



Binary Calculator Circuit

Final Project Report

Fall 2022

12/2/2022

College of Engineering and Computing

Department of Electrical and Computer Engineering

ECE 304: Electronics

Dr. Scott & Prof. Leonard

Riley Taylor

Kieran Thompson

Table of Contents

Introduction	3
Circuit Theory	4
Circuit Description	4
The Half-Adder	6
The OR Gate	6
The AND Gates	8
The Dependent NPN Transistor	9
Half-Adder Propagation	10
The Full Adder	13
The XOR Gates	14
The AND Gates	15
Full Adder Propagation	16
Project Outcome	19
Simulation Model and Results	20
Experimental Results	23
Analysis of Results	27
Conclusion	28
Appendix	29

Introduction

Our project is a binary calculator that will add two two-bit inputs together and will produce a three-bit output. The circuit demonstrates how a binary calculator can be implemented using elements learned from this class as well as previous classes. It showcases how NPN transistors can be used as various switch types such as AND, OR, and XOR and are the building blocks of the circuit. These logic gate outputs are the key elements to our half and full adder argumentation. The circuit is interactive and the user can choose, by the use of switches, what binary numbers they would like to add and the output will be displayed appropriately. We believe that our classmates will be interested in this circuit as full adder and half adder as well as binary as these concepts are covered in multiple classes.

This report is broken down into 5 other sections. Circuit theory will explain the general information in what our circuit is and how it should propagate/function. It will also talk about the outcomes that it addresses. The simulation section will explain how we modeled the circuit using LTspice and its results. The experimental section will explain our created circuit and the results it gave from certain inputs. After this, the next section will explain the comparisons and differences between our circuit theory, simulation results, and experimental results. Finally, the conclusion will explain our results and restate some of the ideas stated in our introduction.

Circuit Theory

Circuit Description

Our circuit operates at 5V using NPN transistors, LED's, mechanical switches, resistors, capacitors, an integrated circuit containing XOR gates, two breadboards, and jumper wires. The circuit is a binary calculator where there will be two two-bit inputs added together to produce a three-bit output. Our two two-bit inputs are represented using mechanical switches that can control whether each bit is on or off along with an LED to represent if they are on and off and our three-bit output is represented by 3 green LEDs that turn on and off depending on if the bit will be one or zero (in our schematic we don't have mechanical switches).

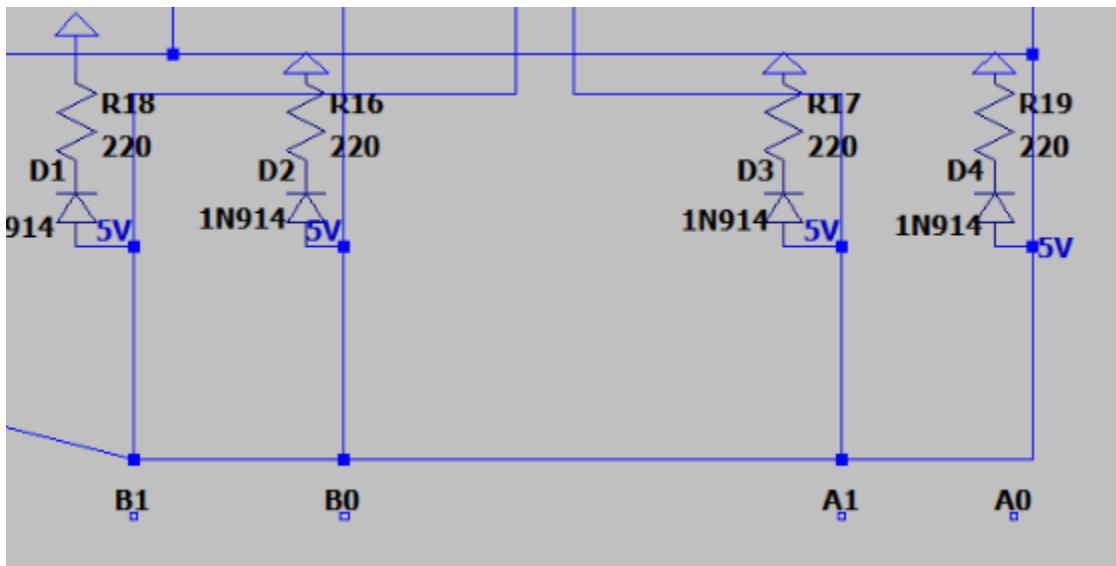


Figure 1 - Circuit Switches

The first part of our circuit represents a half-adder using transistors. The half-adder will take a two-bit input and produce the first bit used in our output and the carry bit which is used in the 2nd part of the circuit. The half-adder is composed of three logic gates which is an XOR gate (made with an OR and AND gate) which will only output an on signal when only one of the two

bits is on, an AND gate where if both bits are on then it will output a signal, and a dependent NPN which will complete the XOR gate. The circuit uses a total of 7 transistors for this.

The 2nd part of our circuit represents a full-adder also using transistors and an integrated circuit for the XOR gates. The full adder will take a three-bit input which includes the most significant bit for each of the two-bit inputs and the carry bit we got from the half adder. A full adder is usually composed of 5 logic gates which are 2 XOR gates, two OR gates, and one AND gate where the XOR gates are used for the 2nd output bit and the OR plus AND gates are used for the carry bit or the 3rd-bit output. We currently have it set up where we use transistors to represent the AND gates, wires joined together to represent the OR gate, and the integrated circuit for the XOR gates although in our schematic we didn't have access to a component of the IC so we just used the XOR gate component to represent it.

The Half-Adder

In this section we will break down the half-adder and the logic that implements our calculator.

