LOG BOOK
Name: TABINA KAMAL
Matric No.: 208651
Assigned Date: 11 November 2021

LAB 2: INTRODUCTION TO DE1-SoC BOARD

1.0 Objectives

- Familiarising with basic features of DE1-SoC board.
- Writing simple Verilog code.

2.0 Results and Simulation

A. Introduction to Quartus Prime Design Software (continuation of Lab 1)

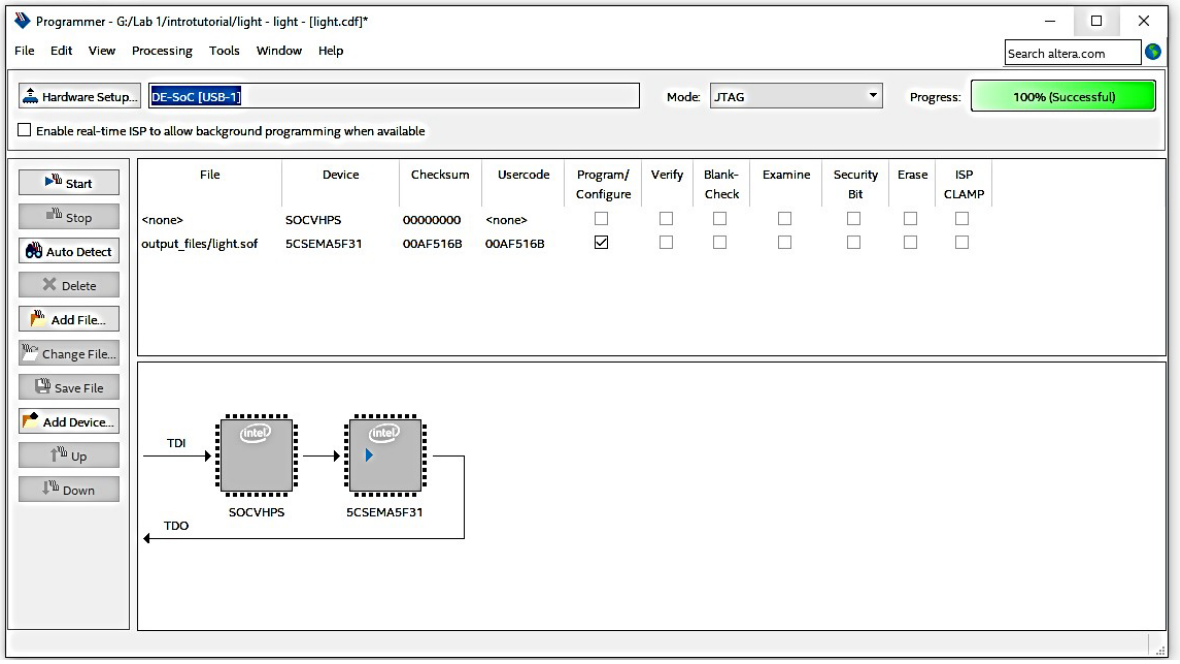

Figure 1: Successful configuration of DE1-SoC board

The truth table was created from the obtained output on the board. The value 0 represents OFF and 1 represents ON. The inputs are SW[0] and SW[1]. The output is LEDR[0].

