Exercise Manual
for
Introduction to VHDL
Using
Quartus II

Introduction to VHDL Lab Overview

Objective: Build a sequential 8 X 8 multiplier

The objective of the following exercises is to build an 8 X 8 multiplier. The input to the multiplier consists of two 8-bit multiplicands (a[7..0], b[7..0]) and the output from the multiplier is a 16-bit result (result[15..0]). Additional outputs are a done bit (DONE_FLAG) and seven signals to drive a seven segment display, (A, B, C, D, E, F, G).

There are several methods of implementing a multiplier; the method chosen for the VHDL labs is the sequential multiplier method. This 8 X 8 multiplier requires four clock cycles to perform the full multiplication. During each cycle, a pair of 4-bit portion of the multiplicands is multiplied in a 4 X 4 multiplier. The multiplication result of these 4 bit slices is then accumulated. At the end of the four cycles, the fully composed 16-bit result can be read at the output.

The following equations illustrate the mathematical principles supporting this implementation:

result[15..0] = a[7..0] * b[7..0]

= ((a[7..4] * 2 ^ 4) + a[3..0] * 2 ^ 0)
* ((b[7..4] * 2 ^ 4) + b[3..0] * 2 ^ 0)

= ((a[7..4] * b[7..4]) * 2 ^ 8)
+ ((a[7..4] * b[3..0]) * 2 ^ 4)
+ ((a[3..0] * b[7..4]) * 2 ^ 4)
+ ((a[3..0] * b[3..0]) * 2 ^ 0)

Figure 1 in the following page illustrates the top-level block diagram of the 8 X 8 multiplier.

The labs are structured as a bottom-up design approach. In each of the first five exercises, you will use targeted features of the VHDL language to build the individual components of the 8 X 8 multiplier. Then, in exercise 6 you will put everything together in a top-level design. You will then compile and simulate to verify the completeness of your design.

Good luck and have fun going through the exercises!

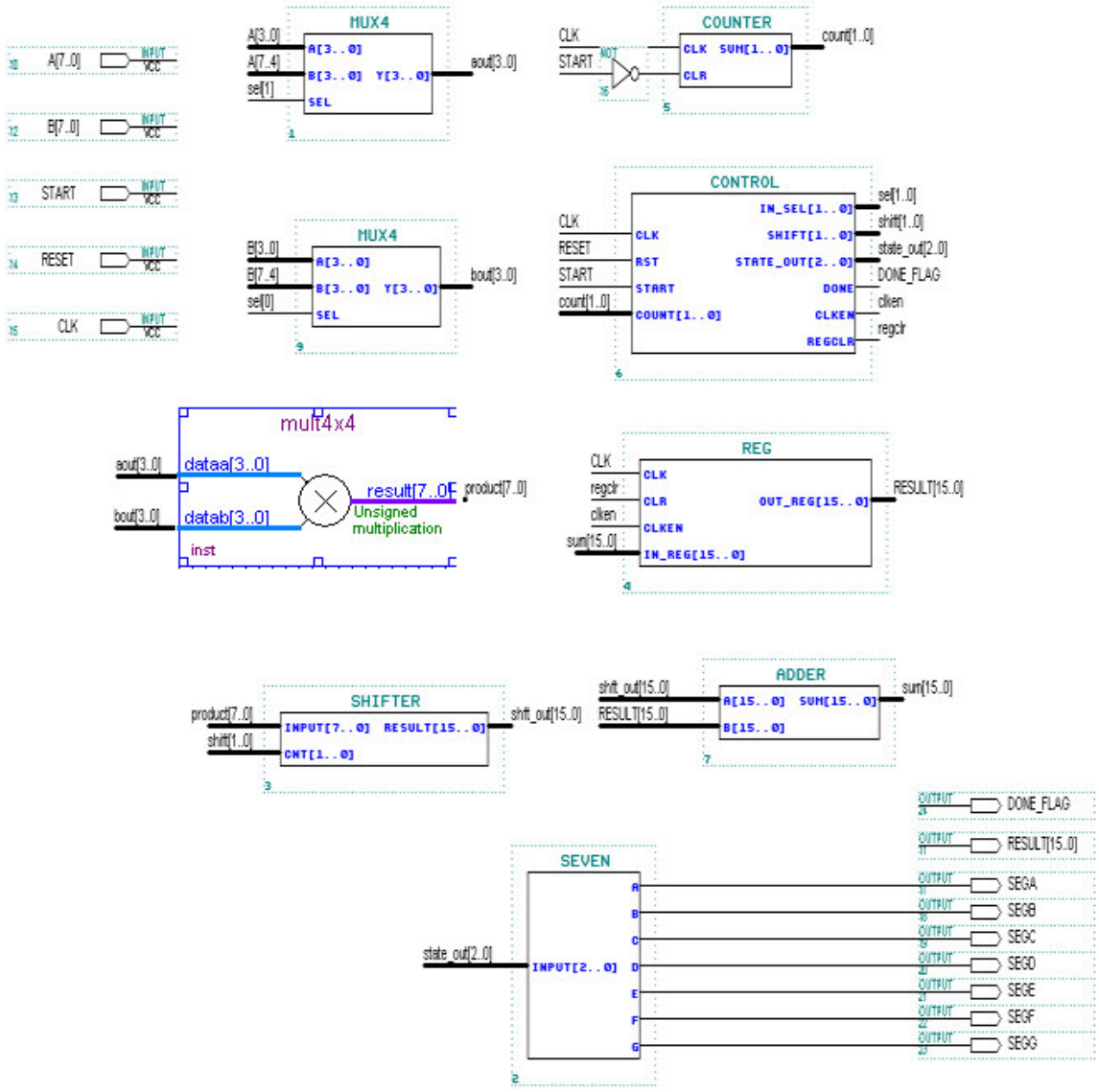


Figure 1 - 8 X 8 multiplier top level design block diagram

Exercise 1

Exercise 1

Objective: The 16-bit adder can be constructed using the + operator.

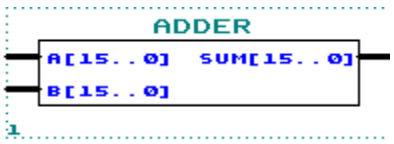


Figure 1-1.

Step 1 (Ask the instructor for the <pathname>)

1. **<pathname>** is_____

2. Use this **<pathname>** pathname for this Lab

Step 2 (Create new project and open VHDL text editor file)

Create a project by using the New Project Wizard. To create a project using the New Project Wizard, please follow these steps:

1. Select **New Project Wizard** from the File menu. The first time you open the New Project Wizard, it shows the introduction page; you can **click Next** to proceed to the first page of the wizard.

2. **Type** the directory name **or** select the directory with **Browse** ... . The directory name has been provided by the instructor in step1.

3. Type a name for the project in the project name box. For this lab, **type adder**.

4. **Type adder** as the name of the top-level design entity of the project in the top-level design entry box.

5. **Click Next**. The second page of the New Project Wizard appears.

6. At this point, there is no file to add because we will create the adder source file later.

7. **Click Next**. The EDA Tools Settings page appears. These exercises will use only Quartus II so the Tool Name should be <NONE> for all the Tool Types.

8. **Click Next**. The Device Family page appears. Select Stratix II as the Device Family and select No, I want to allow the Compiler to choose a device.

9. **Click Next 2 times**. The summary page appears. The summary page gives information about your project.

10. **Click Finish**. You have just finished the project creation. You should see the top-level entity (adder) in the Compilation Hierarchies tab of the Project Navigator window.

11. From the File menu select new or click on  . The New file dialog box will appear; select VHDL file. Click on OK

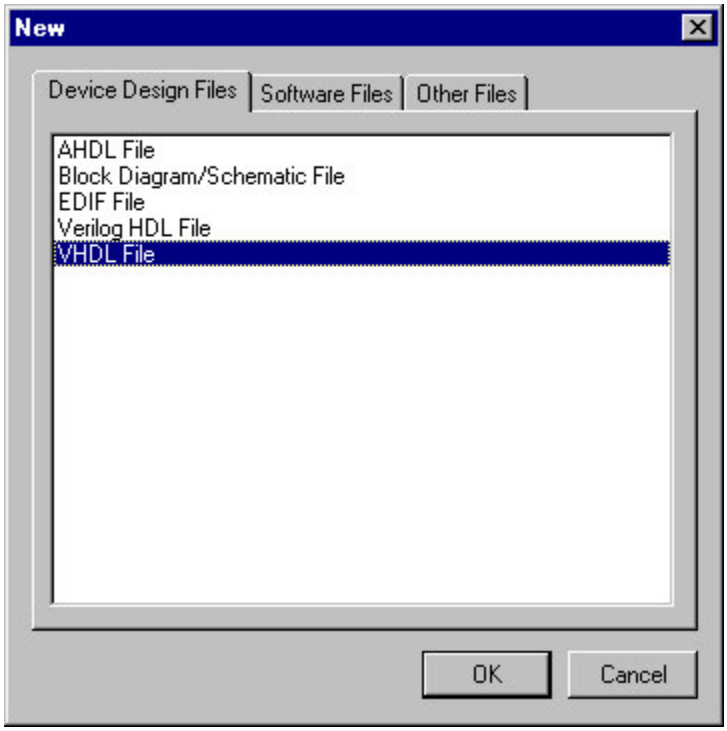


Figure 1-2.

12. VHDL text editor will appear**.**
13. Before the beginning of your code (before the ENTITY), type the following:

    LIBRARY ieee;

    USE ieee.std_logic_1164.all;

    USE ieee.std_logic_unsigned.all;

14. Write your source code.

    Remember to use the same input and output port names as shown in Figure 1-1.

15. ***Go to*** File menu ***and Choose*** Save As**.**

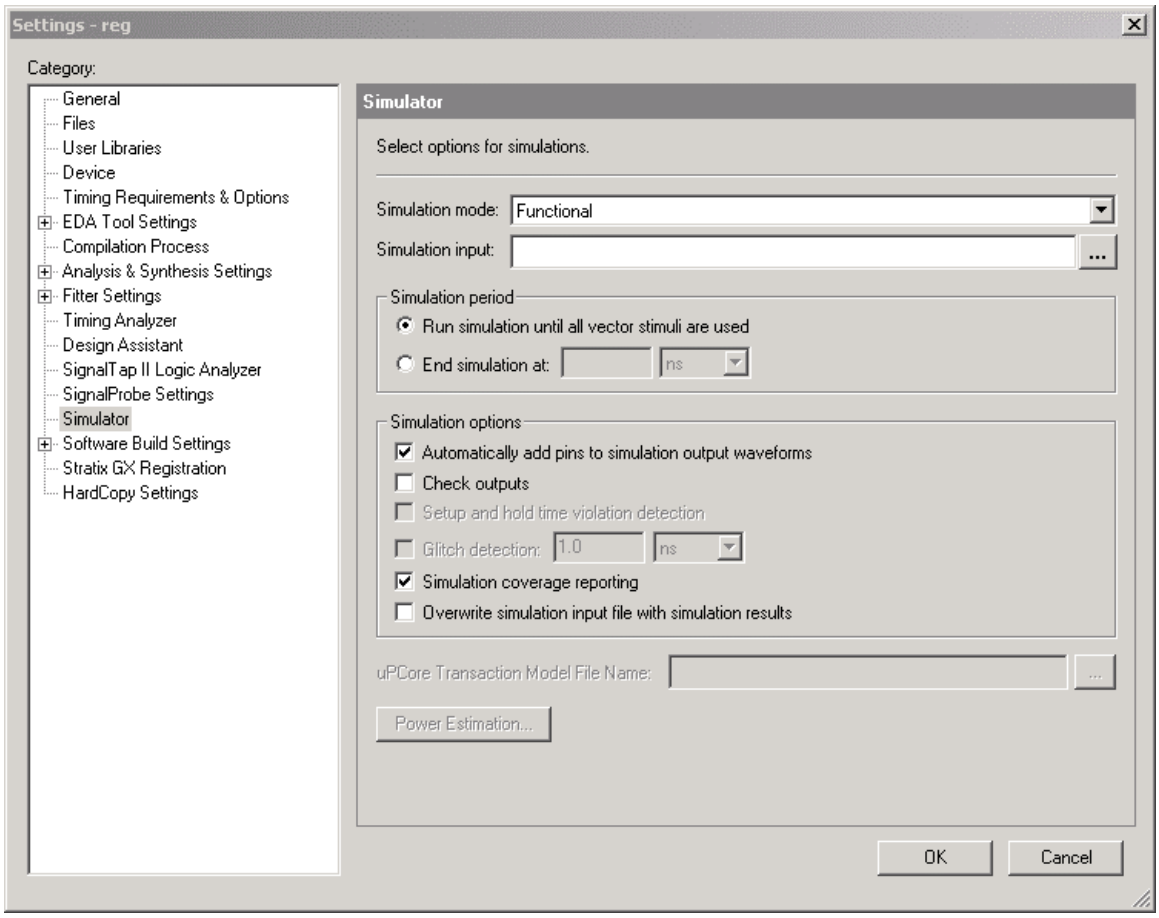
    Save new VHDL text file to <path>\altera_trn\vhdl\lab1\adder.vhd.