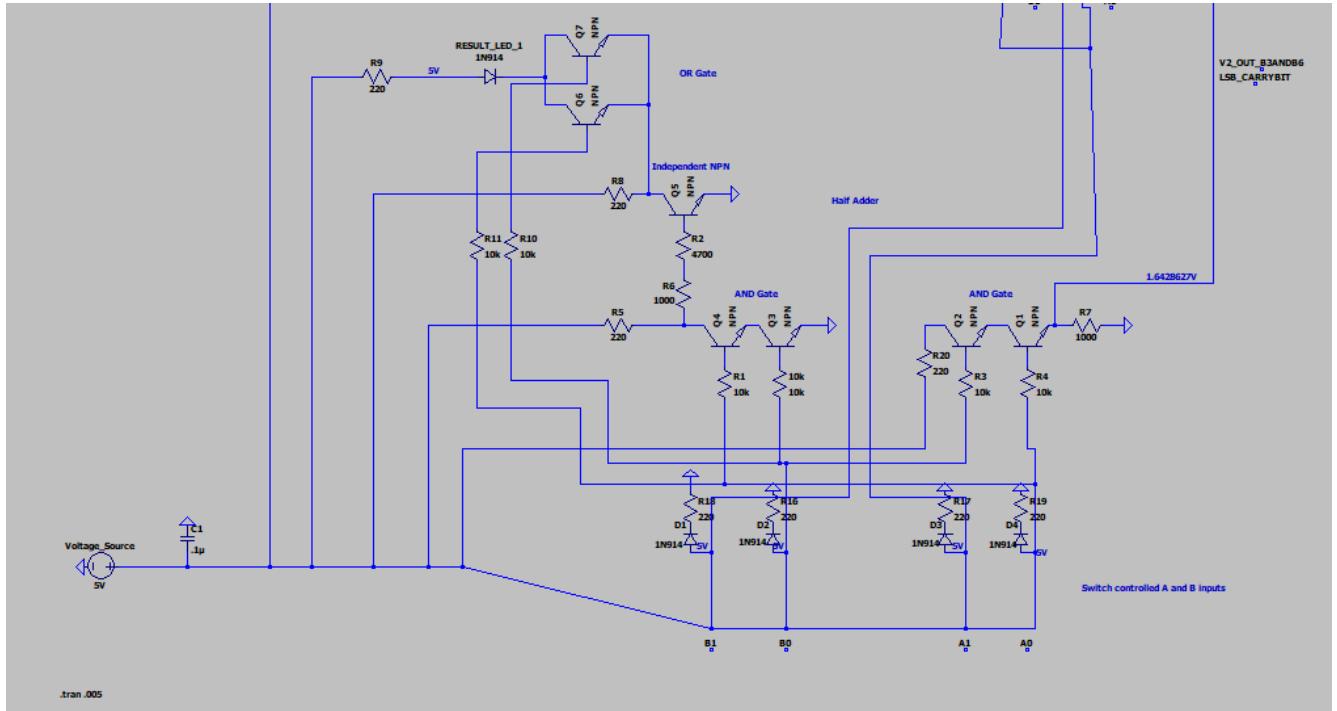


Figure 2 - Half-Adder schematic

The OR Gate

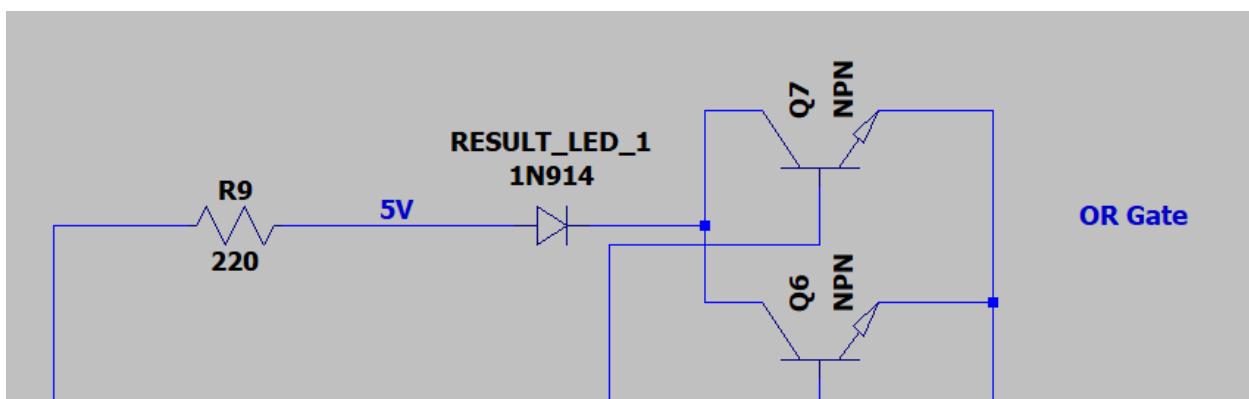


Figure 3 - OR gate representation in schematic

The OR gate allows us to use an LED as the least significant bit in the sum. If the dependent NPN transistor is on, then the current will flow to the ground connected to that

transistor. If both switches are on, then the current will not flow along this path as we will show in the AND gate logic. We are also assuming for now that the transistors have little to no impedance across them. In order for the LED to work safely, it must operate under a 23mA current. In order to achieve this, we used a 220Ω resistor.

$$V = iR$$

$$5V = i * 220\Omega$$

$$i = 22.7mA < 30mA$$

The AND Gates

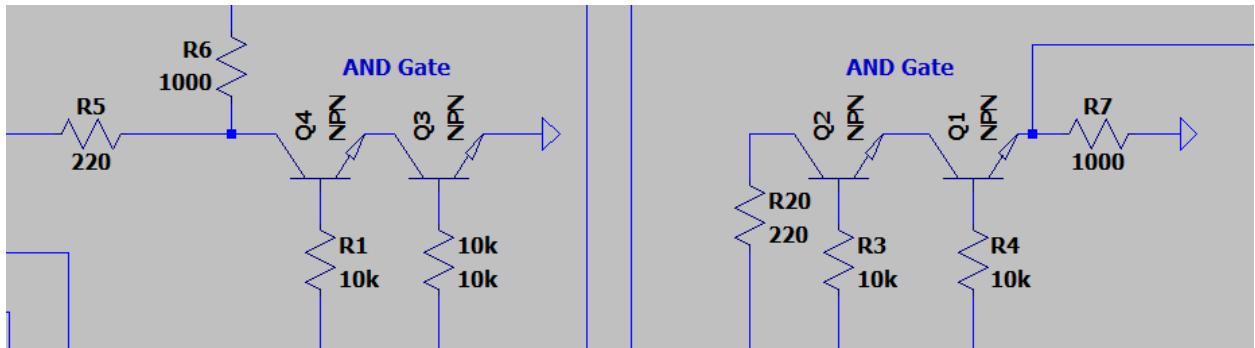


Figure 4 - AND Gate representation in schematic

We used two different AND gates for the half-adder. The first AND gate (shown on the left side) is used in the XOR logic while the other AND gate (shown on the right side) is used for the carry bit. The AND gates are independent of each other, this is because the second AND gate only depends on itself, while the other AND gate depends on the logic in the XOR. The least significant carry bit will be used in the full-adder. The LED follows the same guidelines as the LED in the OR gate in order to operate.

The first AND gate current will flow to the ground if both switches are logic 1, otherwise it will flow upwards into the dependent NPN transistor. Each base will have a $10k\Omega$ resistor, this is to limit as much current as possible to the base in order for the LEDs to properly operate. Assume that if both switches are off then there is no flow in the circuit.

