MÜHENDİSLİK FAKÜLTESİ
FACULTY OF ENGINEERING
ELEKTRİK-ELEKTRONİK MÜHENDİSLİĞİ BÖLÜMÜ
DEPARTMENT OF ELECTRICAL AND
ELECTRONICS ENGINEERING

ORTA DOĞU TEKNİK ÜNİVERSİTESİ
MIDDLE EAST TECHNICAL UNIVERSITY

DUMLUPINAR BULVARI 06800
ÇANKAYA ANKARA/TURKEY
T: +90 312 210 23 02
F: +90 312 210 23 04
ee@metu.edu.tr
www.eee.metu.edu.tr

EXPERIMENT 1. INTRODUCTION TO ALTERA

I. Introduction

I.I Objectives

In this experiment, you will learn computer aided digital design and verification of it using Field Programmable Gate Arrays (FPGA). For programmable logic design, you will learn how to use Quartus II 14.1 software tools.

I.II. Background: Implementation of a logic function with standard gates

Let us design a combinational circuit with three inputs and one output. The output is equal to logic-1 when binary value of the input is less than 3. The output is logic-0 otherwise. Table 1 shows the truth table for this logic function, where x, y, z are the inputs and S is the output. Figure 1 shows the schematic diagram of the designed combinational circuit.

Table 1. Truth table of the design

x	y	z	S
0	0	0	1
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0

$S = x'y'z' + x'y'z + x'yz'$
$S = x'y'(z'+ z)+ x'yz'$ since $(z' + z) = 1$ and $\mu \cdot 1 = \mu$
$S = x'y' + x'yz'$

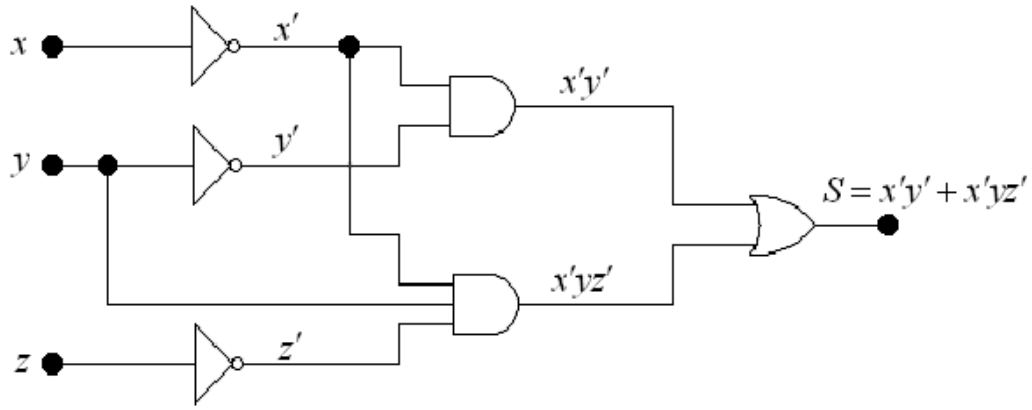


Fig. 1: Logic diagram of the design

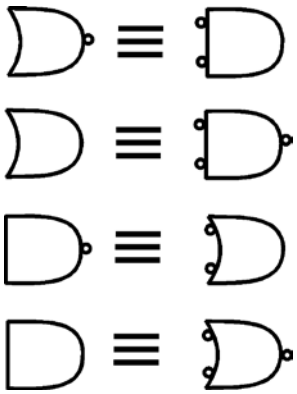
II. Preliminary Work

1) Read the document about ESD and briefly explain your understanding in less than 250 words. (what is ESD, effects of ESD, how to protect from ESD etc…)

2) Read the document about FPGA and explain what FPGA is, applications and limitations in less than 250 words.

3) Read the Background information of the experiment, design an alternative circuit other than the circuit shown in the Fig. 1.

Hint:



4) Bring an at least 8 GB USB to EA314 (or download Quartus II 14.1 web edition) and take the installation files from Yeşim Yücesan in EA 314 during the week. Install Quartus II 14.1 to your computer, and do the tutorial given to you until the end of page 31 (Programming and Configuring the FPGA Device). **Include the screenshots to show that you fulfilled this task fully and completely.**

III. Experimental Work

a) Starting a New Project

1) Create a directory on your desktop with the name of Exp1.

2) Double click on the Quartus II 14.1 icon on your desktop.

3) After opening Quartus II 14.1, you will start a new project. **Select File>New Project Wizard** and click **Next** in the opening window. In the new window set the working directory as Exp1 and write the project name as EE314 as seen from the Figure 2 then click **Next**.

