Jyresa Mae M. Amboang
2020-2591
Laboratory Submitted: April 4, 2023
Laboratory Hours: 216 hrs

Laboratory Activity No. 2

Seven-Segment Display

Purpose: In this laboratory activity you will learn to use the Quartus II, ModelSim, and FPGAs to create a seven segment display hex decoder. Implementing and simulating the seven segment display hex decoder in the Altera DE2-115 and breadboard.

Overview: In laboratory activity 1, you became familiar with the Quartus schematic editor and ModelSim simulator. In laboratory activity 2, you design a more complicated block of combinational logic. To minimize design time, use design practices that will make both designing and debugging efficient. To construct the seven-segment display decoder, skim through the entire laboratory activity first and refer to the "What to Turn In" section. There are also hints on the last page of the laboratory activity.

Theory: A device with seven segments set up in a certain configuration may display the letters A–F as well as the numbers 0–9. A single input controls each segment, and all inputs are merged to show a single character. We may utilize a truth table that associates each potential input combination with a certain output segment value to decode a 4-bit input into a hexadecimal digit. This truth table may then be used to build a logic circuit that produces the right output for each set of inputs. In order to design the 7-segment hex decoder using an FPGA, we first need to build a logic circuit that receives a 4-bit input and produces the appropriate output for every conceivable input combination. This circuit may be built by combining multiplexers and logic gates. After creating the logic circuit, we may use an FPGA (DE2-115) to put it into practice.

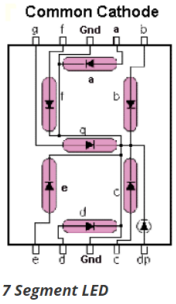


Figure 1. 7 Segment Display Decoder (Image Resource Link: www.thelearningpit.com)

Design: To design your seven-segment display decoder, you will first write the truth table specifying the output values for each input combination. We have started the truth table for you in Figure 2. For example, when the input is $D_{3:0}$ = 0000, all of the segments except g should be on. Because the outputs are active low, they must be $S_{g:a}$ = 1000000. Complete the truth table for the 7-segment display decoder circuit. You will need to turn in your completed truth table.

Hexadecimal Digit	Inputs				Outputs							(in hex)
	D_3	D_2	D_1	D_0	S_g	S_f	S_e	S_d	S_c	S_b	S_a	
0	0	0	0	0	1	0	0	0	0	0	0	40
1	0	0	0	1	1	1	1	1	0	0	1	79
2	0	0	1	0	0	1	0	0	1	0	0	24
3	0	0	1	1	0	1	1	0	0	0	0	30
4	0	1	0	0	0	0	1	1	0	0	1	19
5	0	1	0	1	0	0	1	0	0	1	0	12
6	0	1	1	0	0	0	0	0	0	1	0	02
7	0	1	1	1	1	1	1	1	0	0	0	78

8	1	0	0	0	0	0	0	0	0	0	0	00
9	1	0	0	1	0	0	1	1	0	0	0	18
A	1	0	1	0	0	0	0	1	0	0	0	08
B	1	0	1	1	0	0	0	0	0	1	1	03
C	1	1	0	0	1	0	0	0	1	1	0	46
D	1	1	0	1	0	1	0	0	0	0	1	21
E	1	1	1	0	0	0	0	0	1	1	0	06
F	1	1	1	1	0	0	0	1	1	1	0	0E/14

Figure 2. Truth table for 7-segment display decoder

ModelSim Simulation:

After you are done drawing your seven-segment decoder logic, your next step is to simulate and debug your design in ModelSim. Both input and output bus values can be set to be displayed in either Hexadecimal, Decimal or Binary. Do this by highlighting all of the inputs and outputs. Then right-click and choose **Radix/Binary/Decimal/Hexadecimal.** Choose hex to make your results easy to read. You can use the force command to drive multi-bit inputs.

