CSC 347/ENS 211

Title and Experiment #	Lab3: 1-bit Adder/Subtractor
Name	Gianna Galard
Date Performed	28-Sep-21
Date Submitted	03-Oct-21

The student pledges this work to be their own *Gianna Galard*

Objective:

This week's lab aims to build circuits that carry out essential arithmetic functions, specifically addition and subtraction. Performing 1-bit arithmetic requires considering whether the process will include a carry bit or an extra 1 due to the addition or subtraction. In addition, the output of the arithmetic consists of a carry bit and the sum bit, which is the original answer. The carry bit can be represented as an AND gate since both results are equivalent, where the sum bit is shown as an XOR gate for the same reason. Subtraction considers similar issues as addition, whether a borrow-in bit is included in the arithmetic or if an extra bit is made due to borrow-out. The Bout bit can be represented as the AND gate, and the difference bit is shown as the XOR gate.

Equipment and Chip used:

- TinkerCad
- 74HCT**04** inverter
- 74HCT**08** Quad 2-input AND Gate
- 74HCT**32** Quad 2-input OR Gate
- 74HCT**86** Quad 2-input XOR Gate

Design Procedure:

The student built a full adder to prove that a digital circuit can perform addition in this lab.

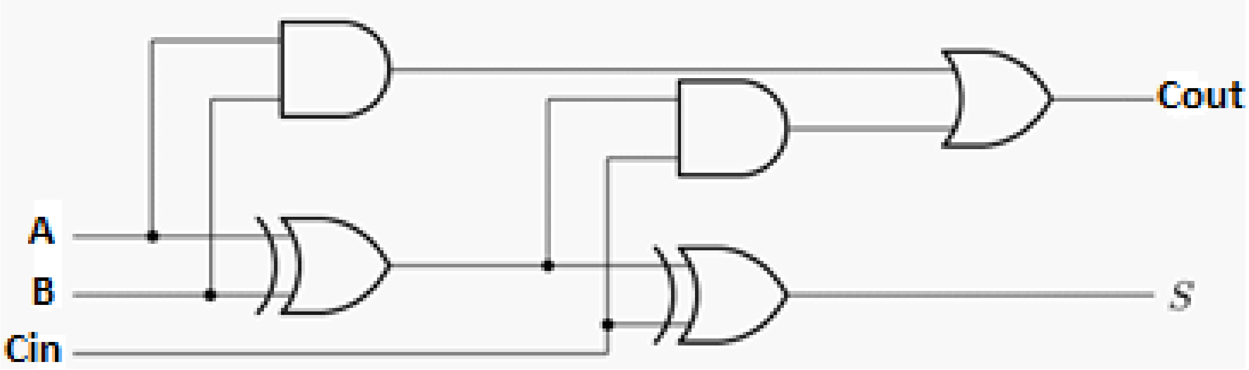
We have three variables, A, B, and Cin. The A and B are data bits; the input carry is the Cin. We have Cout and S columns which stand for the output carry and Sum bit. We filled in our truth table by doing A + B + Cin and outputting 1 as 01, 2 as 10, and 3 as 11. Therefore we had 2^3 = 8 possible variations and therefore rows on the truth table. Cout being the left-most being and S being the right-most bit.

Figure 1. Full Adder Truth Table

A	B	Cin	Cout	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Logic Diagram:

Figure 2. Full Adder Logic Diagram.



Experiment:

After reviewing the logic diagram, the student used TinkerCad to begin circuit construction. It was built on a singular breadboard using the AND, OR, NOT, and XOR Gates. The student approached this in an organized manner by color coordinating the wires: **Cout** is represented by **Purple**, and **S** is represented by **Pink**.

The student applied all eight combinations during testing with the circuit board and proved this truth table to be true.

Figure 3. 0 + 0 + 0 = 00.

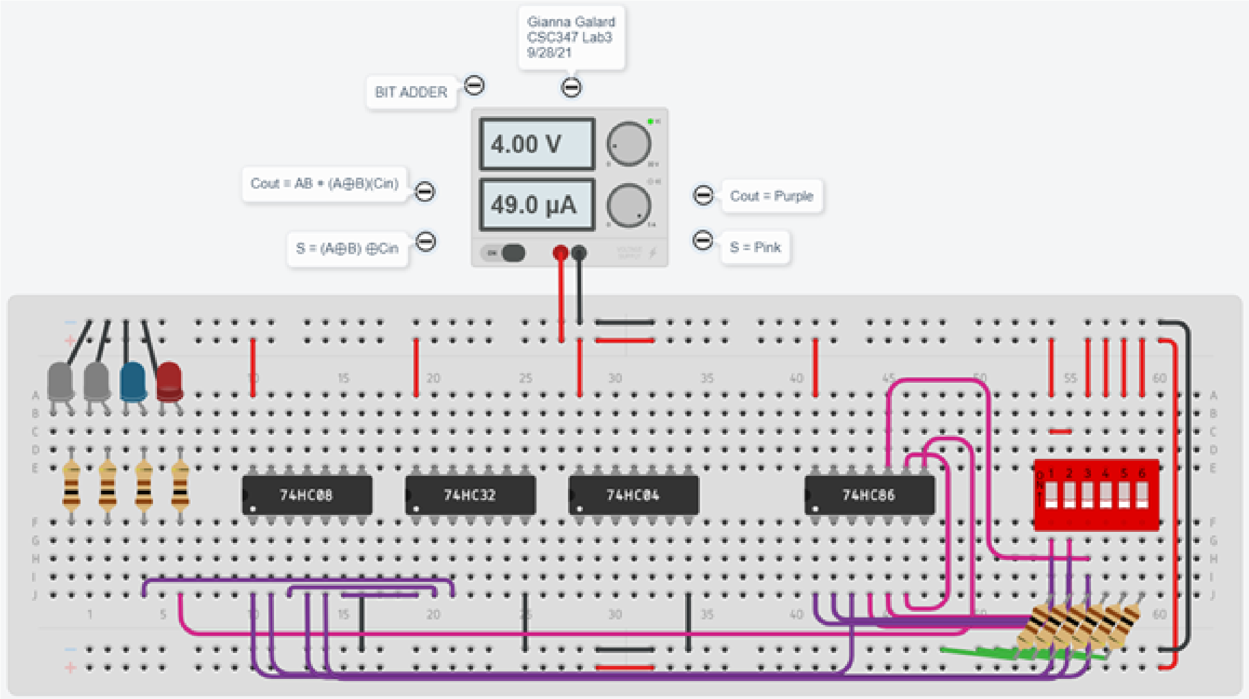


Figure 4. 0 + 0 + 1 = 01.