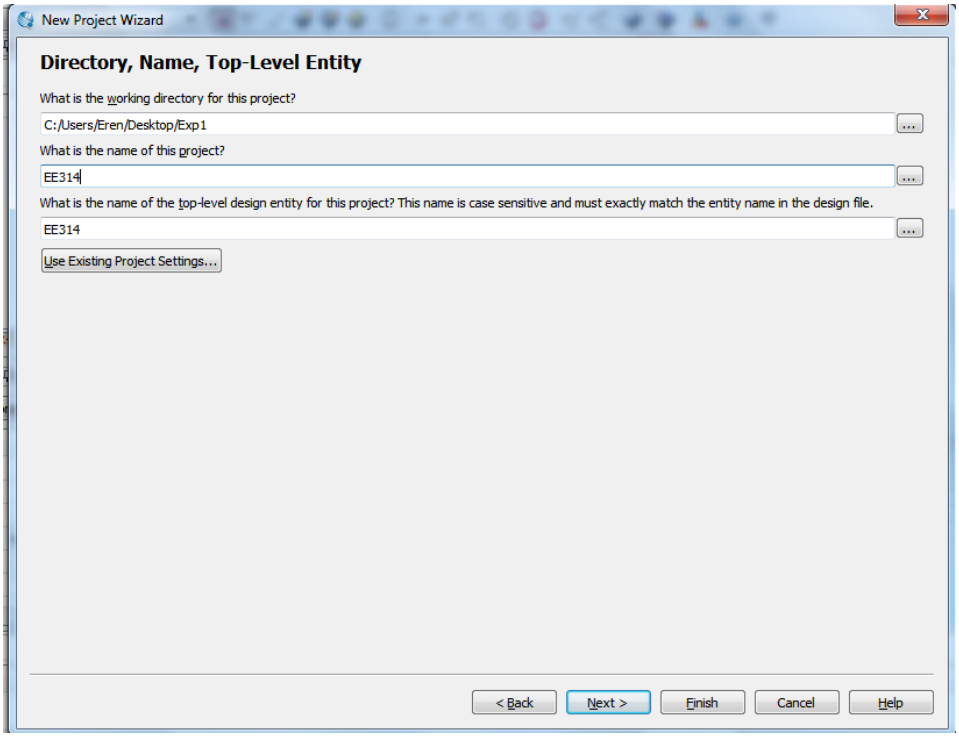


Fig. 2: Assigning working directory and project name

4) Choose **Empty Project** and click **Next**.

5) Click **Next.**

6) Choose Device Family as **Cyclone V** and Choose **5CSEMA5F31C6** (for DE1-SoC) from the list and click **Next**.

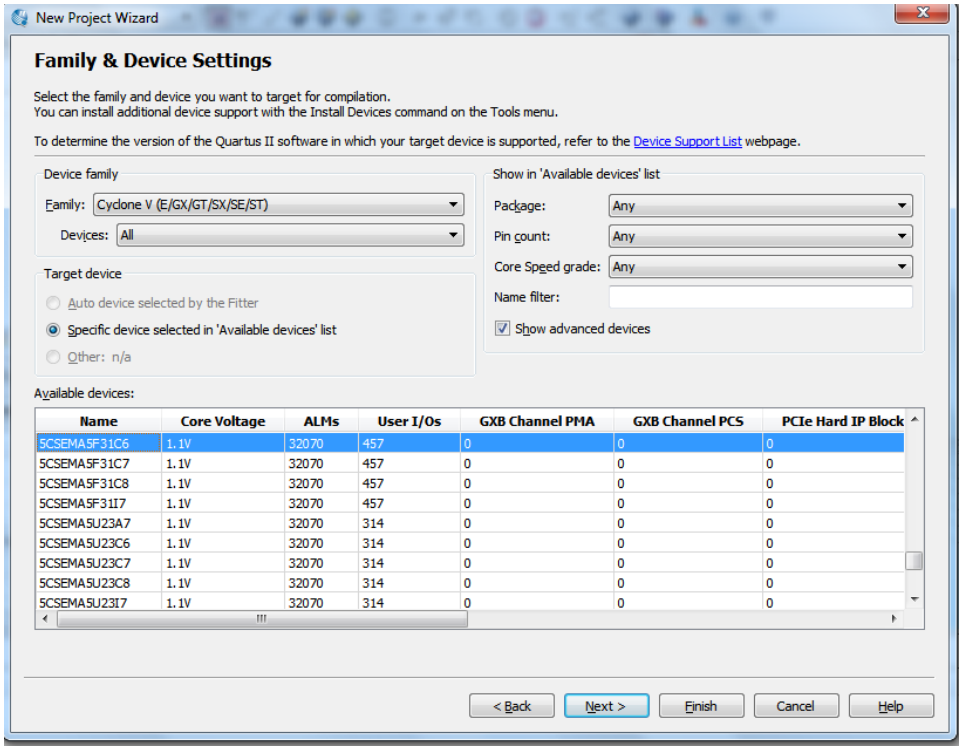


Fig. 3: Device Selection Window

7) Click **Next**.

8) A summary window will be visible. The summary window should be similar to the Fig. 4. Click **Finish**.