The Dependent NPN Transistor

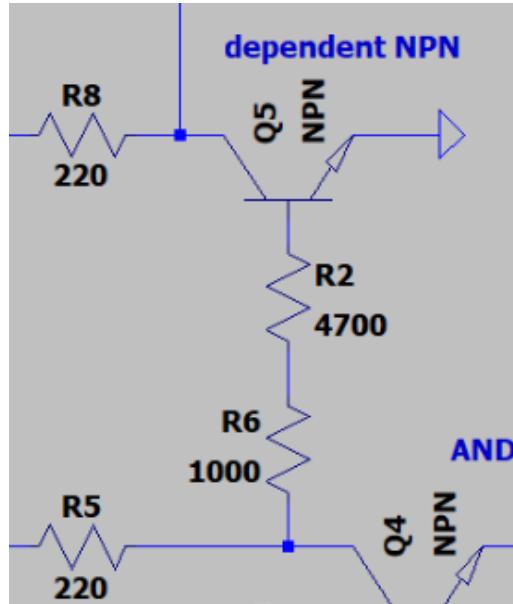


Figure 5 - Dependent NPN Transistor representation in schematic

The dependent NPN transistor is used as our final component in our XOR gate. It connects both the AND gate and the OR gate. Its purpose is to allow the sum LED to path to ground if it has exclusive or logic, otherwise, it will not flow to ground. The major difference between this transistor and the others is that it has a 1000Ω resistor in series with a 4700Ω resistor, this is because we had a dimming problem with the sum bit LED, and to fix this issue we dropped the resistance from $10k\Omega$ to 5700Ω , allowing the LED to receive more current.

Half-Adder Propagation

In order to better understand the propagation of our half-adder, we will show how binary logic is applied with switches.

The following image is switch A is logic 1 and switch B is logic 0. Assume the green path is the flow the current will take and the red x's are blocked paths. The result is the sum bit LED is on, and the carry bit LED is off.

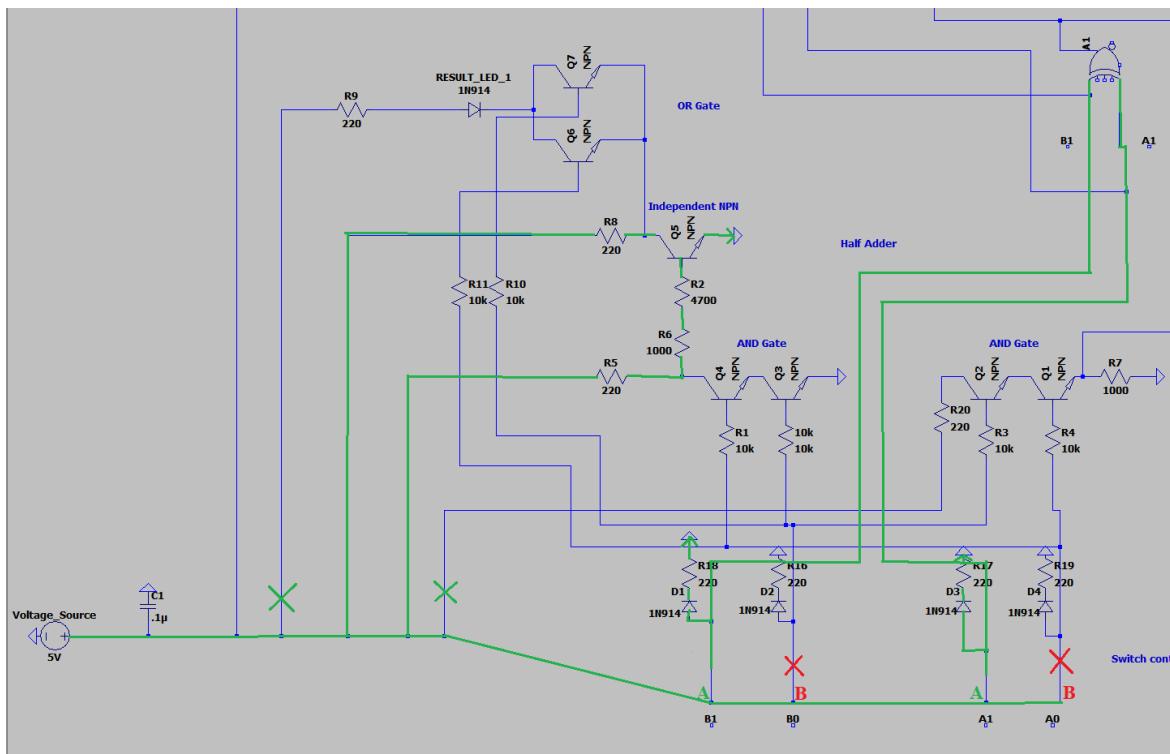


Figure 6 - Example Propagation for the Half-Adder

The following image is switch A is logic 0 and switch B is logic 1. Assume the redpath is the flow the current will take and the green x's are blocked paths. The result is the sum bit LED is on and the carry bit LED is off.

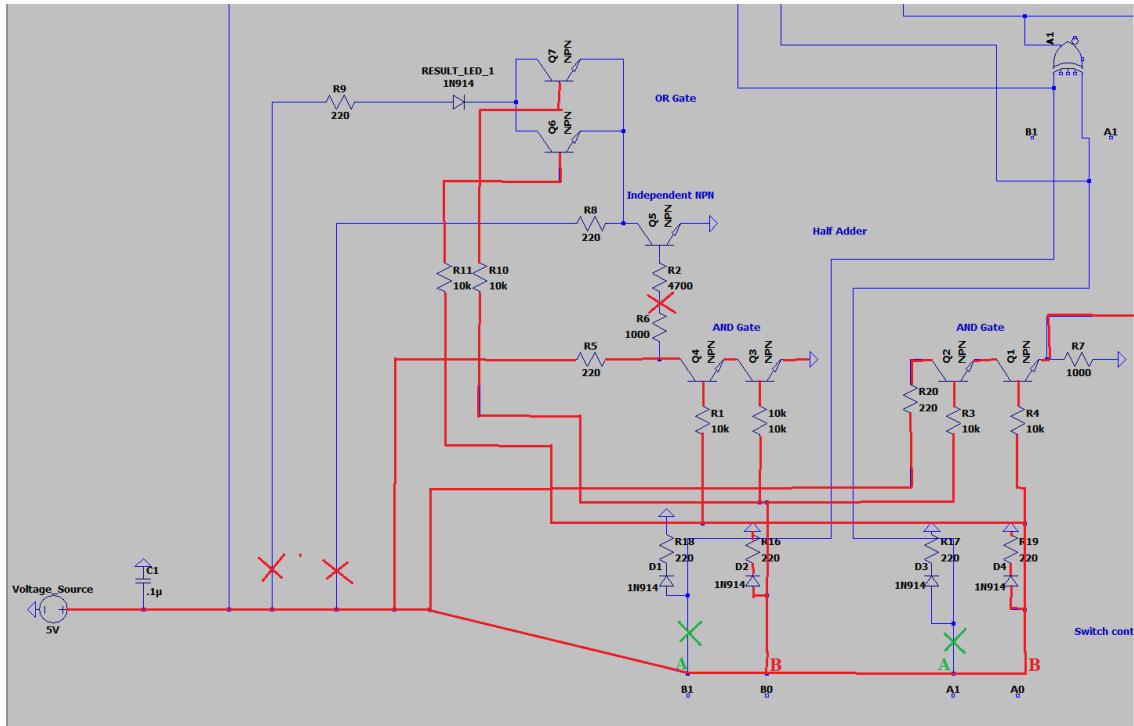


Figure 7 - Example Propagation for the Half-Adder

The last image for the half-adder is switch A is logic 1 and switch B is logic 1. This means that all both switches are open. The blue line will show the resulting current in the circuit and the blue x's will show blocked paths.