Step 3 (Synthesize the design)

1. From the Processing menu choose **Start-> Start Analysis and Synthesis**
2. This will save and check for syntax and semantic errors for the file adder.vhd. It will then synthesize the design. When you see the message "Analysis and synthesis was successful", click OK.

Step 4 (Do a functional simulation)

1. The stimulus file has been created for you to verify the functionality of your design. If you are interested in learning how to create your own stimulus file, please go to the Appendix of this manual.
2. Go to the Assignments menu and choose Simulator. In the Simulator mode section select **Functional.**



3. Choose **OK.**
4. From the Processing menu select the **Generate Functional Simulation Netlist**, Click OK when done.
5. From the Processing menu select Start Simulation or click on  **.** When you see the message, "Simulation was successful", click OK.
6. Check to see if you get the same results shown in Figure 1-6.

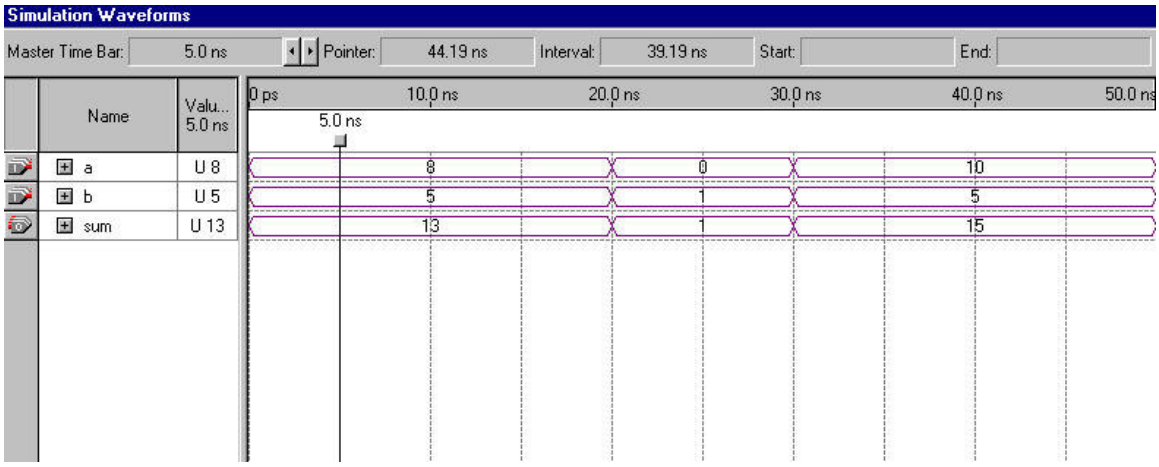


Figure 1-6.

Step 6 (Check operator overloading)

1. In adder.vhd, comment (--) out the LIBRARY and USE clauses in the beginning of the file.

```
-- LIBRARY ieee;
-- USE ieee.std_logic_1164.all;
-- USE ieee.std_logic_unsigned.all;
```

2. Save and check adder.vhd.

3. You should get the following error messages as shown in Figure 1-7.

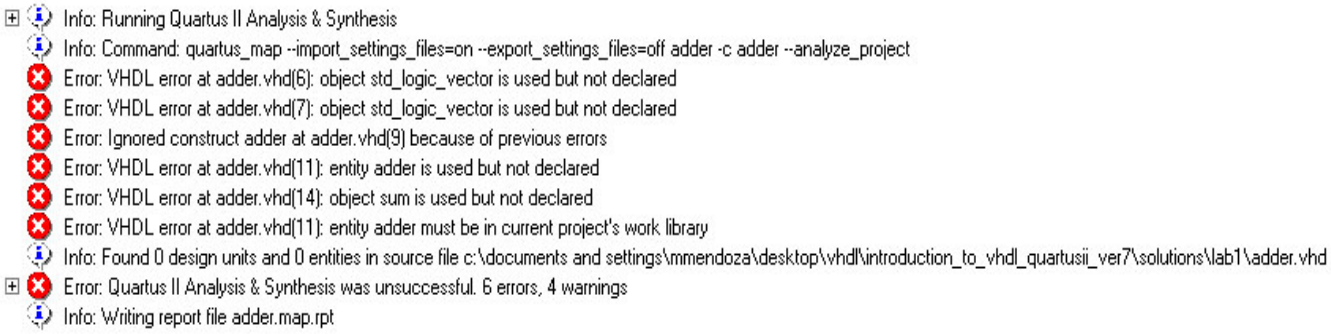


Figure 1-7.

This is due to operator overloading. The VHDL compiler does not understand the arithmetic operation for std_logic_vector data types. The std_logic_unsigned package contains the function that describes this arithmetic operation. Therefore, the library that contains this package and the package itself needs to be referenced in the design file.

5. Change adder.vhd to re-include the declarations then save.

Step 7 (Close the project)

1. From the File menu select Close project option.

END OF EXERCISE 1

Exercise 2

Exercise 2

Objective: Build a four input 2:1 multiplexer using IF-THEN statement.

The input to the multiplexer consists of two 4-bit data buses (a[3..0] and b[3..0]). The output (y[3..0]) is a[3..0] if the select control (sel) is low (0). The output is b[3..0] if sel is high (1).

The four input 2:1 multiplexer will be used in the top level design for selecting the 4-bit slices a[7..4], a[3..0], b[7..4], and b[3..0] as inputs to the 4 X 4 multiplier.

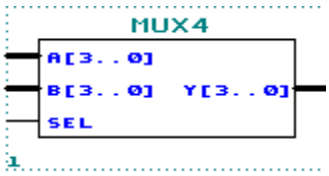


Figure 2-1.

The inputs **a** and **b**, and the output **y** are declared as **STD_LOGIC_VECTOR(3 DOWNTO 0).**
The input **sel** is declared as a **STD_LOGIC.**

Step 1 (Ask the instructor for the <pathname>)

1. **<pathname>** is_____

2. Use this **<pathname>** pathname for this Lab

Step 2 (Create new project and open VHDL text editor file)

Create a project by using the New Project Wizard. To create a project using the New Project Wizard, please follow the following steps:

1. Select **New Project Wizard** from the File menu. The first time you open the New Project Wizard, it shows the introduction page; you can **click Next** to proceed to the first page of the wizard.

2. **Type** the directory name **or** select the directory with **Browse** ... . The directory name has been provided by the instructor in step1.

3. Type a name for the project in the project name box. For this lab, **type mux4**.

4. **Type mux4** as the name of the top-level design entity of the project in the top-level design entry box.

5. **Click Next**. The second page of the New Project Wizard appears.

6. At this point, there is no file to add because we will create the mux4 source file later.

7. **Click Next**. The EDA Tools Settings page appears. These exercises will use only Quartus II so the Tool Name should be <NONE> for all the Tool Types.

8. **Click Next**. The Device Family page appears. Select Stratix II as the Device Family and select No, I want to allow the Compiler to choose a device.
9. **Click Next 2 times**. The summary page appears. The summary page gives information about your project.
10. **Click Finish**. You have just finished the project creation. You should see the top-level entity (mux4) in the Compilation Hierarchies tab of the Project Navigator window.
11. From the File menu select new or click on  . New file dialog box will appear and select VHDL file. Click OK.

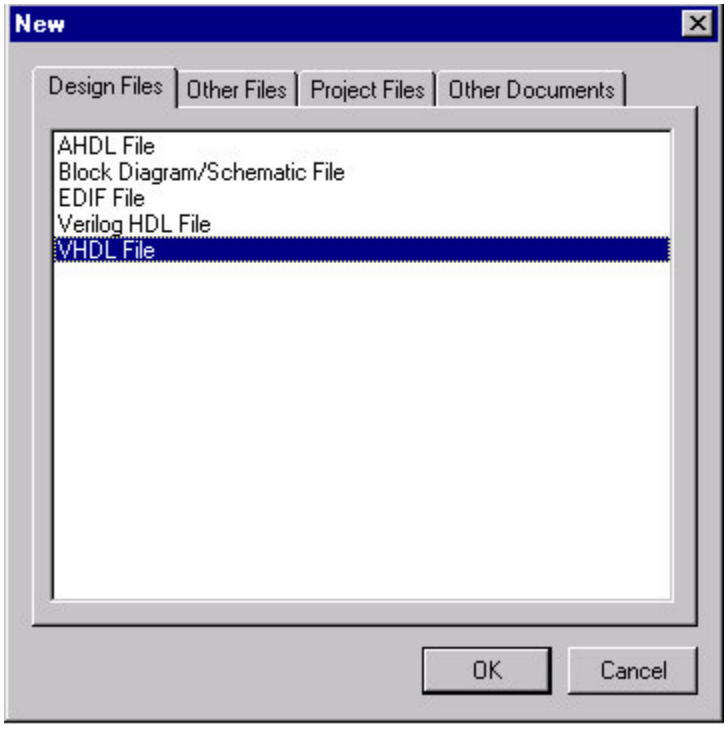


Figure 2-2.

1. VHDL text editor will appear**.**
2. Before the beginning of your code (before the ENTITY), type the following:

   LIBRARY ieee;

   USE ieee.std_logic_1164.all;

   USE ieee.std_logic_unsigned.all;
3. Write your source code.

   Remember to use the same input and output port names as shown in Figure 2-1.
4. ***Go to*** File menu ***and Choose*** Save As**.**

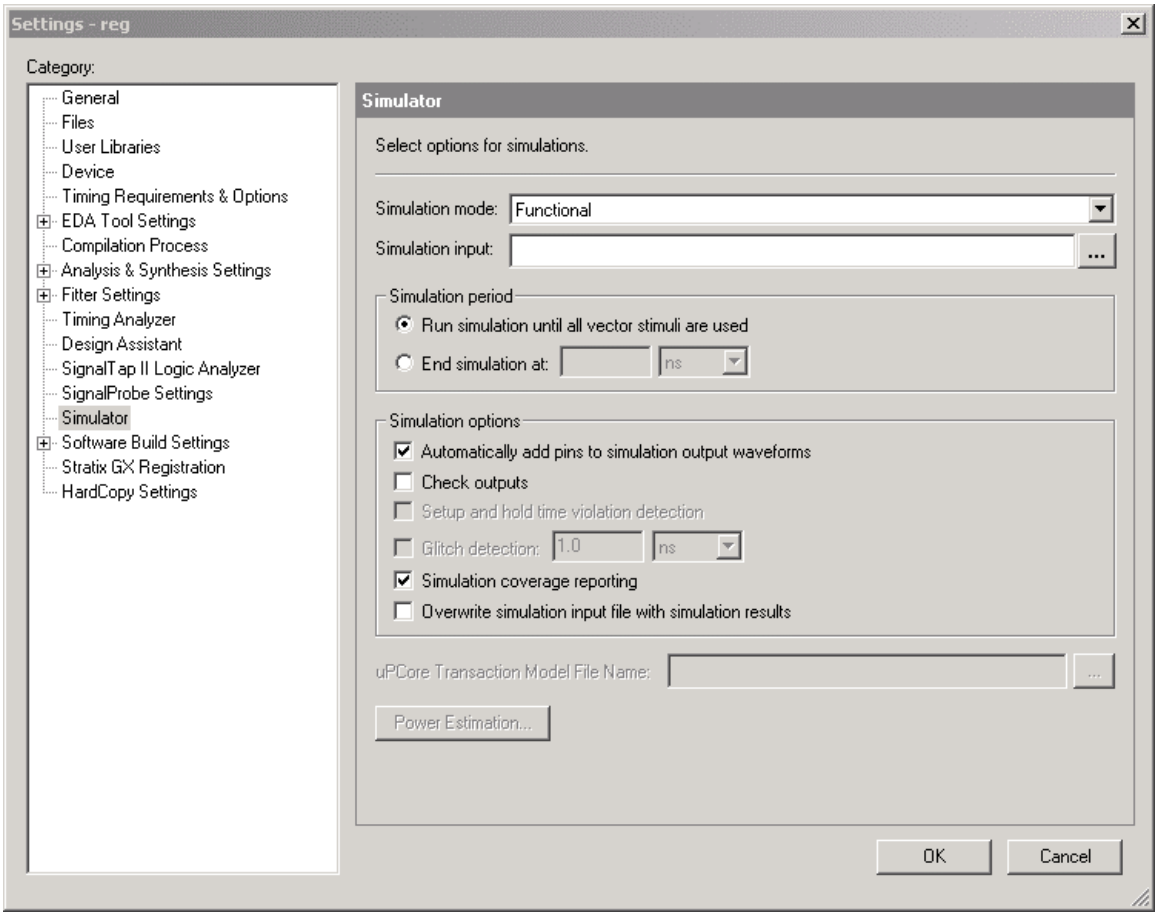
   Save new VHDL text file to <path>\altera_trn\vhdl\lab2\mux4.vhd.