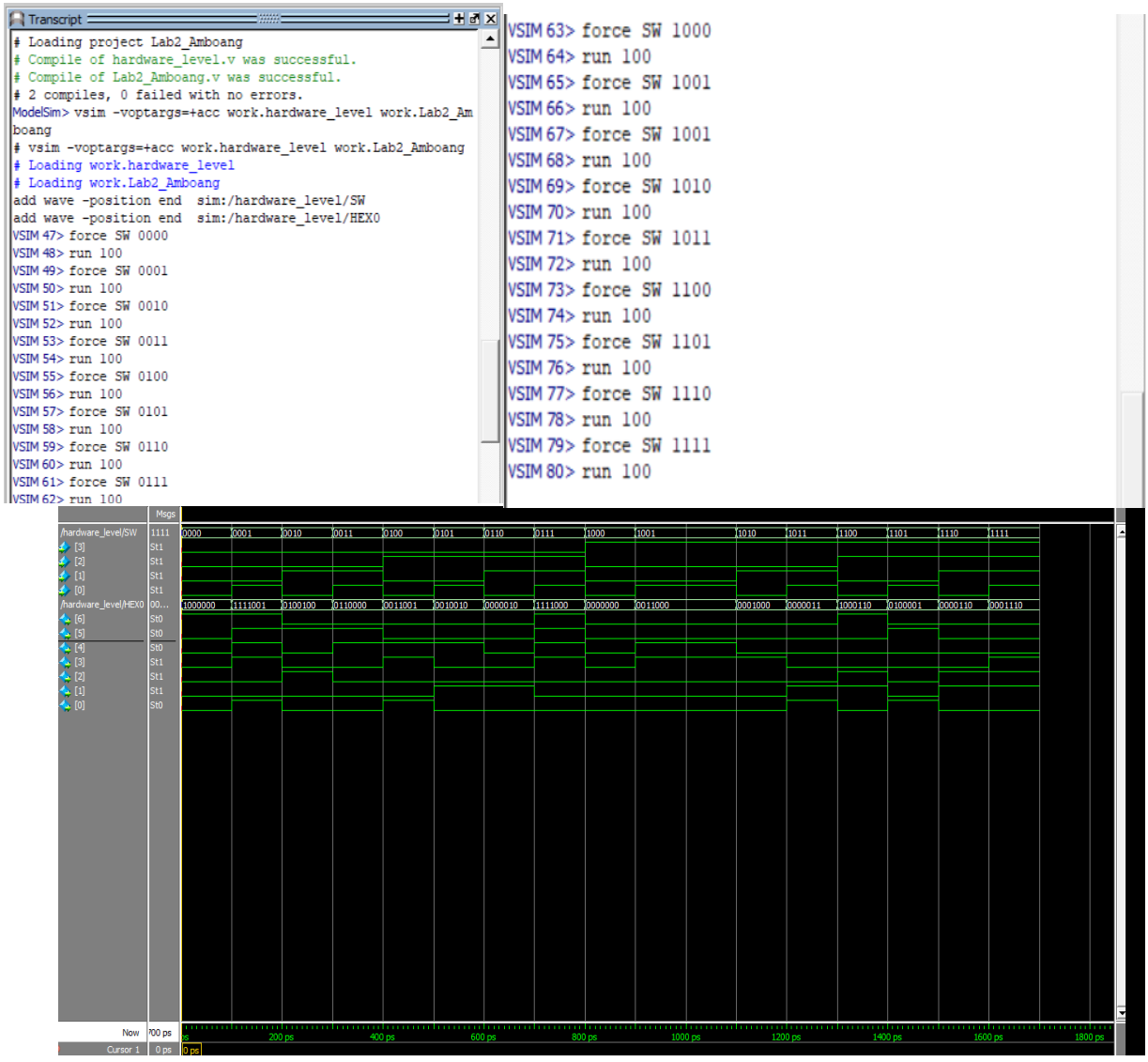


Figure 3. Simulating ModelSim in Binary Form

Hardware Implementation:

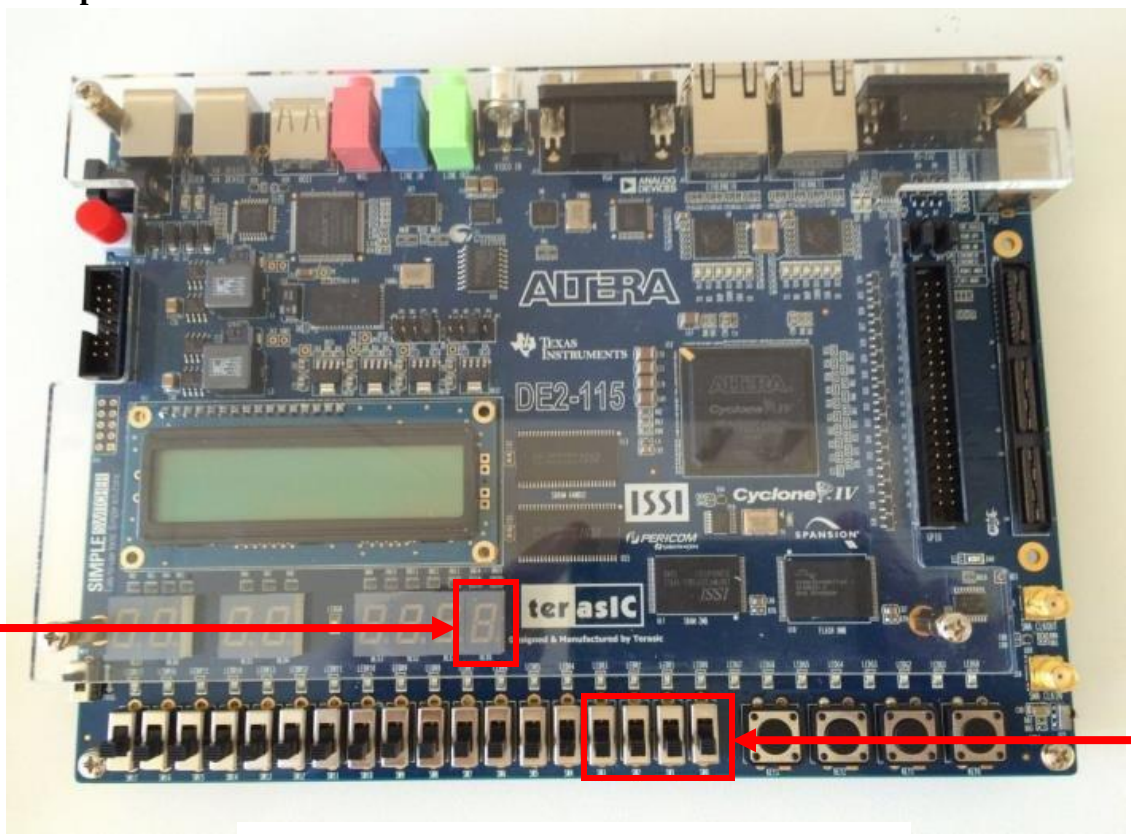


Fig .4 DE2-115 board (image courtesy Terasic

Pin assignment & Compilation

- You need to associate your design with the physical pins of the Cyclone IV FPGA on the DE1-115 board.
- Check that the device is corrected assigned as EP4CE115F29C7 using: Assignments > Device … Click: Processing > Start > Start Analysis and Elaboration. This will work out the input/output port names for your design. But there MUST not be errors, which will be shown in RED.)
- Grab your FPGA manual for your device's pin assignment (https://usermanual.wiki/Pdf/DE2115UserManual.833210098/html). The pin assignment should go through all of your switches and seven segment displays.

ALTERA

Table 4-1 Pin Assignments for Slide Switches

Signal Name	*FPGA Pin No.*	*Description*	*I/O Standard*
SW[0]	PIN_AB28	Slide Switch[0]	Depending on JP7
SW[1]	PIN_AC28	Slide Switch[1]	Depending on JP7
SW[2]	PIN_AC27	Slide Switch[2]	Depending on JP7
SW[3]	PIN_AD27	Slide Switch[3]	Depending on JP7

Table 4-4 Pin Assignments for 7-segment Displays

Signal Name	*FPGA Pin No.*	*Description*	*I/O Standard*
HEX0[0]	PIN_G18	Seven Segment Digit 0[0]	2.5V
HEX0[1]	PIN_F22	Seven Segment Digit 0[1]	2.5V
HEX0[2]	PIN_E17	Seven Segment Digit 0[2]	2.5V
HEX0[3]	PIN_L26	Seven Segment Digit 0[3]	Depending on JP7
HEX0[4]	PIN_L25	Seven Segment Digit 0[4]	Depending on JP7
HEX0[5]	PIN_J22	Seven Segment Digit 0[5]	Depending on JP7
HEX0[6]	PIN_H22	Seven Segment Digit 0[6]	Depending on JP7