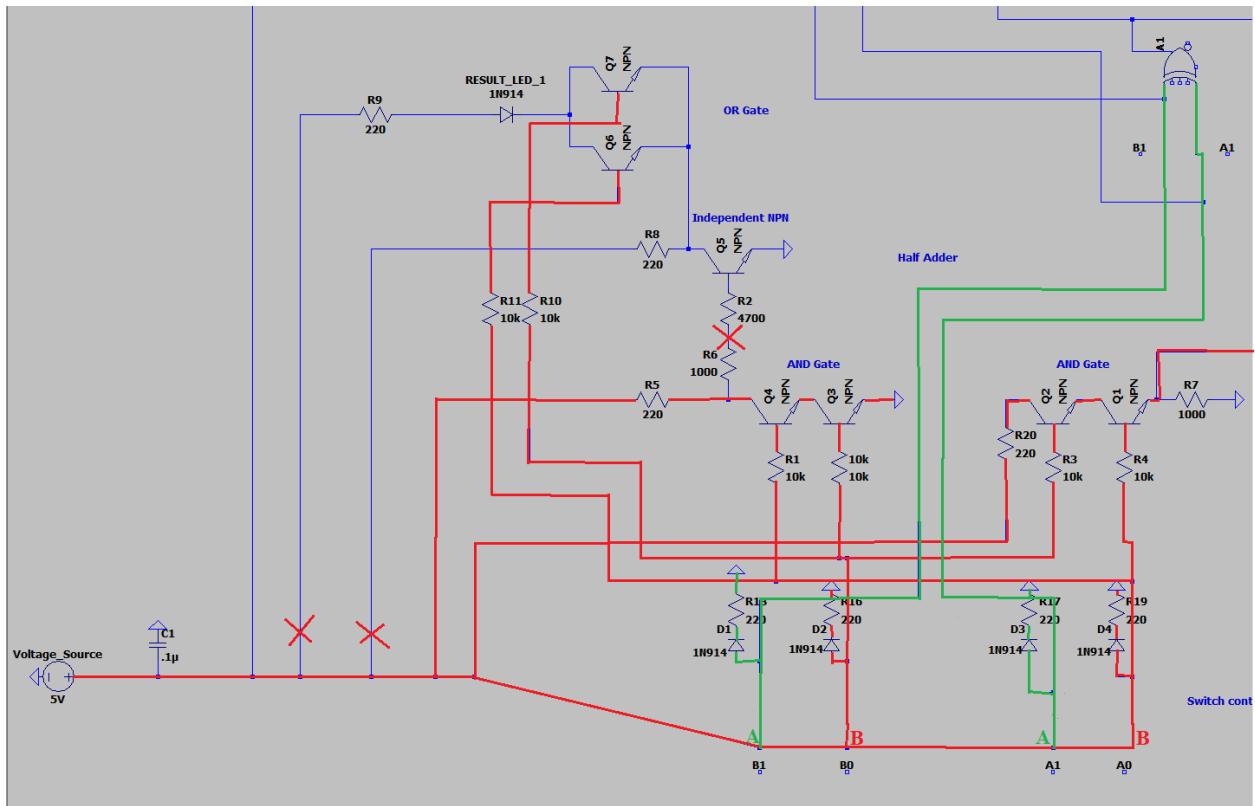


Figure 8 - Example Propagation for the Half-Adder

The Full Adder

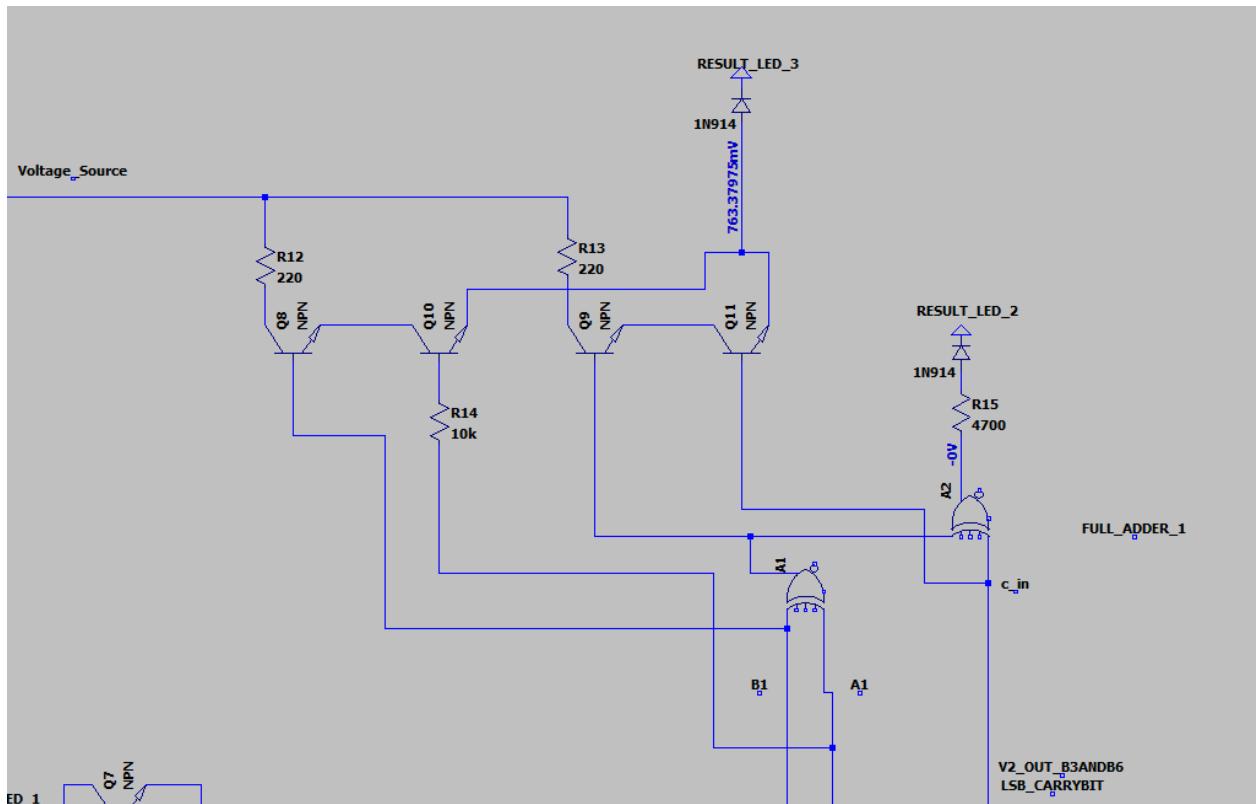


Figure 9 - Full-Adder schematic

In this section we will break down the full adder and its components.