Step 3 (Synthesize the design)

1. From the Processing menu choose **Start-> Start Analysis and Synthesis**
2. This will save and check for syntax and semantic errors for the file mux4.vhd. It will then synthesize the design. When you see the message "Analysis and synthesis was successful", click OK.

Step 4 (Do a functional simulation)

1. The stimulus file has been created for you to verify the functionality of your design. If you are interested in learning how to create your own stimulus file, please go to the Appendix of this manual.
2. Go to the Assignments menu and choose **Simulator**. In the Simulator mode section select **Functional.**



3. Choose **OK.**
4. From the Processing menu select the **Generate Functional Simulation Netlist**.
5. Choose **OK.**

6. From the Processing menu select Start Simulation or click on  . When you see the message, "Simulation was successful", click OK.
7. Check to see if you get the same results shown in Figure 2.6.

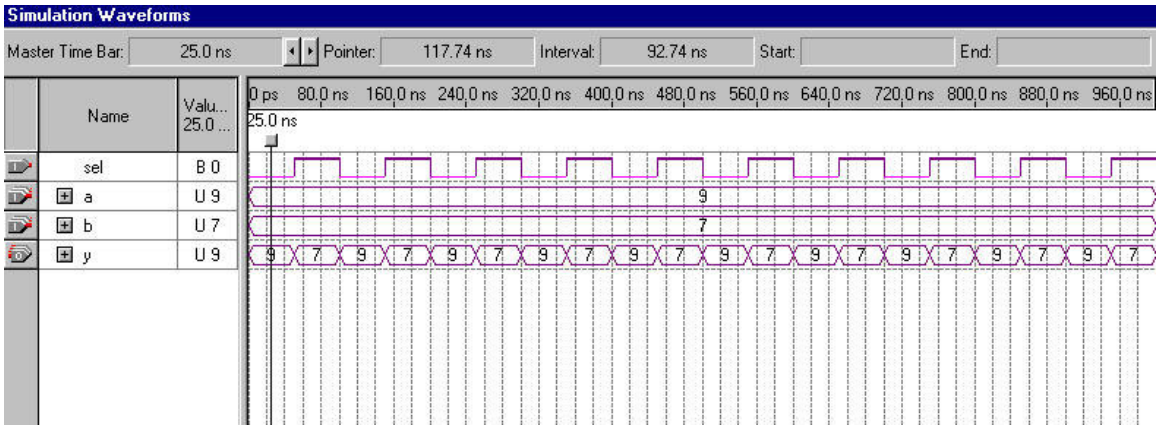


Figure 2-6.

Step 5 (Close the project)

1. From the File menu select Close project option to close the project.

END OF EXERCISE 2

Exercise 3

Exercise 3

Objective: Build a 7-segment display using CASE statement.

The 7-segment display shall display 0, 1, 2, 3, and E.

INPUTS			OUTPUTS							
IN2	IN1	IN0	A	b	c	d	E	F	g	DISPLAY
0	0	0	1	1	1	1	1	1	0	0
0	0	1	0	1	1	0	0	0	0	1
0	1	0	1	1	0	1	1	0	1	2
0	1	1	1	1	1	1	0	0	1	3
1	X	X	1	0	0	1	1	1	1	E

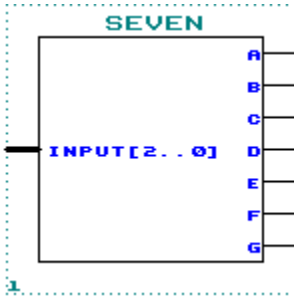


Figure 3-1.

The input **input** is declared as **STD_LOGIC_VECTOR(2 DOWNTO 0).**
The outputs **a, b, c, d, e, f, g** are declared as **STD_LOGIC.**

Step 1 (Ask the instructor for the <pathname>)

1. **<pathname>** is_____

2. Use this **<pathname>** pathname for this Lab

Step 2 (Create new project and open VHDL text editor file)

Create a project by using the New Project Wizard. To create a project using the New Project Wizard, please follow the following steps:

1. Select **New Project Wizard** from the File menu. The first time you open the New Project Wizard, it shows the introduction page; you can **click Next** to proceed to the first page of the wizard.
2. **Type** the directory name **or** select the directory with **Browse** ... . The directory name has been provided by the instructor in step1.
3. Type a name for the project in the project name box. For this lab, **type seven**.
4. **Type seven** as the name of the top-level design entity of the project in the top-level design entry box.
5. **Click Next**. The second page of the New Project Wizard appears.
6. At this point, there is no file to add because we will create the seven source file later.
7. **Click Next**. The EDA Tools Settings page appears. These exercises will use only Quartus II so the Tool Name should be <NONE> for all the Tool Types.
8. **Click Next**. The Device Family page appears. Select Stratix II as the Device Family and select No, I want to allow the Compiler to choose a device.
9. **Click Next 2 times**. The summary page appears. The summary page gives information about your project.
10. **Click Finish**. You have just finished the project creation. You should see the top-level entity (mux4) in the Compilation Hierarchies tab of the Project Navigator window.
11. From the File menu select new or click on  New file dialog box will appear and select VHDL file. Click OK.

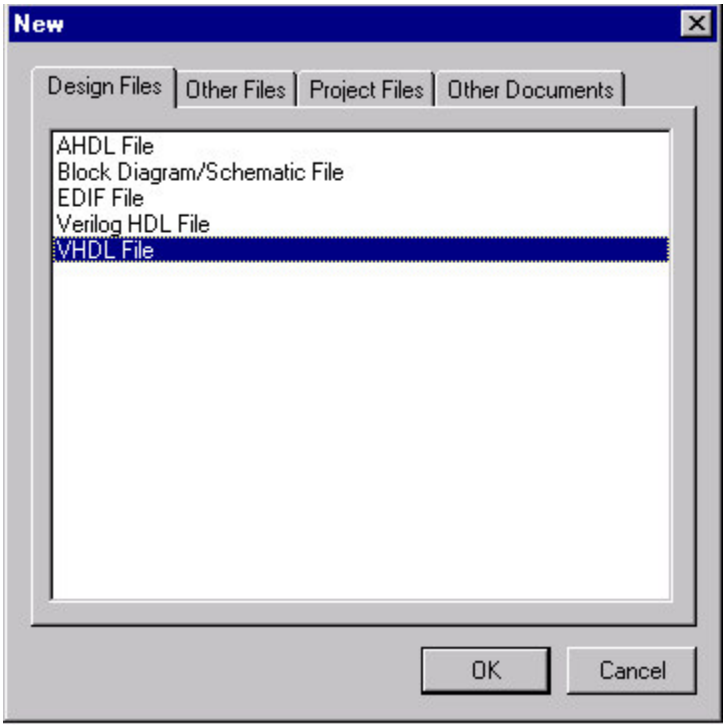


Figure 3-2.

12. VHDL text editor will appear**.**

13. Before the beginning of your code (before the ENTITY), type the following:

    LIBRARY ieee;

    USE ieee.std_logic_1164.all;

    USE ieee.std_logic_unsigned.all;

14. Write your source code.

    Remember to use the same input and output port names as shown in Figure 3-1.

15. ***Go to*** File menu ***and Choose*** Save As**.**

    Save new VHDL text file to <path>\altera_trn\vhdl\lab3\seven.vhd.

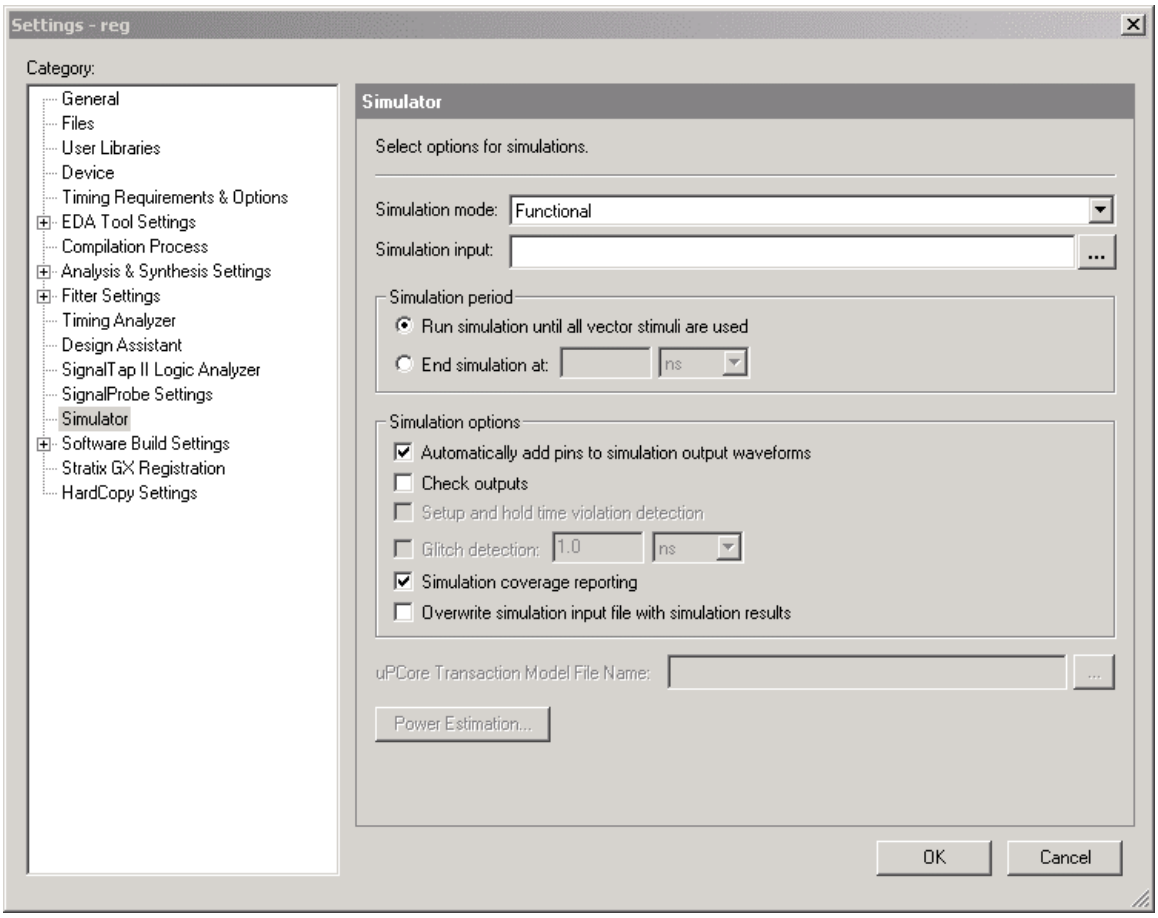
Step 3 (Synthesize the design)

1. From the Processing menu choose **Start-> Start Analysis and Synthesis**
2. This will save and check for syntax and semantic errors for the file seven.vhd. It will then synthesize the design. When you see the message "Analysis and synthesis was successful", click OK.

Step 4 (Do a functional simulation)

1. The stimulus file has been created for you to verify the functionality of your design. If you are interested in learning how to create your own stimulus file, please go to the Appendix of this manual.

2. Go to the Assignments menu and choose . In the Simulator mode section select **Functional.**



3. Choose **OK.**

4. From the Processing menu select the **Generate Functional Simulation Netlist**, Click OK when done.

5. From the Processing menu select Start Simulation or click on  **.** When you see the message, "Simulation was successful", click OK.

