C4C Asher Speicher Lab 2 Report

3 March 2025| C4C Asher Speicher | USAF Academy, CO

Abstract: *The main goal of the 7-segment display lab was to input a 4-bit binary number using switches and display the number as a hex number on the fourth of the 7 segment displays. The way I accomplished this task was by using a decoder to code each of the 16 possible inputs into my desired output. My desired output would then set certain anodes to low, which would cause certain segments on the display to light up, thus causing the lights to display the hex number associated with the 4-bit binary number. It is important to note that to view the one particular digit in the output, I needed to press a button. However, when buttons are pressed, they go from low to high, and the light is an active low, meaning the button input must have been inverted. Throughout this lab report I am going to describe more in depth how the lab worked, how I solved the problems presented towards me, and go into depth about the components I used.*

Introduction

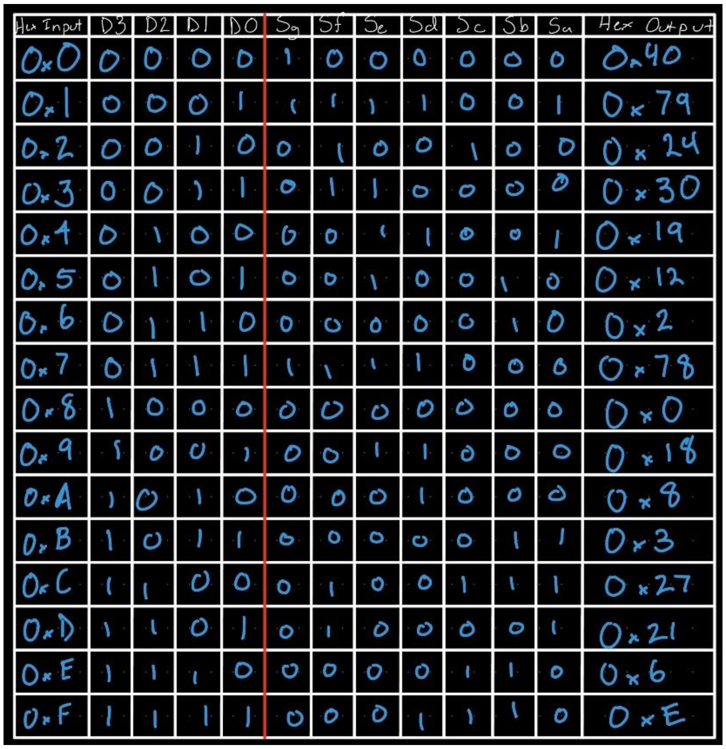
In this section I will describe how the circuitry of the Basys3 board works. The 3 different sections of the Basys3 board I used were the switches, the 4 digit 7-segment display, and button C. The switches are rather straightforward, and I only uncommented and used four of them, switches 0-3. I set switch 0 to be my least significant bit (LSB) and switch 3 to be my most significant bit (MSB). In other words, when I entered a binary number, say 1010, switch 3 was high, switch 2 was low, switch 1 was high and switch 0 was low. With button C, when button C is not pressed it passes a low or a '0'. When button C is pressed, it passes a high or '1'. This is an issue because what I wanted to pass it to, the 7-segment display, is an active low. What an active low means is that it produces the desired output when a '0' is inputted and does not produce the output when a '1' is inputted. This could be compared to an LED. Let's say the desired output for the LED is to have it light up. If it is an active high LED, then passing a '1' would cause it to light up, and passing a '0' would cause it to stay dark. However, if it was an active low, like the 7-segment display, then passing a '1' would cause it to stay dark, and passing a '0' would cause it to light up. The way the 7-segment display is set up is by having a common anode which feeds into each of the 7 different segments, and those 7 segments feed into their own cathode. These cathodes are shared by all 4 of the 7-segment displays, while each 7-segment display has its own common anode (labeled anode 0-3). Due to the fact that these are active low, I can model it with a transistor. When a high signal is passed to the anode, it causes the transistor to open, not allowing current to flow to the cathode, meaning when there is no current, the segment does not light up. However, when a low signal is passed through the anode, it allows whichever segments are activated to light up. This is the general concept behind the active low. So, in order to only have my 4th 7-segment display to light up, I set anodes 1,2, and 3 to high, and set my 0 anode to the inverted of my button input to activate it when the button is pressed. In other words, once the button is pressed it passed a high input, which is inverted to low, which is fed into anode '0' which activates it. When the button is not being pressed, it is passing a '0' which is inverted to a '1' to the anode, keeping it deactivated.