The XOR Gates

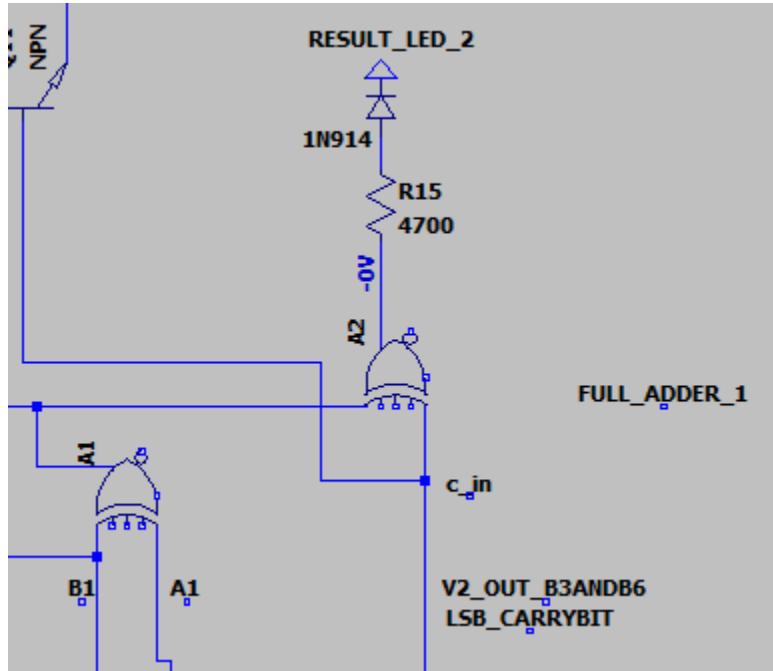


Figure 10 - Full-Adder XOR gates

The XOR gates are used for calculating bit 2. We used an IC for the XOR gates but for this schematic, we represented them with logic gates. The XOR gate on the left determines if only one of the input bits is logic 1 and if not then it outputs a logic 0 which will be used for the 2nd bit as well as the 3rd bit. The XOR gate on the right receives the input from the XOR gate on the left as well as the carry bit and decides if only one is logic 1 then it outputs a logic 0 which goes to the LED representing the 2nd sum bit. To summarize, the output bit will only have logic 1 when either the carry bit from half-adder is logic 1 and the input bits are logic 0, only one of the input bits is logic 1, or when all 3 bits are logic 1.

The AND Gates

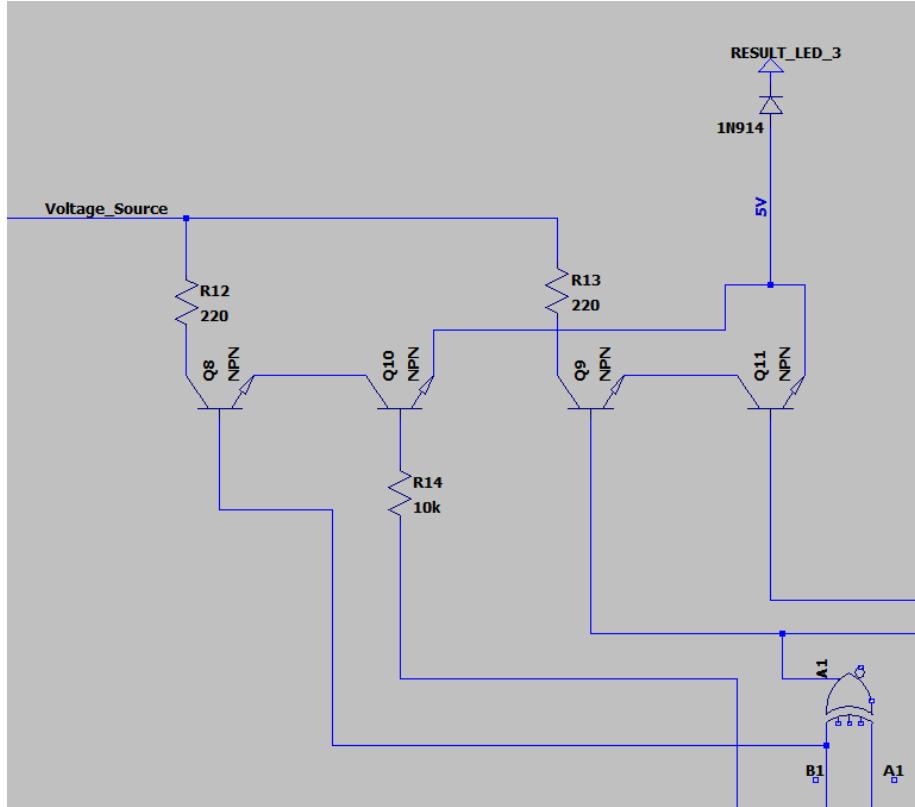


Figure 11 - Full-Adder AND gates

The AND gates as well as the output from the XOR gate explained previously are used for calculating bit 3 or the most significant bit. The first AND gate on the left represented with the first two transistors has the most significant input bits going into it where if both are 1 then the AND gate outputs a logic 1. The 2nd AND gate on the right represented with the last two transistors is used for the carry bit and the output from the XOR gate mentioned previously where if the XOR gate outputs a logic 1 (only one of the input bits is logic 1) and the carry bit is also logic 1 then it outputs a logic 1. We have both of these outputs from the AND gate joined together to output a current for the LED almost like an OR gate. For both cases, if either AND gate outputs a logic 1 then the output at the top will be logic 1 and the LED representing the 3rd

sum bit will turn on. To summarize, if either two or three bits including the carry bit are logic 1 then the third bit will be logic 1.

Full Adder Propagation

To give a better understanding of the circuit I added a couple of scenarios for the circuit showing how the current should flow if certain bits are logic 1.

The first image shown below represents when only one of the input bits is on where the green line shows the current path and the purple X is where the current will stop. As you can see when only one input is logic 1 then only bit 2 will turn on.