6. Check to see if you get the same results shown in Figure 3-6.

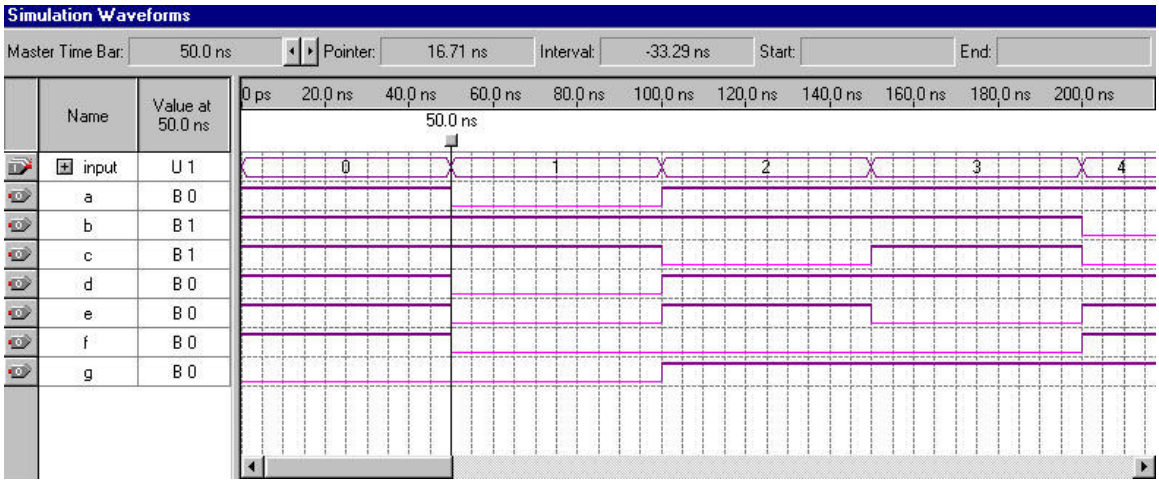


Figure 3-6.

Step 5 (Close the project)

1. From the File menu select Close project option to close the project.

END OF EXERCISE 3

Exercise 4

Exercise 4

Objective: Build an 8-bit to 16-bit shifter using a FOR LOOP statement

In this exercise, you will build an 8-bit to 16-bit shifter. This shifter will be capable of perform three types of shifter operations: no shift, left shift by 4 bit positions, and left shift by 8 bit positions.

The input to the shifter consists of a single 8-bit data bus (in[7..0]). The shift operation is controlled by the control signal cnt[1..0].

When cnt = "00", the no shift operation will be selected. The output (result[15..0]) shall be: result[15..8] = 0 and result [7..0] = in[7..0].

When cnt = "01", the left shift by 4 operation will be selected. The output (result[15..0]) shall be: result[15..12] = 0, result [11..4] = in[7..0], and result[3..0] = 0.

When cnt = "10", the left shift by 8 operation will be selected. The output (result[15..0]) shall be: result[15..8] = in[7..0], and result[7..0] = 0.

When cnt ="11", the no shift operation will be selected. The output (result[15..0]) shall be: result[15..8] = 0 and result [7..0] = in[7..0].

This 8-bit to 16-bit shifter will be used to perform the * 2 ^ 0 (no shift), * 2 ^ 4 (left shift by 4 bit positions), and * 2 ^ 8 (left shift by 8 bit positions) operations of the following equation:

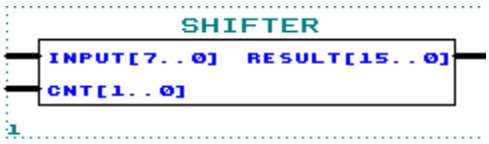


Figure 4-1.

Step 1 (Ask the instructor for the <pathname>)

1. **<pathname>** is_____

2. Use this **<pathname>** pathname for this Lab

Step 2 (Create new project and open VHDL text editor file)

Create a project by using the New Project Wizard. To create a project using the New Project Wizard, please follow the following steps:

1. Select **New Project Wizard** from the File menu. The first time you open the New Project Wizard, it shows the introduction page; you can **click Next** to proceed to the first page of the wizard.
2. **Type** the directory name **or** select the directory with **Browse**  . The directory name has been provided by the instructor in step1.
3. Type a name for the project in the project name box. For this lab, **type shifter**.
4. **Type shifter** as the name of the top-level design entity of the project in the top-level design entry box.
5. **Click Next**. The second page of the New Project Wizard appears.
6. At this point, there is no file to add because we will create the shifter source file later.
7. **Click Next**. The EDA Tools Settings page appears. These exercises will use only Quartus II so the Tool Name should be <NONE> for all the Tool Types.
8. **Click Next**. The Device Family page appears. Select Stratix II as the Device Family and select No, I want to allow the Compiler to choose a device.
9. **Click Next 2 times**. The summary page appears. The summary page gives information about your project.
10. **Click Finish**. You have just finished the project creation. You should see the top-level entity (mux4) in the Compilation Hierarchies tab of the Project Navigator window.
11. From the File menu select new or click on  . New file dialog box will appear and select VHDL file. Click OK.

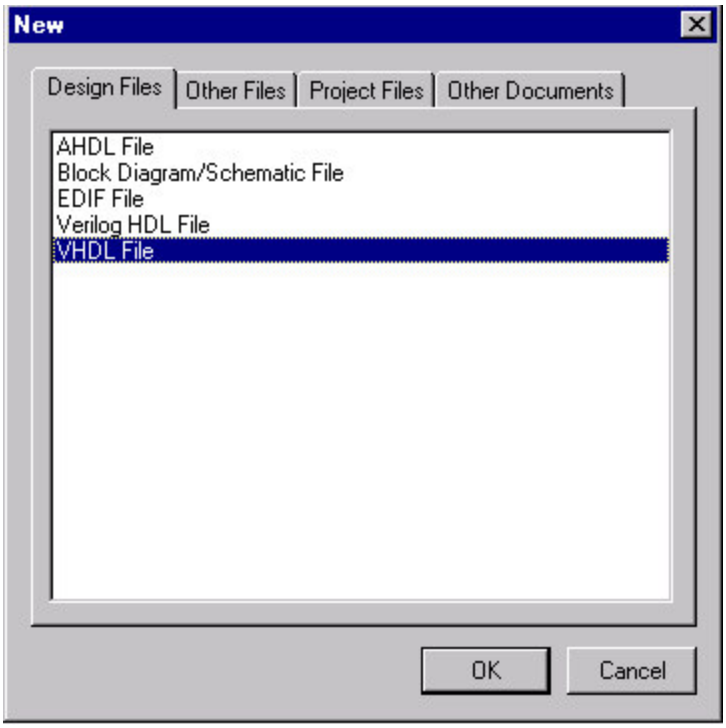


Figure 4-2.

12. VHDL text editor will appear**.**
13. Before the beginning of your code (before the ENTITY), type the following:

    LIBRARY ieee;

    USE ieee.std_logic_1164.all;

    USE ieee.std_logic_unsigned.all;
14. Write your source code.

    Remember to use the same input and output port names as shown in Figure 4-1.

15. ***Go to*** File menu ***and Choose*** Save As**.**

    Save new VHDL text file to <path>\altera_trn\vhdl\lab4\shifter.vhd.

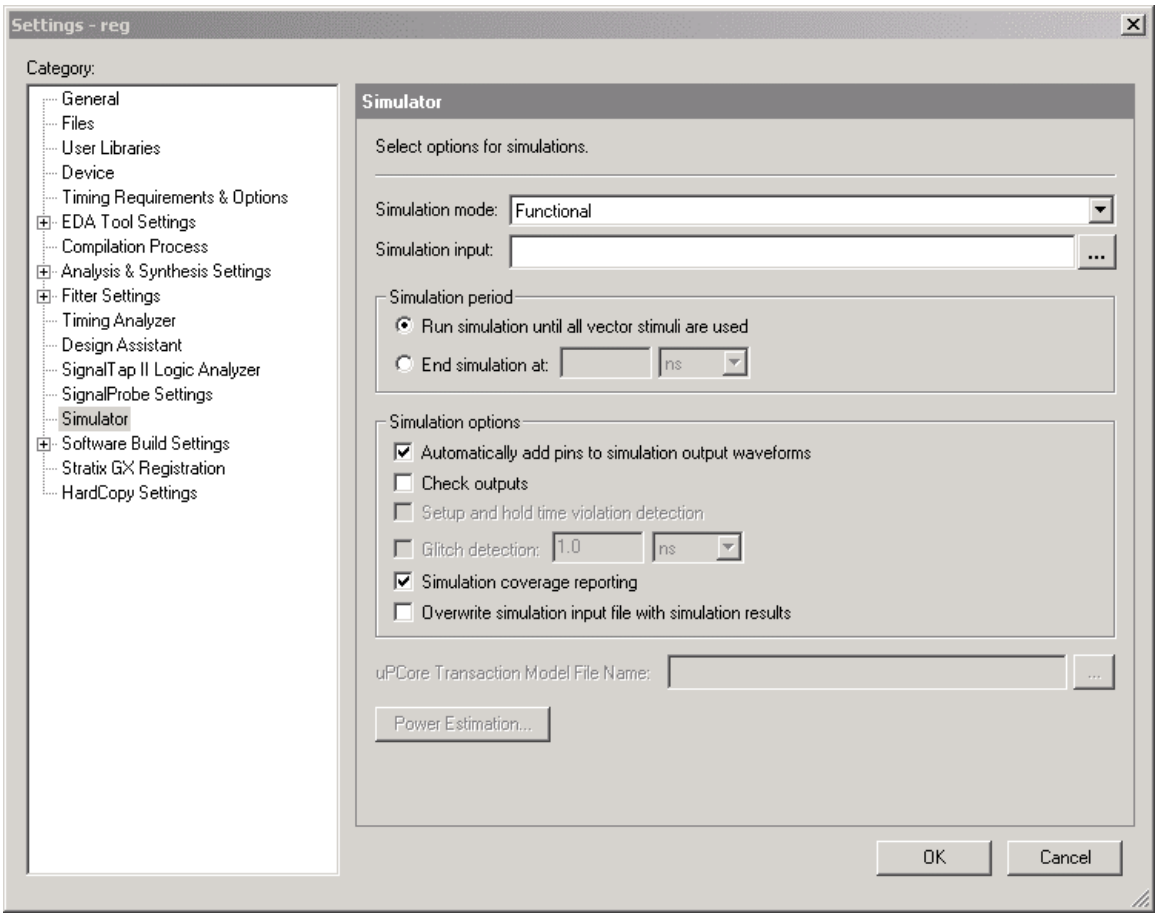
Step 3 (Synthesize the design)

1. From the Processing menu choose **Start-> Start Analysis and Synthesis**
2. This will save and check for syntax and semantic errors for the file shifter.vhd. It will then synthesize the design. When you see the message "Analysis and synthesis was successful", click OK.

Step 4 (Do a functional simulation)

1. The stimulus file has been created for you to verify the functionality of your design. If you are interested in learning how to create your own stimulus file, please go to the Appendix of this manual.

2. Go to the Assignments menu and choose Simulator. In the Simulator mode section select **Functional.**



3. Choose **OK.**

4. From the Processing menu select the **Generate Functional Simulation Netlist**, Click OK when done.

5. From the Processing menu select Start Simulation or click on  **.** When you see the message, "Simulation was successful", click OK.

1. Check to see if you get the same results shown in Figure 4-6.

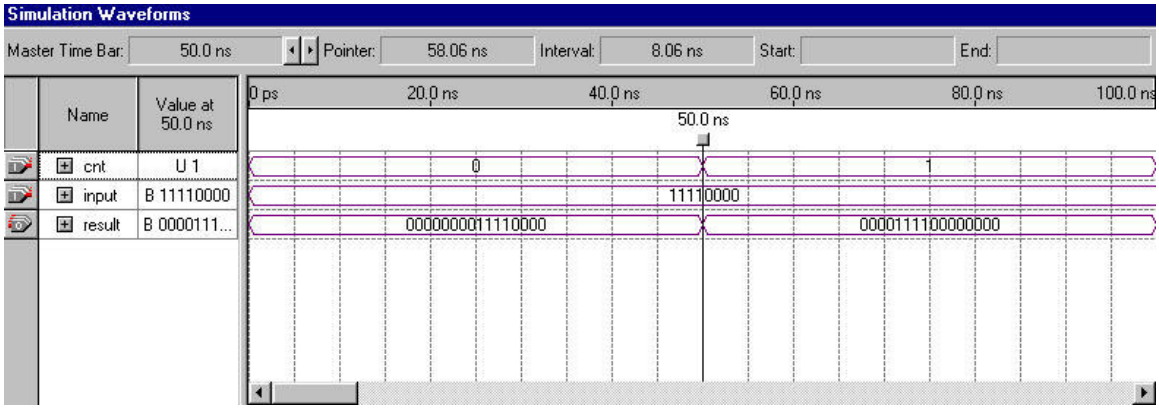


Figure 4-6.

Step 5 (Close the project)

1. From the File menu select Close project option to close the project.

END OF EXERCISE 4

Exercise 5

Exercise 5

Objective: Two-part exercise:

Part A -- Build a 16-bit register with synchronous operation

Part B –Build a 2 bit counter with asynchronous operation

Part A

Functionality of the 16-bit register:

If clr='1' AND clken='1', then in_reg will be loaded into register at the rising edge of clock.

