CprE 381, Computer Organization and Assembly Level Programming

Lab 1 Report

Student Name Zachary Scurlock

Submit a typeset pdf version of this on Canvas by the due date. Refer to the highlighted language in the lab document for the context of the following questions.

[Part 1.c] Think of three more cases and record them in your lab report.

Case 1: Perform a calculation that will result in the number 69

Set iX to 30, iW to 2, iLdW to 1, and iY to 9, which outputs the number 69.

Case 2: Output days since January 8, 2022 (As of August 23, 2023).

Set iX to 250, iW to 2, iLdW to 1, and iY to 92, which outputs the number 592.

Case 3: Output birth year

Set iX to 1000, iW to 2, iLdW to 1, and iY to 2, which outputs the number 2002.

[Part 1.e] For labels 9, 20, 32, and 33, specify where (VHDL file and line number) these values are located – some will be found in more than one place. Also attempt to explain the functionality of each label as it occurs in the code

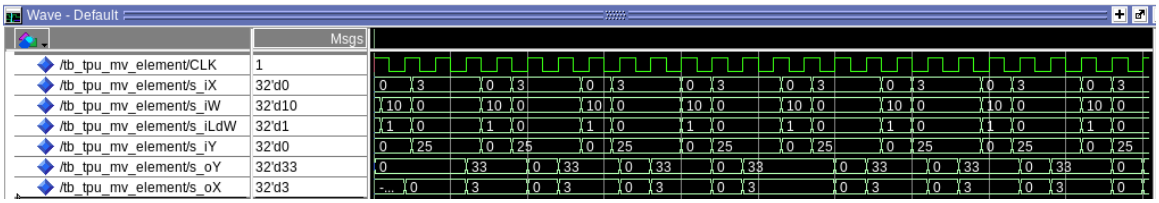
Label 9: In the attached diagram area, (9) is the oQ output of the g_Weight module which serves as the input of the multiplier module g_Mult1.

Label 20: In the attached diagram area, (20) is the iD input of the g_Delay3 module which is a D flip flop serving as a delay for the circuit.

Label 32: In the attached diagram area, (32) is the adder module of the circuit which takes an input from the multiplier module (g_Mult1) and the delay module (g_Delay2), and outputs the final value.

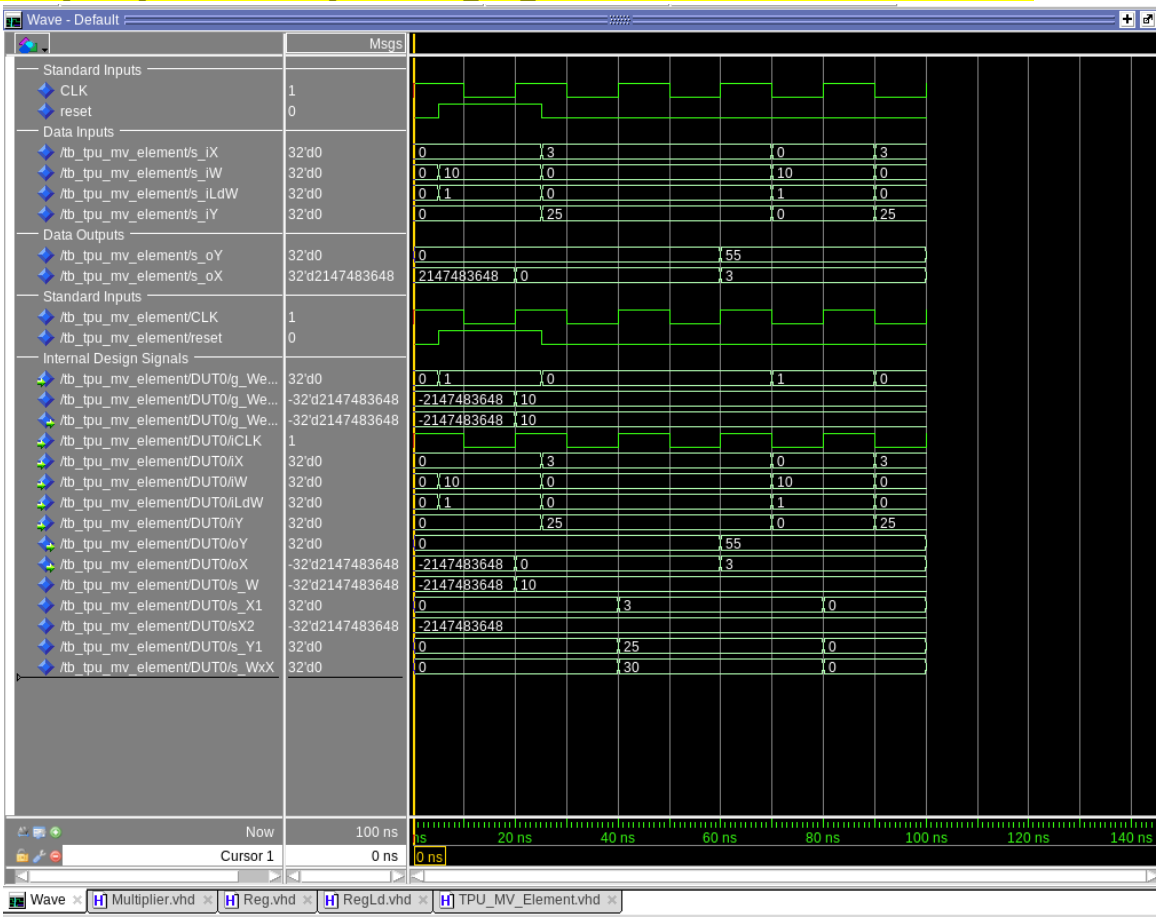
Label 33: In the attached diagram area, (33) is the circuit in its entirety known as TPU_MV_Element. The purpose of this circuit is to take four inputs from the user, iX, iW, iLdW, and iY, and output them as oX, and oY. In this circuit three delay modules can be found serving to delay the clock while another part of the circuit completes calculations. A multiplier module can also be found which multiplies two of the user inputs and outputs them into the adder module.