Methodology

In this lab, I primarily followed the instructions on the lab, however I deviated in the creation of my test bench. I created truth tables and k-maps in the prelab, then created files in the manner directed. After studying the textbook and the lab itself I created my required entities and components in VHDL and following that simulated a test bench. Finding my values to be correct, I uploaded my code to my Basys3 board and ran the program.

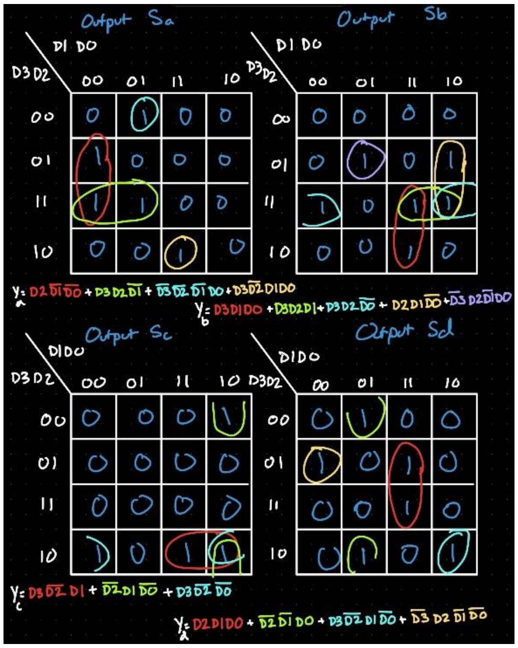
Design

For the design, I started in the pre-lab and progressed onwards into my actual VHDL code and finally the physical Basys3 board. Starting with the pre-lab, I completed the pre-lab before the document was updated. What that means is I created an active-low truth table with segment g as my MSB and segment a as my LSB (see figure 1). Additionally, I completed k-maps for each segment and came up with the simplified Boolean algebra equation (see figure 2 and 3).



Hex Input	D3	D2	D1	D0	Sg	Sf	Se	Sd	Sc	Sb	Sa	Hex Output
0x0	0	0	0	0	1	0	0	0	0	0	0	0x40
0x1	0	0	0	1	1	1	1	1	0	0	1	0x79
0x2	0	0	1	0	0	1	0	0	1	0	0	0x24
0x3	0	0	1	1	0	1	1	0	0	0	0	0x30
0x4	0	1	0	0	0	0	1	1	0	0	1	0x19
0x5	0	1	0	1	0	0	1	0	0	1	0	0x12
0x6	0	1	1	0	0	0	0	0	0	1	0	0x2
0x7	0	1	1	1	1	1	1	1	0	0	0	0x78
0x8	1	0	0	0	0	0	0	0	0	0	0	0x0
0x9	1	0	0	1	0	0	1	1	0	0	0	0x18
0xA	1	0	1	0	0	0	0	1	0	0	0	0x8
0xB	1	0	1	1	0	0	0	0	0	1	1	0x3
0xC	1	1	0	0	0	1	0	0	1	1	1	0x27
0xD	1	1	0	1	0	1	0	0	0	0	1	0x21
0xE	1	1	1	0	0	0	0	0	1	1	0	0x6
0xF	1	1	1	1	0	0	0	1	1	1	0	0xE

Figure 1



Output Sa

D3D2 \ D1 D0	00	01	11	10
00	0	1	0	0
01	1	0	0	0
11	1	1	0	0
10	0	0	1	0

Output Sb

D3D2 \ D1 D0	00	01	11	10
00	0	0	0	0
01	0	1	0	1
11	1	0	1	1
10	0	0	1	0

$y_a = D2\overline{D1}\,\overline{D0} + D3D2\overline{D1} + \overline{D3}\,\overline{D2}\,\overline{D1}D0 + D3\overline{D2}D1D0$

$y_b = D3D1D0 + D3D2D1 + D3D2\overline{D0} + D2D1\overline{D0} + \overline{D3}D2\overline{D1}D0$

