ECC3304 [Kumpulan 1] Digital Systems Design

LAB REPORT [Lab 7]

MODELSIM-INTEL SIMULATOR USING VERILOG TESTBENCH

Name: TABINA KAMAL

Matric No.: 208651

Submission Date: 14 January 2022

Lecturer: PUAN ROSLIZAH BINTI ALI

Demonstrator: SITI AQILAH BINTI AZMAN

A. AIMS

- Implementing a testbench and simulating the design using ModelSim software
- Performing basic testbench simulation on an 8-bit binary up counter
- Designing and simulating a 16-bit full adder

B. INTRODUCTION

A testbench consists of pieces of code that are used for the functional simulation of FGPA and ASIC circuits. They provide stimulus to the actual design to carry out the simulation, meaning that no actual inputs are provided nor are actual outputs obtained. Using a testbench ensures that the circuit functions correctly before the synthesis process. It is much more convenient and cheaper to detect bugs in simulation rather than in actual hardware. To ensure the thorough checking of the design, all possible inputs of the design should be tested using the testbench.

ModelSim is a tool that simulates behavioural, RTL, and gate-level codes. For this experiment, the behavioural models of two simple designs; 8-bit binary up counter and 16-bit full adder, were tested using ModelSim.

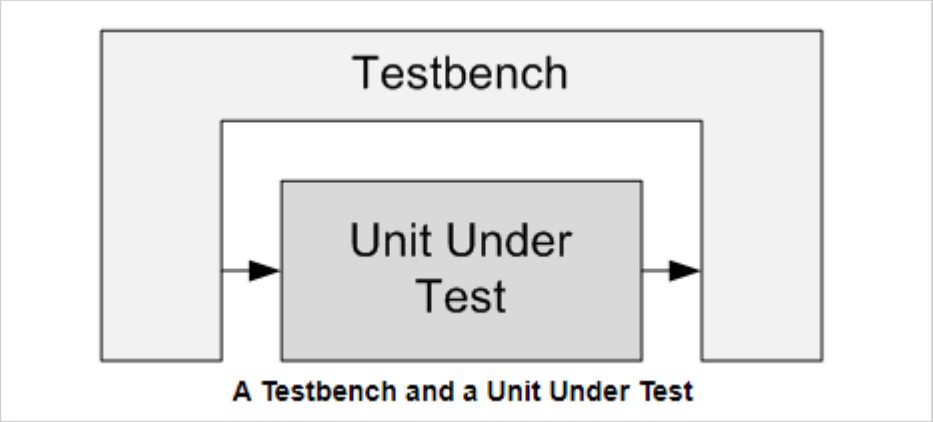

Figure 1: A block diagram showing how a testbench 'wraps' a module or unit under test.

C. METHODS

1) Basic Simulation [8-bit up counter]

i. The ModelSim-Intel FGPA Simulator software was started.

ii. The instructions in the existing documentation ModelSim were referred to for the simulation. *[Help → PDF Documentation → Tutorial → Chapter 3: Basic Simulation]*

iii. The procedures were implemented and the simulation results were recorded.

2) Design and Simulation of a 16-bit Full Adder

i. A 16-bit full adder was designed on Quartus Prime using Verilog entry. The project was named *stb_adder*.

ii. The Verilog design was compiled and verified. Any errors were rectified.

iii. For the same project, another Verilog entry file was created as the testbench called *tstb_adder*. The file was analysed and any errors found were rectified.

iv. Following similar steps as in the *'Basic Simulation'*, the functionality of the 16-bit full adder was simulated and tested on ModelSim using given values (Section D.2)

D. RESULTS

1) Basic Simulation [8-bit up counter]