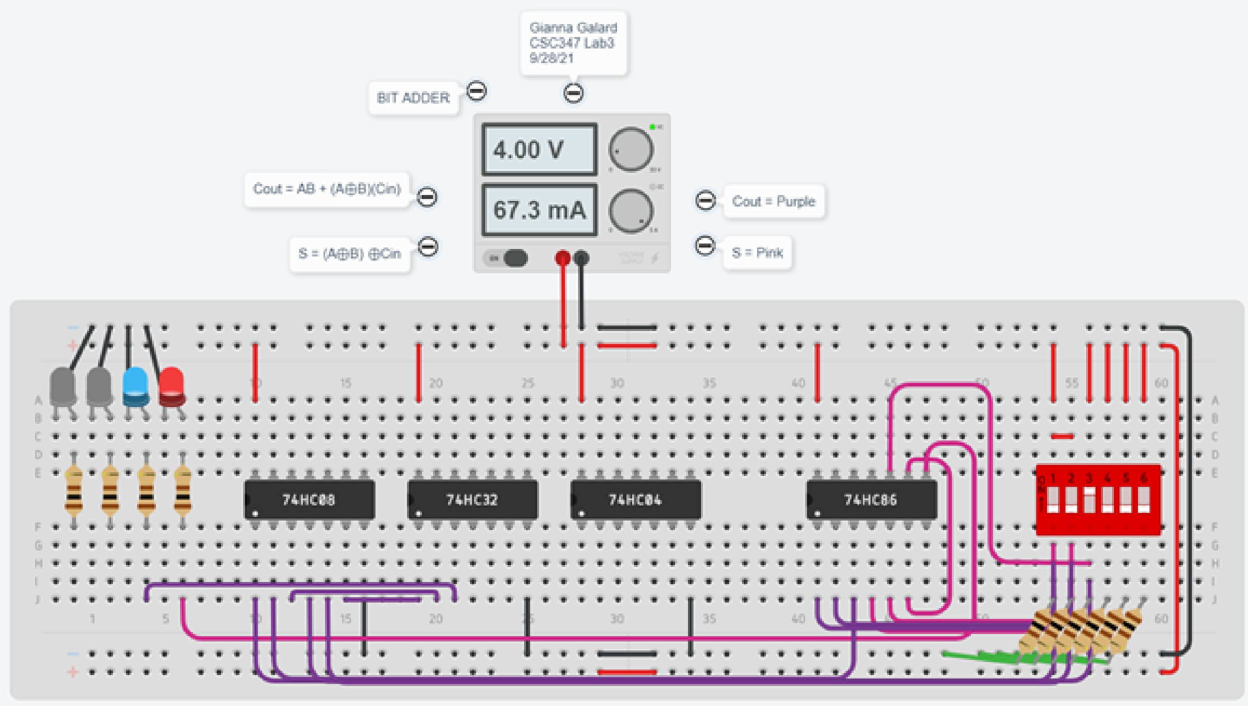


Figure 5. 0 + 1 + 0 = 01.

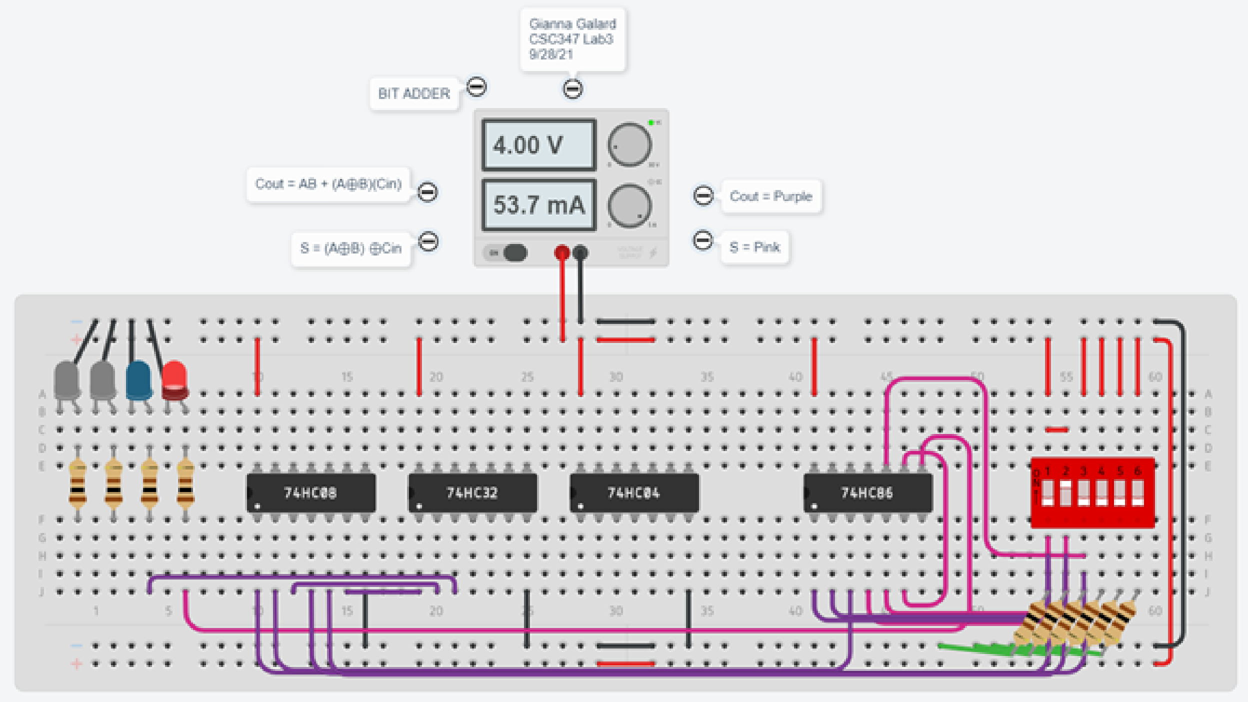


Figure 6. 0 + 1 + 1 = 10.