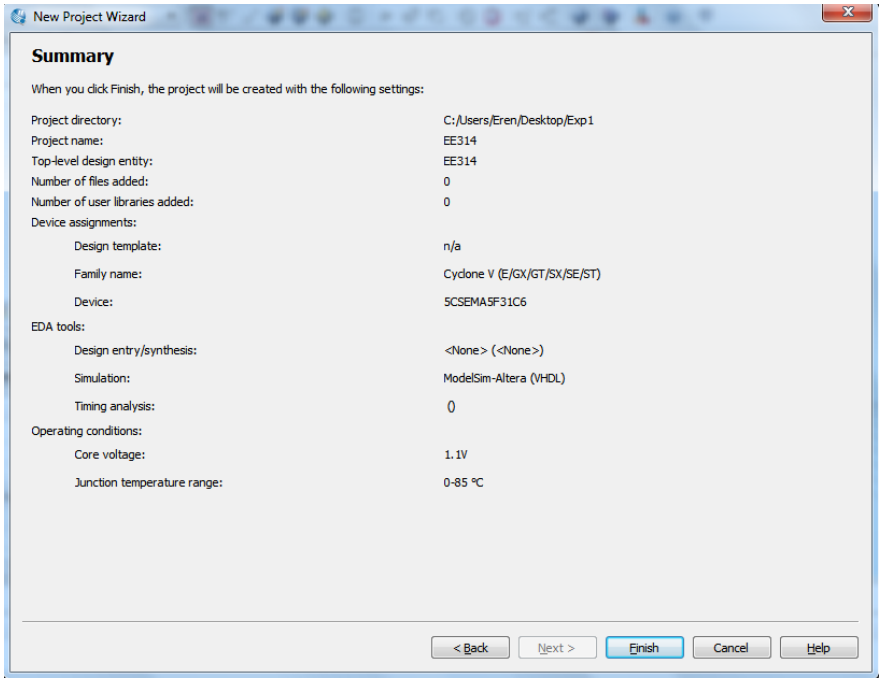


Fig. 4: Summary Window

b) Entering The Design

1) Choose **File>New >Block Diagram/Schematic File** and click **OK**. Then Choose **File>Save as** and enter the name of the file as EE314. Ensure that "**Add the file to current projec**t" is selected and then click **Save**.

2) Click on  icon and expand the **library** directory and then expand **primitives** directory. In primitives, expand **logic** directory and choose **and2** gate as shown in Fig. 5.

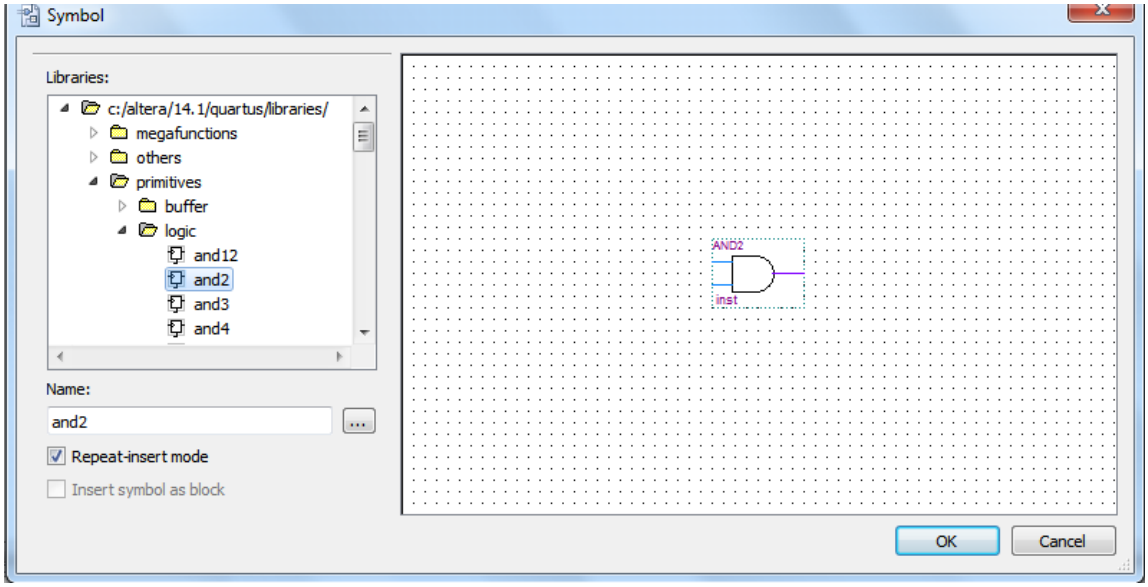


Fig. 5: Choosing a symbol from library

3) Place **and2** gate into the sheet and then choose and place **and3, or2** and **not** gates as you choose **and2** gate from the library.

4) Before wiring, **Input and output pins** are needed to be placed. Apply same procedure like choosing a gate but now expand **pin** directory instead of logic directory. Then choose and place **Input and Output** pins respectively.

5) Right click on the input pin at the top of the schematic and choose properties. Write x as pin name and click OK. Do the same thing for the other input and output pins and name the remaining pins as y, z and S respectively.

