

Digital Design Lab
ECE 315

Lab/Project #2

Group #7

Karysse Hay, Nikeem

Dunkelly-Allen and Dibyanshi Mishra

Zeinab Ramezani, TA

University of Miami

2/21/2022

Overview

A seven- segment display is a form of electronic display device for displaying decimal numerals. These are widely used in digital clocks, electronic meters, basic calculators, and other electronic devices that display numerical information. The aim of this lab is to implement a circuit that will display the decimal numbers (0 to 9) on a 7 segment display from a 4- bit binary input on a DE1 board. It displays integers by turning on and off the desired LEDS out of the seven it has. The DE1 board requires a low-level voltage to turn on and a Vcc to turn off the LEDs. The objective of this lab was to determine and implement Boolean functions for the display circuit on Quartus Prime (circuit simulator used). After the circuit was simulated, it was downloaded to the DE1 board.

Equipment

- Quartus prime Software
- DE1-SoC board
- ModelSim
- AND 3 gates
- OR 3 gates
- OR 2 gates
- Input pins
- Output pins
- DE1 SoC Data Sheet

Description

Firstly, the 7 Segment Display Truth Table was used to create 7 different K-Maps for each one of the outputs which corresponded to the seven segments ($D_0, D_1, D_2, D_3, D_4, D_5, D_6$ and D_7). This truth table had 4 inputs and 7 outputs as stated above. From this, the boolean equations were found and recorded. A minimal circuit was designed and was then checked to ensure that was in fact the minimum.

On Quartus Prime, a new project was created and New Project Wizard was selected. Using a personal USB, the location and name of the file was chosen. After an empty project was chosen and the flowing wizard was left blank, the DE1 SoC board was selected. Under simulation, the Tool Name was changed to simulations and the format was changed to VHDL. In the Project Navigator Window, a new Block Diagram/ Schematic File was opened.

When the project opened, inputs, outputs, AND gates and OR gates were added according to the boolean equations obtained. The inputs and outputs were labeled correctly. After all components were connected and labelled, the circuit was compiled. If no errors were found, the simulation was performed.

A waveform vector file was created and the inputs and outputs: University Program, VWF. The node finder was clocked and all input and output pins were added to the selected nodes window. A new window popped up which consisted of all the pins. In the VWF file, under simulation, The Quartus I simulator option was selected. The End Time was set for each one of the I/Os. Since there were four inputs, 16.0 microseconds was set as the End Time ($2^4=16$) and 1.0 microsecond was set as the Grid Size. The most significant bit (X_3) was selected and the value was edited to overwrite the clock. In this window, the value was set to 16.0 microseconds with a 50% duty cycle. The same was done for the remaining 3 input settings (X_2, X_1 and X_0) with the respective period times: 8.0 microseconds, 4.0 microseconds and 2.0 microseconds. The inputs were then Grouped after highlighting the inputs then selecting Edit then Grouping then Group then naming them as Inputs and keeping the Radix set to Binary. The simulation was ran by clicking ‘Run Functional Simulation’. When the design looked complete and functional, it was uploaded to the board.

In order to use the switches and LEDs on the board, the inputs and outputs had to be associated to the switches and LEDs on the board. To do this, pins had to be assigned. In the Quartus Primes window and the Assignments tab, the Pin Planner was selected and a new window popped up. The DE1-SoC Data Sheet was used to specify the ‘Location’ column for each row. The input pins: X₃, X₂, X₁ and X₀ were assigned the pins PIN_AF10, PIN_AF9, PIN_AC12 and PIN_AB12 respectively. The design was compiled and the DE1-SoC was turned on. After the compilation finishes, Program Device was opened and the DE1-SoC board was selected as the Hardware Output. This window was closed and Auto-Detect was selected and left as default. Finally, the 5CSEBA5 device was selected and the correct file was uploaded to the board. The Start button was clicked and the design was tested using the toggle switches on the board.