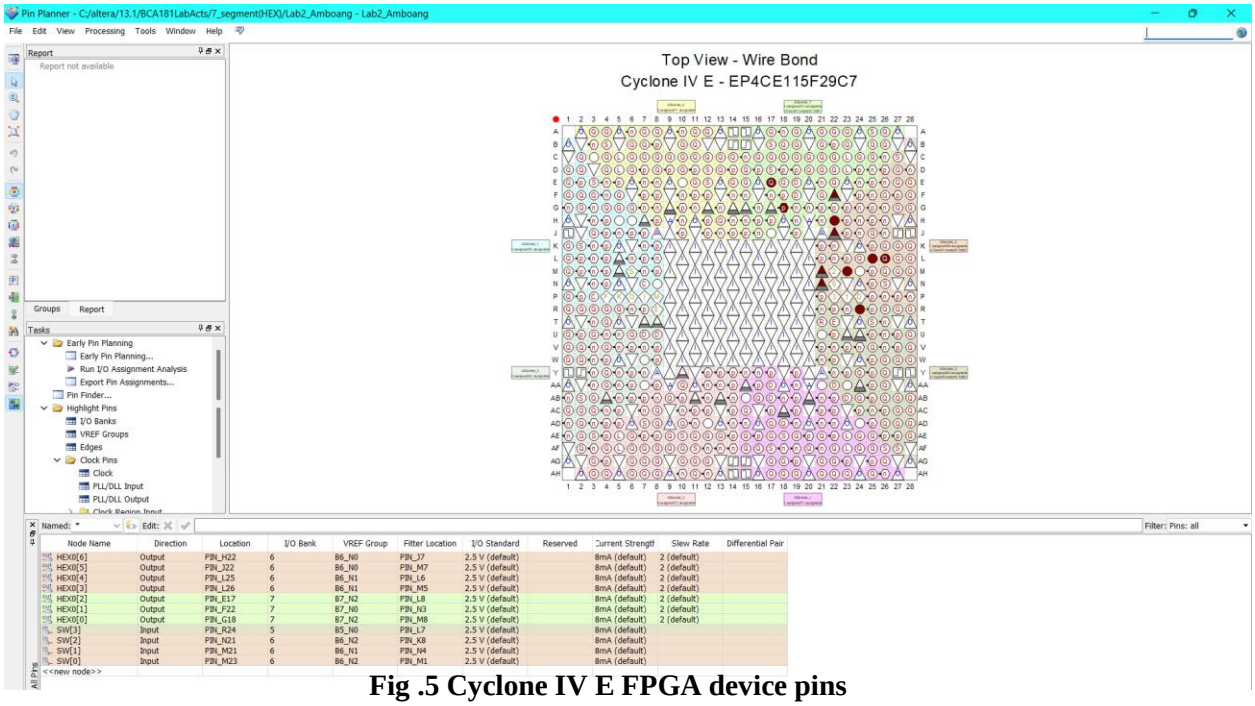
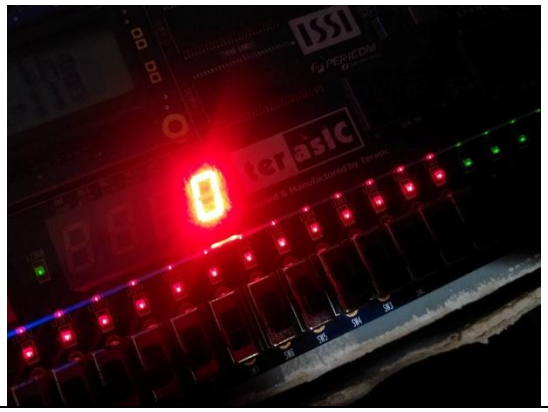
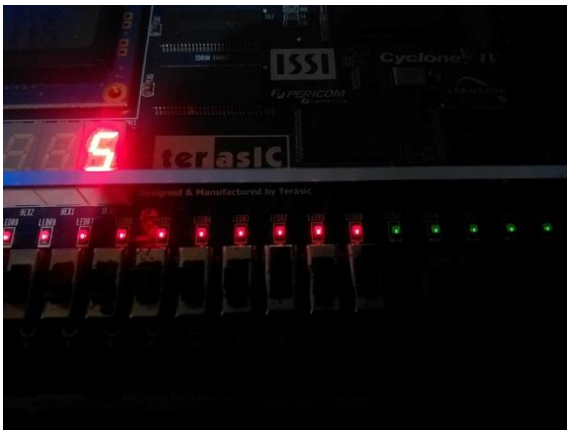