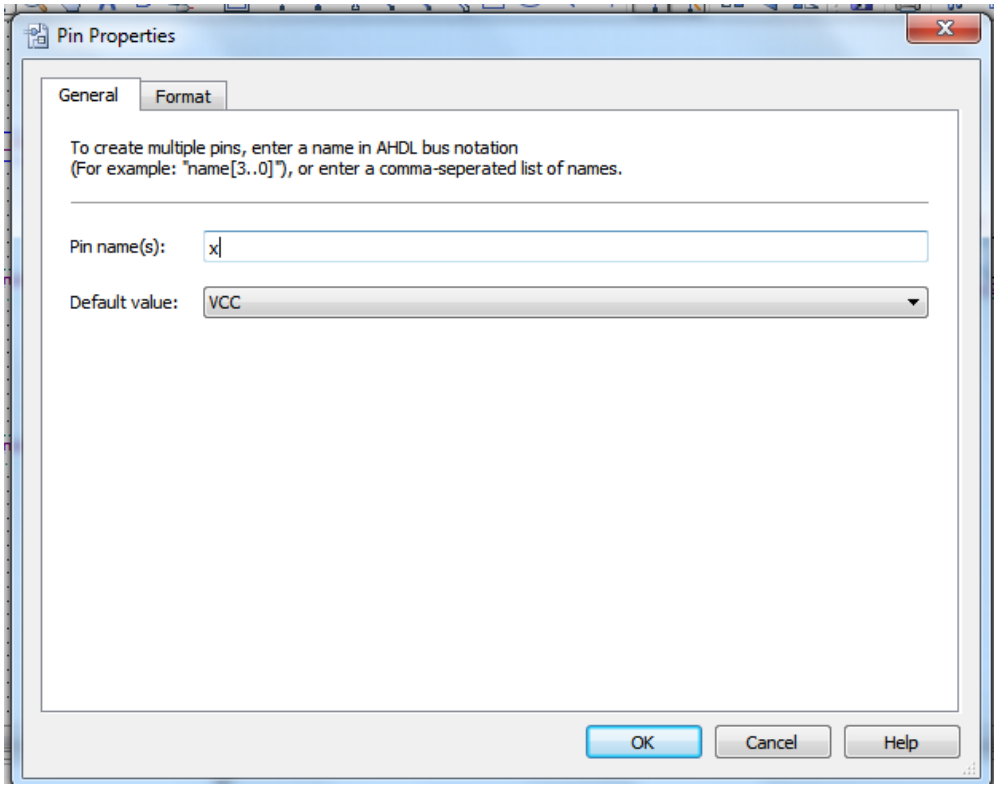


Fig. 6: Assigning Pin Names

6) Now we are ready to make wiring. Click on (Orthogonal Node Tool) icon. Position the mouse pointer over the right edge of the *x* input pin. Click and hold the mouse button and drag the mouse to the right until the drawn line reaches the pinstub on the top input of the not gate. Release the mouse button when you see a box appear, which leaves the line connecting the two pinstubs. Apply same procedure for remaining wires and construct the circuit shown in Fig. 7. After wiring save the schematic.

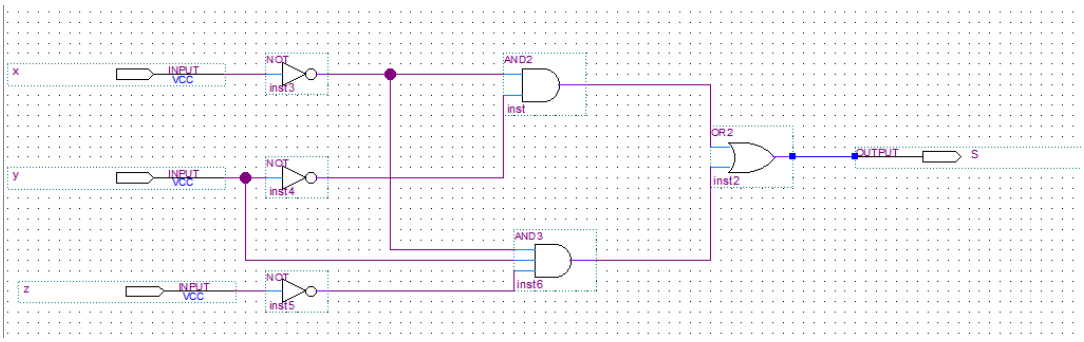


Fig. 7: The completed circuit diagram

c) Compiling the Designed circuit

The entered schematic diagram file, EE314.bdf, is processed by several Quartus II tools that analyze the file, synthesize the circuit, and generate an implementation of it for the target chip. These tools are controlled by the application program called the Compiler.

1) Run the compiler selecting **Processing > Start Compilation** or click on icon. You are supposed to save your project before compiling. As the compilation is continuing, the progress is reported at the left of the schematic. When the compilation finishes, compilation report is produced. A tab showing the report is opened automatically and you can close it. To open it whenever you want, click **Processing>Compilation Report**. There are several messages are in the message window. When compilation becomes unsuccessful, appropriate messages will be given in there. The interface looks like in Fig. 8 after a successful compilation.

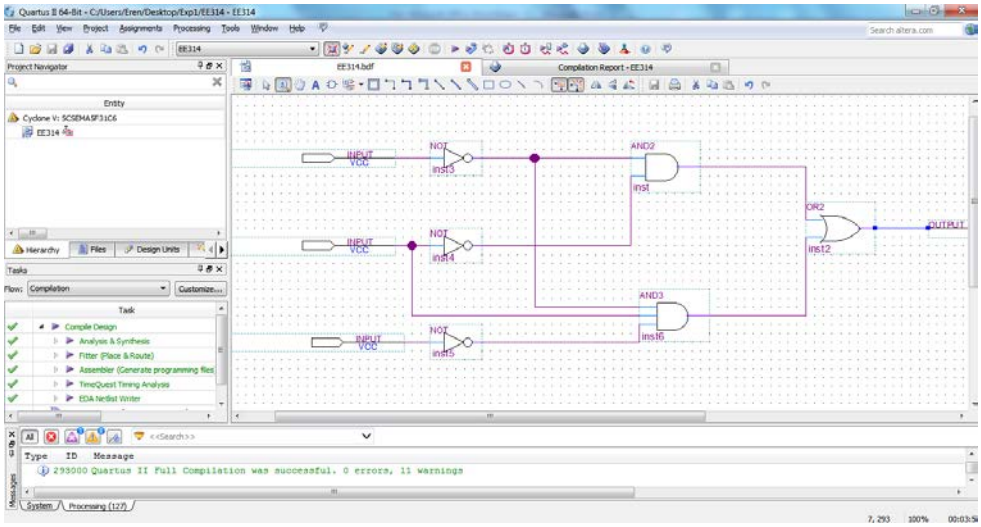


Fig. 8: The screen after a successful compilation

2) Now we will see what happens if the compiler cannot compile the designed circuit. Click on the  icon and right click on the wire between **output of and2** and **input of or2 gate** and then click on **Delete**. Then compile your design. You will see a screen similar to the Fig. 9.