Otherwise, if clr='0', then the register is cleared at the rising edge of clock.

Note: This is a synchronous clear and clock enable register. Therefore, the IF-THEN that checks for clear and clock enable should be inside the IF-THEN statement that that checks for the clock condition.

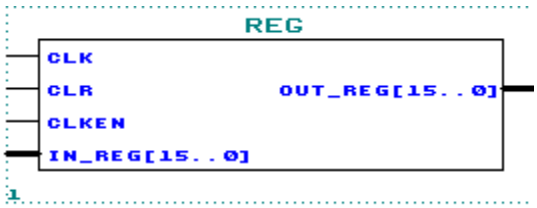


Figure 5-1.

Step 1 (Ask the instructor for the <pathname>)

1. **<pathname>** is_____

2. Use this **<pathname>** pathname for this Lab

Step 2 (Create new project and open VHDL text editor file)

Create a project by using the New Project Wizard. To create a project using the New Project Wizard, please follow the following steps:

1. Select **New Project Wizard** from the File menu. The first time you open the New Project Wizard, it shows the introduction page; you can **click Next** to proceed to the first page of the wizard.

2. **Type** the directory name **or** select the directory with **Browse** ... . The directory name has been provided by the instructor in step1.
3. Type a name for the project in the project name box. For this lab, **type reg**.
4. **Type reg** as the name of the top-level design entity of the project in the top-level design entry box.
5. **Click Next**. The second page of the New Project Wizard appears.
6. At this point, there is no file to add because we will create the reg source file later.
7. **Click Next**. The EDA Tools Settings page appears. These exercises will use only Quartus II so the Tool Name should be <NONE> for all the Tool Types.
8. **Click Next**. The Device Family page appears. Select Stratix II as the Device Family and select No, I want to allow the Compiler to choose a device.
9. **Click Next 2 times**. The summary page appears. The summary page gives information about your project.
10. **Click Finish**. You have just finished the project creation. You should see the top-level entity (mux4) in the Compilation Hierarchies tab of the Project Navigator window.
11. From the File menu select new or click on . New file dialog box will appear and select VHDL file.

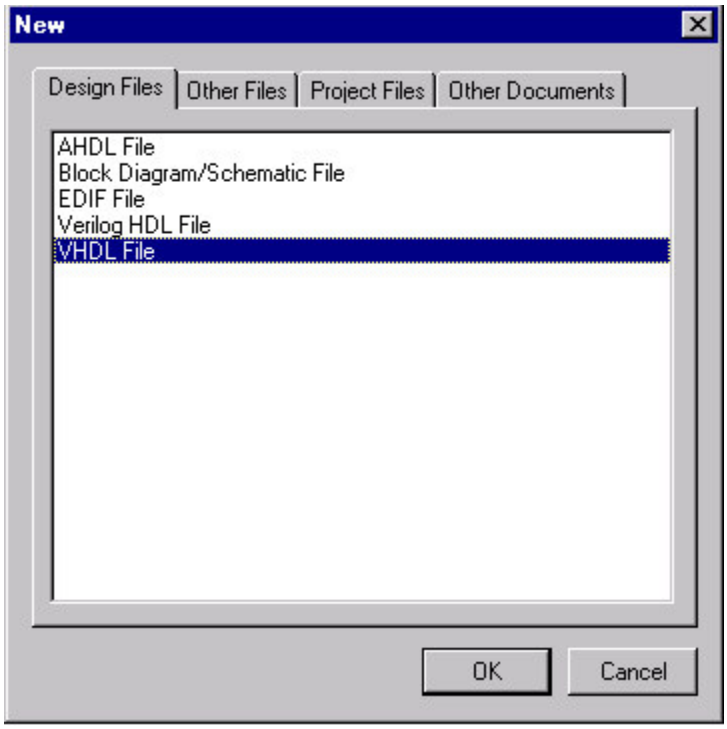


Figure 5-2.

12. VHDL text editor will appear**.**
13. Before the beginning of your code (before the ENTITY), type the following:

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_unsigned.all;
```

14. Write your source code.

    Remember to use the same input and output port names as shown in Figure 5-1.

15. ***Go to*** File menu ***and Choose*** Save As**.**

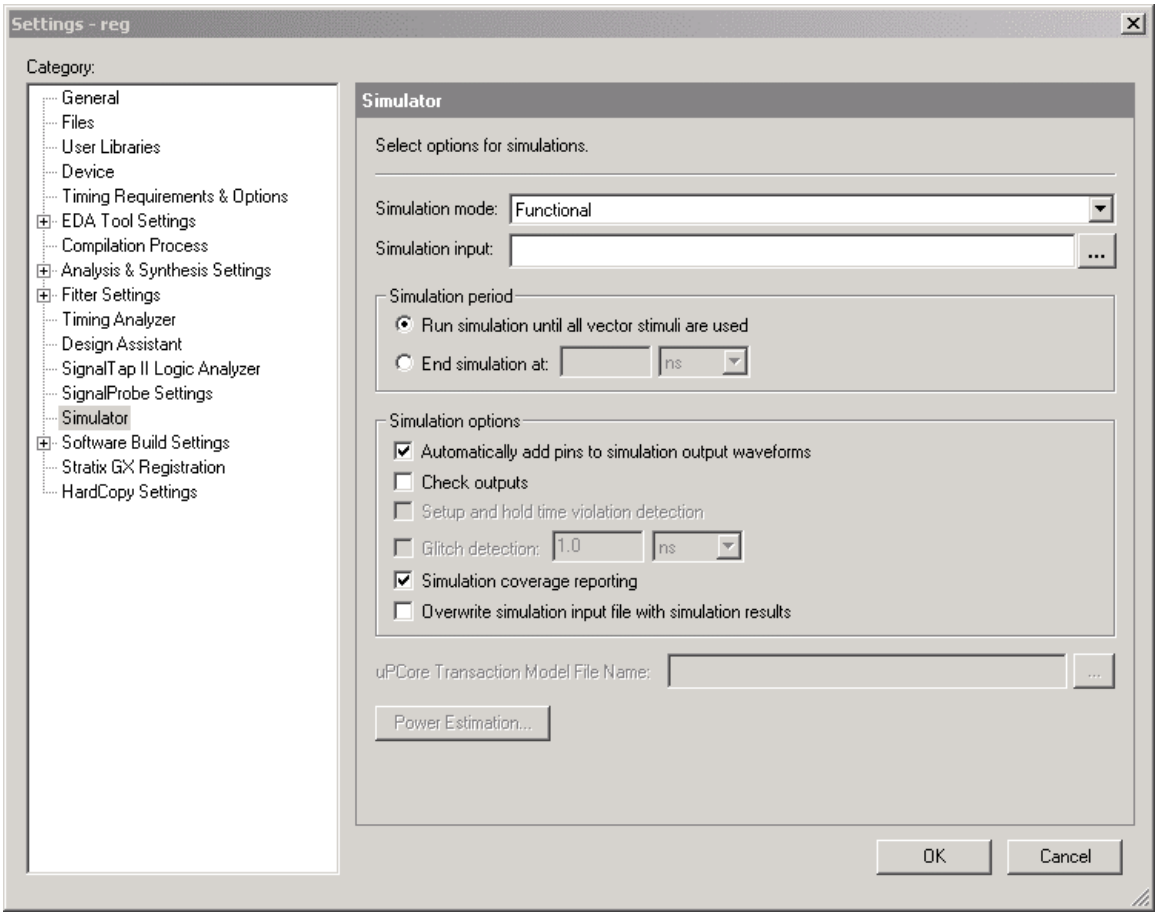
    Save new VHDL text file to <path>\altera_trn\vhdl\lab5a\reg.vhd.

Step 3 (Synthesize the design)

1. From the Processing menu choose **Start-> Start Analysis and Synthesis**
2. This will save and check for syntax and semantic errors for the file reg.vhd. It will then synthesize the design. When you see the message "Netlist Analysis and synthesis was successful", click OK.

Step 4 (Do a functional simulation)

1. The stimulus file has been created for you to verify the functionality of your design. If you are interested in learning how to create your own stimulus file, please go to the Appendix of this manual.
2. Go to the Assignments menu and choose Simulator. In the Simulator mode section select **Functional.**



3.

4. Choose **OK.**

5. From the Processing menu select the **Generate Functional Simulation Netlist**, Click OK when done.

6. From the Processing menu select Start Simulation or click on  . When you see the message, "Simulation was successful", click OK.

7. Check to see if you get the same results shown in Figure 5-6.

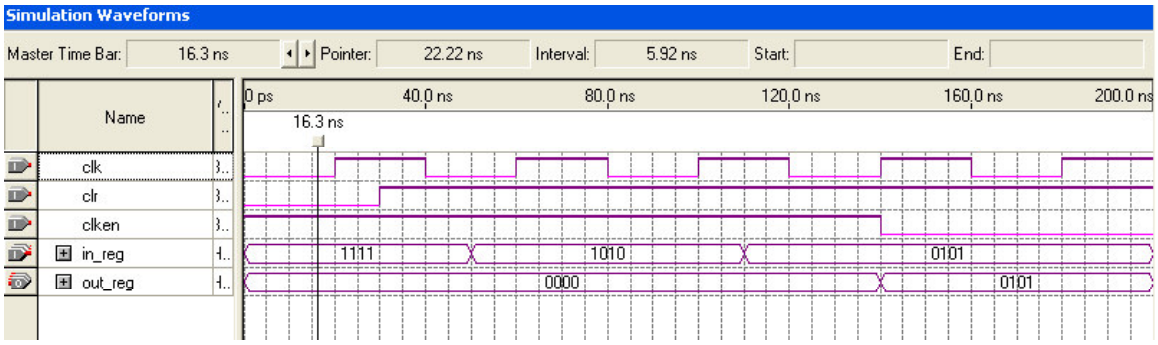


Figure 5-6.

Step 5 (Close the project)

1. From the File menu select Close project option to close the project.

Part B

The 2-bit counter can be constructed using the +1 operation. It is used to help the state machine track the cycles of the sequential multiplication.

Functionality of the 2-bit counter:

If clr='0', the counter is cleared immediately.

Otherwise, if clr='0', then the count increments by 1 at the rising edge of clock.

Note: This is an asynchronous clear.

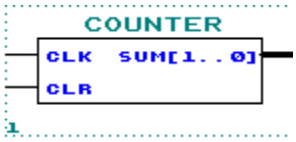


Figure 5-7.

Step 1 (Ask the instructor for the <pathname>)

1. **<pathname>** is_____

2. Use this **<pathname>** pathname for this Lab

Step 2 (Create new project and open VHDL text editor file)

Create a project by using the New Project Wizard. To create a project using the New Project Wizard, please follow the following steps:

1. Select **New Project Wizard** from the File menu. The first time you open the New Project Wizard, it shows the introduction page; you can **click Next** to proceed to the first page of the wizard.