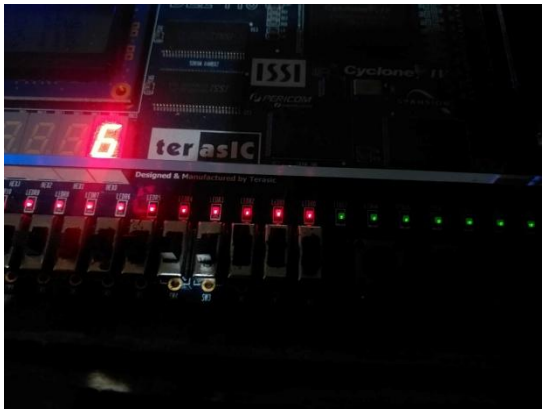
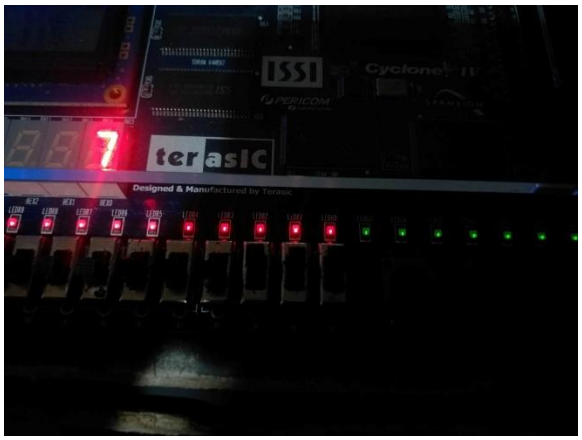
Fig .5 Cyclone IV E FPGA device pins

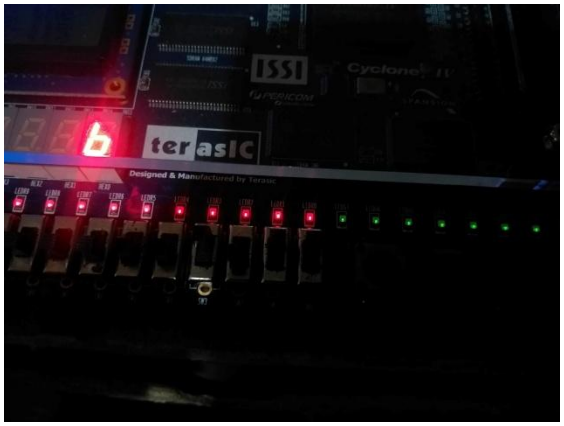
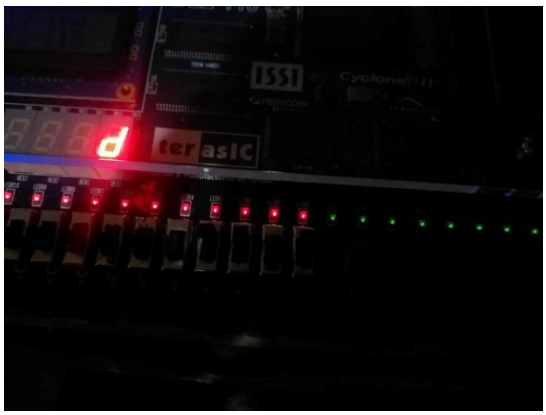
Output in Altera DES-115 FPGA

Download your design to the Altera DE2-115 FPGA board using the steps from Lab 1. Toggle switches 0-3 and check that the display labeled HEX0 cycles through the correct outputs. The Seven-Segment Displays unit and their output corresponding segment are represented in their Boolean expression as shown in the Figure 2., the information in the truth table has been provided "ON means 0" and "OFF means 1".

DECIMAL	BINARY	ACTUAL SIMUALTION
0	**0000**	
1	**0001**	

2	**0010**	
3	**0011**	
4	**0100**	
5	**0101**	

6	**0110**	
7	**0111**	
8	**1000**	
9	**1001**	

10 / A	**1010**	
11 / B	**1011**	
12 / C	**1100**	
13 / D	**1101**	

14/ E	**1110**	
15 / F	**1111**	

Conclusion:

As a result of this implementation, the input using switch and output of his HEX [0] using FPGA device "Altera DE2-115 Board". The solution is to understand the concept of binary to hexadecimal conversion, as shown in Figure 2. The BCD to Seven Segment converter requires a four bit binary input to display the equivalent decimal value on the seven-segment LED. This means that numbers from zero to fifteen cannot be displayed, so it is programmed to be off for those numbers. To create a BCD to 7-segment decoder, the first step is to create a truth table. Boolean expressions are formulated for each LED in the display. A program is then created in Quartus II to be embedded on the Altera De2 Board. In the Quartus II simulation, it was concluded that the schematic diagram the ideal simulation of logic circuits. The schematic diagram shows many delays in different parts of the signal output, caused by the number of logic gates used. "Altera DE2-115 Board" is relevant to many applications in today's market, so you can freely apply Verilog files, use switches to simulate various counter designs and designs on FPGA devices, by implementing, and you gain implementation knowledge in the 7 segment display.