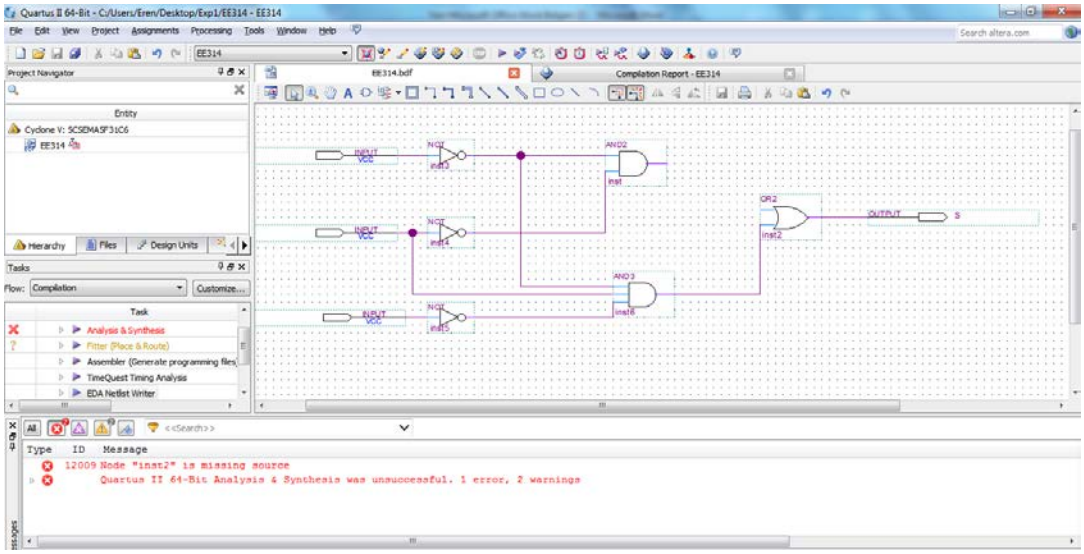


Fig. 9: Unsuccessful Compile Screen

3) Fix the circuit and recompile your design again.

d) Pin Assignment

We are supposed to choose input and output pins to be able to test the circuit implemented on FPGA. As input we will use switches and as output we will use a LED.

1) Select **Assignments>Assignment Editor**.

2) In the **Category** drop-down menu, select All. Click on the **<<new>>** button located near the top left corner to make a new item appear in the Table 2. Double click the box under the column labeled **To** so that Node Finder button  appears.

3) Click on the button to reach the window in Fig. 10. Click on  to reach more search options. In Filter drop down menu select **Pins: all**. Then click the List button to display output and input pins to be assigned: (S, x, y, z). Click on **>>** button and click OK. Now all pins are labeled under the column **To**.

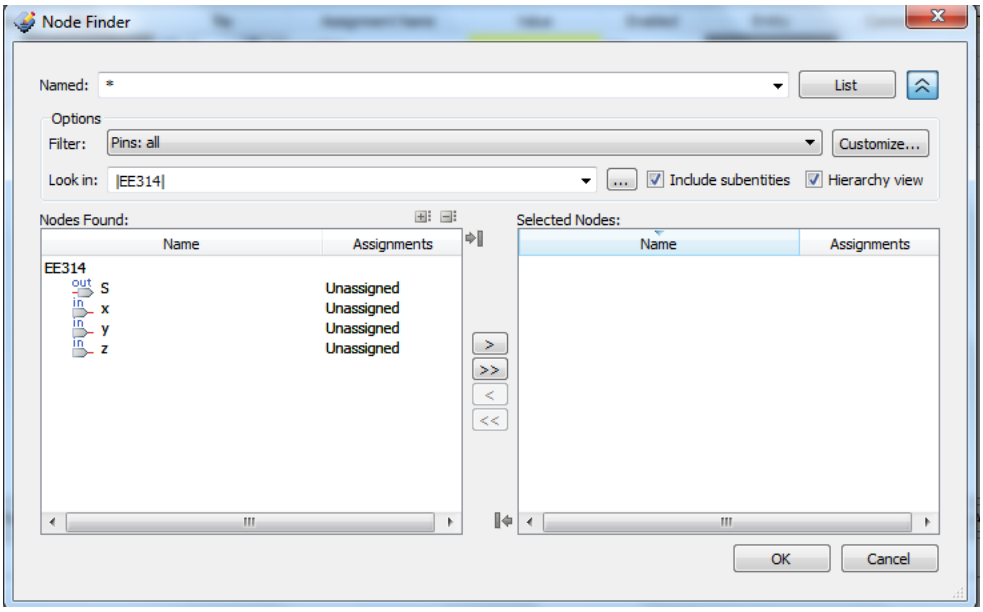


Fig. 10: Node Finder window

4) Now open the drop down menu under **Assignment Name** column and select **Location(Accepts wildcards/group)** as shown in Figure 11.

