Lab Assignment 2
Controlling the 7-Segment Displays with Digital Logic

Zhenming Yang, Connor McEleney

yang.zhenm@northeastern.edu, mceleney.c@northeastern.edu

Instructor: Prof. Zagieboylo

d.zagieboylo@northeastern.edu

Due Date: September 28th, 2023

Submit Date: September 27th, 2023

Abstract

In this lab a 7-Segmen Display controller was constructed using the Quartus Prime Schematic tool to simulate the behavior of the circuit and upload it to the DE1-SoC to test the circuit.

Introduction

The FPGA circuit can activate individual segments in the 7-segment displays by using a logic 0 signal (active low). To deactivate them, a logic 1 signal is applied. This behavior is a result of the 7-segment displays featuring a common anode configuration, where all the positive sides (anodes) of the LED segments are connected electrically, while each LED's negative side (cathode) has its dedicated pin [1].

In this lab, we will create a digital circuit exclusively for displaying digits from 0 to 9 using only digital logic components. This circuit, known as a Binary Coded Decimal (BCD) decoder, accepts a 4-bit input that represents the binary counterpart of a digit ranging from 0 to 9. It then activates the appropriate segments to showcase the corresponding digit on the display.

Lab Setup

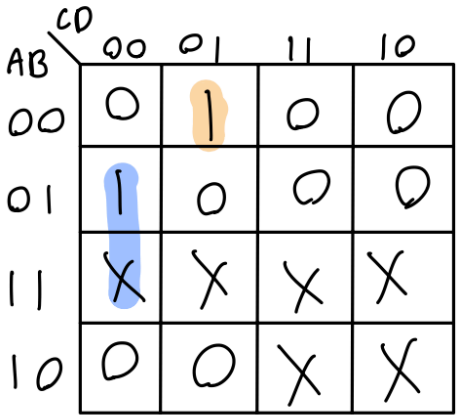
Pre-Lab

The truth table and K-maps for this experiment are attached below.

Decimal #	A	B	BCD Inputs C	D	7-Segment Outputs						
	i[3]	i[2]	i[1]	i[0]	a	b	c	d	e	f	g
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

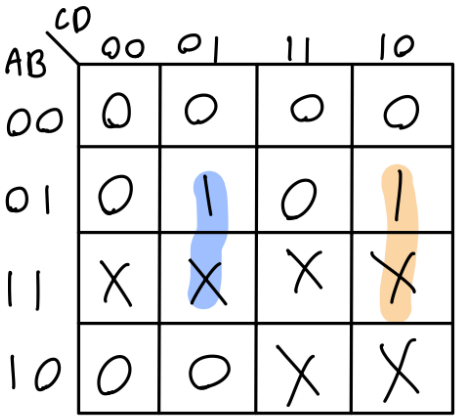
Table 1: Digits to display on the 7-segment displays