[Part 1.g.v] In your lab report, include a screenshot of the waveform. Describe, in plain English, any differences between what you expected and what the simulation showed.



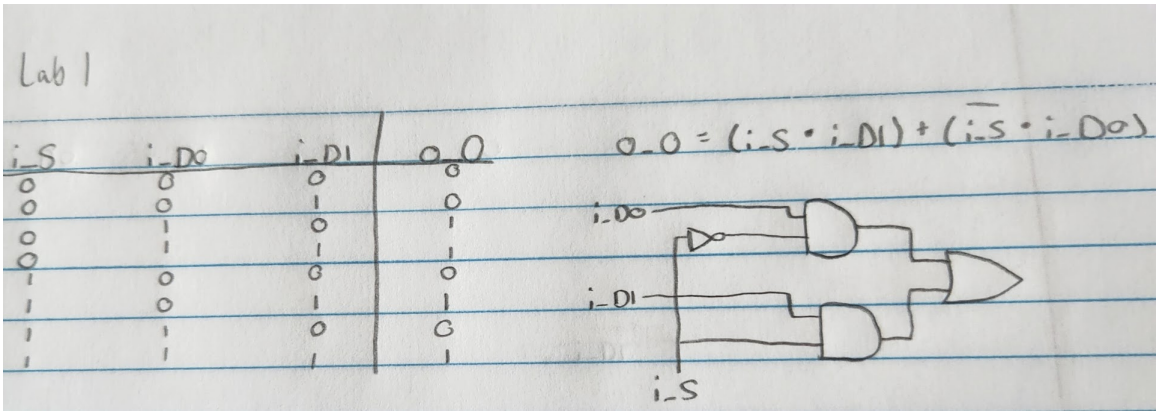
The MAC is outputting 33 instead of 55, because iX is being used as an input to the g_Add1 module instead of iY.

[Part 1.h] In your lab report, include a screenshot of the waveform. Describe, in plain English, how your waveform matches the expected result (e.g., reference the specific cycles and times). In your submission zip file, provide the completed *TPU_MV_Element.vhd* file in a folder called 'MAC'.



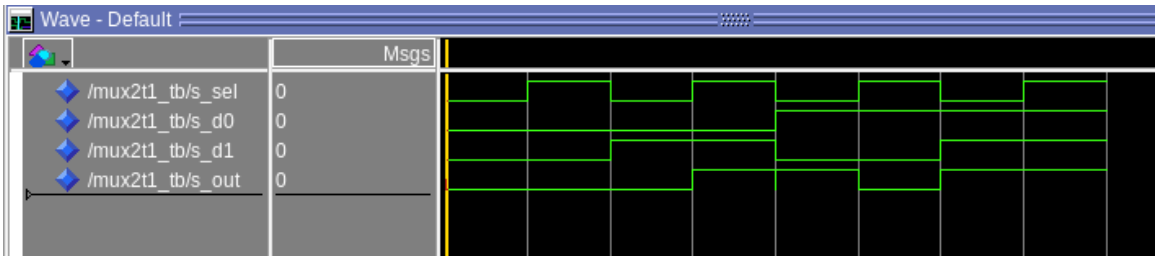
This waveform matches the expected results because the output oY is 55 rather than 33 as seen before.