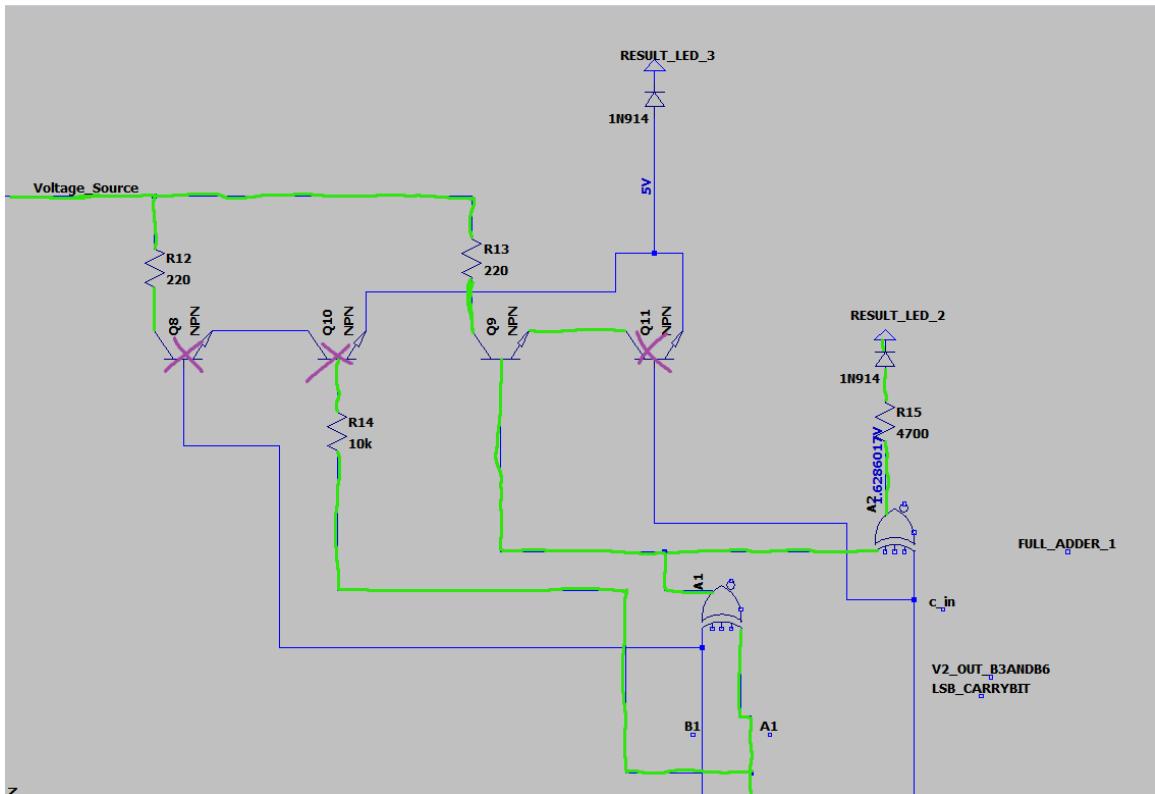


Figure 12 - Full-Adder propagation

The second image shown below is if the carry bit is on and one of the input bits is on where the green line shows the current path for the input bit and the carry bit, and the purple X for where the current will stop. As you can see when two inputs are logic 1 then only bit 3 is on.

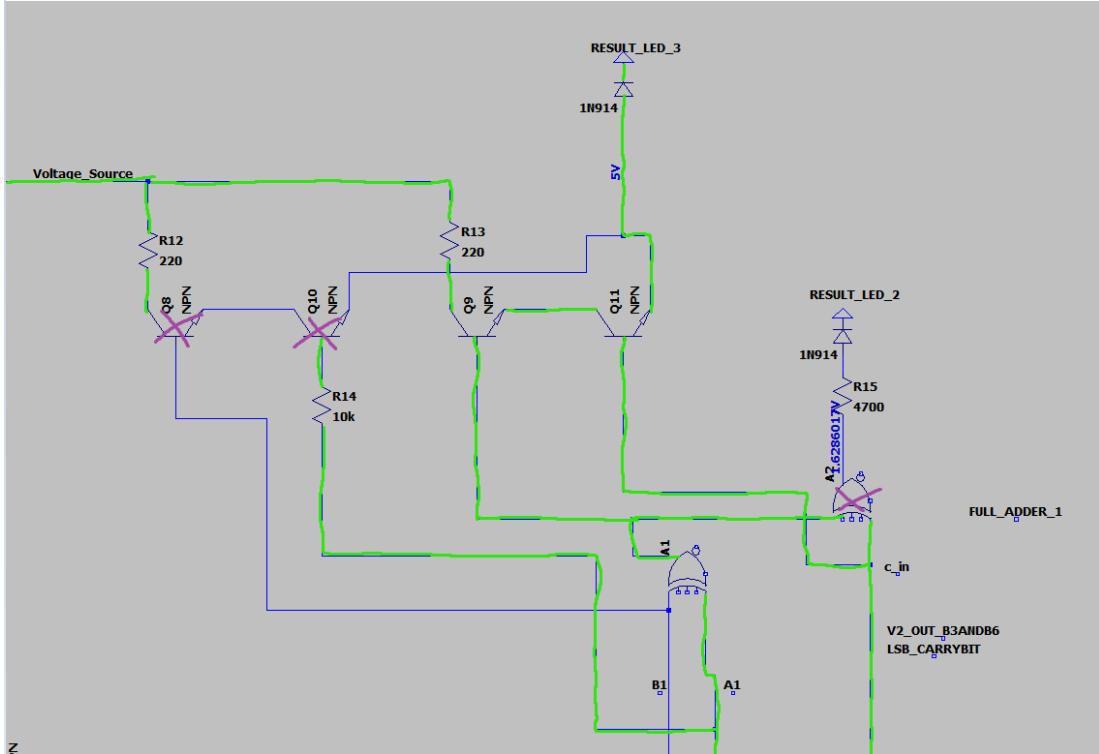


Figure 13 - Full-Adder propagation

The third image shown below is if all 3 inputs are on where the green path is for input bit 1, input bit 2, and input bit 3 while the purple X's are where the current stops. As you can see when all 3 inputs are logic 1 then bits two and three will be on. As you can see when all the 3 bits are logic 1 then bits 2 and 3 are also logic 1.