for a



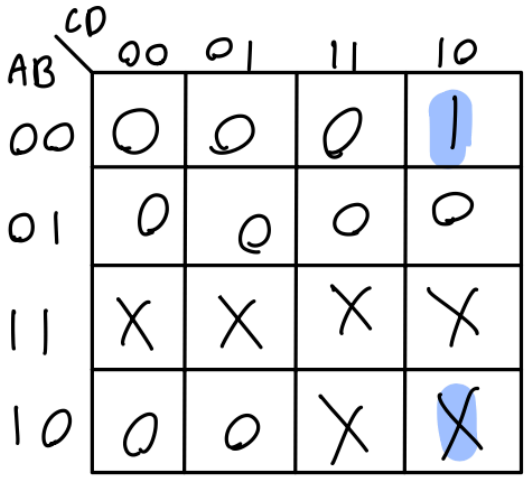
$B\bar{C}\bar{D} + \bar{A}BC\bar{D}$

for b



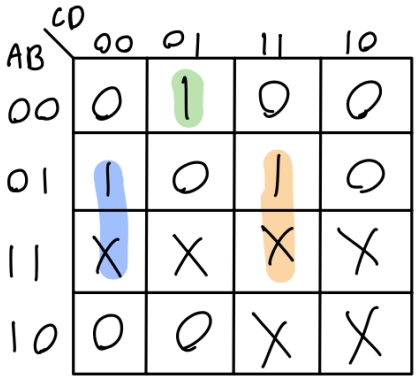
$B\bar{C}D + BC\bar{D}$

for c



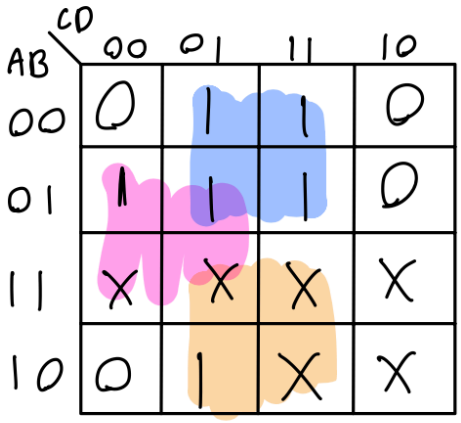
$\bar{B}C\bar{D}$

for d



$B\bar{C}\bar{D} + BCD + \overline{A}BCD$

for e

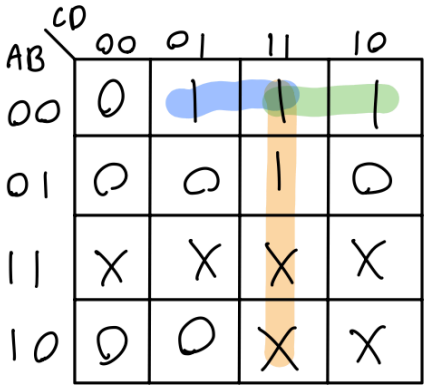


$\bar{A}D + AD$

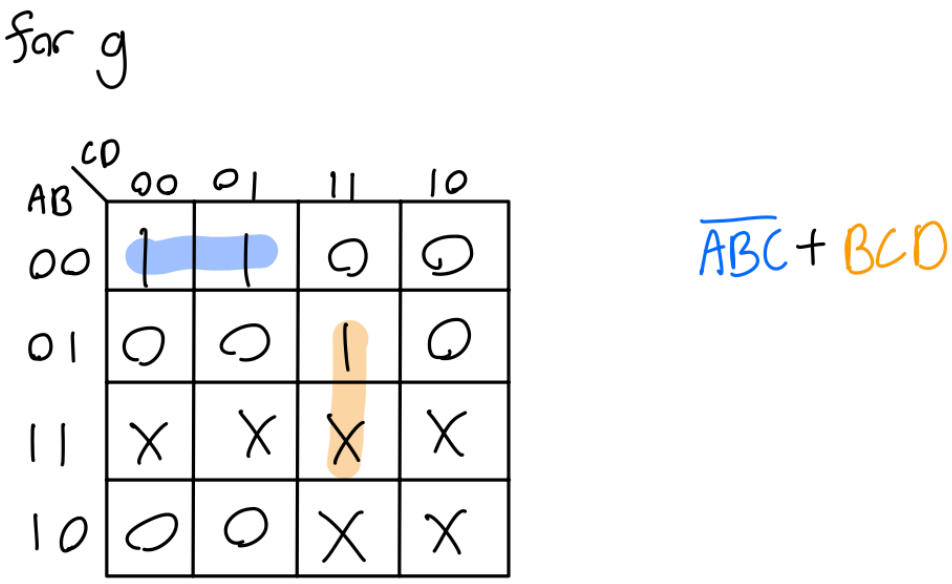
$D(\bar{A} + A)$

$D + B\bar{C}$

for f



$\bar{A}\bar{B}D + CD + \bar{A}\bar{B}C$



Equipment

DE1-SoC:

- The DE1-SoC is a hardware design platform built around the Altera System-on-Chip (SoC) FPGA. The DE1-SoC is designed for experiments on computer organization and embedded systems. It includes embedded processors, memory, audio and video devices, and some simple I/O peripherals.

