# **Getting Started with ModelSim Student Edition**

ModelSim is a simulation and debugging environment created by Mentor Graphics. ModelSim allows you to check the syntax and verify the functionality of VHDL programs. This tutorial teaches the basic capabilities of ModelSim.

## I. Starting ModelSim

To start ModelSim, click Start → Programs→ ModelSim → ModelSim.

## **II. Creating a Project**

The first step in using ModelSim is creating a project. Projects ease the interaction with ModelSim and are useful for organizing files and simulation settings.

When you started ModelSim, a welcome dialog may have appeared (see Figure 1). If the welcome Dialog did not appear, you can display it by selecting **Help**  $\rightarrow$  **Welcome Menu**.

- 1. Click the "Jumpstart" button.
- 2. Click "Create a Project."



Figure 1: Welcome Menu

Instead of using the welcome menu, you can create a new project by selecting File  $\rightarrow$  New  $\rightarrow$  Project from the menu bar in the main window.

3. A "Create Project" window appears (See Figure 2). Enter an appropriate name for your project in the "Project Name" field, set the location where you would like the project to be created, and do not modify any of the other options. For this tutorial, name your project "Tutorial." Hit OK.

🕅 Create Project	×
Project Name	
Tutorial	
Project Location	
D:/Temp	Browse
Default Library Name work	
Copy Settings From	
odeltech_pe_edu_6.2e/modelsim.ini Brow	vse
🔹 Copy Library Mappings 🔘 Reference Library	/ Mappings
OK	Cancel

Figure 2: Create Project Dialog

4. After hitting OK, an "Add items to the Project" dialog box appears (See Figure 3).

Add items to the Project	×
Click on the icon to add items of that ty	be:
Create New File Add Existing Fi	le
Create Simulation Create New Fold	der
	Close

Figure 3: Add Items to the Project Dialog

There are three options to add files to the project:

- We can create new VHDL files (from scratch) and add them to the project.
- We can add already existing files to the project.
- We can do a combination of the two operations by combining the two previous steps.

We will first illustrate the method of adding new files to the project we have just created.

# Creating new VHDL files

In the "Add items to the Project" dialog box click on "Create New File." If you have closed the "Add items to the Project" dialog box, then select Project → Add to Project → New File.

2. The "Create Project File" dialog box appears. Enter an appropriate file name for your new file, choose "VHDL" in the "Add file as type" field, and choose "Top Level" in the "Folder" field (See Figure 4). For the tutorial, name your new file "DFF." Hit OK.

M Create Project File	×
File Name	
DFF	Browse
Add file as type	Folder
VHDL	Top Level 🗨
	OKCancel

Figure 4: Create Project File Dialog

- 3. Click "Close" in the "Add items to the Project" dialog.
- 4. In the workspace window of the main interface (see Figure 5), double-click on the file you have just created (DFF.vhd).

	ModelSim PE Student Edition 6.2e					
👘 Menu Bar 🔶	File Edit View Compile Simulate Add Source Tools Layout Window Help					
	📙 🗋 😅 💭 🐇 🍋 🛍 🕰 😂 🖾 🕴 🛤 🏗 🧞 📙 🤣 🎬 👰 🔀 🗍 Contains 🦳					
	X 🗱 🗈 🖪 🛛 Layout NoDesign		Compile All Button			
🔹 Workspace 🔶	Workspace 💴 📶 🖻 🛋 📕 🖬 🚬 📊 D:/Temp/DFF.vhd	+ 🗗 🗙	<ul> <li>Compile Button</li> </ul>			
	Name Statu Type In #	^				
	DFF.vhd YVHDL 1		- Source Window			
Compile Status —						
	Project Library					
Transcript 🗕	Transcript					
	ModelSim>					
	R Transcript	<u></u>				
	Project : Tutorial <no design="" loaded=""></no>	1.				
Figure 5: ModelSim's Interface						

5. The source code of your file appears in a new source window. In this tutorial, we will use the VHDL code for a simple D flip-flop from the text.