2. **Type** the directory name **or** select the directory with **Browse** ... . The directory name has been provided by the instructor in step1.

3. Type a name for the project in the project name box. For this lab, **type counter**.

4. **Type counter** as the name of the top-level design entity of the project in the top-level design entry box.

5. **Click Next**. The second page of the New Project Wizard appears.

6. At this point, there is no file to add because we will create the counter source file later.

7. **Click Next**. The EDA Tools Settings page appears. These exercises will use only Quartus II so the Tool Name should be <NONE> for all the Tool Types.

8. **Click Next**. The Device Family page appears. Select Stratix II as the Device Family and select No, I want to allow the Compiler to choose a device.

9. **Click Next 2 times**. The summary page appears. The summary page gives information about your project.
10. **Click Finish**. You have just finished the project creation. You should see the top-level entity (mux4) in the Compilation Hierarchies tab of the Project Navigator window.
11. From the File menu select new or click on  . New file dialog box will appear and select VHDL file.

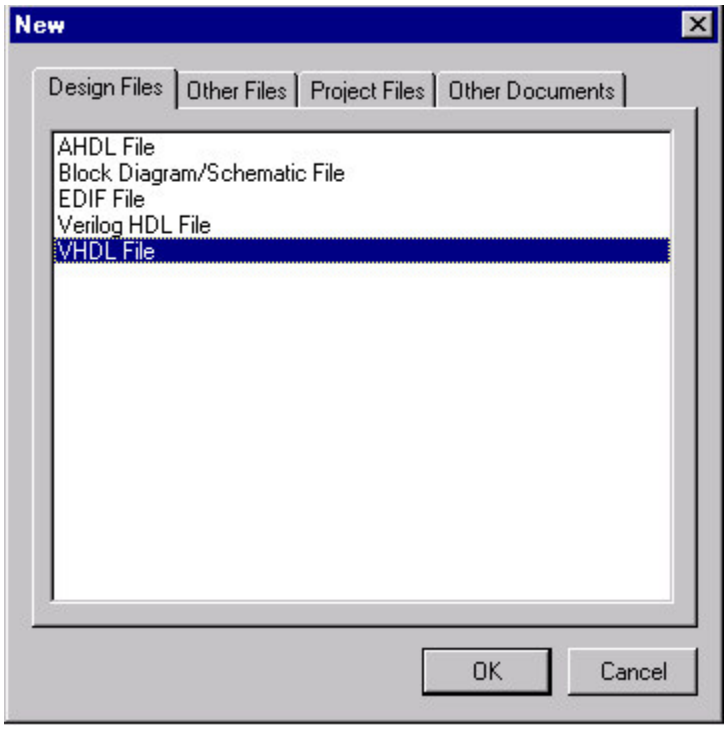


Figure 5-8.

12. VHDL text editor will appear**.**
13. Before the beginning of your code (before the ENTITY), type the following:

    LIBRARY ieee;

    USE ieee.std_logic_1164.all;

    USE ieee.std_logic_unsigned.all;

14. Write your source code.

    Remember to use the same input and output port names as shown in Figure 5-7.

15. ***Go to*** File menu ***and Choose*** Save As**.**

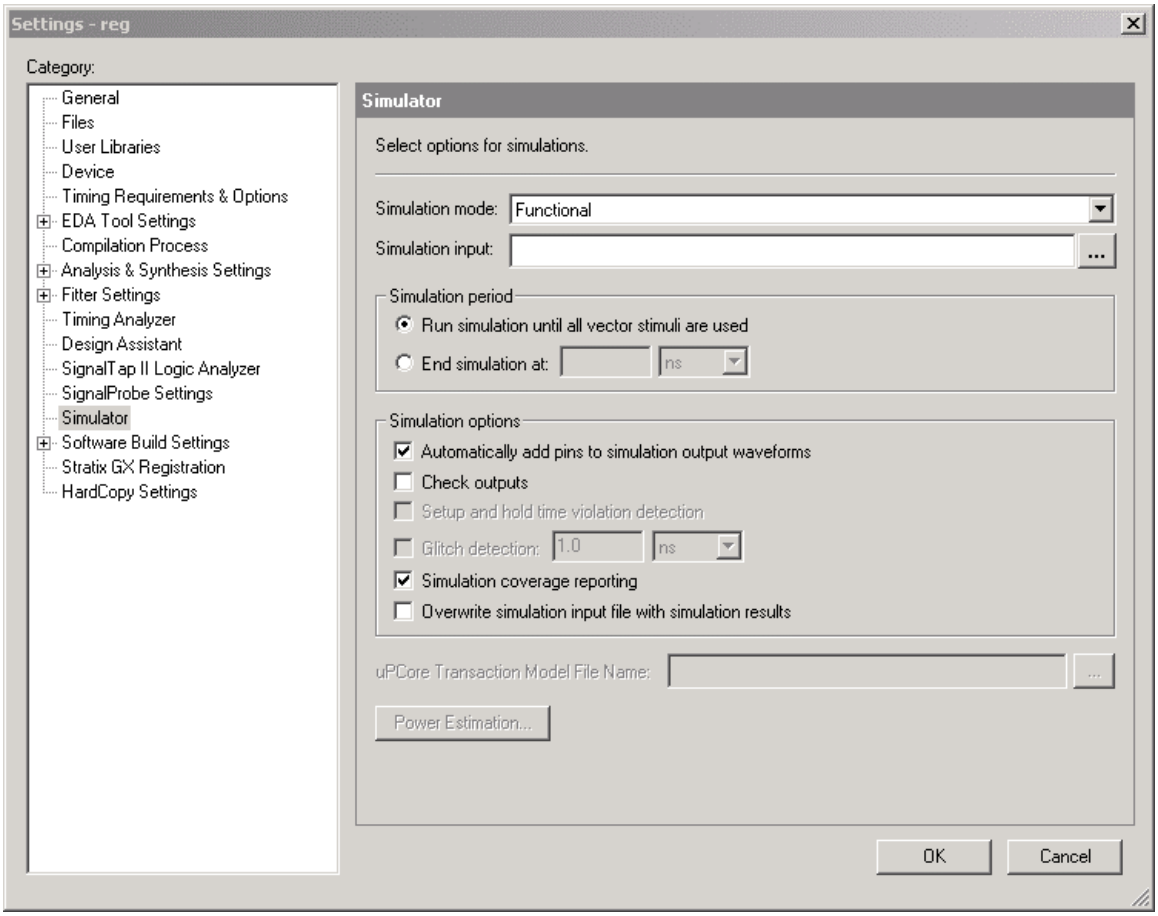
    Save new VHDL text file to <path>\altera_trn\vhdl\lab5b\counter.vhd.

Step 3 (Synthesize the design)

1. From the Processing menu choose **Start-> Start Analysis and Synthesis**
2. This will save and check for syntax and semantic errors for the file counter.vhd. It will then synthesize the design. When you see the message "Analysis and synthesis was successful", click OK.

Step 4 (Do a functional simulation)

1. The stimulus file has been created for you to verify the functionality of your design. If you are interested in learning how to create your own stimulus file, please go to the Appendix of this manual.
2. Go to the Assignments menu and choose choose Simulator. In the Simulator mode section select **Functional.**



3. Choose **OK.**
4. From the Processing menu select the **Generate Functional Simulation Netlist**, Click OK when done.
5. From the Processing menu select Start Simulation or click on  . When you see the message, "Simulation was successful", click OK.
6. Check to see if you get the same results shown in Figure 5-12.

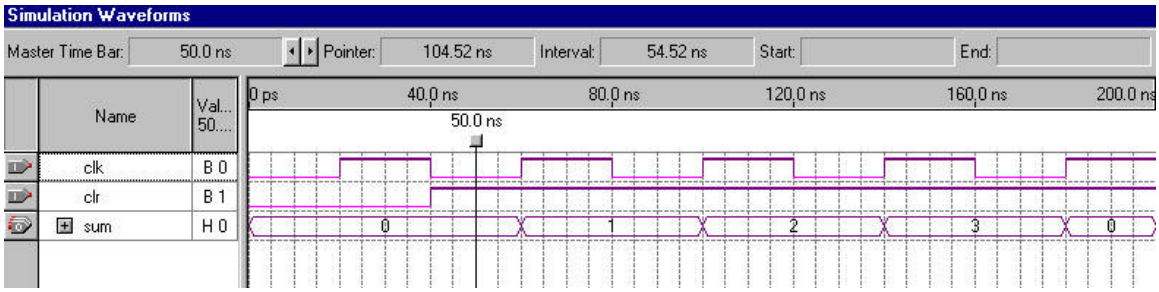


Figure 5-12.

Step 5 (Close the project)

1. From the File menu select Close project option to close the project.

END OF EXERCISE 5

Exercise 6

Exercise 6

Objective: Three-part exercise:

Part A – Examine the controlling state machine

Part B – Build a 4x4 multiplier using LPM_MULT

Part C – Putting it all together by declaring and instantiating the lower-level components

Part A

You have now completed building all of the components necessary to build the 8x8 multiplier, except for the controlling state machine and the 4x4 multiplier (Part B). Due to time, the controlling state machine has been written for you and is located in <path>\altera_trn\vhdl\lab6a\control.v.

This state machine will manage all the operation that occurs within the 8 X 8 multiplier.

The state machine will perform the ((a[3..0] * b[3..0]) * 2 ^ 0) multiplication in the first cycle (LSB state) after the input signal *start* becomes a '1'. This intermediate result is saved in an accumulator.

In the second clock cycle (MID state), the ((a[3..0] * b[7..4]) * 2 ^ 4) multiplication is performed. The multiplication result is added with the content of the accumulator and clocked back into the accumulator.

In the third clock cycle (MID state), the ((a[7..4] * b[3..0]) * 2 ^ 4) multiplication is performed. The multiplication result is added with the content of the accumulator and clocked back into the accumulator.

In the fourth clock cycle (MSB state), the ((a[7..4] * b[7..4]) * 2 ^ 8) multiplication is performed. The multiplication result is added with the content of the accumulator and clocked back into the accumulator. This result is the final result:

result[15..0] = a [7..0] * b[7..0]