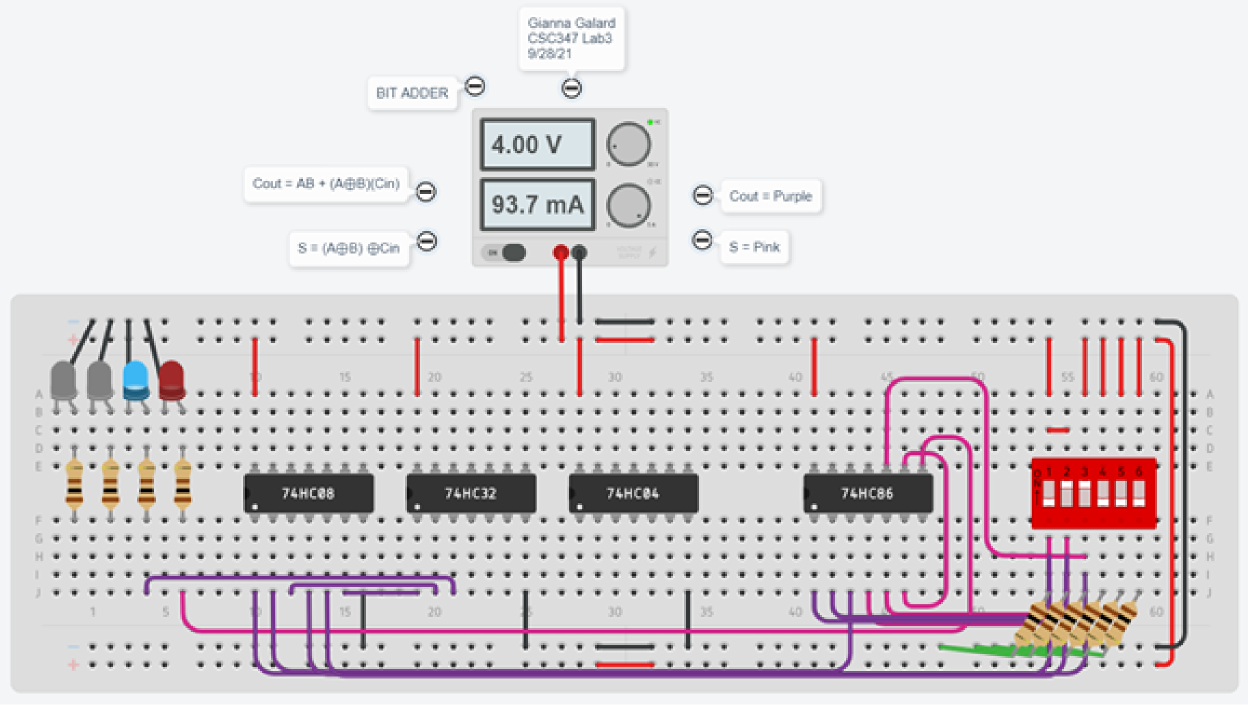


Figure 7. 1 + 0 + 0 = 01.

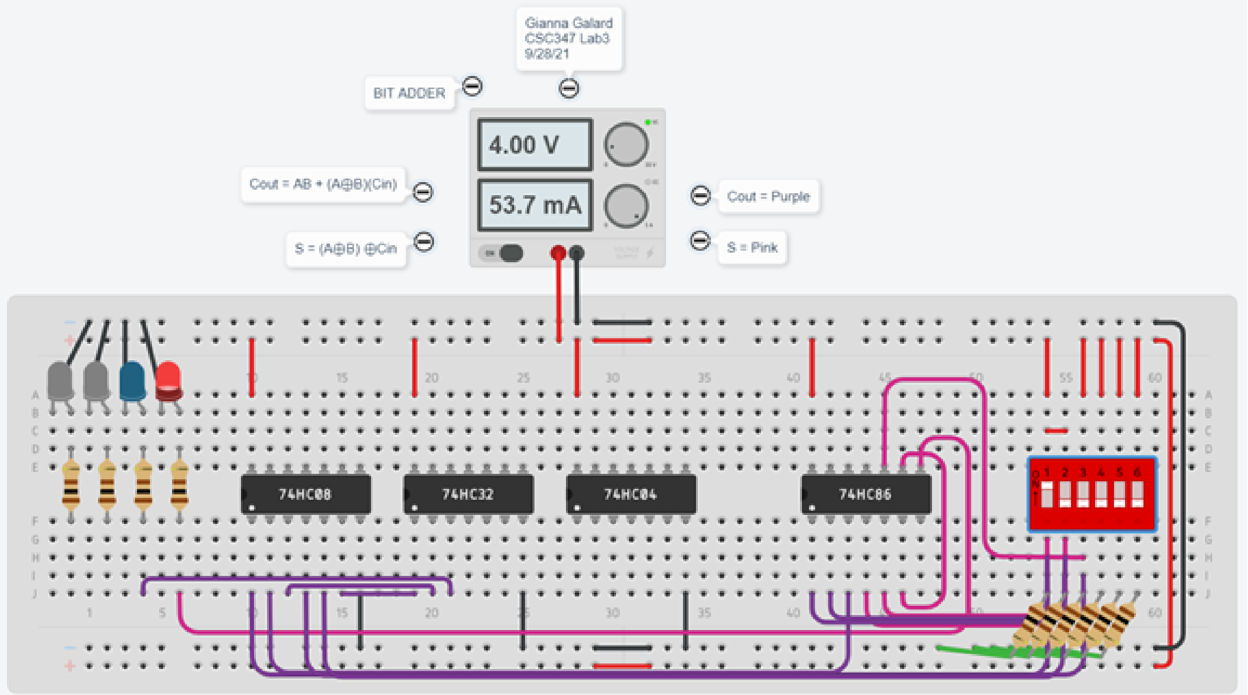


Figure 8. 1 + 0 + 1 = 10.