[Part 3.a] Draw the truth table, Boolean equation, and Boolean circuit equivalent (using only two-input gates) that implements a 2:1 mux. Include this in your lab report.



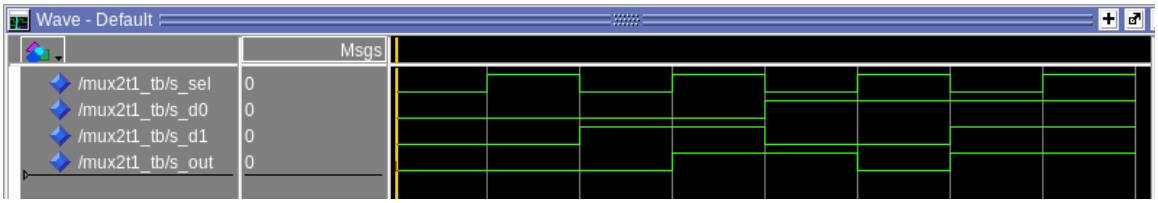
[Part 3.d] In your lab report, include a screenshot of the waveform. Make sure to label the screenshot with which module it is testing.

Testing the 2 to 1 mux



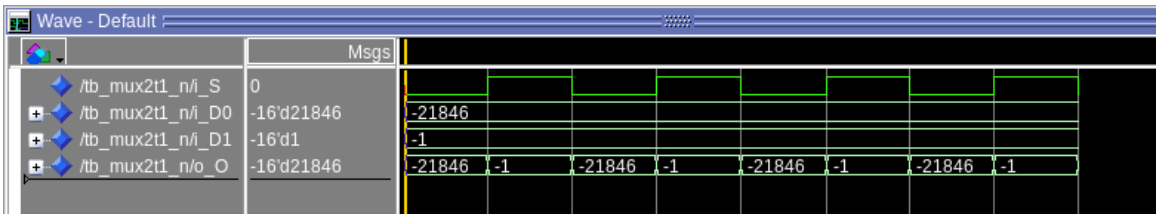
[Part 3.e] Again, in your lab report, include a labeled screenshot of the waveform showing the dataflow mux implementation working.

Testing the dataflow 2 to 1 mux

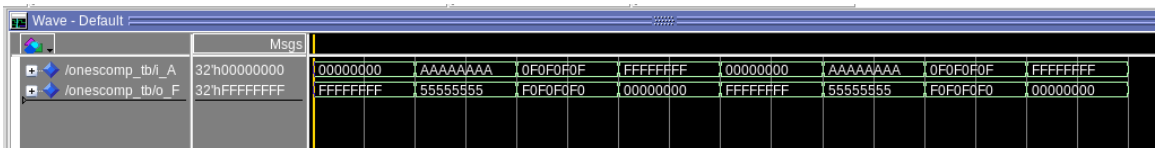


[Part 4] Include a waveform screenshot and corresponding description demonstrating it is working correctly.

Testing the N-Bit 2 to 1 mux

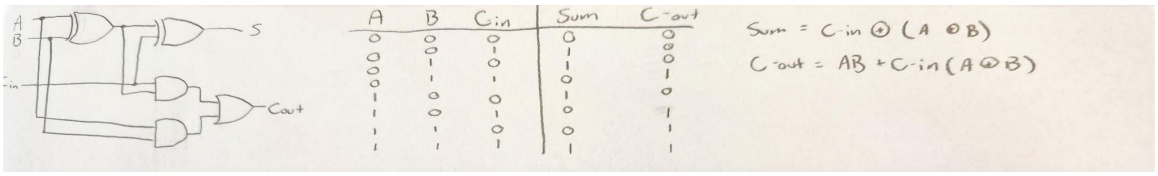


[Part 5.b] Include a waveform screenshot and description in your lab report.