Before you begin entering code, make sure read-only mode is turned off by right-clicking in the source window and making sure "Read Only" in the menu that appears does not have a checkmark by it.

6. Type (or copy and paste) the following code in the source window; <u>however, leave out</u> <u>the last semicolon of this code</u>. This will illustrate error correction using the ModelSim compiler.

```
entity DFF is
  port(D, CLK: in bit;
        Q: out bit; QN: out bit := '1');
end DFF;
architecture SIMPLE of DFF is
begin
  process(CLK)
  begin
        if CLK'event and CLK = '1' then
        Q <= D after 10 ns;
        QN <= not D after 10 ns;
        end if;
        end process;
end SIMPLE;
```

7. Save your code (File  $\rightarrow$  Save).

#### Adding files to the project

Instead of starting with a new file, you can add an existing file to the project. We do not need to add any more files to the project for this tutorial; however, the steps below can be used if you need to add a file in the future.

- Click anywhere in the workspace window. Select Project → Add to Project → Existing File.
- 2. An "Add File to Project" dialog pops up. Select the file that you want to add to the project. Also, make sure that you select "VHDL" from in the "Add file as type" field, and click on "Copy to project directory." Hit OK.

Add file to Project	
File Name	
D:/Temp/DFF.vhd	Browse
Add file as type	Folder
C Reference from current location	Copy to project directory
	OK Cancel

Figure 6: "Add file to Project" Window

3. You should now see the file that you have just added in the workspace window of ModelSim's main interface.

#### **III. Compiling and Debugging Project Files**

- 1. Select Compile  $\rightarrow$  Compile All.
- 2. The compilation result is shown in the transcript window. A red message indicates that there is an error in the code.

ModelSim PE Student Edition 6.2e	
File Edit View Compile Simulate Add Source Tools Layout Window Help	
📙 🗋 😂 🛛 🖧 🦄 🛍 🕰 🕰 🗍 📥 🚉 🖏 🗍 🕸 🕮 🛺 🌋 🗍 Contains 🦳	<i>?</i>
X≪ »X 🖻 🖹 Layout NoDesign 🗨	
Workspace ==== :::::::==== # # Z	+ 🗗 🗙
Name Statu Ty ln #	-
DFF.vhd VI - 1 Antity DEFie	•
Project Library	
Transcript	==== <b>=</b> ∎ ₫
# Compile of DFF.vhd failed with 1 errors.	-
ModelSim>	•
A Tra	nscript
Project : Tutorial <no design="" loaded=""></no>	1.

Figure 7: Compilation Result in the Transcript Window

3. Double-click on the compilation results in the transcript window. This will open a new window that describes the nature of the error. In our case, the error message is as follows:

🕅D:/Temp/DFF.vhd Unsuccessful Compile	×
vcom -work work -2002 -explicit D:/Temp/DFF.vhd Model Technology ModelSim PE Student Edition vcom 6.2e Compiler 2006.11 Nov 16 2006 Loading package standard Compiling entity dff Compiling architecture simple of dff ** Error: D:/Temp/DFF.vhd(14): near "EOF": expecting: '/	
Close	

Figure 8: Error

4. Correct the above error by adding the semicolon after the "end SIMPLE" statement. Hit save, and then recompile the file. Repeat steps 2-4 until the code compiles with no errors.

# IV. Simulating the Design

This section covers the basics for simulating a design using ModelSim.

1. Click on the "Library" tab at the bottom of the workspace window and then click on the (+) sign next to the work library. You should see the entity name of the code that we have just compiled, "dff" (See Figure 9).

ModelSim PE Student Edition 6.2e	×
File Edit View Compile Simulate Add Library Tools Layout Window Help	
📙 🗋 😹 🌆 🍓 🏠 🔍 🛔 🛤 🚉 😘 📙 🧇 🕮 🌠 🛐 🖉 Contains 🦳 🧷	
X∢ ≫X 🖹 🖹 Layout NoDesign 💌	
Workspace === # # X HD:/Temp/DFF.vhd	×
Name Ty - In #	-
Image: Work work work work work work work here       Libit Ent         Image: Image	•
Transcript #	æ
ModelSim>	-
Transcript 4	>
Project : Tutorial	1.