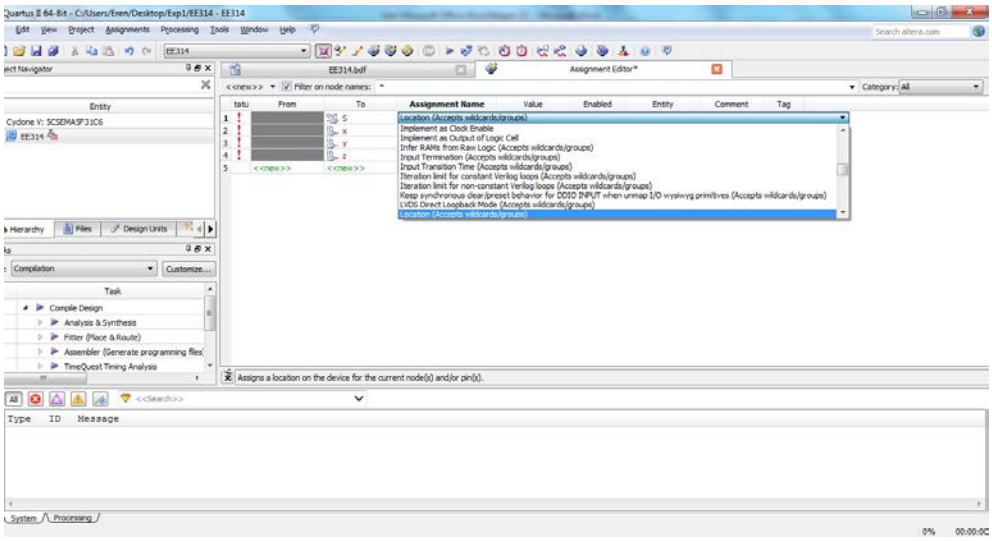


Fig. 11: Assignment Name selection window

5) We will assign **LEDR[0]** as output and **SW[0], SW[1], SW[2]** as x, y, z respectively. Write the FPGA Pin No. expression in Appendix A to the column under the Value. For example to assign SW[0] switch to x, PIN_AB12 should be written to the row under the Value column. The resulting window should be similar to the window in Fig. 12.

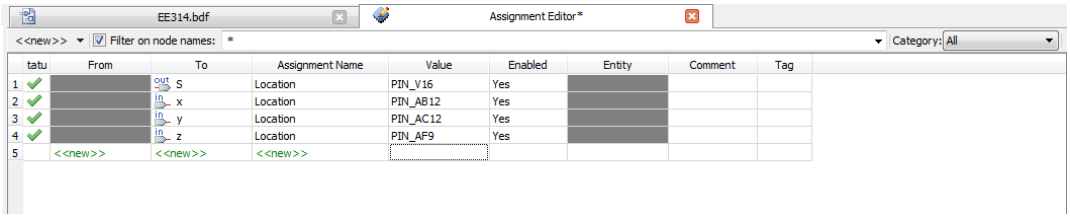


Fig. 12: Pin Assignment window after pin values are written

6) You can export the pin assignment clicking on **Assignment>Export Assignments**. Then click **OK** on the window and then click on **Yes** on the pop up window. Moreover it is possible to import a pin assignment to the current project.

e) Simulating the Designed Circuit

In this phase we will check the functionality of the designed circuit.

1) Click on **File>New>Verification/Debugging Files>University Program VWF** to open the **Simulation Waveform Editor**. The simulation window is opened. Now save it as **EE314.vwf**.

2) In the simulation window, set the desired simulation from 0 to 400 ns selecting **Edit>Set End Time** and entering **400 ns** in the dialog box that pops up. Then select **View>Fit in Window** to display the entire simulation range of 0 ns to 400 ns in the window shown in Figure 13.

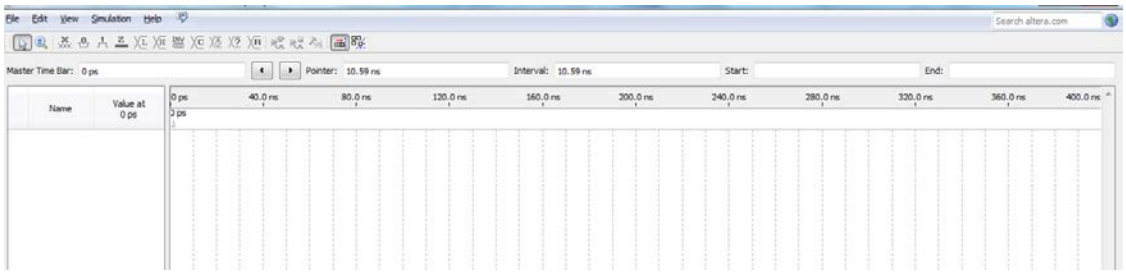


Fig. 13: The augmented waveform editor window

3) Now input and output nodes are included into the simulation. Click **Edit>Insert>Insert Node or Bus** and then click **Node Finder** on the opened window. Then like in pin assignment in Filter menu choose Pins: all and click on

the List. Then click on **>>** and then **OK.** Then click **OK** again on the opened window again.

4) Now, input waveforms are needed to be arranged. To make it easy to draw waveforms, waveform editor displays vertical guidelines and provides a drawing feature that snaps on these lines which can be invoked clicking **Edit>Snap to Grid** (a tick sign is needed to be seen). Click on waveform for x node and choose the time interval between 200 and 400 ns clicking on mouse when the pointer is near 200 ns and drag it until 400 ns. Then choose **Edit> Value>Force High**. Apply similar procedure to y and z to obtain the waveforms similar to Fig. 14. Then save the file.