Results and Analysis

Results

Part 1: BCD Circuit

Based on the truth table and the k-map we created for the pre-lab, we created a most optimized logic to control the 7-segment display using the switches. We then **put our logic into our schematic using the "NOT", "AND," and "OR" gates** (Figure 1). We can see here that each output (a-g), they all have their corresponding logic that we got from our individual k-map generated in pre-lab.

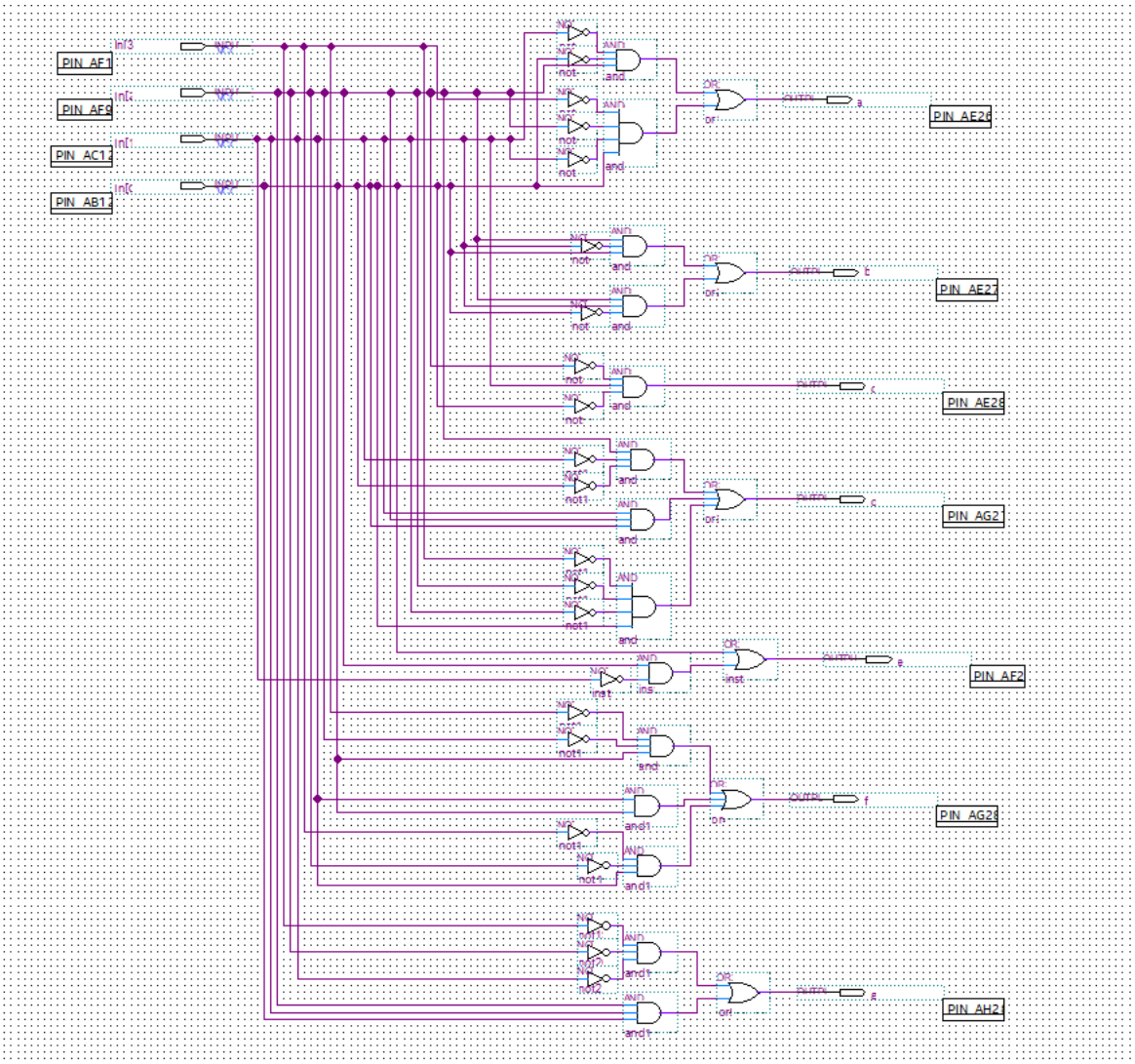


Figure 1: The schematic of the 7-segment display controller based on our optimized logic generated from the k maps.