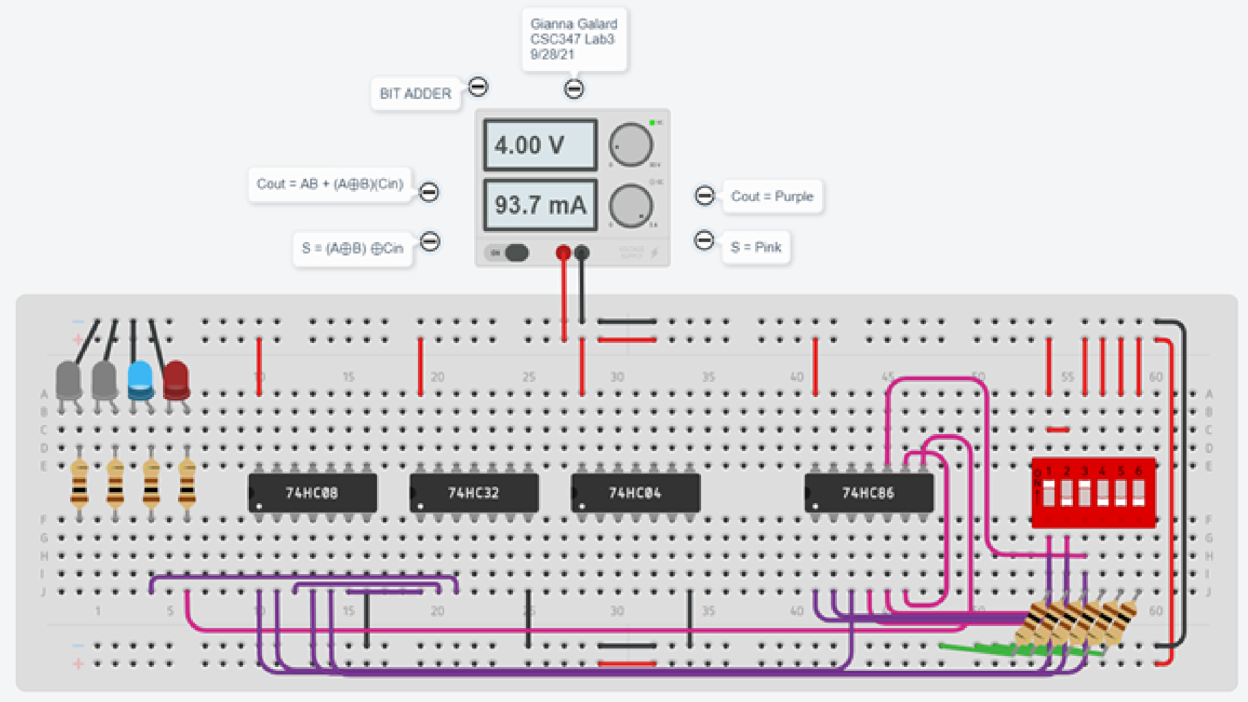


Figure 9. 1 + 1 + 0 = 10.

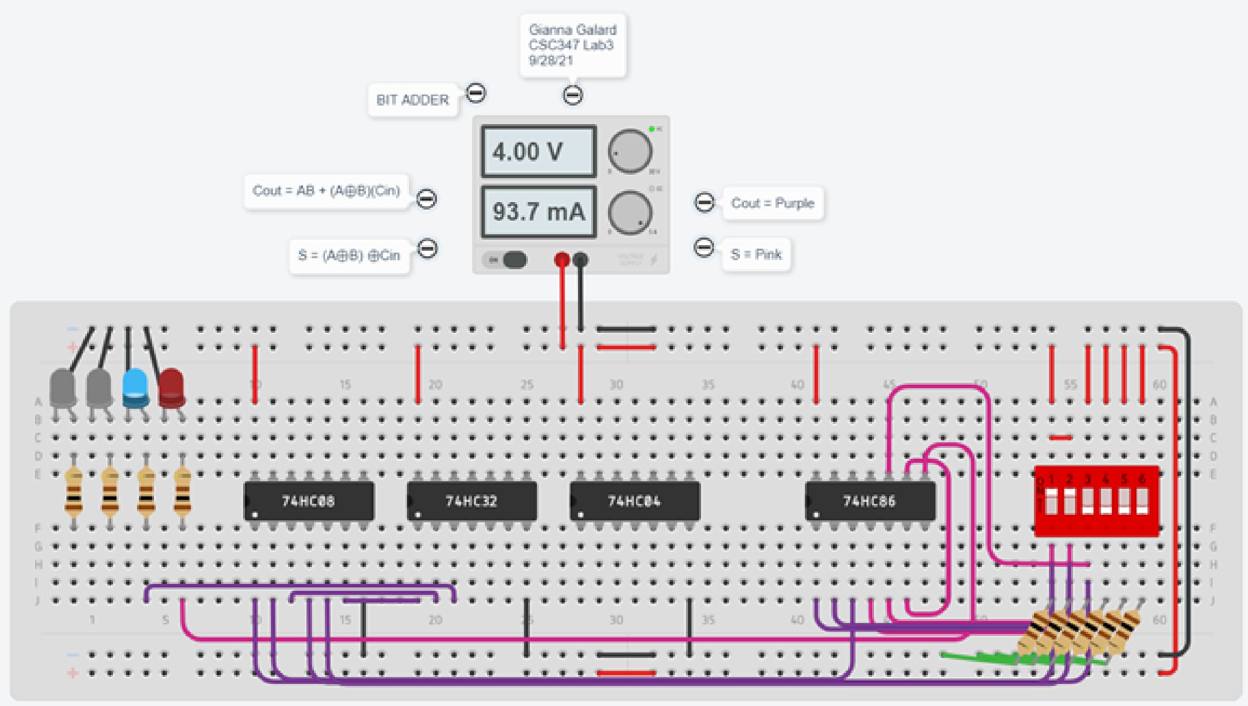
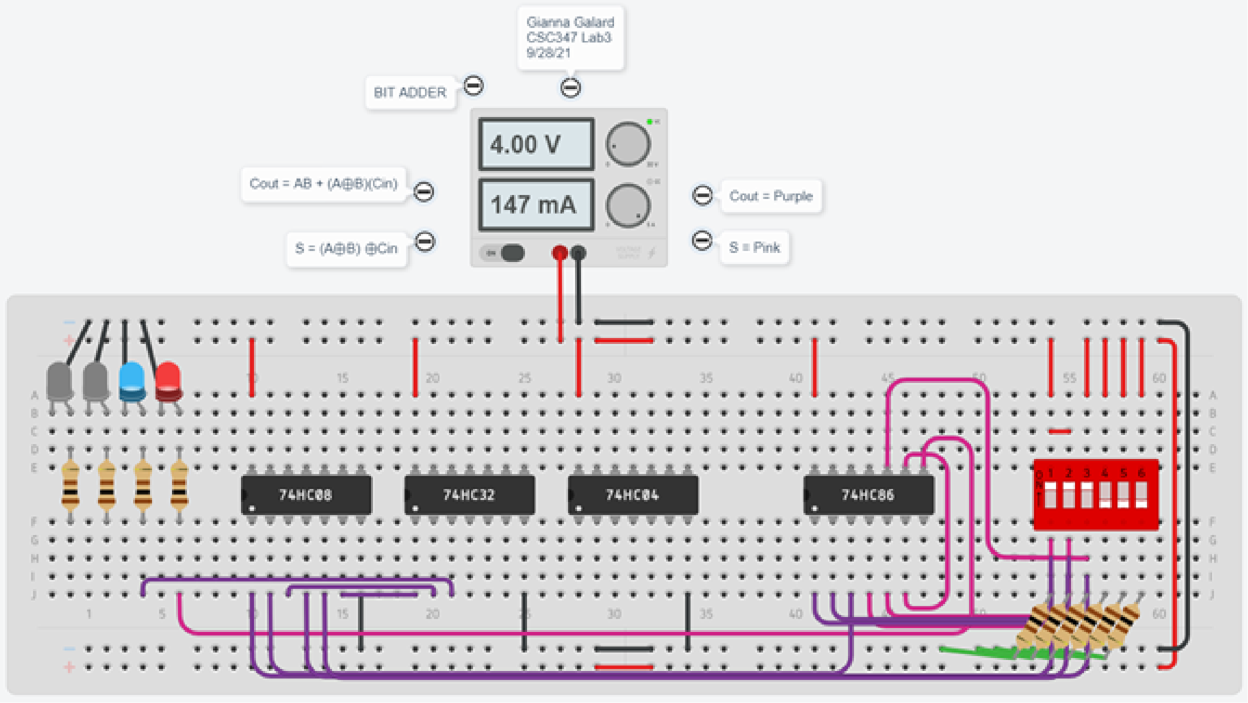


Figure 10. 1 + 1 + 1 = 11.



