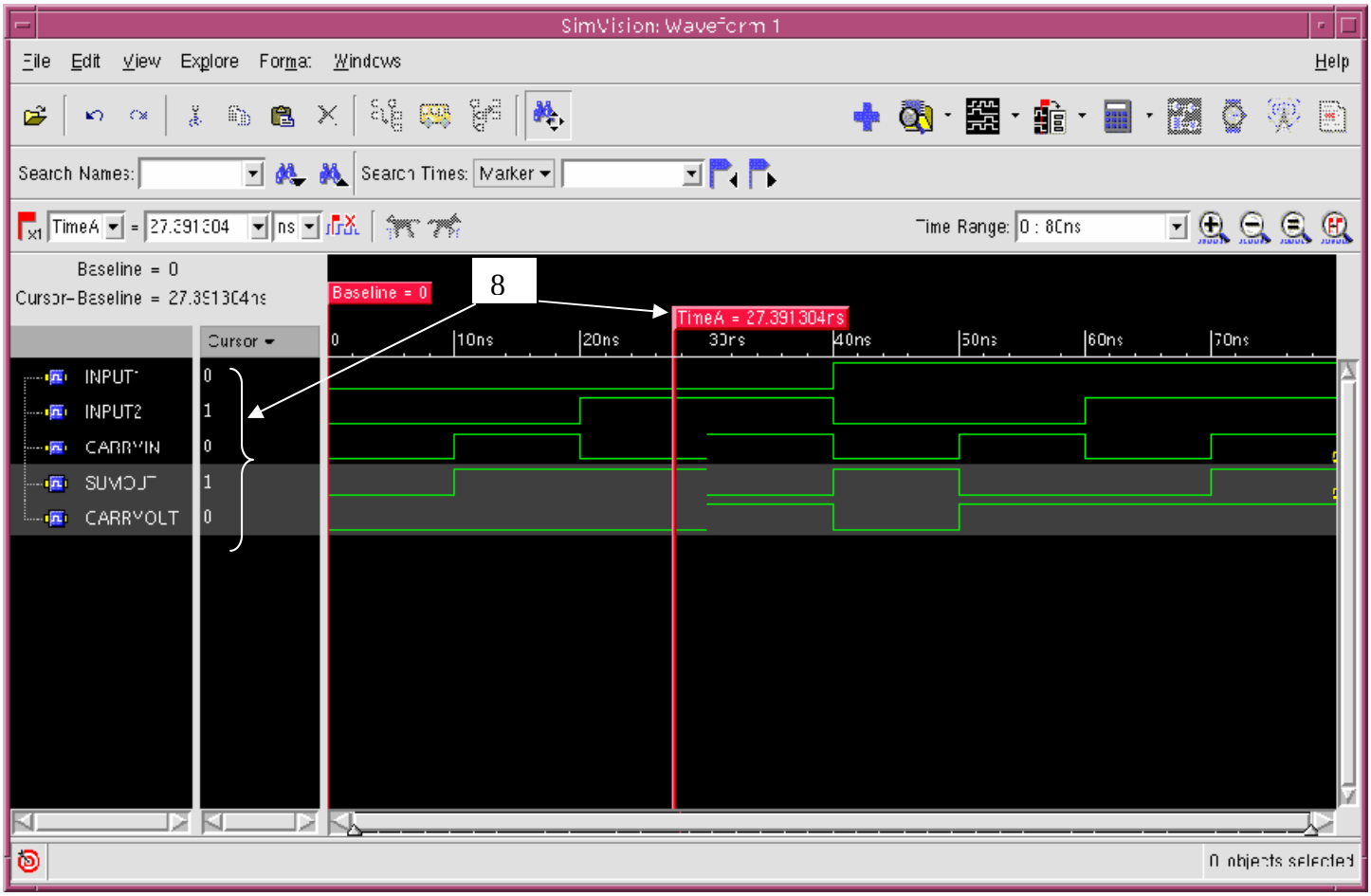

Fig 19: Generated waveform.

Fig 19 shows the waveform generated for the design unit under test.

8. By clicking specific positions on the waveform, the signal values associated at that instance of time is indicated at the signal watch window.