- 2. Double-click on "dff" to load the simulator. This should switch the workspace view to a third tab, named "sim." An objects window and additional buttons will also appear.
- 3. Select View → List and View → Wave from the menu bar. This will open two commonly used debugging windows.
- 4. Locate the objects window and select the signals that you want to monitor for simulation purposes. For this tutorial, select all of the signals by holding down the shift key and clicking on them.

ModelSim PE Student E	dition 6.2e					
File Edit View Compile S	Simulate Add Wave	Tools Layout W	/indow	Help		
📙 🗅 🚅 🖶 🖨   🐰 🛍	🛍 🎦 🔔 🛛 🖊 🖺	°5 🛛 🍪 🏙 i	<b>),</b> 🕅	📙 🛧 🐳 🗼 📑 🗐 100 r		0 🛐
Contains 🧷 🦪	' X∢≫X ⊡ ⊠	Layout Simulate	•	ਤ 🛛 🕹 🕺 🗠 🛨 🗍 💽	🔁 🔝 🛛 🗱 😽	। 🀔 🛛 🍳 🍳 🔍
Workspace = 🕬 🖬 🛋 💌	Objects ==== /////	—————————————————————————————————————	💼 wave	e - default		+ Z X
▼ Instance	▼ Name	Value				
⊡-,⊒ dff	🔷 d	0				
L_ line8	🔷 clk	0				
📕 standard	• q	0				
	🔷 qn	1				
				Now	Ons	200 400
				Cursor 1	0 ns <mark>0 ns</mark>	
I I			4	F 4		
🚺 Libraru 🕼 sim 🕕						
	•					<u> </u>
Transcript						: + a
VSIM 5>						<b></b>
						<u>ا</u>
						<u> </u>
Project : ch2_project No	w:Ons Delta:O			sim:/dff - Limited Visibility R	egion 0 ns to 428 ns	3

## **Getting Started with ModelSim Student Edition**

Figure 10: The Objects Window

- 5. Click and drag the signals into the wave window, using the left mouse button. You can also use the menu and click on Add → Wave → Selected signals when the objects window is selected.
- 6. Click and drag the signals into the list window, using the left mouse button, or click Add
   → List → Selected signals in the menu bar when the objects window is selected.
- 7. We are now ready to simulate our design. At the "VSIM" prompt in the transcript window, type "**force clk 0 0 ns, 1 10 ns -repeat 20 ns**" and then hit enter. This statement forces the *clk* signal to take the value of 0 at 0ns and 1 at 10 ns. The statement repeats itself every 20 ns, creating a clock signal with a 20 ns period.

ModelSim PE Student Ed	ition 6.2e							
File Edit View Compile S	imulate Add Transci	ript Tools Layout	Window Help					
📙 🗅 🚅 🔛 🚭   👗 🐚	📽 🖄 😂   🗚 🖁	- °5 🛛 🕸 🖽 (	👰 🚨 📔 🛧 🐳	≧ <b>†</b> 100 ns	ŧ <u>∎</u> ∎‡   <del>(</del> } (	4 <u>1</u>		
📔 Contains 📃 🦪	X∢ >X ⊡ ⊗	Layout Simulate		🗠 🚽 🛛 💽	🛱 🗈   🗱 🐎   🤅	<u>K</u> ( ) Q Q <b>Q Q</b> ( )		
Workspace = 🕬 🖬 🛋 🗙	Objects ==== >>>>>	========	💶 wave - default			+ <b>2</b> ×		
Instance	▼ Name	Value		0				
⊡- <b>⊒</b> dff	d	<u> </u>	🔥 /dff/clk	0				
Leg line_8		0	🔶 /dff/q	0				
standard	v y ▲ an	1	🔷 /dff/qn	1				
			N	ow	0 ns			
			Curs	or 1	Ons One	400 400		
		·	1					
Library 😽 sim 🕮	•	•	H DFF.vhd	wave wave		<u> « &gt;</u>		
Transcript								
- Handonpe								
VSIM 5> force clk 0 0 ns, 1 10	Ins-repeat 20 ns					-		
Project : ch2_project Now: 0 ns Delta: 0 sim:/dff - Limited Visibility Region								