```
=    ( ( a[7..4] * b[7..4] ) * 2 ^ 8 )
   + ( ( a[7..4] * b[3..0] ) * 2 ^ 4 )
   + ( ( a[3..0] * b[7..4] ) * 2 ^ 4 )
   + ( ( a[3..0] * b[3..0] ) * 2 ^ 0 )
```

NOTE: There are two inputs to the state machine *start* and *count*[1..0]. The *start* signal is a single cycle high-true signal. When *start* becomes a '1', it indicates that multiplication can begin at next clock cycle. The *start* signal can only be asserted for one clock cycle. The *start* signal shall stay a '0' until next 8 x 8 multiplication is to be performed. The *count*[1..0] signal is the output of a free running 2-bit counter. The *count*[1..0] signal is synchronously initialized by the *start* signal. *Count*[1..0] is used by the state machine to track the cycles of the multiplication.

Please also note that this is NOT the optimal design. The state machine design as you see it is intended for exercising your VHDL skills and not the ability to perform optimum solution.

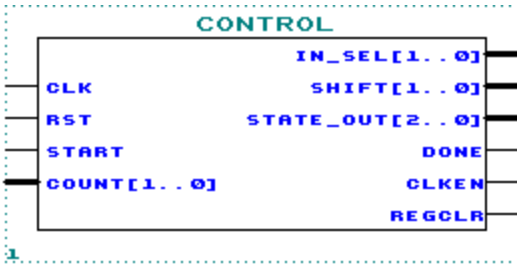
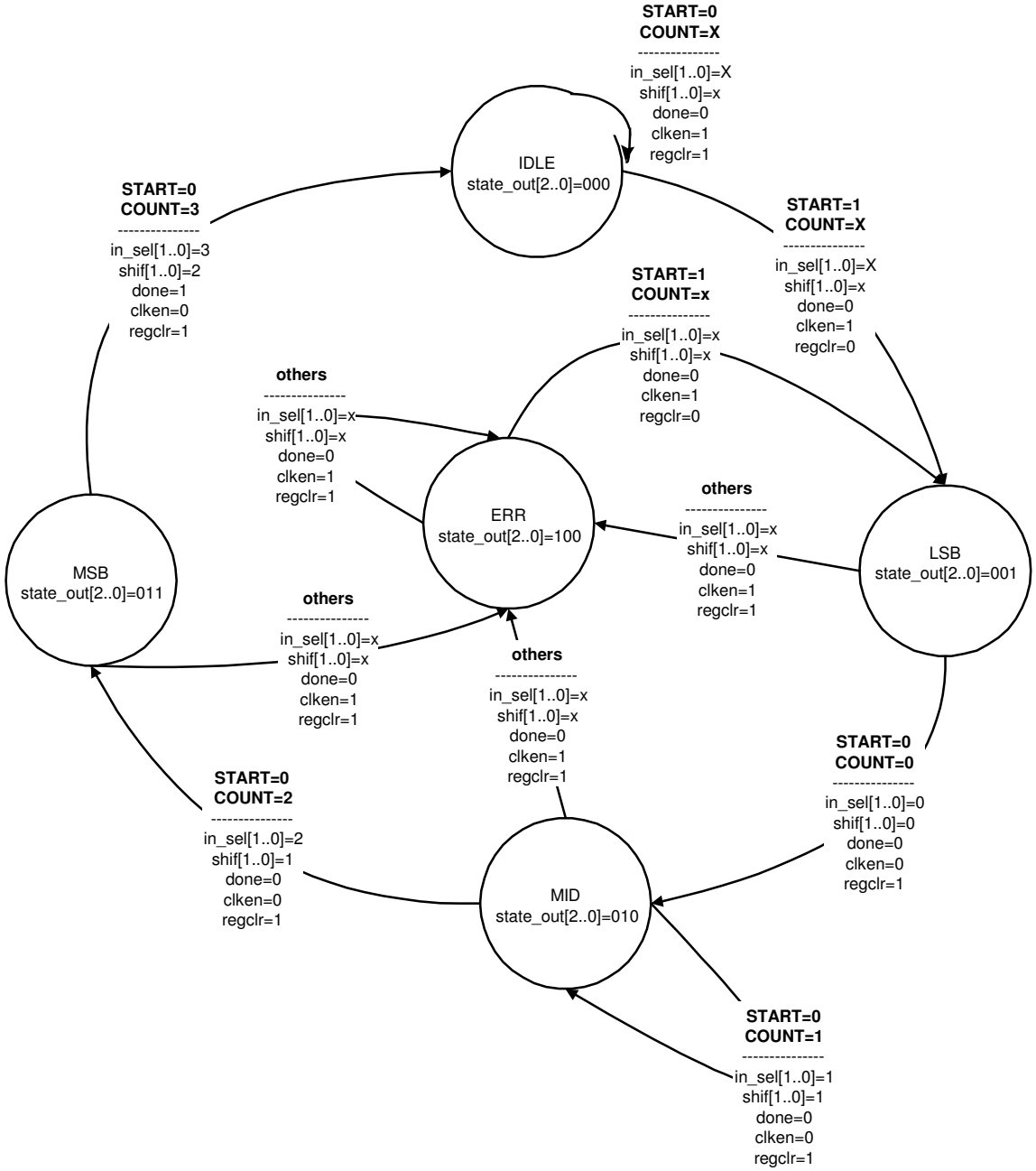


Figure 6-1.



X means a “don’t care” in this state diagram.

Part B

The last component you will build is a 4x4 multiplier.

You will use the LPM_MULT supplied by Altera to build a multiplier with a 4-bit multiplicands and an 8-bit result output.

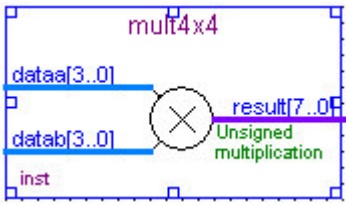


Figure 6-2.

Step 1 (Ask the instructor for the <pathname>)

1. **<pathname>** is_____

2. Use this **<pathname>** pathname for this Lab

Step 2 (Use the MegaWizard Plug-In Manager to Create the Multiplier)

To create a VHDL file which instantiates the LPM_MULT, please follow the following steps:

1. Choose **Tools>MegaWizard Plug-In Manager**. In the window that appears, select **Create a new custom megafunction variation**. Click on **Next**.

2. In the window that appears, expand the **arithmetic** folder and select **LPM_MULT**. Choose **VHDL** output. Name the **output file <pathname>\lab6b\mult4x4**. Click on **Next**.

3. Set the width of the **dataa** bus to be **4** bits and the width of the **datab** bus to be **4** bits. For the remaining settings in this window, use the defaults that appear. Select **Next**.

4. For the next 2 windows that appear, simply use the default settings by selecting **Next.**

5. Select **Finish** in the final window that appears.

The multiplier is built. A VHDL file (mult4x4.vhd) has been created which instantiates LPM_MULT. To view this file, go to the File menu, select Open and select mult4x4.vhd.

Step 3 (Create a Project for the Multiplier)

Create a project by using the New Project Wizard. To create a project using the New Project Wizard, please follow the following steps:

1. Select **New Project Wizard** from the File menu. The first time you open the New Project Wizard, it shows the introduction page; you can **click Next** to proceed to the first page of the wizard.
2. **Type** the directory name **or** select the directory with **Browse**  . The directory name has been provided by the instructor in step1.
3. Type a name for the project in the project name box. For this lab, **type mult4x4**.
4. **Type mult4x4** as the name of the top-level design entity of the project in the top-level design entry box.
5. **Click Next**. The second page of the New Project Wizard appears.
6. **Click Next**. The EDA Tools Settings page appears. These exercises will use only Quartus II so the Tool Name should be <NONE> for all the Tool Types.
7. **Click Next**. The Device Family page appears. Select Stratix II as the Device Family and select No, I want to allow the Compiler to choose a device.
8. **Click Next 2 times**. The summary page appears. The summary page gives information about your project.
9. **Click Finish**. You have just finished the project creation. You should see the top-level entity (mux4) in the Compilation Hierarchies tab of the Project Navigator window.

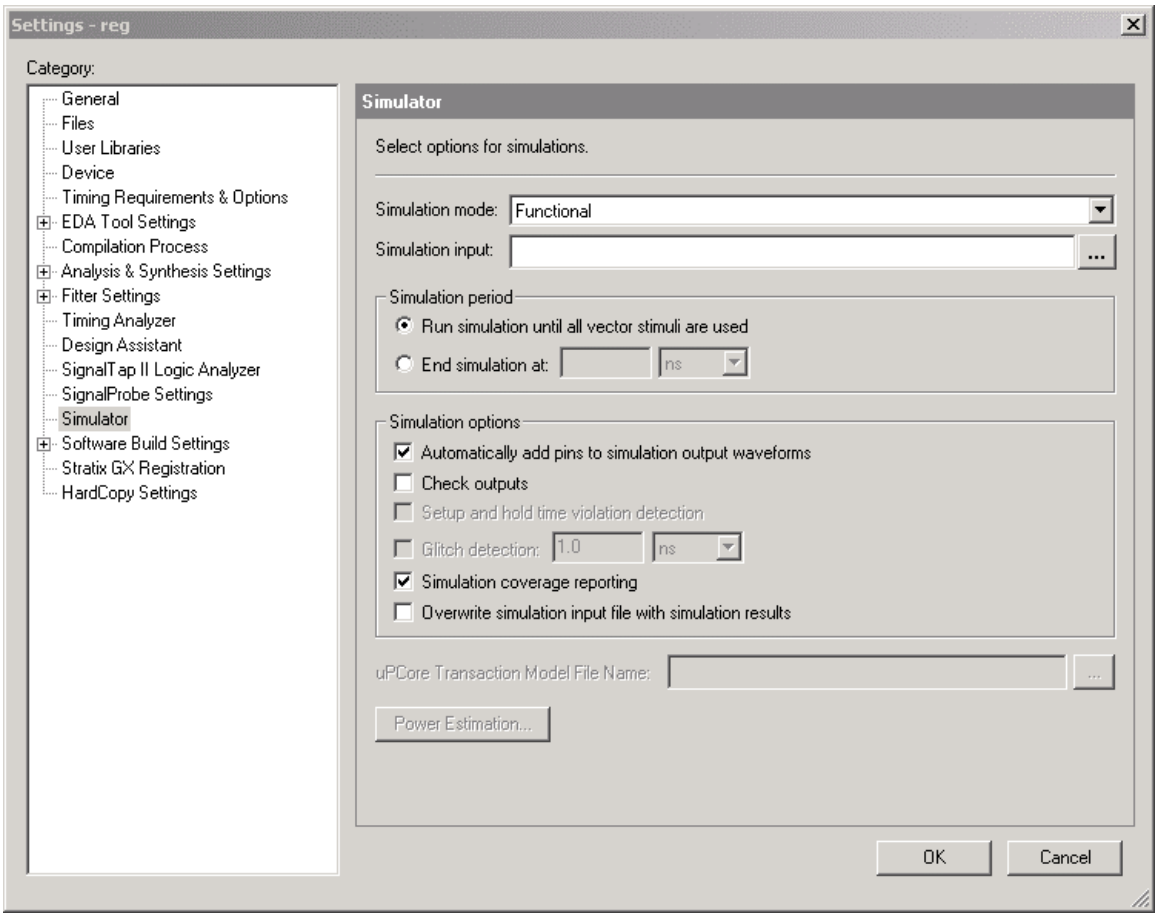
Step 4 (Synthesize the design)

1. From the Processing menu choose **Start-> Start Analysis and Synthesis**
2. This will save and check for syntax and semantic errors for the file mult4x4.vhd. It will then synthesize the design. When you see the message “Analysis and synthesis was successful”, click OK.

Step 5 (Do a functional simulation)

1. The stimulus file has been created for you to verify the functionality of your design. If you are interested in learning how to create your own stimulus file, please go to the Appendix of this manual.

2. Go to the Assignments menu and choose Simulator. In the Simulator mode section select **Functional.**



3. Choose **OK.**
4. From the Processing menu select the **Generate Functional Simulation Netlist**, Click OK when done.
5. From the Processing menu select Start Simulation or click on  . When you see the message, "Simulation was successful", click OK.
6. Check to see if you get the same results shown in Figure 6-6.

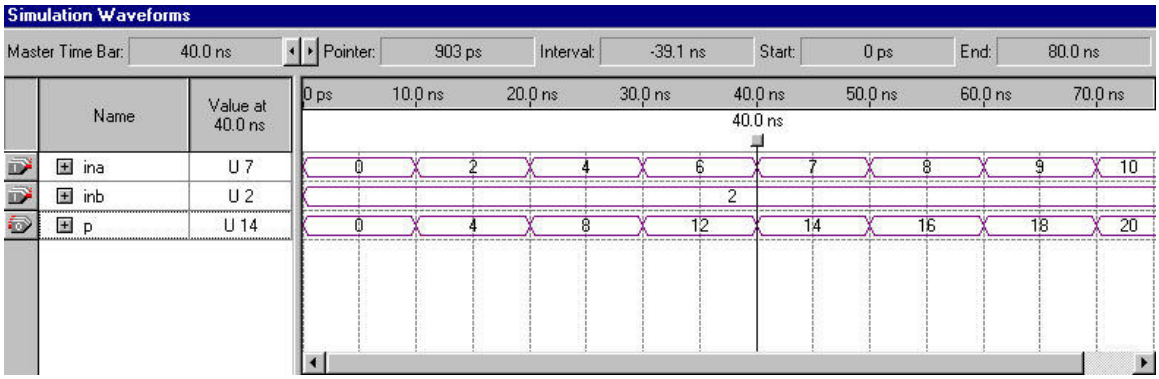


Figure 6-6.

Step 6 (Close the project)

1. From the File menu select Close project option to close the project.

Part C

You have now completed building all of the components necessary to build the 8x8 multiplier.

Making use of the knowledge you have gained up to this point, you should instantiate each component in a top-level design and connect all signals as shown in Figure 6-7. To save time, we have completed part of this for you. Please finish this task by instantiating mult4x4 and shifter. You will also need to declare the PRODUCT signals. You have successfully completed the Introduction to VHDL class once your top-level design is compiled and simulated correctly.

Congratulations!

You have completed the implementation of the following 8x8 multiplier.

result[15..0] = a [7..0] * b[7..0]