We then run Functional Simulation and verify the correct and expected output for inputs representing several decimal digits in Waveform (Figure 2). **We generated numbers from 1 to 9 from the switches in binary (0000 to 1001)**. The result corresponds to the truth table we generated for the pre-lab.

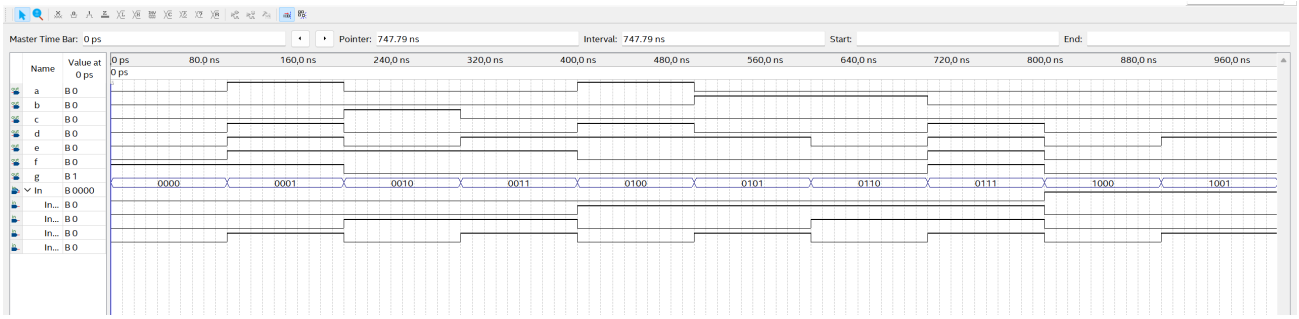


Figure 2: The Waveform simulation of the 7-segment display controller schematics. The binary number represented with the switches are displayed on the "in" row.

We then assigned the inputs and outputs to the correct pins on the DE1-SoC board. After checking all the pins are correct, we upload our design into the DE1-SoC board and test it on the first 7-segment display (HEX0) (Figure 3). **We can see the switches represent the binary number of the decimal number displayed on the 7-segment display**.

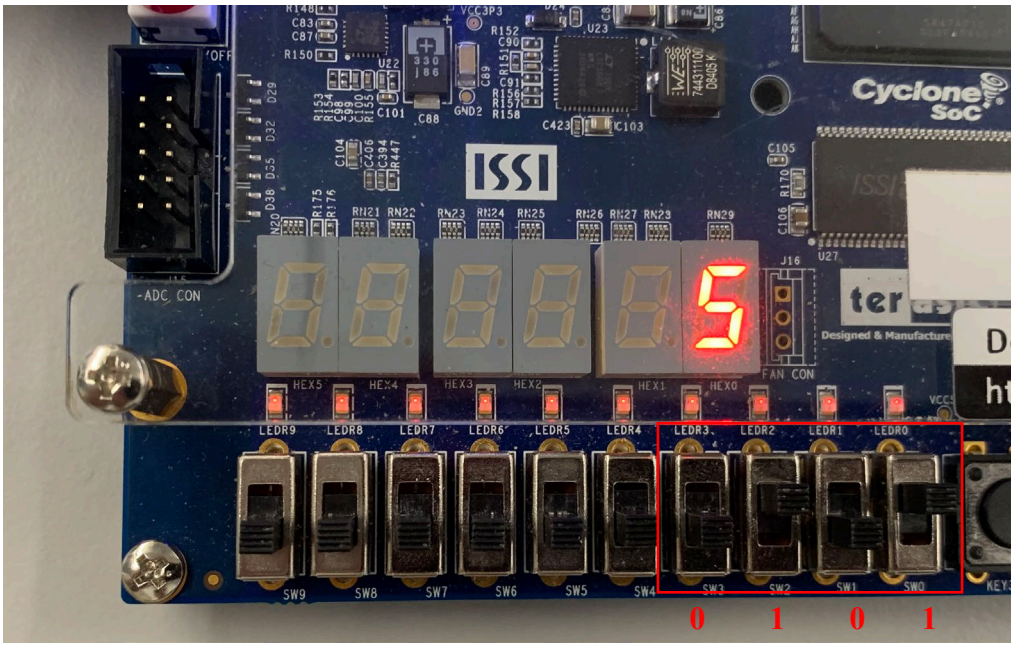


Figure 3: The 7-segment Display correctly showed the corresponding decimal number (5) from the binary number (0101) represented using the switches.

