Yuan Wang and Francesca Narea
Laboratory #6
ELEN/COEN 21L
2-24-17

Introduction:

In this lab, we used hierarchical design to incorporated the use of multiple lower-level symbol files into our final ALU design. In our prelabs, we created the schematic for this ALU design and filled in a truth table of values for the 4-Bit inputs, Carry In and Outputs based on given arithmetic functions.

Procedures:

First, we created and compiled the schematic and symbol for the 4-Bit Vcc. Next, we created the symbol for the 4-Bit 8-1 Mux. This schematic consists of four 1-Bit 8-1 Muxes, with Enable connected to power, three selectors and eight 3-Bit inputs, labeled A-H. In order to accommodate the 4-Bits, we made use of the Orthogonal Bus tool. After this symbol set-up, we created the ALU schematic with two 4-Bit inputs and using two 4-Bit inverters, two 4-Bit 8-1, one 1-Bit 8-1 Mux, one 4-Bit Vcc, one 4-Bit Gnd and one Ripple Carry Adder. The outputs of the two 4-Bit 8-1 Muxes connected to the two inputs for the Ripple Carry and we used the 1-Bit 8-1 Mux as the Cin. The After a successful compilation, we created pin assignments in order to display our schematic on the 7-Segment display on the Altera FPGA.