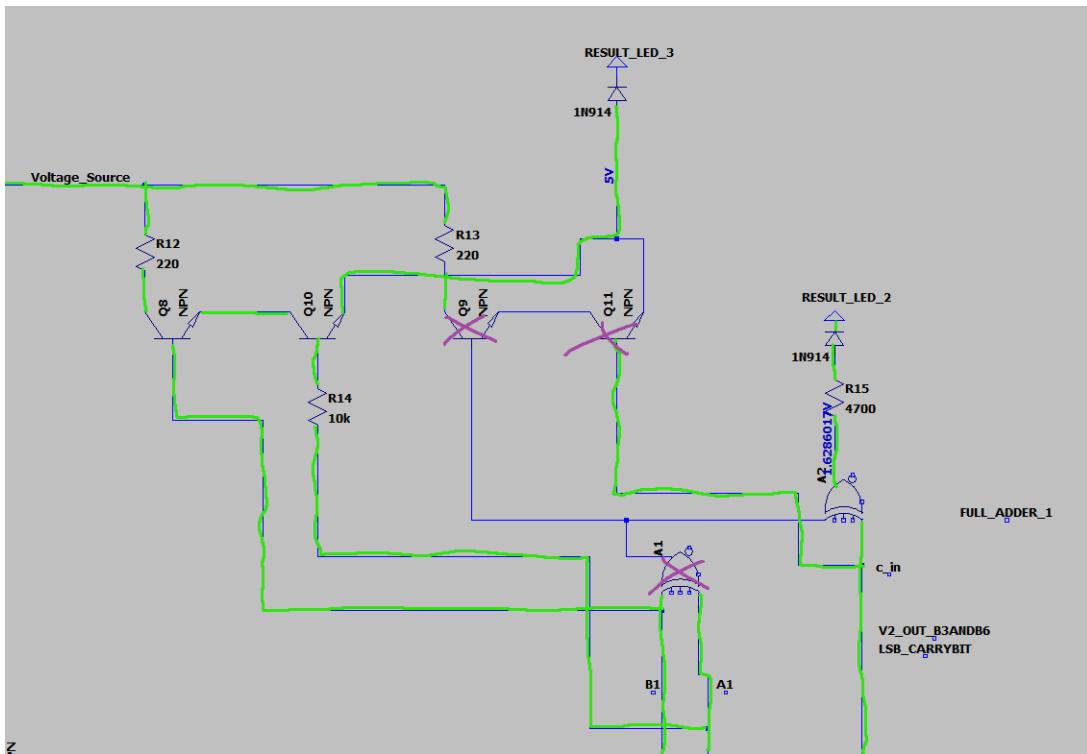


Figure 14 - Full-Adder propagation

Project Outcome

Our project looks at the course objective of having a good understanding of the basic electronic devices from giving an understanding of how logic gates are utilized/set up in most modern technology as well as understanding binary in circuits. This project also addresses the ABET learning outcome of being able to work effectively with a team. With this project we are working in a group where we each have assigned roles while also communicating with each other and completing certain tasks together.

Our project may not address public issues directly but we believe that this circuit can help with public economics. Our circuit is basically a simple calculator that could benefit every common person especially when it comes to their personal finances. The circuit can help people complete their taxes or solve personal expenditure issues. Overall, with the circuit representing a calculator it can be used in a lot of aspects of daily life but we believe it benefits the most in the area of economics.

We believe that this project will be very helpful for freshmen college students as binary is a very important concept for almost any ECE student so being able to physically see it and logically understand it in a circuit will give students a better understanding of most general electronics and how they implement numbers in the system. This project is also helpful as it gives an understanding of logic gates and how they are used/represented in a circuit which is an important concept to understand if you are an ECE major.

Simulation Model and Results

We used LTSpice for our simulation. In order for the simulation to work, a0, a1, b0, b1 wires which represent the switches for the bits that will need to be cut and replaced (on the bottom of the simulation), depending on the inputs that will be simulated. In the following image, you can see that a0 is cut while the other three wires are connected, this will allow the simulation to act as 11 + 10.

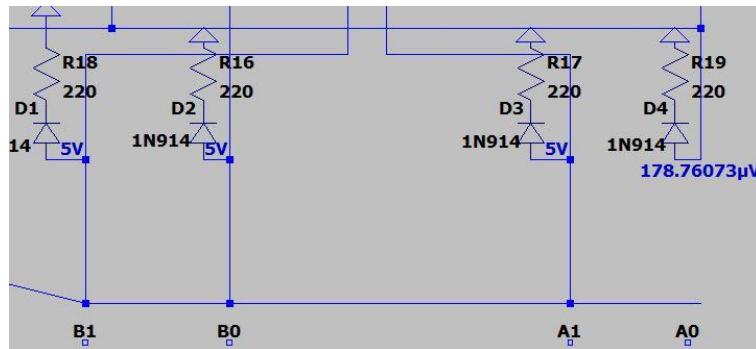


Figure 15 - Circuit Switches, A0 off

We then measured the voltage at each LED to see if it's on (logic 1) or off (logic 0) based on the defined inputs. We determined that lower threshold readings from the result LEDs would indicate an off reading while higher threshold readings would indicate an on reading. There is a slight error in some of the readings as explained below.

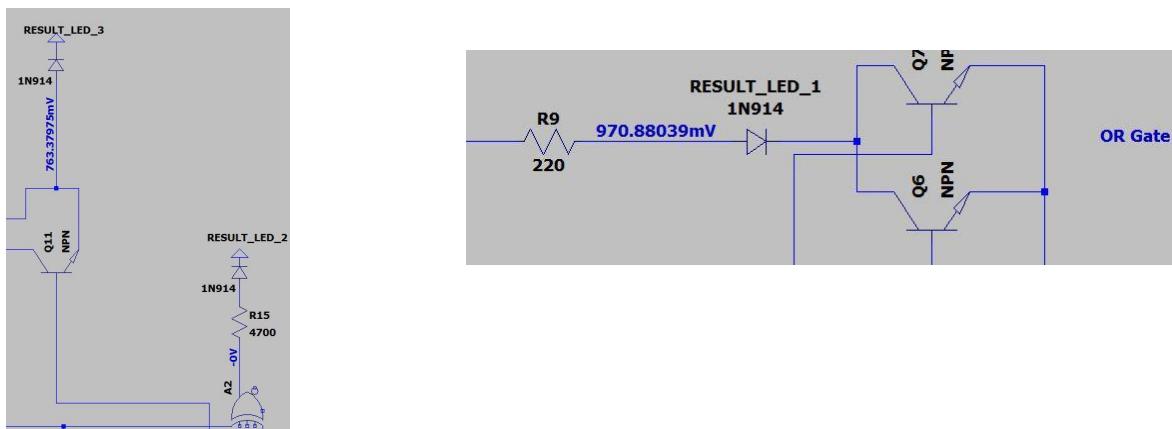


Figure 16 and 17 - Result LED simulation

Overall the simulation will not be exact because our actual circuit utilizes an IC for two XOR gates.

b1	b0	a1	a0		MSB (V)	Mid B (V)	LSB (V)
0	0	0	0		10p (0)	0 (0)	5 ₁ (0)
0	0	0	1		2m (0)	0 (0)	970m (1)
0	0	1	0		539m (0)	1 (1)	5 ₁ (0)
0	0	1	1		539m (0)	1 (1)	970m (1)
0	1	0	0		10p (0)	0 (0)	970m (1)
0	1	0	1		709m ₂ (0)	1 (1)	5 (1)
0	1	1	0		539m (0)	1 (1)	970m (1)
0	1	1	1		710m (1)	0 (0)	5 ₁ (0)
1	0	0	0		5.1n (0)	1 (1)	5 ₁ (0)
1	0	0	1		4m (0)	1 (1)	970m (1)
1	0	1	0		763m (1)	0 (0)	5 ₁ (0)
1	0	1	1		763m (1)	0 (0)	970m (1)
1	1	0	0		5.1n (0)	1 (1)	970m (1)
1	1	0	1		709m(1)	0(0)	5(1)
1	1	1	0		763m(1)	0 (0)	970m(1)
1	1	1	1		785m (1)	1(1)	5 ₁ (0)

1. LSB for 0,0,0,0 case is 5V because the simulation has transistors acting like very high resistance. In demonstration the LED will be off because it will be open circuit.
2. There will be slight errors with our simulation because our actual circuit uses an IC chip while the simulation uses logic gates provided in the software. The XOR gate in LTSpice is not correctly outputting 0 when it should during some of the simulations.