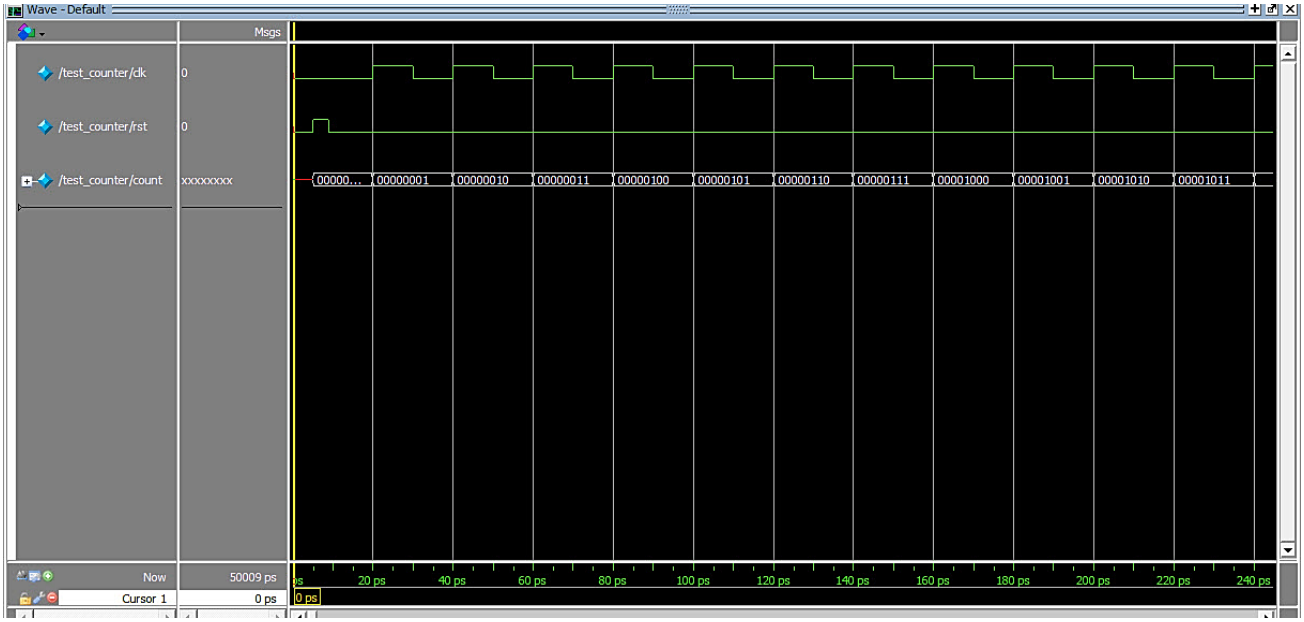

Figure 2: ModelSim waveform simulation for 8-bit binary counter

```
45
46  always @ (posedge clk or posedge rst)
47    if (rst)
48       count = 8'h00;
49    else
50       count <= count + 8'h01;
```

Figure 3: Breakpoint reached for 8-bit binary counter simulation

2) Design and Simulation of a 16-bit Full Adder

```
module stb_adder (A, B, Sum, Cin, Cout);

   input Cin;
   input [15:0] A, B;

   output reg [15:0] Sum;
   output reg Cout;

   always @(A or B or Cin)
      begin
         {Cout,Sum} = A + B;
      end

endmodule
```

Figure 4: 16-bit full adder design Verilog code

```
module tstb_adder;

   //INPUTS
   reg Cin;
   reg [15:0] A;
   reg [15:0] B;

   //OUTPUTS
   wire [15:0] Sum;
   wire Cout;

   //Instatiating the Verilog Design
   //Unit under test [UUT]
   stb_adder UUT(A, B, Sum, Cin, Cout);

   initial begin

      A = 4'hABBA; B = 4'hBABA; Cin = 4'h0000;
      #100;
      A = 4'hFFFF; B = 4'hFFFF; Cin = 4'h0001;
      #100;
      A = 4'h109E; B = 4'h6D99; Cin = 4'h0000;
      #100;
      A = 4'hABCD; B = 4'hCDAB; Cin = 4'h0000;
      #100;
      A = 4'h0000; B = 4'h0180; Cin = 4'h0001;
      #100;

      $finish;

   end


endmodule
```