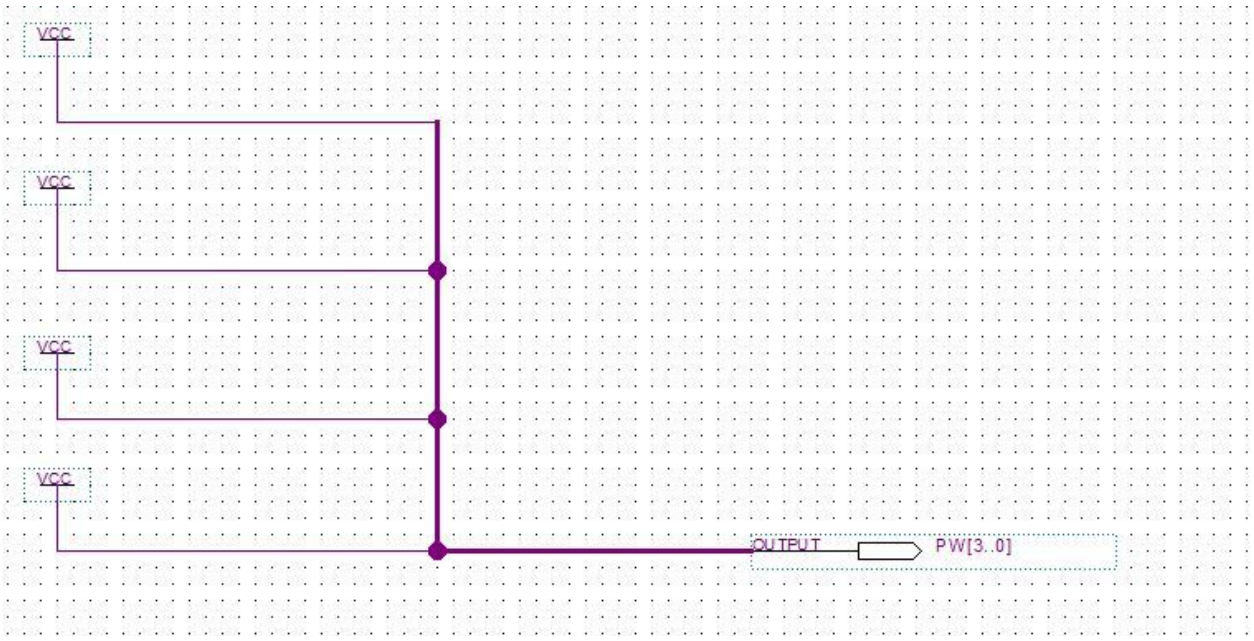

Figure 1: Schematic for 4-Bit Vcc

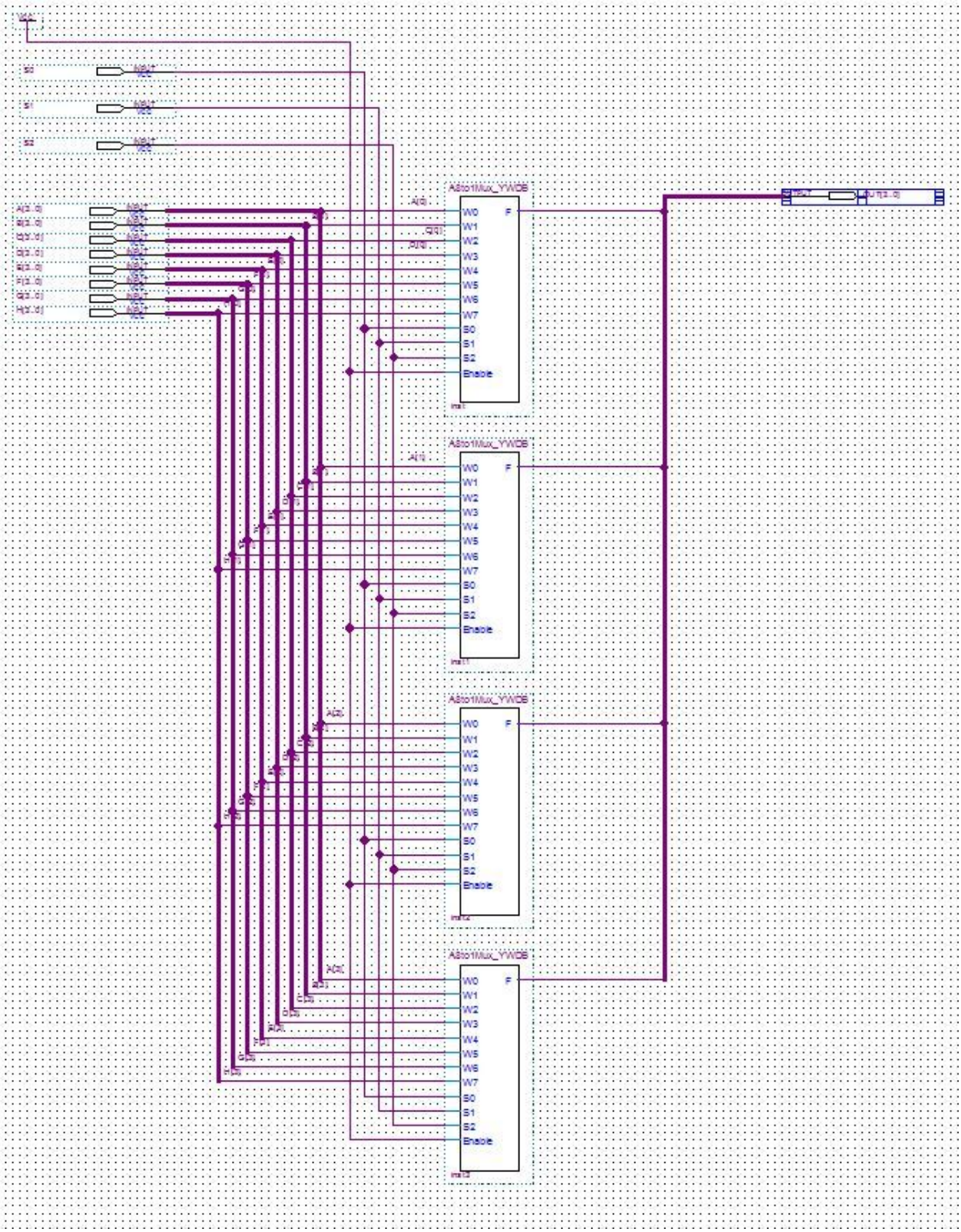

Figure 2: Schematic for 4-Bit 8-1 Mux

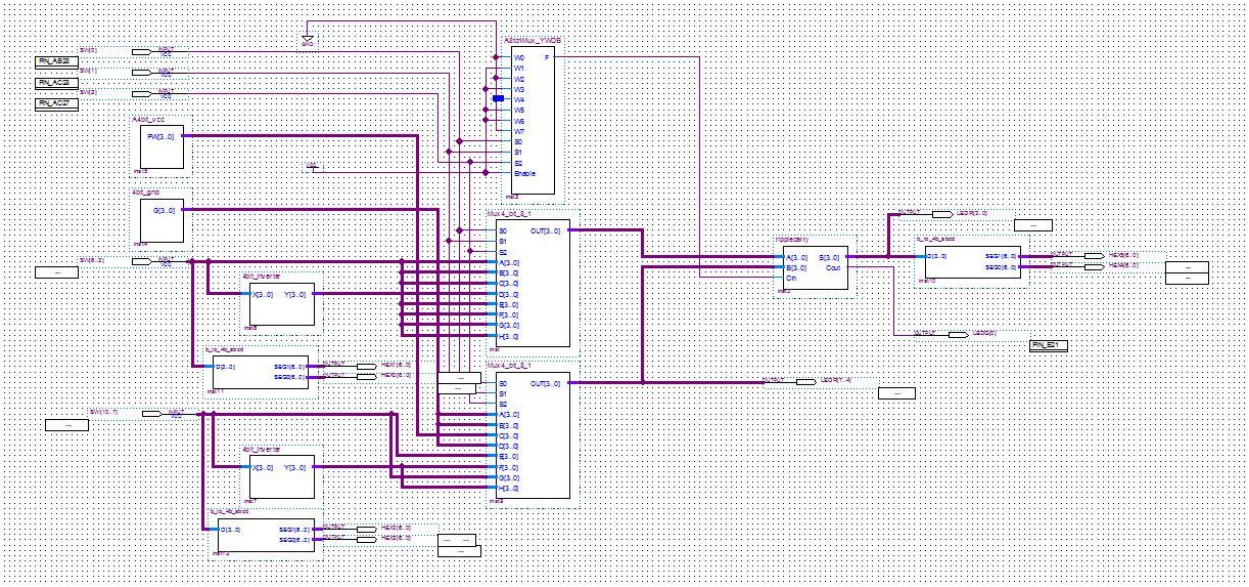

Figure 3: Schematic for ALU

Test Plan

Test 1:
Set switches 0-2 to 0. Set switches 3-6 to 1010. Set switches 7-10 to 0.
Expected Result: Output is 5 because it is transferred from A.
Purpose: Test Function A.

Test 2:
Set S[0] = 0, S[1] = 0, S[2] = 1 and S[3..6] = 1010. S[7..10] = 0.
Expected Result: Output is 6 because it is incrementing A.
Purpose: Test Function B

Test 3:
Set switches S[0] = 0, S[1] = 1, S[2] = 0, S[3..6] = 1010
S[7..10] = 0.
Expected Result: Output is 4 because it is decreased by 1
Purpose: Test Function C.

Test 4:
Set switches S[0] = 0, S[1] = 1, S[2] = 1, S[3..6] = 1010
S[7..10] = 0.
Expected Result: Output is -5 because it is negated.
Purpose: Test Function D.

Test 5:
Set S[0..2] =100 , S[3..6] = 1010, S[7..10] = 0001.
Expected Result: Output is 6 because 5 +1 = 6
Purpose: Test Function E.

Test 6:
Set S[0..2] =101 , S[3..6] =1010 , S[7..10] = 0001.
Expected Result: Output is 4 because 5-1 = 4
Purpose: Test Function F.

Test 7:
Set S[0..2] =110 , S[3..6] =1010 , S[7..10] = 0001.
Expected Result: Output is 7 because 5 +1+1=7
Purpose: Test Function G.

Test 8:
Set S[0..2] =111 , S[3..6] = 1010, S[7..10] = 0001.
Expected Result: Output is 3 because 5-1-1 =3
Purpose: Test Function H.

How would you add logic to your circuit to detect the overflow conditions?

- Add an LED connecting to
- XOR or XNOR A&B

How would you connect two 4-bit ALUs to make an 8-bit ALU?

- Separate 8 bit input as two 4 bit inputs. Add 0-3 bits of two inputs together and then the carry our will be the carry in of the other 4 bit ALUs. Then add 4-7 bits of two inputs with carry in from the previous 4-bit ALUs. Then the results contains the sums from two 4-bit ALUs.

Conclusion:

During this lab, we learned how to incorporate multiple symbol files into a complex schematic for an ALU. We further increased our skills in learning how to break down hierarchical design through these symbol files. Additionally, we have been gradually adding on more and more bits on our inputs and outputs, so we have learned the importance and necessity of the bus tool as well as the value of Muxes, especially given that some individuals implemented the problem statement in the prelab using AND, OR, inverters, etc. For example, we were able to incorporate multiple bit Muxes in our ALU schematic, which allowed us to easier grasp the concept of the ALU, as we abstracted from the more detailed and time consuming method with AND, OR, inverters, etc. We also learned about the importance of a test plan, as we were required to report our values and tests to see if our schematic worked as described.

We realized that as our problem statements become more and more complex, the testing process also becomes more complex, which means that we have to build our skills in developing a logically minimized plan to confirm that our lab performs correctly.