Design Synthesis

Truth Table

	<i>Input</i> X ₃ X ₂ X ₁ X ₀	Display D ₀ D ₁ D ₂ D ₃ D ₄ D ₅ D ₆
0	0 0 0 0	0 0 0 0 0 0 1
1	0 0 0 1	1 0 0 1 1 1 1
2	0 0 1 0	0 0 1 0 0 1 0
3	0 0 1 1	0 0 0 0 1 1 0
4	0 1 0 0	1 0 0 1 1 0 0
5	0 1 0 1	0 1 0 0 1 0 0
6	0 1 1 0	0 1 0 0 0 0 0
7	0 1 1 1	0 0 0 1 1 1 1
8	1 0 0 0	0 0 0 0 0 0 0
9	1 0 0 1	0 0 0 0 1 0 0
10	1 0 1 0	X X X X X X X X
11	1 0 1 1	X X X X X X X X
12	1 1 0 0	X X X X X X X X
13	1 1 0 1	X X X X X X X X
14	1 1 1 0	X X X X X X X X
15	1 1 1 1	X X X X X X X X

K-Maps

Segment D0	X3, X2 00	01	11	10
X1, X0 00	0	1	-	0
01	1	0	-	0
11	0	0	-	-
10	0	0	-	-

Segment D1	X3, X2 00	01	11	10
X1, X0 00	0	0	-	0
01	0	1	-	0
11	0	0	-	-
10	0	1	-	-

Segment D2	X3, X2 00	01	11	10
X1, X0 00	0	0	-	0
01	0	0	-	0
11	0	0	-	-
10	1	0	-	-

Segment D3	X3, X2 00	01	11	10
X1, X0 00	0	1	-	0
01	1	0	-	0
11	0	1	-	-
10	0	0	-	-

Segment D4	X3, X2 00	01	11	10
X1, X0 00	0	1	-	0
01	1	1	-	0
11	1	1	-	-
10	0	0	-	-

Segment D5	X3, X2 00	01	11	10
X1, X0 00	0	0	-	0
01	1	0	-	0
11	1	1	-	-
10	1	0	-	-

Segment D6	X3, X2 00	01	11	10
X1, X0 00	1	1	-	0
01	1	0	-	0
11	0	1	-	-
10	0	0	-	-

Boolean Equations

$$D0: X_3'X_2X_1'X_0' + X_3'X_2'X_1'X_0$$

$$D1: X_3'X_2X_1'X_0' + X_3'X_2X_1X_0'$$

$$D2: X_3'X_2'X_1X_0'$$

$$D3: X_3'X_2X_1'X_0' + X_3'X_2'X_1'X_0 + X_3'X_2X_1X_0$$

$$D4: X_3'X_2X_1' + X_2'X_1'X_0 + X_3'X_0$$

$$D5: X_3'X_1X_0 + X_3'X_2'X_0 + X_3'X_2'X_1$$

$$D6: X_3'X_2'X_1' + X_3'X_2X_1X_0$$

Steps:

In order to start a truth table must be made, this table can be obtained by using the X3, X2, X1, and X0 inputs from 0000 to 1111 in binary. The numbers needed on the 7 segment display are 0-9 which is 0000 to 1001 in binary, 10 to 15 will not be needed. The outputs D0 to D6 are determined by looking at the 7 segment display diagram. A 0 is used if the segment is needed to be active, a 1 turns the segment off (e.g. 1 requires segments D1 and D2 to be on). Each D segment is represented by its appropriate K-Map, from there simplification methods are used to obtain the minimum expressions for each segment