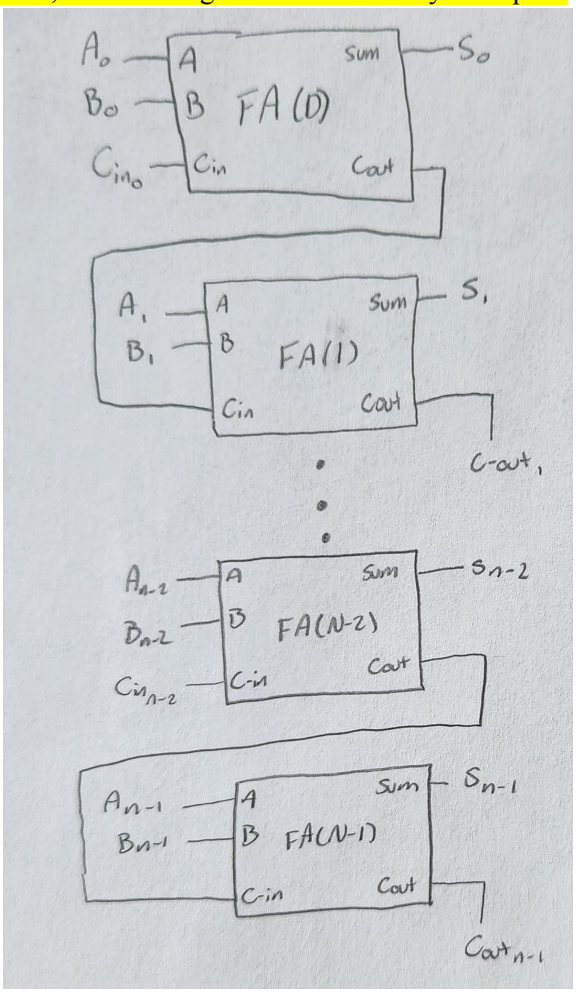


Testing N-Bit Inverter

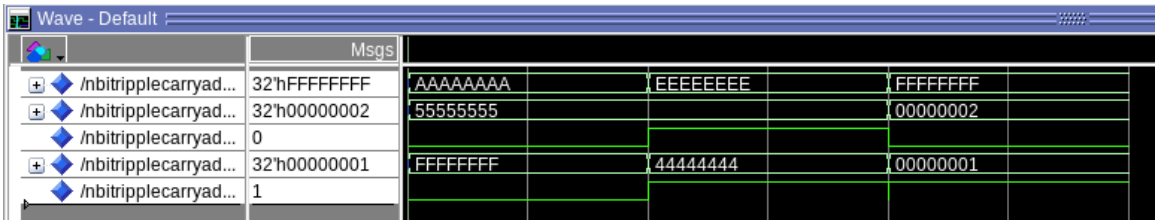
[Part 6.a] A full adder takes three single-bit inputs and produces two single-bit outputs – a sum and carry for the addition of the three input bits. Draw the truth table, Boolean equation, and Boolean circuit equivalent (using only two-input gates) that implements a 1-bit full adder. Include this in your report.