Conclusion:

After the student tested each input for both the adder and subtractor circuits, they proved that the expected outputs for the sum and carry bits and the difference and Bout bits match their actual outputs. By confirming the outputs for each resulting bit, it is then proven that the sum and difference bit is like an XOR operation, whereas the carry and Bout bit have a strict correlation to the AND gate. Although each input goes through more than 1 level of gates, the result still indicates 1 accurate output, showing that each circuit is not just one dimensional, as exemplified in previous labs. The third input, z, and the Bin, accurately records the possibility of an extra bit added to the equation to show the possible outcome if such a situation arises. By having a separate indicator for the extra carry or Bout bit, each circuit can successfully carry out 3-bit arithmetic.

Homework:

Follow the same procedure for the 1-bit adder to build a 1-bit subtractor. Derive the truth table for 1-bit full subtractor (A- B – Bin), write down the functions, and build the subtractor using the given logic diagram.

Figure 11. Full Sub Truth Table.

A	B	Bin	Bout	D
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Figure 12. Full Sub Logic Diagram.

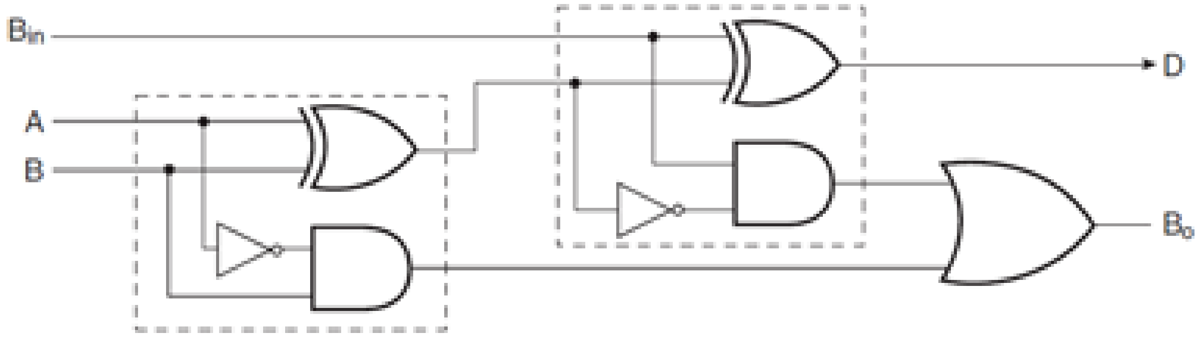


Figure 13. 0 - 0 - 0 = 00.

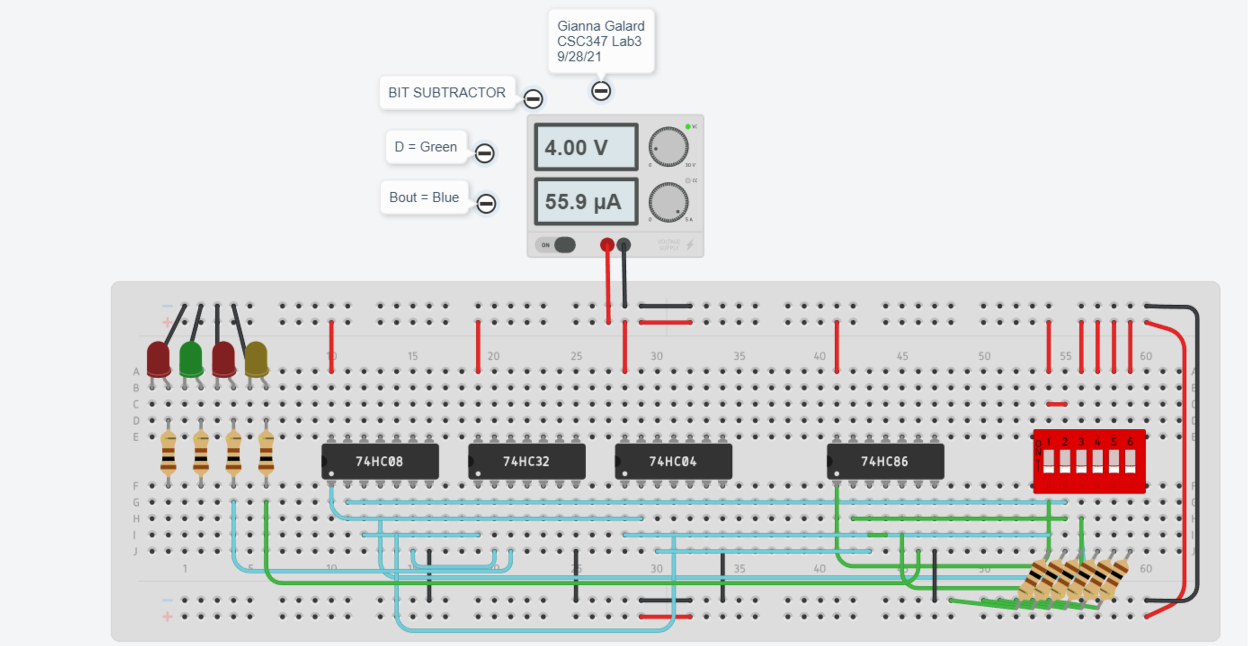


Figure 14. 0 - 0 - 1 = 11.