We tried a few different binary numbers and they all showed up fine on the 7-segment display (Figure 4).

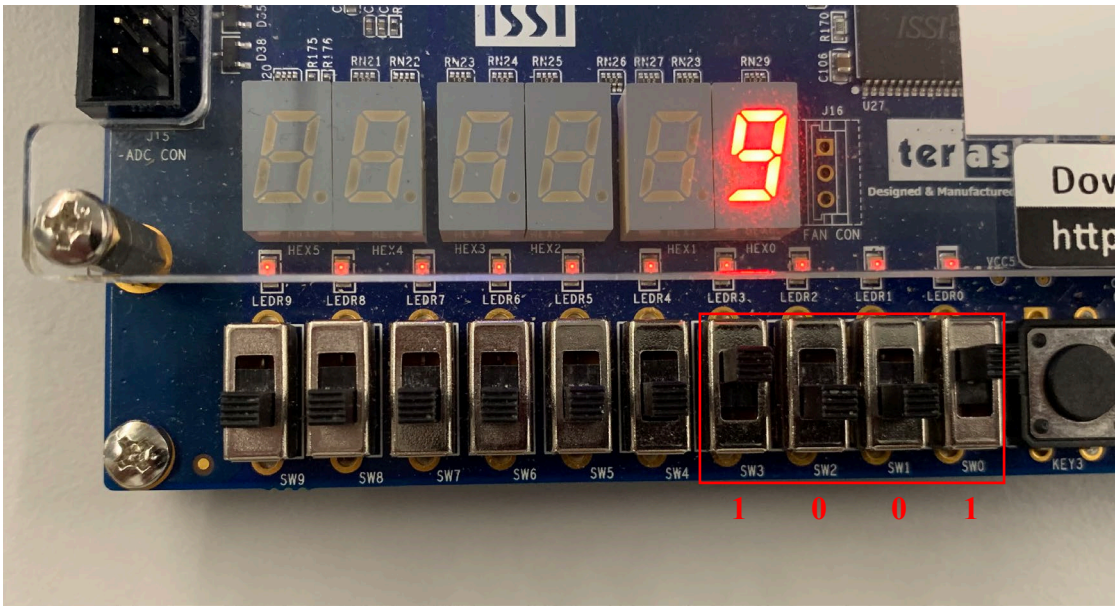


Figure 4: The display changes number based on the binary number we inputted using the switches.

We finished up by bundling the control into a bsf file and saved it for Part 2.

Part 2: Enabling the 7-Segment Display

To integrate a new button that enables the screen display, we first **created a truth table** for just one segment of the screen (since the other 6 are the same). The screen segment only lights up (when it is 0) when the enable switch is on and the output from the controller is 0.

Enable Switch	Output from our controller		Screen segment
0	0		1
0	1		1
1	1		1
1	0		0

We can see this is just **a "NAND" gate with "NOT" output from the controller**. Based on this intuition, we created our schematic (Figure 5).