SW[0] / x_1	SW[1] / x_2	LEDR[0] / f
0	0	0
0	1	1
1	0	1
1	1	1

B. Testing a simple Verilog design

Part 1 to 3

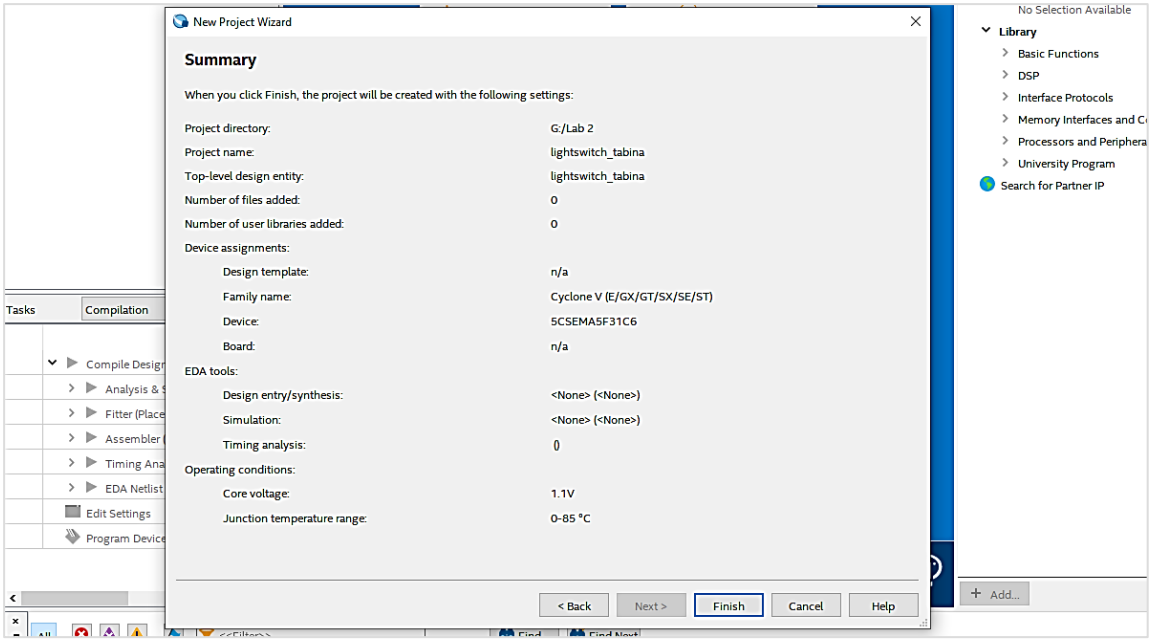

Figure 2: Creation of new project

The truth table was created from the obtained output on the board. The value 0 represents OFF and 1 represents ON. The inputs are SW[0], SW[1], SW[2] and SW[3]. The output is LEDR[0], LEDR[1], LEDR[2] and LEDR[3].

SW[0] / x_1	SW[1] / x_2	SW[2] / x_3	SW[3] / x_4	LEDR[0] / f_1	LEDR[1] / f_2	LEDR[2] / f_3	LEDR[3] / f_4
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	1
0	1	0	0	0	1	0	0
0	1	0	1	0	1	0	1
0	1	1	0	0	1	1	0
0	1	1	1	0	1	1	1
1	0	0	0	1	0	0	0
1	0	0	1	1	0	0	1
1	0	1	0	1	0	1	0
1	0	1	1	1	0	1	1
1	1	0	0	1	1	0	0
1	1	0	1	1	1	0	1
1	1	1	0	1	1	1	0
1	1	1	1	1	1	1	1