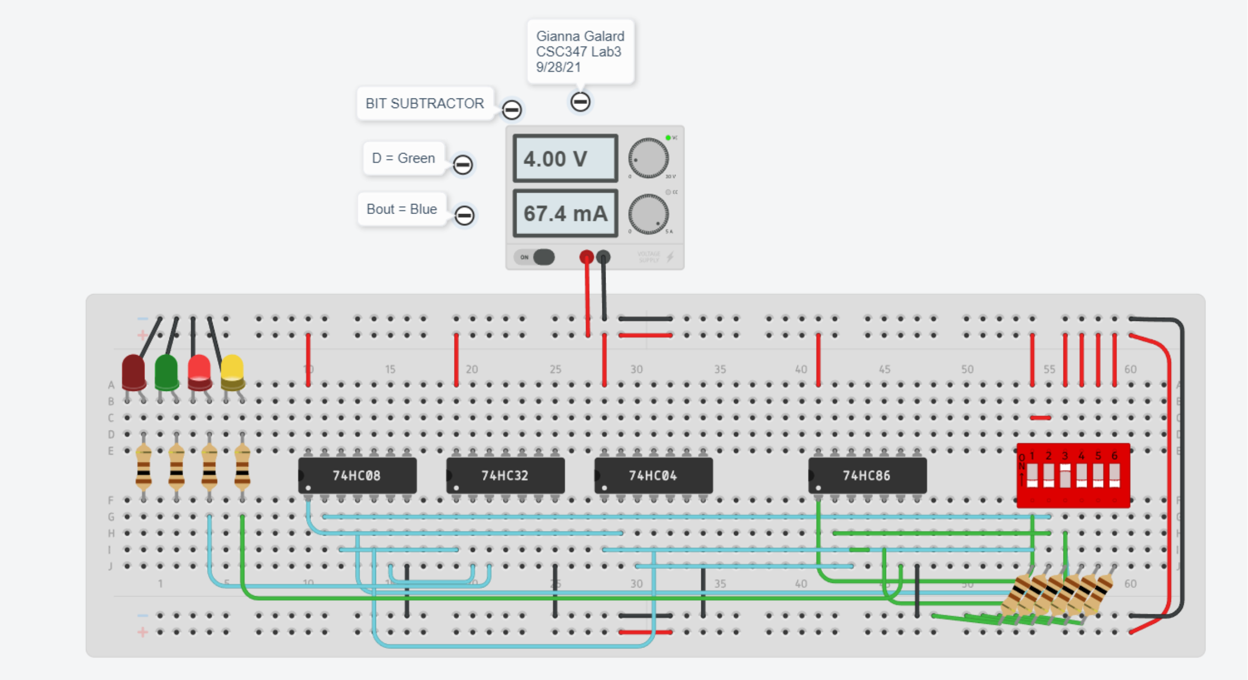


Figure 15. 0 - 1 - 0 = 11.

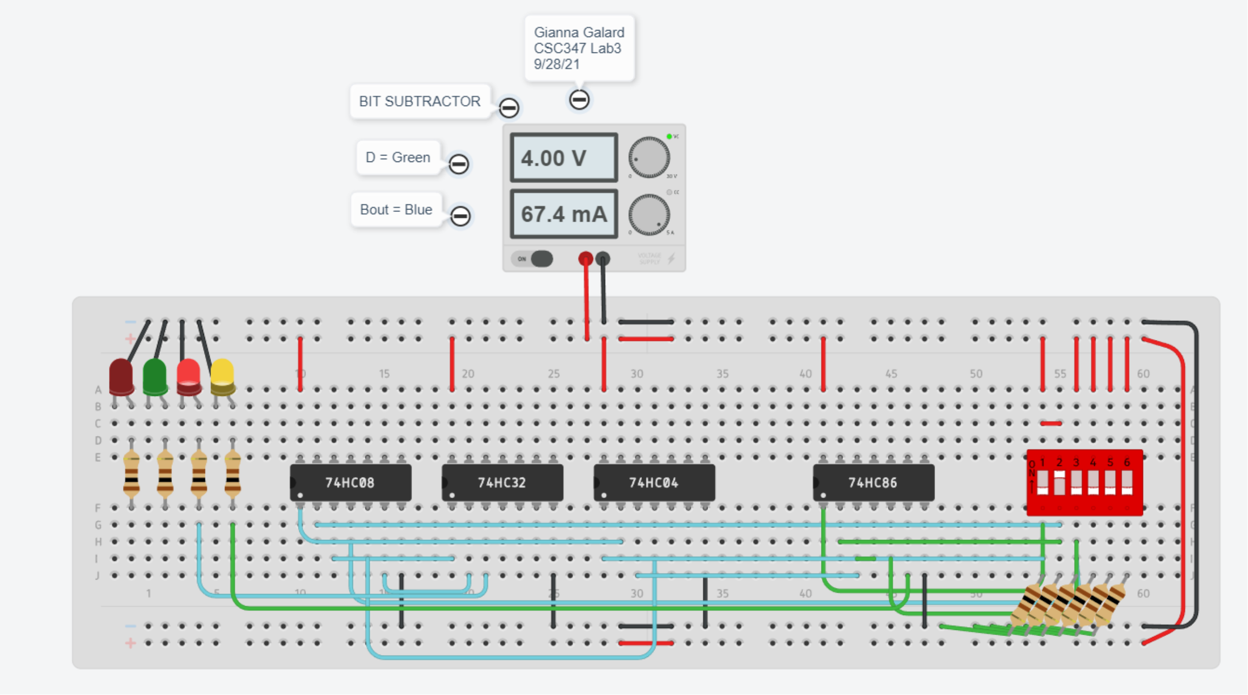


Figure 16. 0 - 1 - 1 = 10.