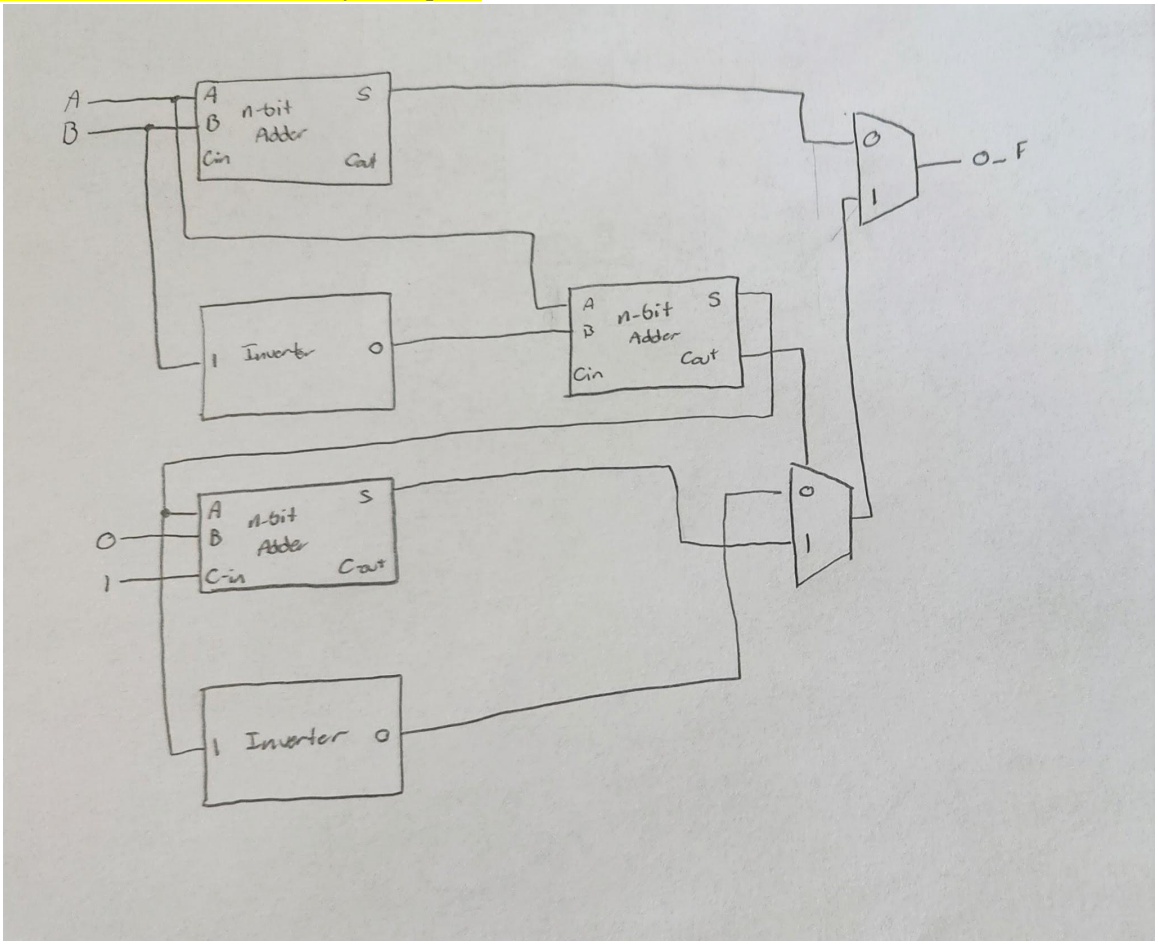
[Part 6.c] Then draw a schematic of the intended design, including inputs and outputs and at least the 0, 1, N-2, and N-1 stages. Include this in your report.



[Part 6.d] Include an annotated waveform screenshot in your write-up.

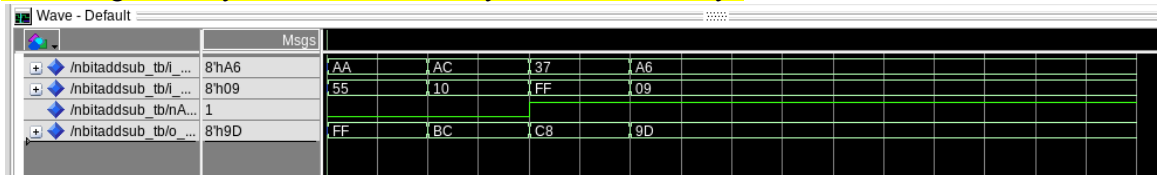


[Part 7.a] Draw a schematic (don't use a schematic capture tool) showing how an N-bit adder/subtractor with control can be implemented using only the three main components designed in earlier parts of this lab (i.e., the N-bit inverter, N-bit 2:1 mux, and N-bit adder). How is the 'nAdd_Sub' bit used? Include this in your report.



n_Add_Sub is used towards the end of the circuit at the last N-bit multiplexor select line to determine if the output should be the added or subtracted numbers.

[Part 7.c] Provide multiple waveform screenshots in your write-up to confirm that this component is working correctly. What test-cases did you include and why?



--Test 1--

```
i_A_tb <= "10101010";                    Testing adding two numbers
i_B_tb <= "01010101";
wait for 100 ns;

--Test 2--
i_A_tb <= x"AC";                         Testing adding two number
i_B_tb <= x"10";
wait for 100 ns;

--Test 3--
i_A_tb <= "00110111";                  Testing subtracting a large number from a small number
i_B_tb <= "11111111";
nAdd_Sub_tb <= '1';
wait for 100 ns;

--Test 4--
i_A_tb <= x"A6";                     Testing subtracting a large number from a small number
i_B_tb <= x"09";
wait for 100 ns;
```