

ELC 2137 Lab #4: Subtractor

Maya Martin & Xingpeng Yi

February 12, 2020

Summary

The subtractor is similar to the 2-bit adder in the previous lab. When dealing with a subtraction of the binary, easiest way is adding the 2's complement of the second number to the first. In order to calculate the 2's complement of the second number, XOR gates are added between the Mode terminal and each bit of the second number. A XOR gate is also placed between the mode terminal and the out carry of the two full adders, and the result is the out carry of the subtractor.

Q&A

1. Why did we use two full adders instead of a half adder and a full adder?

Although half and full adders are both combinational logic circuits they hold different roles regarding how they process their inputs. In this lab we focused on the function of two full adders instead of a full and half adder combination because it allows our circuit the flexibility to build on each other. By using a full adder we are able to add three 1-bit digits that consist of OR/AND gates which in return allows us to add and carry along other inputs. Unlike half adders which would not have kept any of our addition throughout the lab.

2. How many input combinations would it take to exhaustively test the adder/subtractor?

By squaring the binary conditions (n^2), we are able to exhaustively test the adder /subtractor (23) giving us the value 529 binary combinations. With these values we can find the exhaustive test to be equal to 4,232 input combinations.

3. Why were the combinations given in the truth table chosen?

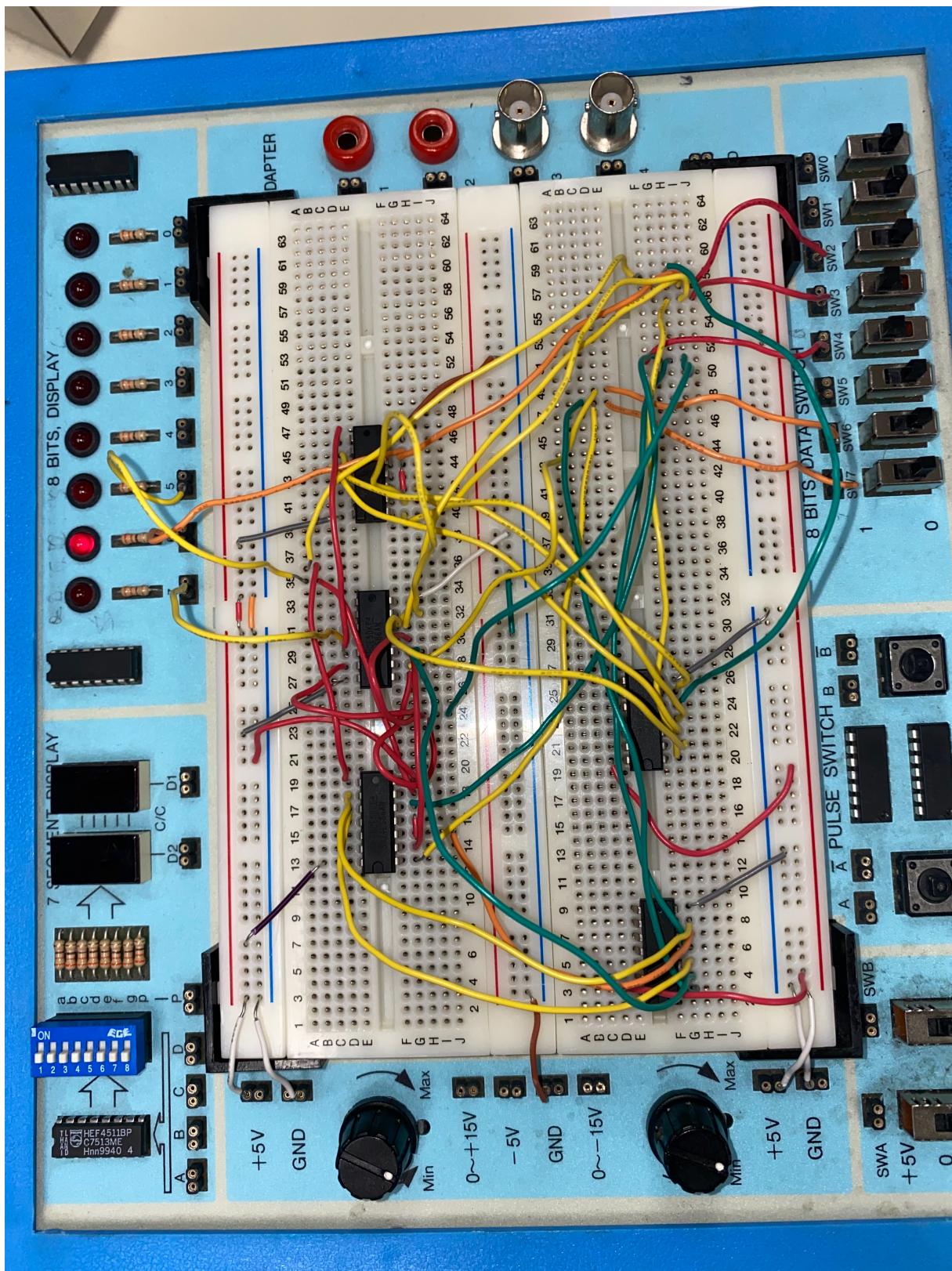
The combination that were given on the truth table were chosen to expand our concept of how to flip the carry/borrow output and solidify the concept of a subtractor circuit and its overall purpose.