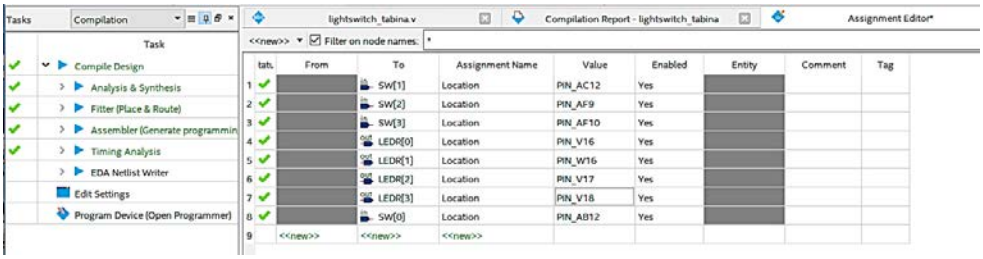

Figure 3: Pin assignment

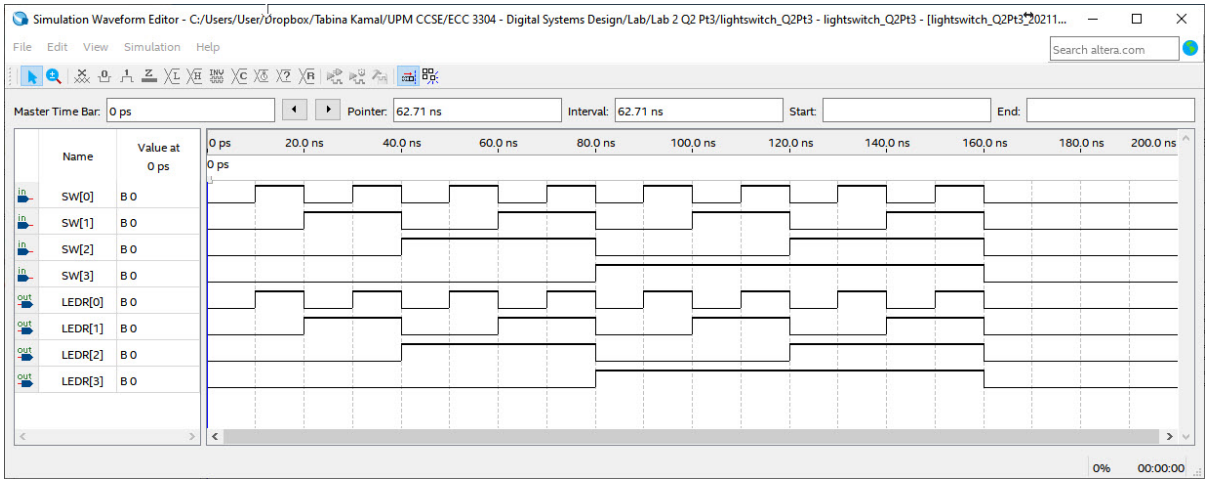

Figure 4: Functional simulation of code

Part 4

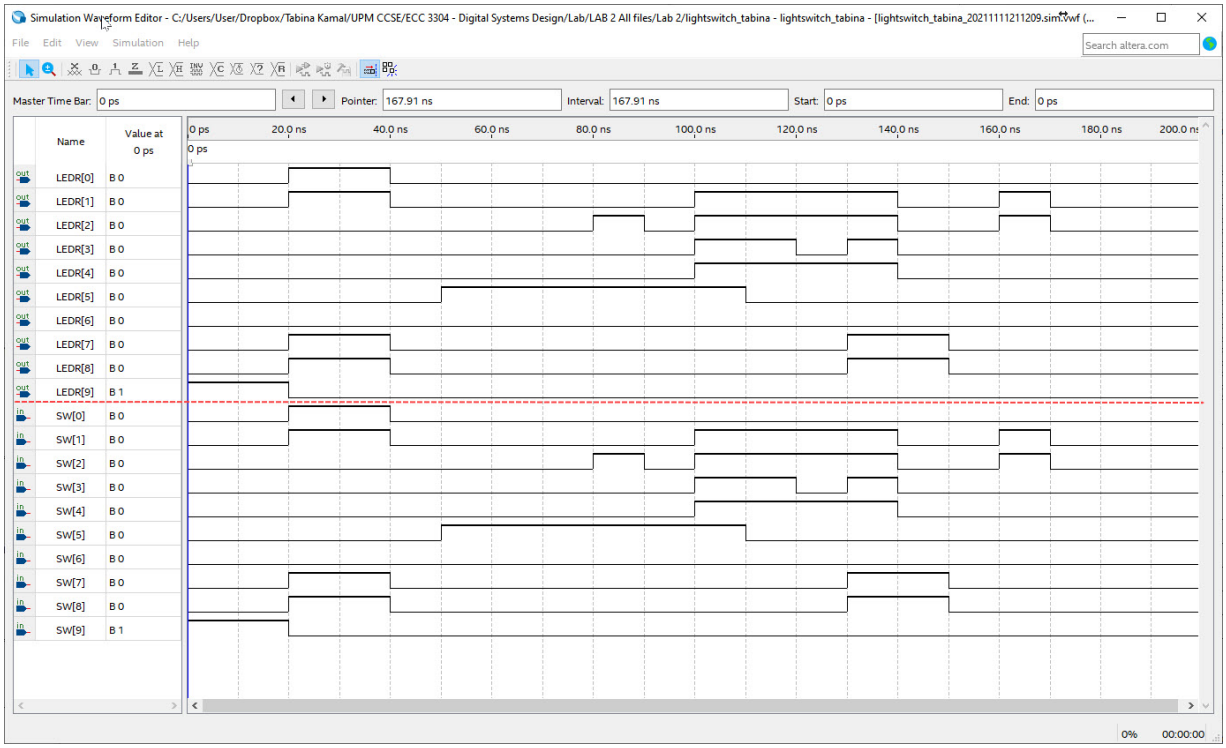

Figure 5: Functional simulation of modified code

Part 5

Single_Segment_7_Display_Tabina.v | Compilation Report - Single_Segment_7_Display_Tabina | Assignment Editor

```
1
2  // Module that connects the SW switches to the 7 Segment lights
3
4  module Single_Segment_7_Display_Tabina (SW, HEX0);
5  input [6:0] SW; // toggle switches
6  output [6:0] HEX0; // red LEDs
7     assign HEX0[0] = SW[0];
8     assign HEX0[1] = SW[1];
9     assign HEX0[2] = SW[2];
10    assign HEX0[3] = SW[3];
11    assign HEX0[4] = SW[4];
12    assign HEX0[5] = SW[5];
13    assign HEX0[6] = SW[6];
14 endmodule
```

Figure 6: Code for single 7-segment display

Part 6

DisplayCCSE_Tabina.v* | Compilation Rep

```
1  //ACTIVE LOW INPUT
2  module DisplayCCSE_Tabina (HEX0, HEX1, HEX2, HEX3);
3  output [6:0] HEX0, HEX1, HEX2, HEX3; // red LEDs
4     assign HEX0[0] = 0;
5     assign HEX0[1] = 1;
6     assign HEX0[2] = 1;
7     assign HEX0[3] = 0;
8     assign HEX0[4] = 0;
9     assign HEX0[5] = 0;
10    assign HEX0[6] = 1;
11
12    assign HEX1[0] = 0;
13    assign HEX1[1] = 1;
14    assign HEX1[2] = 1;
15    assign HEX1[3] = 0;
16    assign HEX1[4] = 0;
17    assign HEX1[5] = 0;
18    assign HEX1[6] = 1;
19
20    assign HEX2[0] = 0;
21    assign HEX2[1] = 1;
22    assign HEX2[2] = 0;
23    assign HEX2[3] = 0;
24    assign HEX2[4] = 1;
25    assign HEX2[5] = 0;