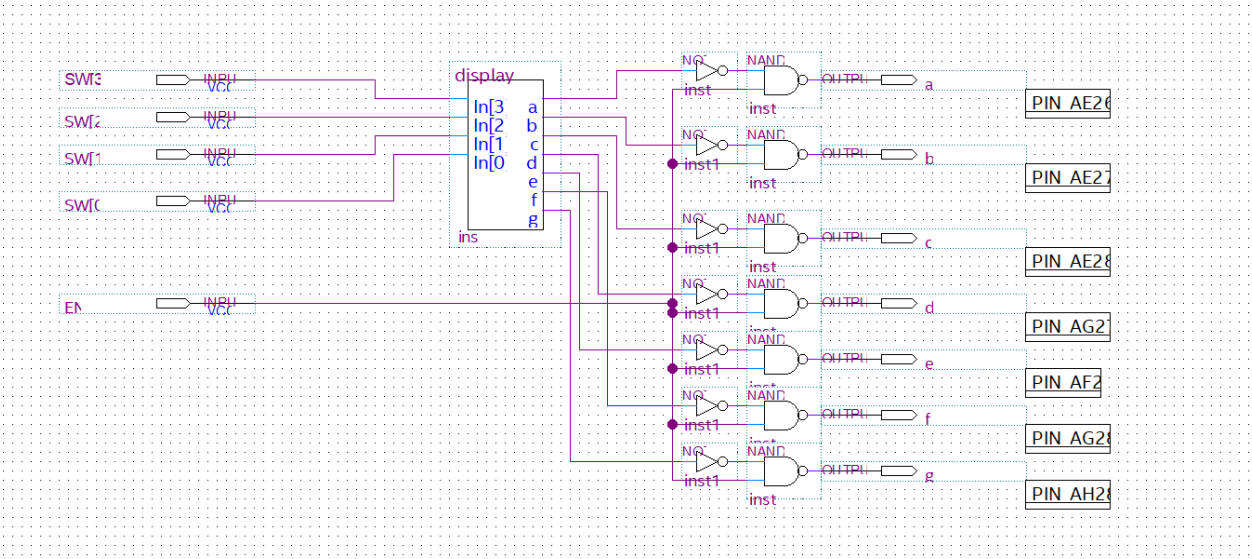


Figure 5: Schematic of the enable 7-segment display controlled using 5 switches.

We then uploaded our new design on the board with their corresponding pins (only 1 more switch is added) and tested our new logic with the enabling pin. You can see in Figure 6: **the display is on and displays the correct number from the switches 1-4 when switch 5 is on**. And when switch 5 is off, the display is off (Figure 7).

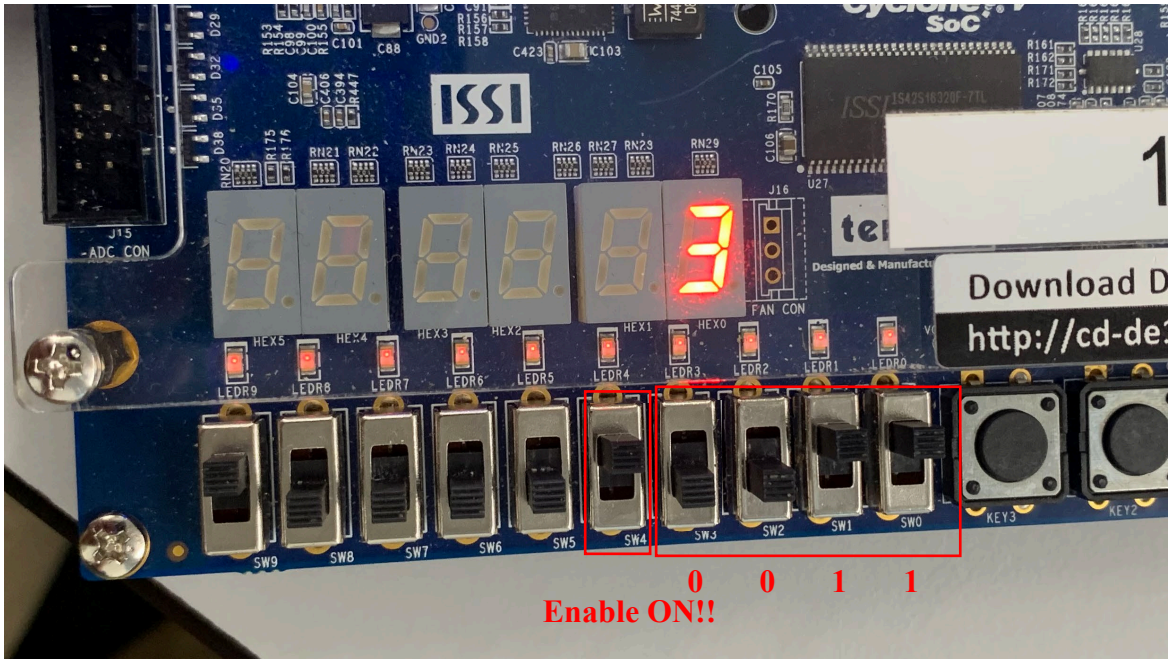


Figure 6: The display shows the correct number when the enable switch is on.