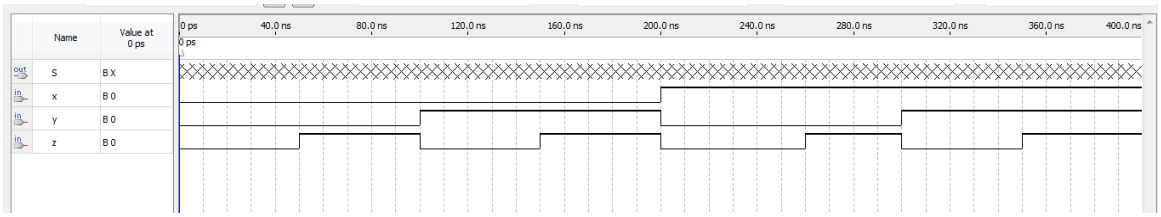


Fig. 14: Desired input waveforms

5) Before running functional simulation **Analysis and Synthesis** is required. Therefore click on  on main Quartus II screen. Then click on  icon. Then output waveform will be observed in a new window. In Fig. 15, functional simulation result is shown.

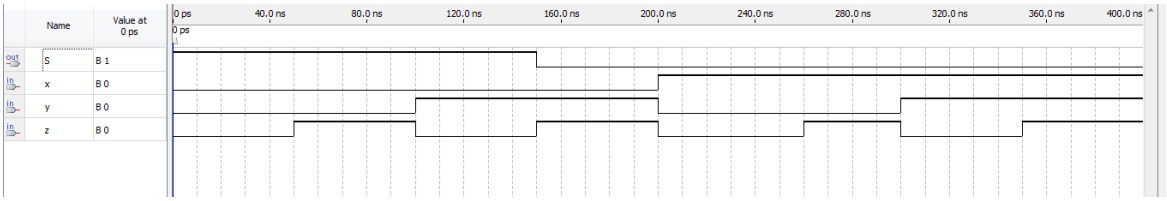


Fig. 15: Functional Simulation Result

As seen from Fig. 15, the simulation results are consistent with the truth table used while designing this circuit. The next phase is programming the FPGA.

f) Programming the Device

1) Recompile the design.

2) Click on **Program Device (Open Programmer)**.

3) If DE1-SoC is not chosen default, click on **Hardware Setup** and choose **DE1-SoC** on the opened window then click on Close.

4) Click on Auto Detect and then click on **5CSEMA5** and click on OK. Click **Yes** on the pop up window. After clicking **Yes** the screen should be similar to Fig. 16.

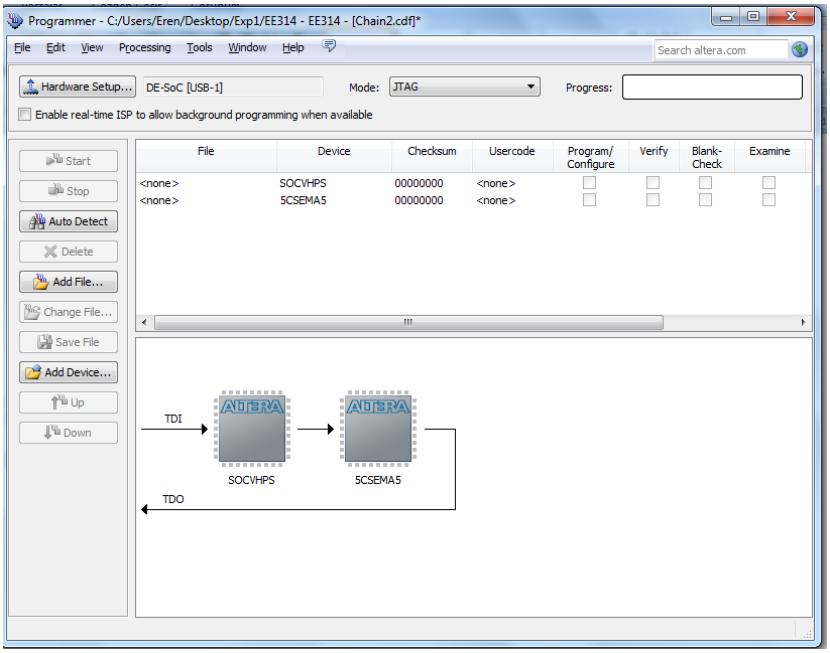


Fig. 16: Programmer window after step 4

5) Double Click the **column under the file** on **5CSEMA5 row** and click on **output_files** directory and then click on **EE314.sof** file. This file is your programming file as shown in Fig. 17.