## **Getting Started with ModelSim Student Edition**



- 8. Type "**run 40 ns**" at the "VSIM" prompt in the transcript window and then hit enter. This command will run the simulation for 40 ns. You can see changes in signal values in the wave and list windows.
- 9. Next, change the value of the signal *d* to 1 by typing "**force d 1**" and then hit enter. The change in *d* will take place at the current simulation time, 40 ns.
- 10. Type "**run 40 ns**" and then hit enter to simulate for another 40 ns. The simulator will continue from the current simulation time of 40 ns and finish at 80 ns.
- 11. Select the wave window, and click on the "zoom full" button (See Figure 12). Your simulation output should look similar to Figure 12.

#### ModelSim PE Student Edition 6.2e - 🗆 🗵 Layout Window Help File Edit View Compile Simulate Add Wave Tools 100 ns 븆 🚉 🚉 😫 😯 🚱 🌇 🗋 🚅 🖬 🎒 | 🐰 🖻 🛍 🖄 🔔 | 🖊 🎥 0<sub>0</sub> 🗇 🛗 🎑 🎽 🧼 🗼 | Ef 4 ତ୍ର୍ ସ୍ ସ୍ ତ୍ର Layout Simulate 🔈 👗 🛨 🛨 Contains X∢⊁X B Bà ◄ 🔨 🕞 🔝 I 📴 📯 I 🌋 Zoom Workspace = 🛲 🖬 🖻 🗙 Objects 3 비폐지 💼 wave - default Value 🔻 Name /dff/d 🖲 Instance \_**-,∏** dff d /dff/clk 0 -**@** line\_\_8 clk /dff/q 1 q 🗾 standard /dff/qn 0 🔶 qn Now 80 ns Cursor 1 0 ns • 4 ▶ ∢ 🔊 sim 💵 Library Η DFF.vhd 🛛 🐺 list « » 💼 wave 🛛 • Þ Transcript 土团 • VSIM 9> -« » 🔒 Transcript sim:/dff - Limited Visibility Region 0 ns to 84 ns Project : ch2\_project Now: 80 ns Delta: 1

# Getting Started with ModelSim Student Edition



The above waveforms show that q changes to the value of d 10 ns after the rising edge of the clock, as expected. The same result is confirmed using the list window.

# **Printing results**

- <u>Printing the list window</u>
  - In order to print the list window, first select it, and then select File → Export → Tabular list from the menu bar, and then save the file as "simulation.txt" to a convenient location.
  - 2. Start Notepad and open the file that you saved above. You can then print the file.
- <u>Printing the wave window</u>

To print the wave window, simply select **File**  $\rightarrow$  **Print** from the menu bar when the wave window is selected. Make sure that your printed waveform can be read. (Numbers may be printed too small if you try to print a long simulation waveform on one page.)

# V. Notes

- When you use the force command, statements in your VHDL code that modify any signal you have forced will no longer have any effect. For example, if your code has a signal Z that you wish to initialize to 0 at the beginning of the simulation, you might use the command "force Z 0 0 ns." If you do this, Z will never change during simulation due to assignments in your program, even if you use the statement "z <= '1';" as your first line of code. To overcome this problem, change the above statement to: "force –deposit Z 0 0 ns." The "deposit" option will set the value of Z to 0 at 0 ns and allow it to be changed by statements within your code.</li>
- You can create a text file of simulator commands (such as **force**, **run**...) and save it with a ".do" extension. You can then simply click on **Tools** → **TCL** → **Execute Macro**, and then select your ".do" file to execute all of the commands in the file. Macros are extremely useful when you debug a large program and need to execute a large number of simulator commands each time you fix a bug in the program and wish to retest your code.