Complete Logic Diagram

Block Diagram

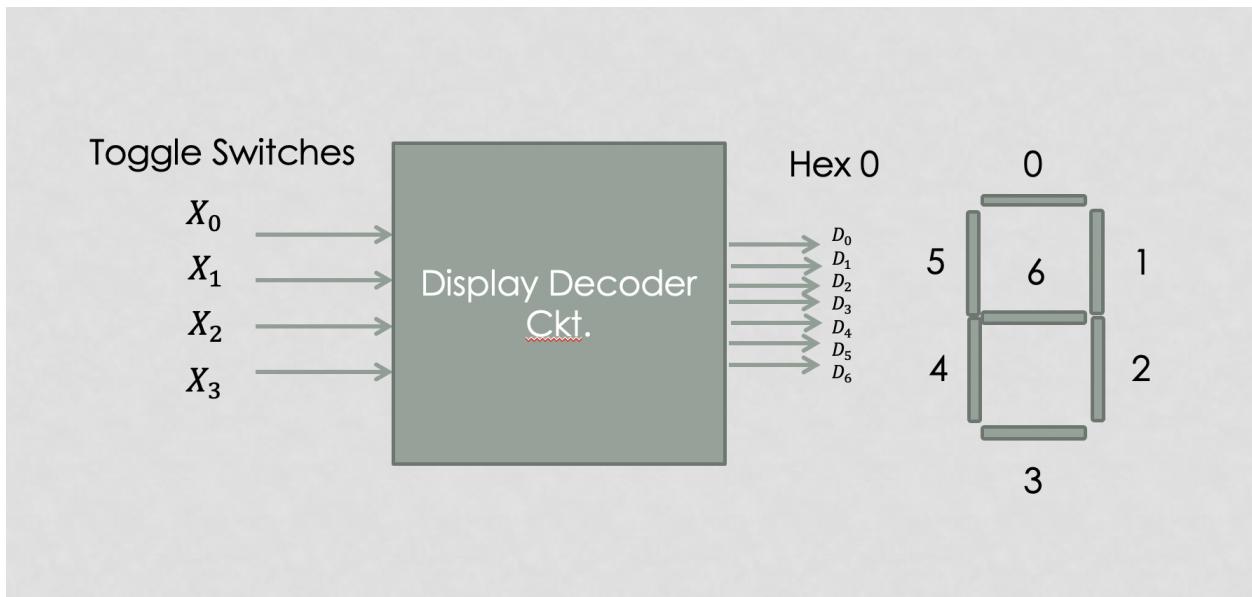


Figure 1: Block Diagram for Circuit

Logic Diagram

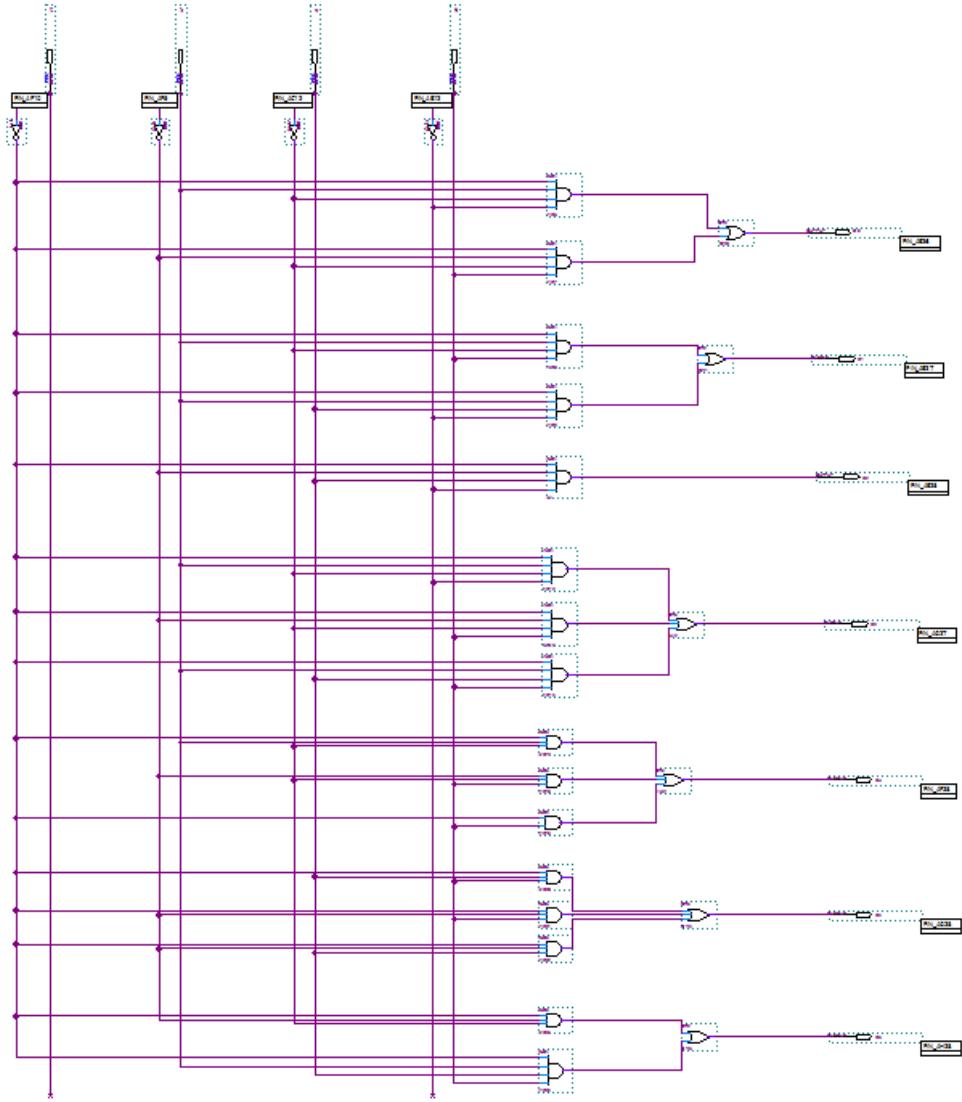


Figure 2: Complete Logic Diagram on Quartus Prime

Results and Simulations

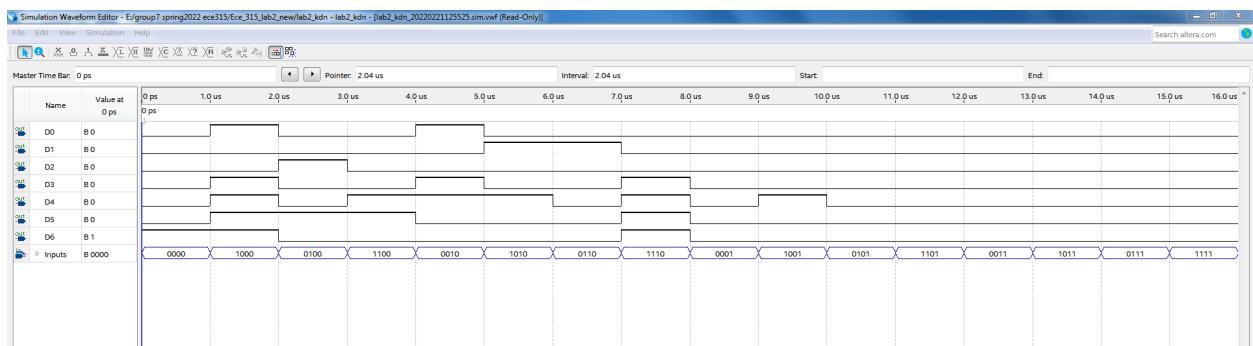


Figure 3: Seven Segment Display Quartus Prime Simulation Results

The seven segment display was implemented through a circuit that displays decimal numbers 0 to 9 on the 7 segment display from 4 bit binary input. The boolean functions and the circuit were implemented on Quartus prime, and then further simulated through simulation waveform. Figure 1 shows the Quartus simulator results where each pin was added and assigned the value '1' or '0'. The stimulation was further downloaded to the DE1 board. The DE1 has 6 seven segment displays with toggle switches that displayed the decimal numbers.