26    assign HEX2[6] = 0;
27
28    assign HEX3[0] = 0;
29    assign HEX3[1] = 1;
30    assign HEX3[2] = 1;
31    assign HEX3[3] = 0;
32    assign HEX3[4] = 0;
33    assign HEX3[5] = 0;
34    assign HEX3[6] = 0;
35 endmodule
```

Figure 7: Code for displaying CCSE

LOG BOOK
Name: TABINA KAMAL
Matric No.: 208651
Assigned Date: 11 November 2021

3.0 Discussion

A. Introduction to Quartus Prime Design Software (continuation of Lab 1)

The simulation was carried out successfully in Lab 1 and the same results were obtained on the DE1-SoC board.

There was a minor issue when configuring the device. The following steps were taken:
Open 'Programmer' window > Delete the current device > Auto-detect devices > Choose the option 5CSEMAC5 > Delete the 5CSEMAC6 > Select 'Add file' > Add *.sof* file of the project.

B. Testing a simple Verilog design

Part 1 to 3

The code was successfully run on the simulation as well as the board. When running it was discovered that the module name needed to be the same as that of the file name.
In the given code, the first line is: `module Lab_1a (SW, LEDR);`
The code was modified to: `module lightswitch_tabina (SW, LEDR);`

Part 4

Code was written and successfully compiled. The functional simulation was successful. The expected result is that the input values are equal to the outputs.

Part 5

Code was written and successfully compiled.

Part 6

Code was written and successfully compiled.

4.0 Conclusion

Part 1 to 3: Completed and demonstrated

Part 4: Not demonstrated

Part 5: Not demonstrated

Part 6: Not demonstrated