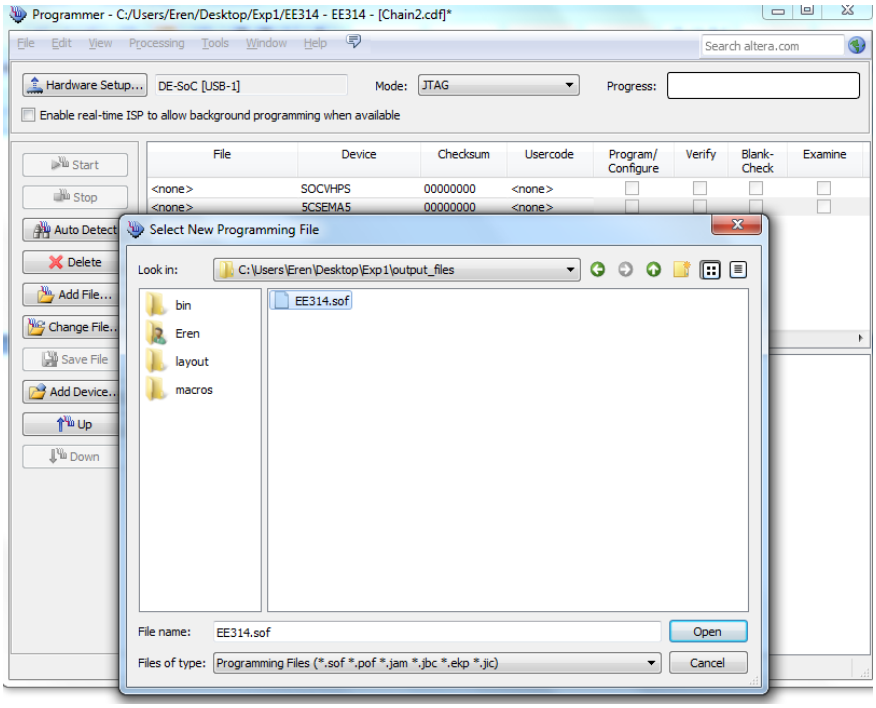


Fig. 17: Finding the programming file

6) Click on OK. Then click on **Program/Configure**. Programmer window should be similar to Fig. 18.

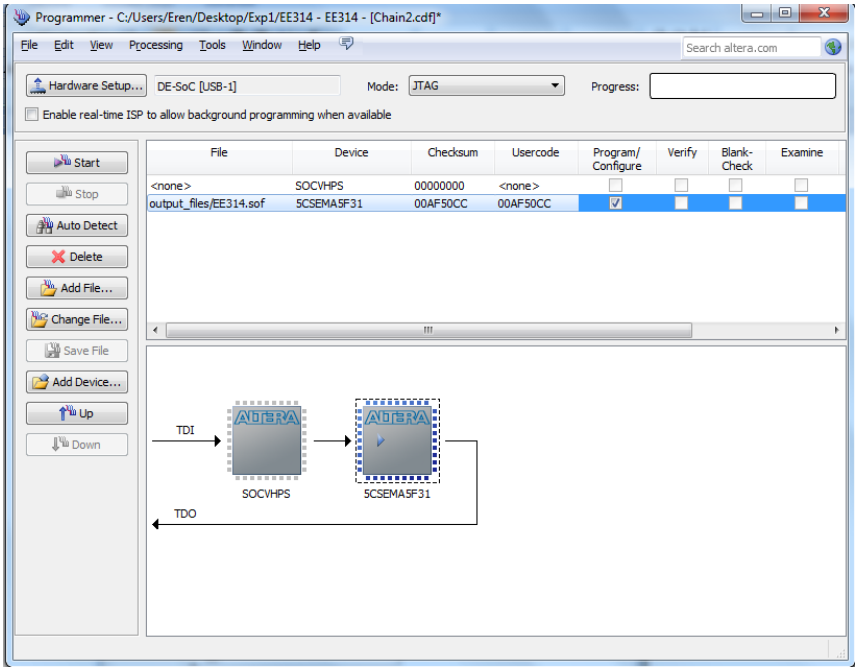


Fig. 18: Programmer window after step 6

7) Click on Start. If the Progress Bar is similar to Fig. 19, it means that your device is programmed successfully.

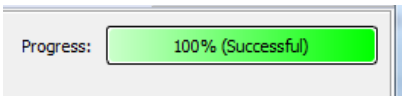


Fig. 19: Progress Bar after programming the device

g) Testing the Circuit

Test the designed circuit applying different input combinations and compare your results with the truth table and simulation results.

Appendix A: FPGA switch, button and LED description table

Signal Name	*FPGA Pin No.*	*Description*	*I/O Standard*
SW[0]	PIN_AB12	Slide Switch[0]	3.3V
SW[1]	PIN_AC12	Slide Switch[1]	3.3V
SW[2]	PIN_AF9	Slide Switch[2]	3.3V
SW[3]	PIN_AF10	Slide Switch[3]	3.3V
SW[4]	PIN_AD11	Slide Switch[4]	3.3V
SW[5]	PIN_AD12	Slide Switch[5]	3.3V
SW[6]	PIN_AE11	Slide Switch[6]	3.3V
SW[7]	PIN_AC9	Slide Switch[7]	3.3V
SW[8]	PIN_AD10	Slide Switch[8]	3.3V
SW[9]	PIN_AE12	Slide Switch[9]	3.3V

Signal Name	*FPGA Pin No.*	*Description*	*I/O Standard*
KEY[0]	PIN_AA14	Push-button[0]	3.3V
KEY[1]	PIN_AA15	Push-button[1]	3.3V
KEY[2]	PIN_W15	Push-button[2]	3.3V
KEY[3]	PIN_Y16	Push-button[3]	3.3V

Signal Name	*FPGA Pin No.*	*Description*	*I/O Standard*
LEDR[0]	PIN_V16	LED [0]	3.3V
LEDR[1]	PIN_W16	LED [1]	3.3V
LEDR[2]	PIN_V17	LED [2]	3.3V
LEDR[3]	PIN_V18	LED [3]	3.3V
LEDR[4]	PIN_W17	LED [4]	3.3V
LEDR[5]	PIN_W19	LED [5]	3.3V
LEDR[6]	PIN_Y19	LED [6]	3.3V
LEDR[7]	PIN_W20	LED [7]	3.3V
LEDR[8]	PIN_W21	LED [8]	3.3V
LEDR[9]	PIN_Y21	LED [9]	3.3V