4. Do the results from your adder/subtractor match what you would expect from theory? Explain any discrepancies.

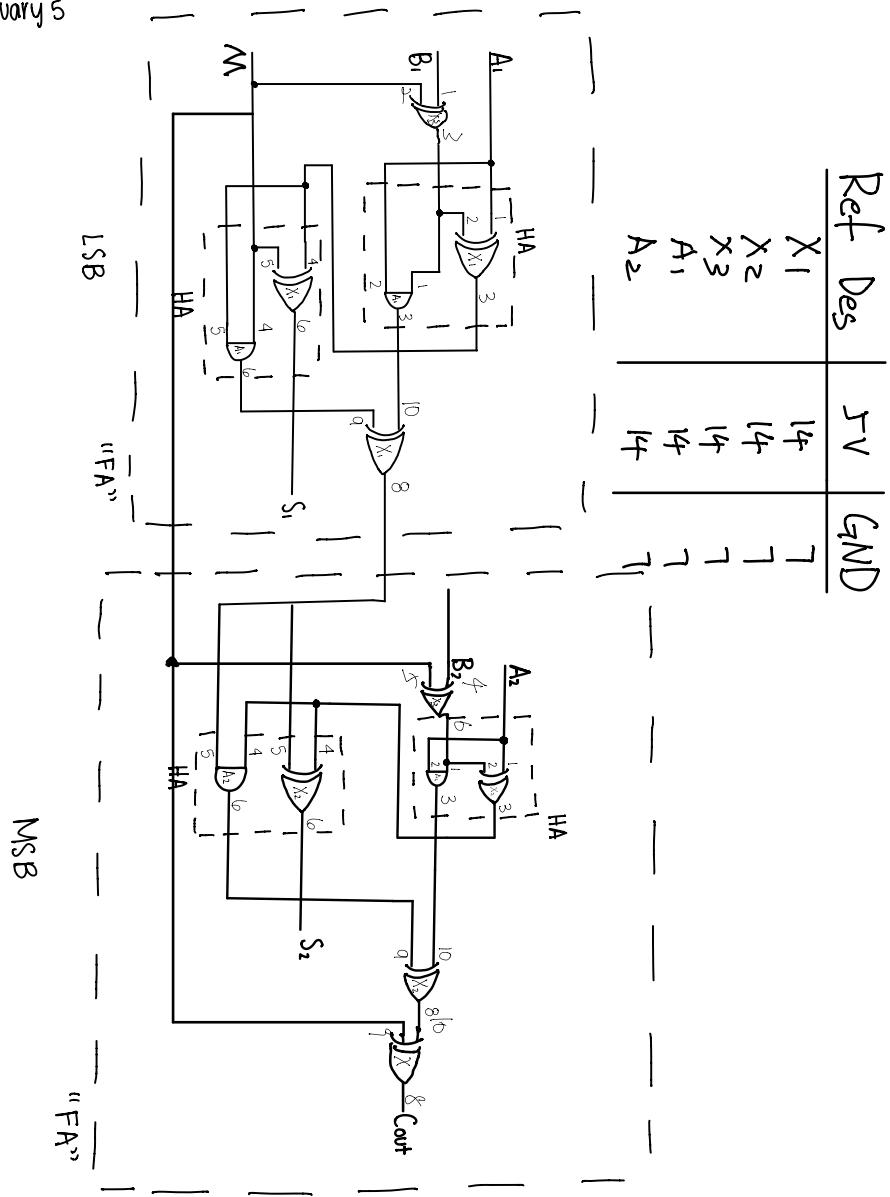
The results from our theory did not reflect in our results, in such that our results flipped at the first digit in our expected results. For example for our expected results we calculated 011 but ended up getting 111, but flipping the a digit this resulted in us getting the correct results. A discrepancy that we encountered was the of the result is opposite to the expected result. 0s in the expected result table become 1s in the actual results, 1s become 0s. The reason is that a XOR gate is placed before the out carry. The mode is used to control the negation. The out carry of the two full adders and the mode goes through a XOR gate, and

the output of the XOR gate is the out carry of the whole subtractor. When the mode is turned on, negation will turn the output to the opposite result.

Results



maya martin
Pre-lab #4
February 5



Circuit Demonstration Page

Student names:

maya martinxingpeng yi**Instructor Signatures**

Separate Full Adders

Gil Buffman

Two-Bit Adder

Gil Buffman

Adder/Subtractor

Gil Buffman

Inputs		Expected Results			Actual Results	
A	B	B 2's comp	Sub	Dec	Sub	
00	01	11	011	-1	111	
00	10	10	010	-2	110	
00	11	01	001	-3	101	
01	01	11	100	0	000	
10	01	11	101	1	001	
10	00	100	110	2	010	

Conclusion

The subtraction of two binary numbers are simplified to minuend add the 2's complement of the subtrahend. To get the 2's complement of the subtrahend, XOR gates are placed between mode terminal of each bit of the subtrahend. The out carry of the whole circuit is the output of a XOR gate which has the out carry of the 2-bit adder and the mode terminal as the inputs. When mode is turned off, which is 0, the whole circuit is working as a 2-bit adder. Reversely, if the mode is on, which is 1, the circuit becomes a subtractor. Based on the truth table of XOR gate, if both of inputs are 1s, the output is 0. One of the inputs is 1, the output is 1. Under the subtraction mode, mode input is 1. If the input is 1, it becomes 0 after the XOR gate. If the input is 0, it becomes 1 after the XOR gate. The mode is also connected to the whole circuit, which adds 1 to the circuit numbers. The whole process is how the 2's complement is found. Flip each bit of the number and add 1 at the last. The 2's complement of the subtrahend is added to the minuend which is the expected result. The actual result differs from the expected result. The XOR gate after the out carry negates the out carry of the whole circuit, so the first bit of actual result is opposite of the first bit of the expected result.