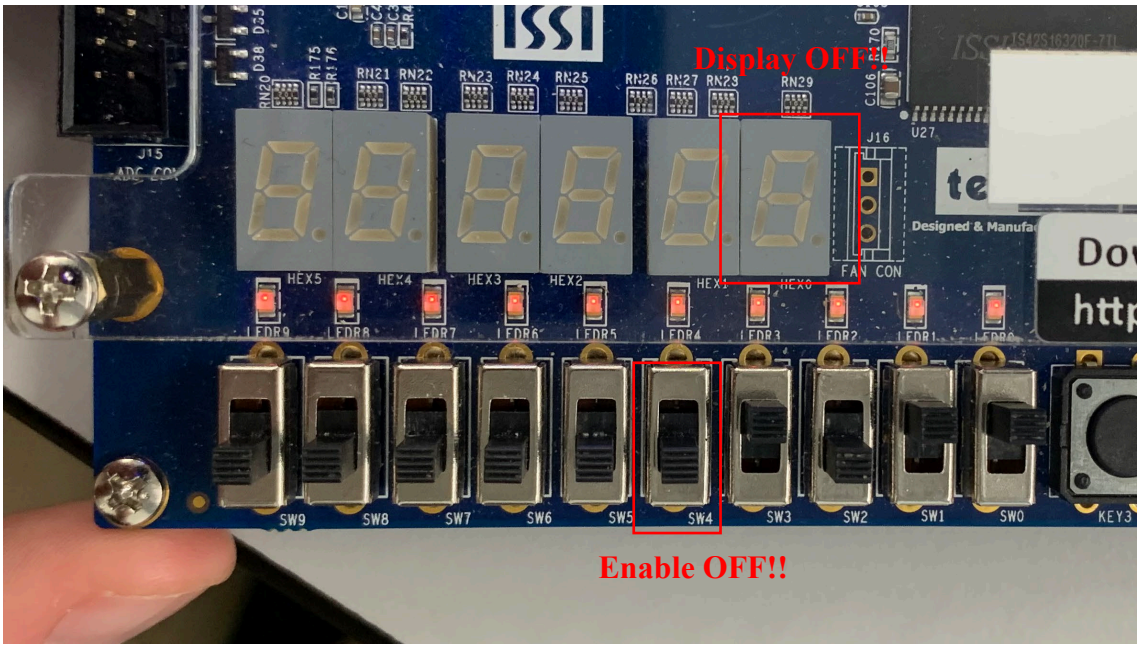


Figure 7: The display turns off when the enable switch is off.

Analysis

The result of Part 1 corresponds to the pre-lab tables and k-maps for the better algorithm. And the simulation and board result reflect our k-map and truth table to be correct. The enable bit is also working as it switches the entire display off when it is off, and it turns back on when it is on. The only trouble we had was figuring out the logic for the control since the board takes 0 as 1 and 1 as 0 in common sense (0 is turn on the part of the display and 1 is turning it off) but we figured it out in our pre-lab quickly.

Conclusion

This experiment demonstrated the principles of circuit optimization using K-maps and how to assemble the circuits given the optimized Boolean formula. Each segment of the seven-segment display was assigned an output bit that would default to on at 0 and off at 1. The desired states were calculated, and each outputs truth table and K-map filled out to find the optimal formula. The compiling and compacting of circuit elements to simplify complex design was also demonstrated. The circuit that controls the number displayed was compacted into a block to be used to construct a circuit that could also turn the entire system on and off. The lab was a good demonstration of integrating two different control circuits together, in this case the binary representation of switched to display its represented number and the control circuit that turns the entire system off/on. It is implied that the on/off system can be scaled up with various other control systems connected to a single on/off control line.

References

[1] DE1-SoC User Manual