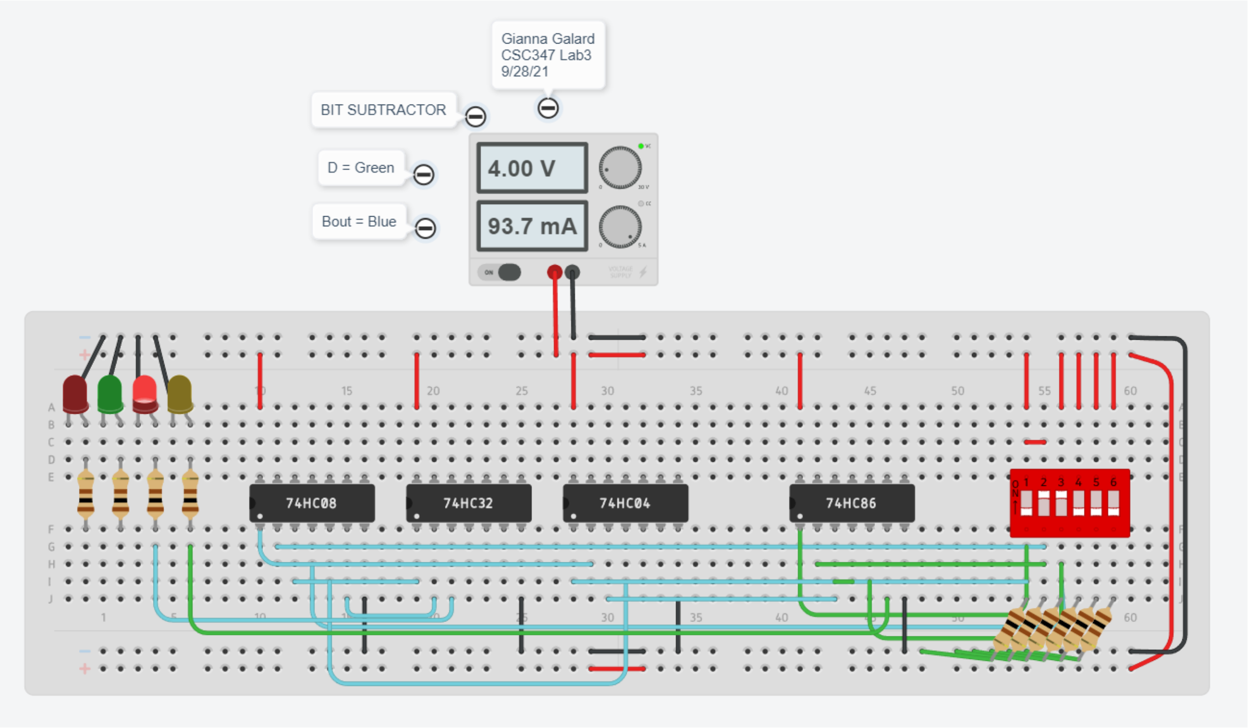


Figure 17. 1 - 0 - 0 = 01.

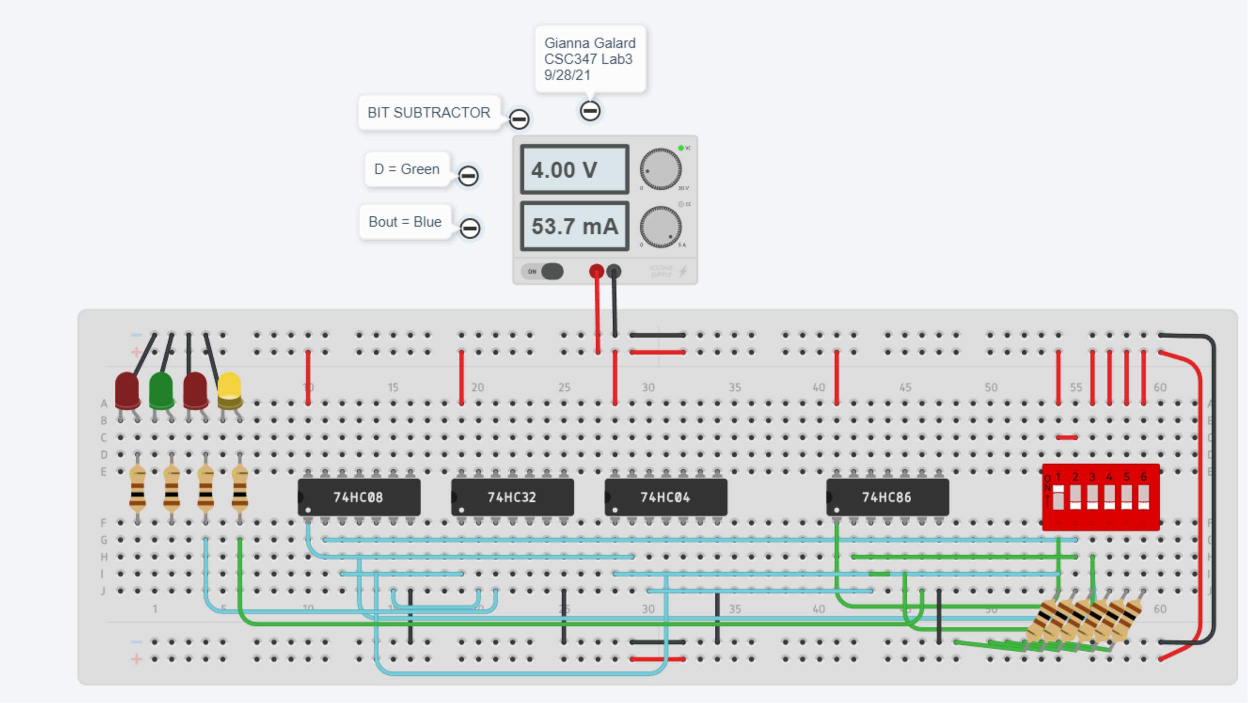


Figure 18. 1 - 0 - 1 = 00.

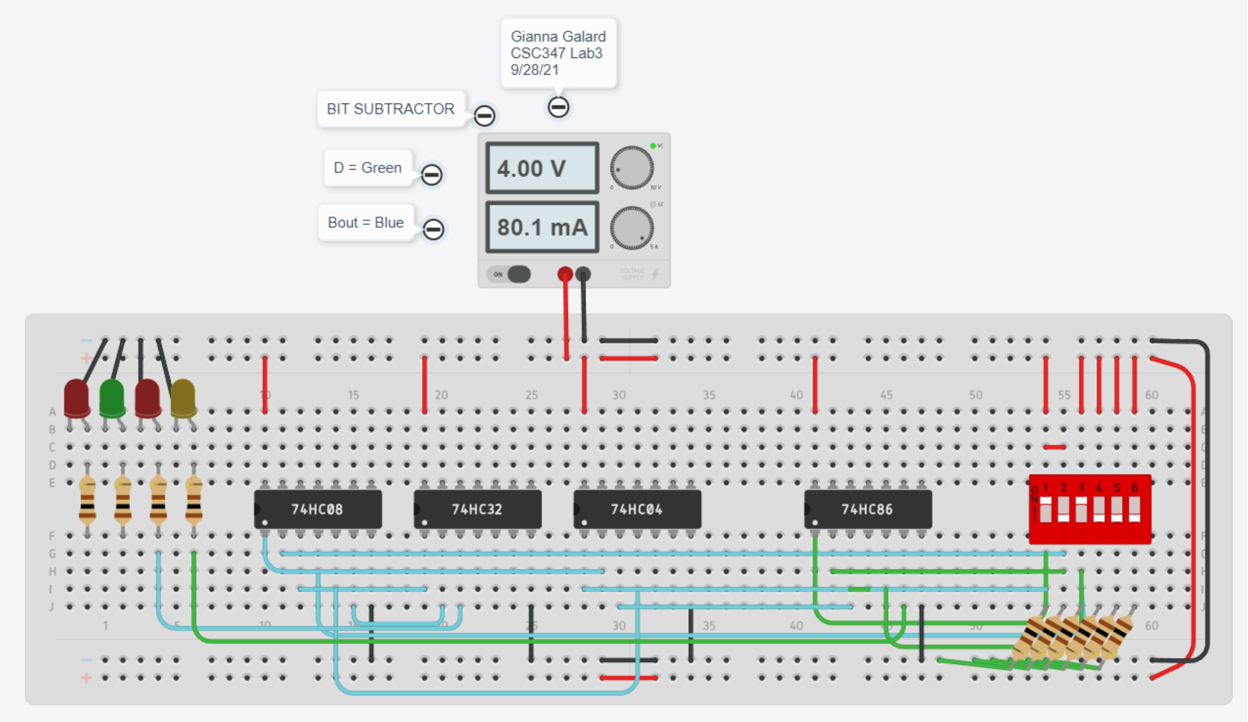


Figure 19. 1 - 1 - 0 = 00.