```
=    ( ( a[7..4] * b[7..4] ) * 2 ^ 8 )
   + ( ( a[7..4] * b[3..0] ) * 2 ^ 4 )
   + ( ( a[3..0] * b[7..4] ) * 2 ^ 4 )
   + ( ( a[3..0] * b[3..0] ) * 2 ^ 0 )
```

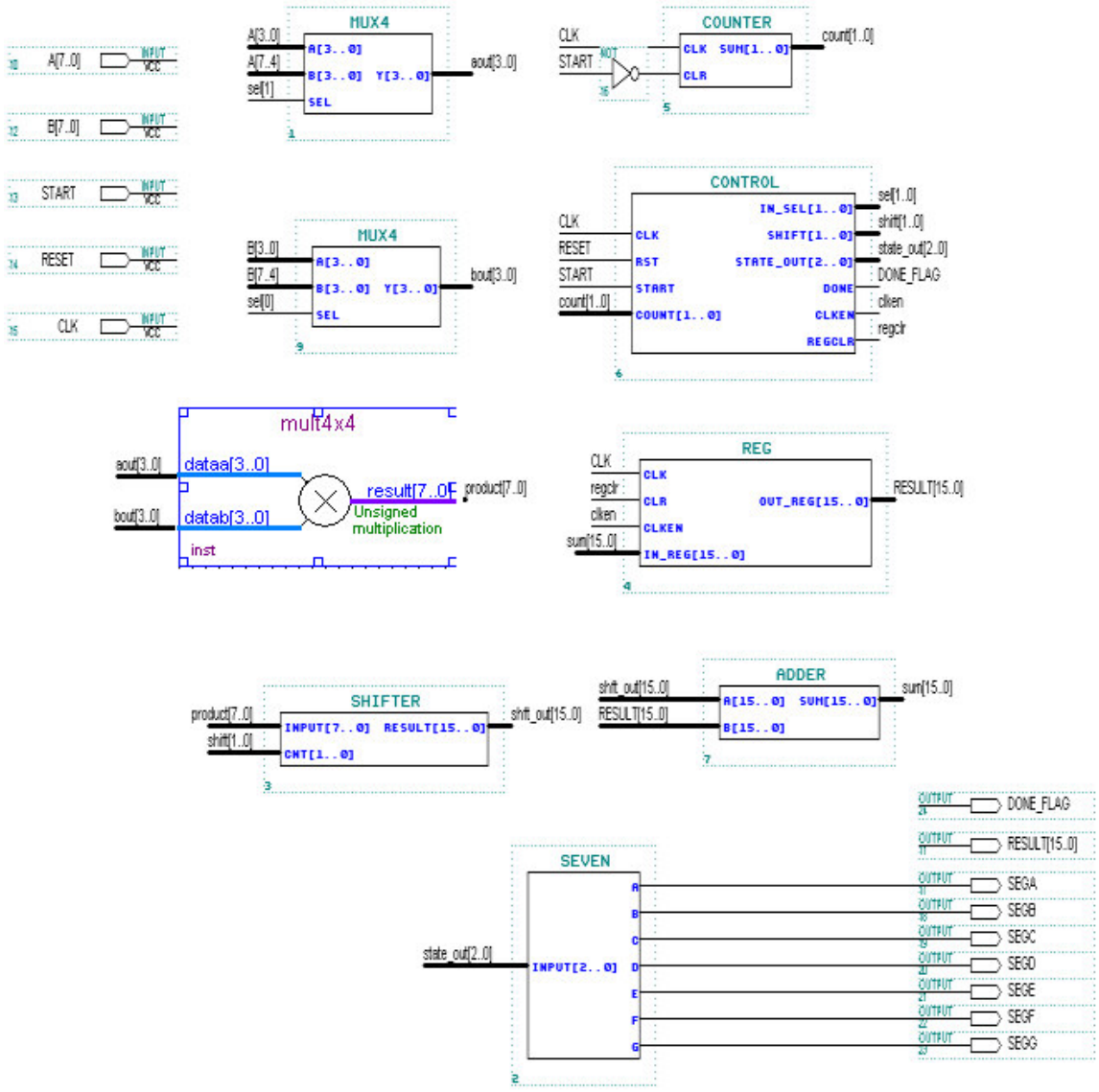


Figure 6-7.

Step 1 (Ask the instructor for the <pathname>)

1. **<pathname>** is_____

2. Use this **<pathname>** pathname for this Lab

Step 2 (Create new project and open VHDL text editor file)

Create a project by using the New Project Wizard. To create a project using the New Project Wizard, please follow the following steps:

1. Select **New Project Wizard** from the File menu. The first time you open the New Project Wizard, it shows the introduction page; you can **click Next** to proceed to the first page of the wizard.
2. **Type** the directory name **or** select the directory with **Browse** ... . The directory name has been provided by the instructor in step1.
3. Type a name for the project in the project name box. For this lab, **type mult8x8**.
4. **Type mult8x8** as the name of the top-level design entity of the project in the top-level design entry box.
5. **Click Next**. The second page of the New Project Wizard appears.
6. Click Add All to add the partially completed mult8x8.vhd file.
7. **Click Next**. The EDA Tools Settings page appears. These exercises will use only Quartus II so the Tool Name should be <NONE> for all the Tool Types.
8. **Click Next**. The Device Family page appears. Select Stratix II as the Device Family and select No, I want to allow the Compiler to choose a device.
9. **Click Next 2 times**. The summary page appears. The summary page gives information about your project.
10. **Click Finish**. You have just finished the project creation. You should see the top-level entity (mult8x8) in the Compilation Hierarchies tab of the Project Navigator window.
11. From the File menu select Open. Open mult8x8.vhd.
12. The VHDL text editor will appear**.**
13. Complete the source code to instantiate mult4x4, shifter, and the PRODUCT signals.

    Remember to use the same input and output port names as shown in Figure 6-7.

14. ***Go to*** File menu ***and Choose*** Save**.**
15. **The lower-level components have been created in different directories than the current directory.**

    In order for Quartus II to find these lower-level components, it must have a search path.

    Therefore, you must do the following:

    - Go to **Assignments menu** and Choose **Settings.** From the **Files & Directories** category choose

      User Libraries tab. Browse the file locations:

      a) Choose the directory structure **<path>\lab1**. Click on **open.**

      b) Choose the directory structure **<path>\ lab2**. Click on **open.**

      c) Choose the directory structure **<path> \lab3**. Click on **open.**

      d) Choose the directory structure **<path>\lab4**. Click on **open.**

      e) Choose the directory structure **<path>\ lab5a**. Click on **open.**

      f) Choose the directory structure **<path>\lab5b**. Click on **open.**

g) Choose the directory structure **<path>\lab6a**. Click on **open.**

h) Choose the directory structure **<path>\lab6b**. Click on **open.**

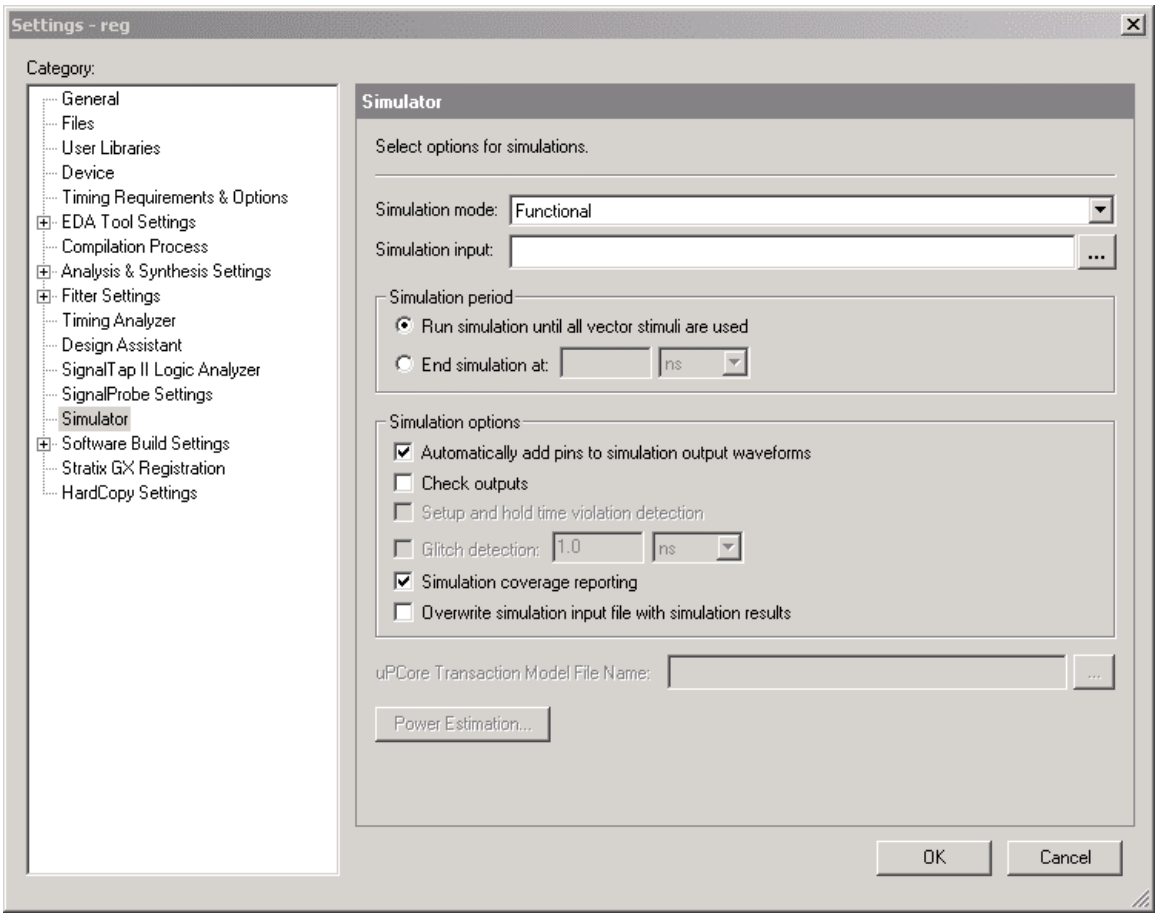
i) Click **OK.**

Step 3 (Synthesize the design)

1. From the Processing menu choose **Start-> Start Analysis and Synthesis**
2. This will save and check for syntax and semantic errors for the file mult8x8.vhd. It will then synthesize the design. When you see the message "Analysis and synthesis was successful", click OK.

Step 5 (Do a functional simulation)

1. The stimulus file has been created for you to verify the functionality of your design. If you are interested in learning how to create your own stimulus file, please go to the Appendix of this manual.
2. Go to the Assignments menu and choose Simulator. In the Simulator mode section select **Functional.**



3. Choose **OK.**
4. From the Processing menu select the **Generate Functional Simulation Netlist**, Click OK when done.

5.

6. From the Processing menu select Start Simulation or click on  . When you see the message, "Simulation was successful", click OK.
7. Check to see if you get the same results shown in Figure 6-11.

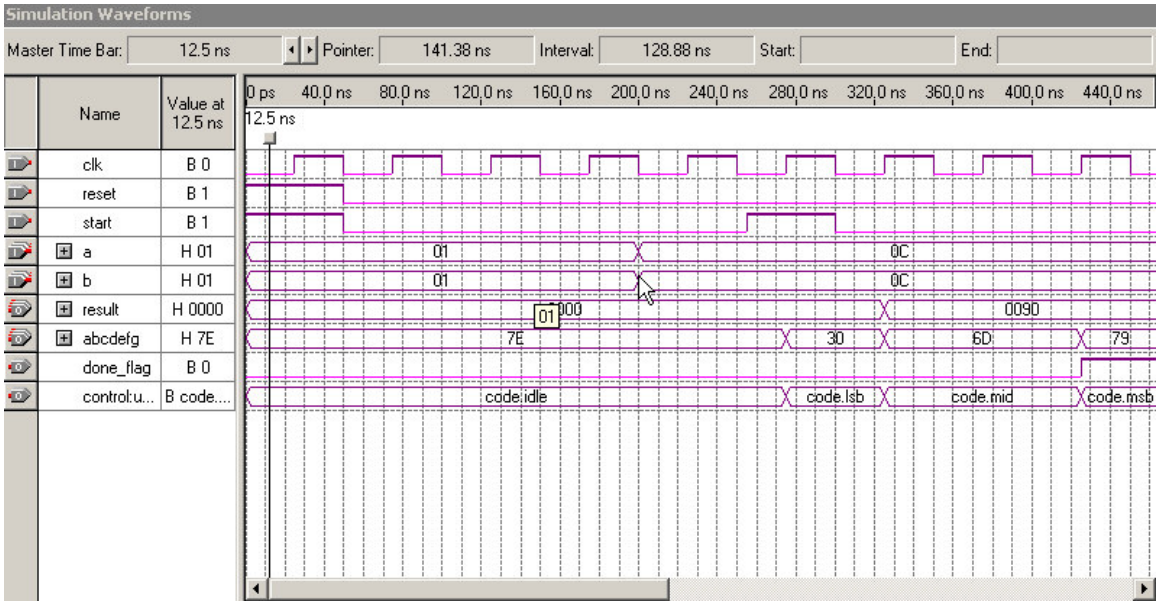


Figure 6-11.

Step 7 (Close the project)

1. From the File menu select Close project option to close the project.

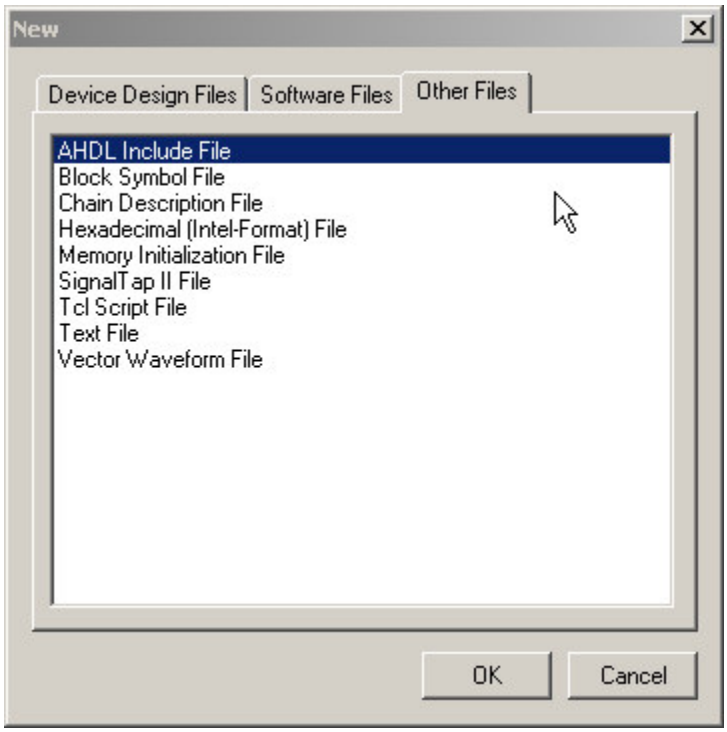
END OF EXERCISE 6

APPENDIX

How to create a stimulus file:

1. Create a new waveform stimulus for simulation. Click on 

2. Under ***Other Files tab:*** Choose **Vector Waveform File**



3. Click **OK.**
4. Enter nodes into the waveform file. ***Edit menu -> Insert Node or Bus***
5. Set the inputs to the appropriate values.
6. Save the file.