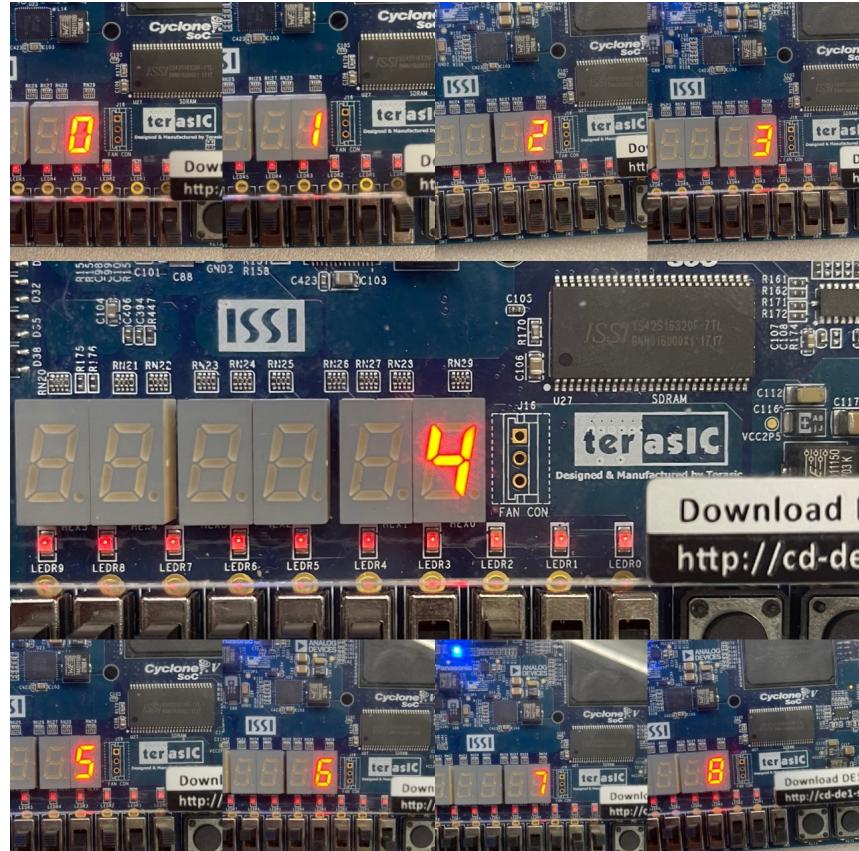


Figure 4: Photo of Decimal numbers on DE1-SoC Board

This photo illustrates every decimal number on the seven-segment display. They are in logical order for easy reference. These numbers were achieved by toggling the switches according to the binary conversion of the decimal numbers. For instance, in the larger photo, the '4' was achieved by assigning LEDR2 as a high input and both LEDR1 and LEDR0 as low inputs. This reads as '100' in binary which is '4' in decimal.

Conclusion

The 7 Segment Display was created by designing a circuit that would display the integers by simply turning on or off the desired LEDs. In order to do so, a truth table and 7 different K-maps were made which were used to derive the boolean equations. These boolean equations were used to design a circuit that was created on the Quartus Prime software. The circuit was designed using the input, output, AND, and OR gates based on these boolean equations. The circuit was

then further simulated and downloaded to the DE1 board where it displayed the decimal numbers. Nevertheless, we encountered some problems during the design process. When we first designed the circuit on Quartus Prime, our design did not work and kept on showing error, stating that design supplied two input values for one output value. Unable to detect the error in circuit, we created a new project and recreated the design based on the K-maps. Fortunately, the circuit worked the second time and stimulated perfectly. There were no limitations or flaws in the design, the design was able to successfully display decimal number 0 to 9 on the DE1 board. We learned many new skills through this lab. We were introduced to the DE1 board and Quartus Prime, and learned how to create circuits on a software as opposed to designing circuits on breadboards.