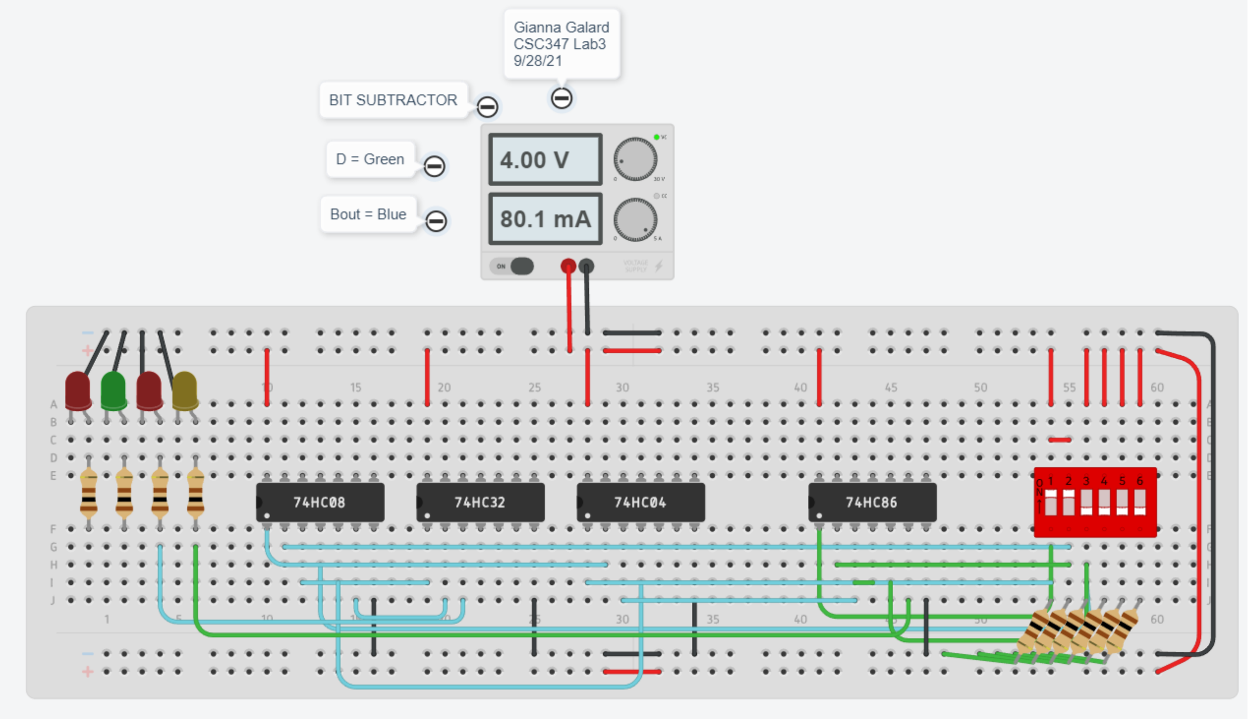
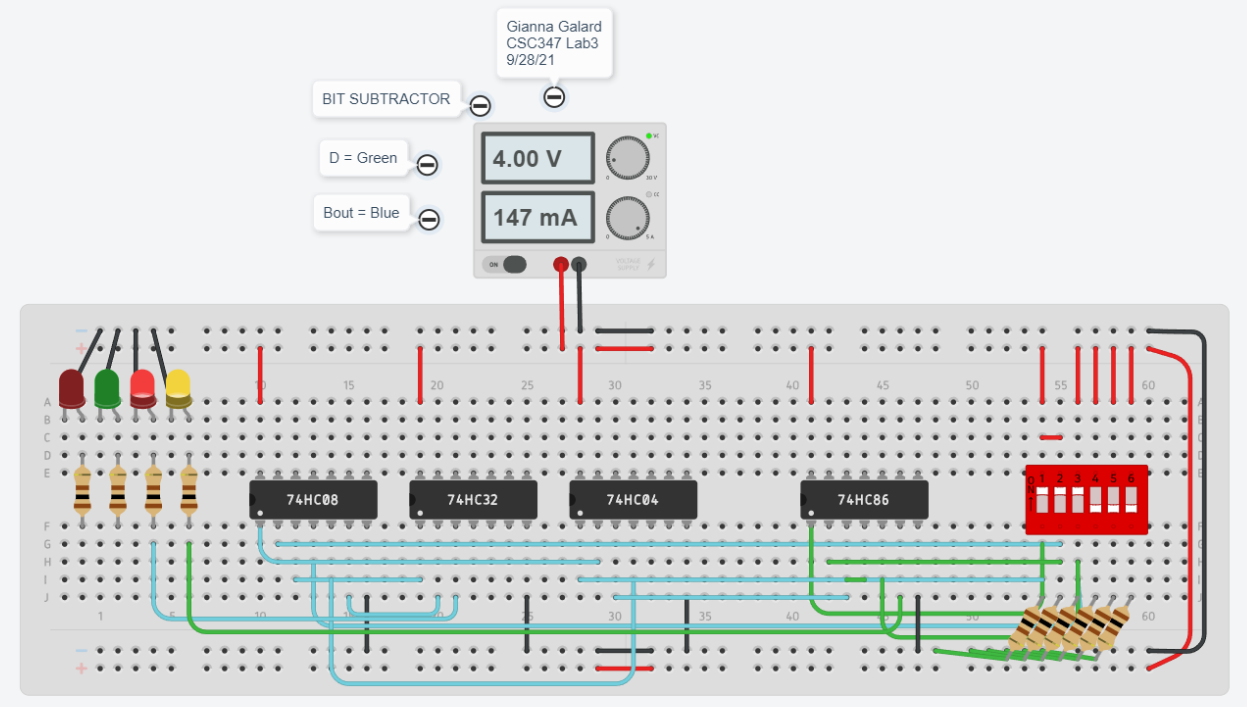


Figure 20. 1 - 1 - 1 = 11.



Resources:

Figure 10 -

https://www.tinkercad.com/things/gBXckLXPbDP-csc347-lab-3-bit-adder/editel?sharecode=oyZmVblK-T2dit9uwlwnLRqQ78dbGgljYun-9uUs6s0

Figure 20 -

https://www.tinkercad.com/things/b7eFjYkmT7w-copy-of-csc-347-starter-kit/editel?sharecode=f5bjOS1_I1fqkGl9U9GmPL43bpEk0d8ZGtq2IAx5kEU