Figure 5: 16-bit full adder testbench Verilog code

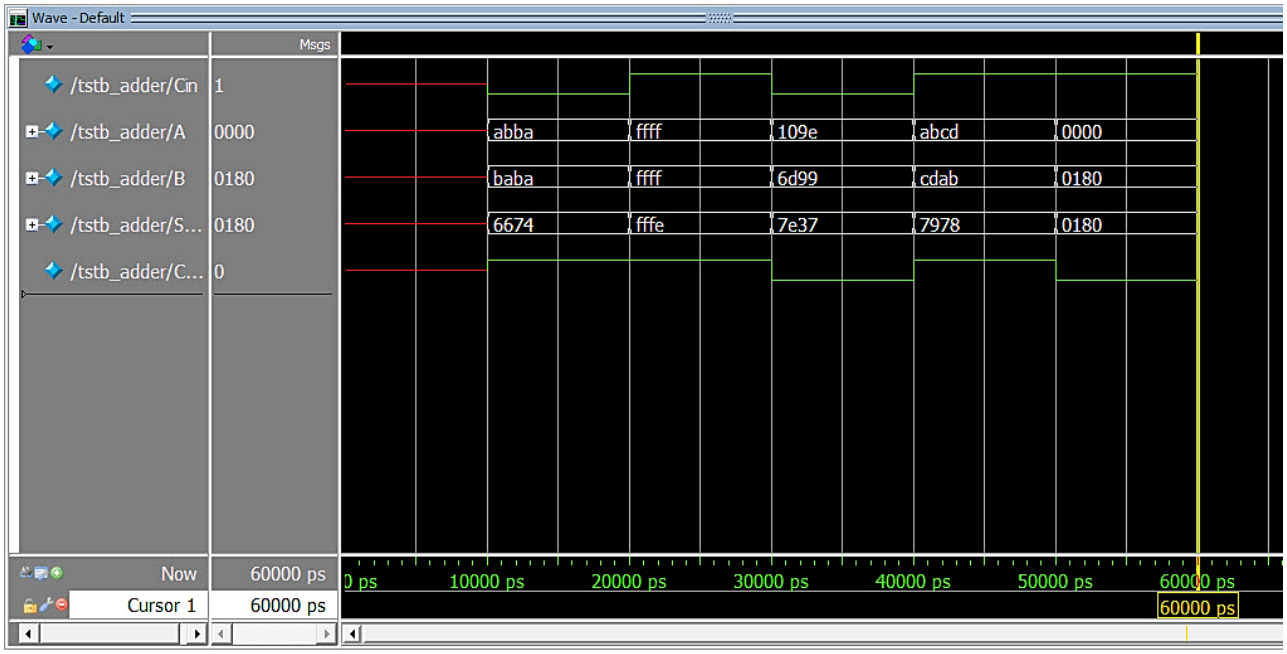

Figure 6: ModelSim waveform simulation for 16-bit full adder

Table 1: 16-bit full adder table of inputs and outputs obtained from simulation

[15:0] A	ABBA	FFFF	109E	ABCD	0000
[15:0] B	BABA	FFFF	6D99	CDAB	0180
Cin	0	1	0	1	1
[15:0] Sum	6674	FFFE	7E37	7978	0180
Cout	1	1	0	1	0

E. DISCUSSION

A testbench file is used for the verification of a digital hardware design. It is used to simulate and analyse a design without the need for any physical hardware. Complicated circuits can be analysed easily with the testbench file on ModelSim unlike the waveform functional simulation in Quartus Prime which would be very tedious and error-prone. Any unexpected results can be easily spotted in the output of the testbench simulation. The use of testbench allows better detection of errors in the Verilog code before being implemented on hardware. In the industry, this will prevent producing faulty hardware, thus ensuring lesser cost and wastage of resources.

1) Basic Simulation [8-bit up counter]

The expected results were obtained from the 8-bit counter. From Figure 2, it is observed that the design counts from binary number 00000000 to 11111111 at the positive edge of the clock input.

2) Design and Simulation of a 16-bit Full Adder

The expected results were obtained from the 16-bit full adder. The Verilog design code was compiled successfully and the testbench code produced the correct results as shown in the waveforms in Figure 6.

F. CONCLUSION

Overall, the experiment was successful as all the expected results were obtained. The objectives of the experiment were fulfilled. The purpose and usage of the testbench were observed. The circuits tested were quite ‘large’ to be tested using the Quartus Prime waveform functional simulation. It is not an impossible process however it would be tedious and an inefficient method to debug the circuit. The testbench simulation using ModelSim allowed for the two ‘large’ designs to be tested using input stimulus given by the user.

G. REFERENCES

[1] "Tutorial - What is a Testbench (simulation)", *Nandland.com*, 2022. [Online]. Available: https://www.nandland.com/articles/what-is-a-testbench-fpga.html. [Accessed: 14- Jan- 2022].

[2] "ModelSim", *Siemens Digital Industries Software*, 2022. [Online]. Available: https://eda.sw.siemens.com/en-US/ic/modelsim/. [Accessed: 14- Jan- 2022].