Output Sc

D3D2 \ D1D0	00	01	11	10
00	0	0	0	1
01	0	0	0	0
11	0	0	0	0
10	1	0	1	1

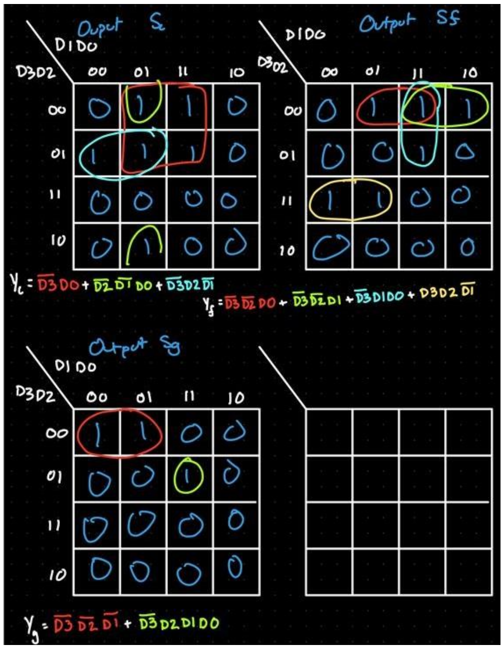
Output Sd

D3D2 \ D1D0	00	01	11	10
00	0	1	0	0
01	1	0	1	0
11	0	0	1	0
10	0	1	0	1

$y_c = D3\overline{D2}D1 + \overline{D2}D1\overline{D0} + D3\overline{D2}\,\overline{D0}$

$y_d = D2D1D0 + \overline{D2}\,\overline{D1}D0 + D3\overline{D2}D1\overline{D0} + \overline{D3}D2\overline{D1}\,\overline{D0}$

Figure 2



Output Se

D3D2 \ D1D0	00	01	11	10
00	0	1	1	0
01	1	1	1	0
11	0	0	0	0
10	0	1	0	0

Output Sf

D3D2 \ D1D0	00	01	11	10
00	0	1	1	1
01	0	0	1	0
11	1	1	0	0
10	0	0	0	0

$Y_e = \overline{D3}D0 + \overline{D2}\,\overline{D1}D0 + \overline{D3}D2\overline{D1}$

$Y_f = \overline{D3}\,\overline{D2}D0 + \overline{D3}\,\overline{D2}D1 + \overline{D3}D1D0 + D3D2\overline{D1}$

Output Sg

D3D2 \ D1 D0	00	01	11	10
00	1	1	0	0
01	0	0	1	0
11	0	0	0	0
10	0	0	0	0

$Y_g = \overline{D3}\,\overline{D2}\,\overline{D1} + \overline{D3}D2D1D0$

Figure 3

Following this original prelab caused the actual lab to be reduced in complexity, because I did not need to reverse the values of my truth table to swap the MSB and LSB, and invert all of my values to go from active high to active low. Considering how the Basys3 board requires an active low, and my truth table was already in active low, I did not need to convert. Because of this, I saved myself a lot of work and directly implemented what I had done in the prelab into my code.

Discussion

This lab contributed a lot to my learning about VHDL. On a personal level it contributed to understanding more about myself. I get really frustrated and stressed when I do not understand something and I have to do it, versus when I do understand something and I have to do it. When I do understand it and have to do it, I like doing it. When I do not understand it and try to do it, I get immensely frustrated and stressed. Due to the fact that I could not understand how to do this lab for 2 weeks, I had a lot of excess stress for PROG. But once I understood it (which at the time of writing this I still do not know), I will no longer be stressed. I truly want to finish out all my work for PROG and look back and say "wow, I am proud of the work I put in and the results I got". I am hoping that will be the case depending on my PROG grades. In regards to what I learned about ECE 281, I learned Dr. York is not as scary as I thought he was for seeking EI from him. I also learned more about creating actual files in VHDL, and can (hopefully) recreate that for future labs, which is what I assume we will be doing.

Time log

The prelab was quite simple and took me about an hour. The lab itself was very complex because I am lacking in VHDL skills, and across 2 weeks took me about 14 hours. The lab report was long and took me about 3 hours. All in all, I spent about 18 hours doing the lab, and another 24 hours actively stressing about whether I would finish it or not.

Documentation Statement

For this lab I received a lot of EI from Lt Col Wyche. He helped me understand on multiple occasions (at least 3) how the VHDL code worked. Additionally, I sought EI from Dr. York in regards to my VHDL test bench, because I was consistently getting errors. Dr. York helped me fix my test bench and helped me realize I was doing a lot of things that VHDL did not like and that was causing the errors.