Since our simulation is not an exact replica of the physical circuit, we have run into problems with very specific cases. The first case is when the circuit is at rest, the LSB should be off but in the simulation it is receiving 5V. This is a contradiction because in demonstration the

LED will be an open circuit. The second case is that we could not implement our exact IC into LTSpice, which ultimately will result in some propagation error in the final result. We also ran into a problem with the switches, the default switches given in the software acted as closed switches regardless of input, so we decided to use the wire cutting method instead. Regardless of these issues, our simulation still produced somewhat accurate results to what we expected.

Experimental Results

The figure below shows the entire circuit. The binary inputs are controlled using mechanical switches with the green LEDs showing if the bit is on or off. The transistors on the front breadboard represent the half adder while transistors and IC of XOR gates on the back breadboard represent the full adder. There are three LEDs to represent the output bits with the red LED being the MSB and the two blue LEDs being the second bit and the LSB.

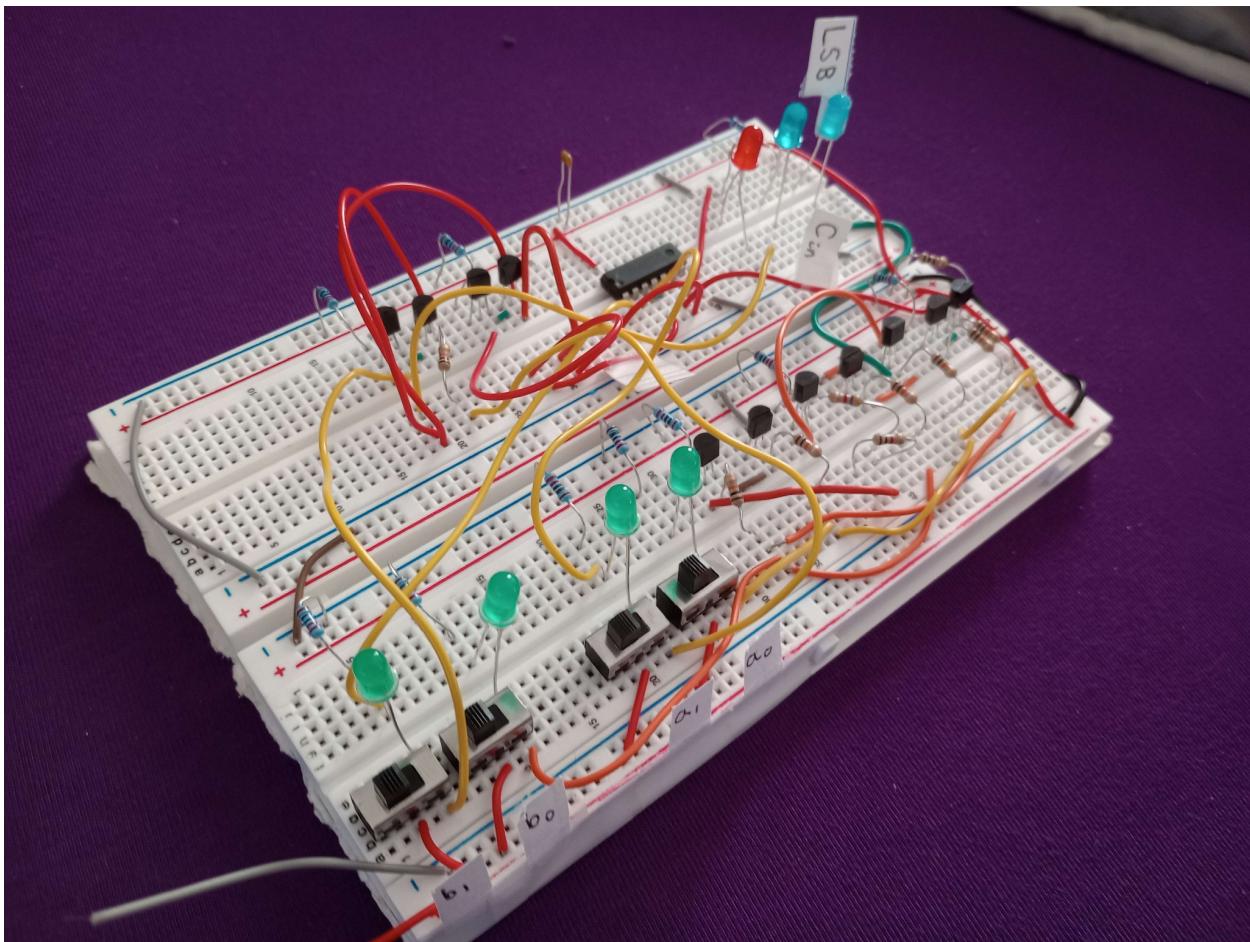


Figure 18 - Full complete circuit at rest

For our experiment, we expected that when the certain binary inputs were on/off that the corresponding binary outputs would be on/off. For example if the 1st and 3rd switches are

switched on then 1st and 3rd green LED should be on and with that representing the binary addition of $10 + 10 = 100$ then only the red output LED should be on. The truth table below shows all of our expected results based on the inputs (0 being the LED should be off and 1 being the LED should be on).

b1 (Far left green LED)	b0 (2nd green LED from left)	a1 (3rd green LED from left)	a0 (Far right green LED)		MSB (Red LED)	Second Output Bit (Blue LED in middle)	LSB (Blue LED on the right)
0	0	0	0		0	0	0
0	0	0	1		0	0	1
0	0	1	0		0	1	0
0	0	1	1		0	1	1
0	1	0	0		0	0	1
0	1	0	1		0	1	0
0	1	1	0		0	1	1
0	1	1	1		1	0	0
1	0	0	0		0	1	0
1	0	0	1		0	1	1
1	0	1	0		1	0	0
1	0	1	1		1	0	1
1	1	0	0		0	1	1
1	1	0	1		1	0	0
1	1	1	0		1	0	1
1	1	1	1		1	1	0

Our actual results from the experiment proved successful. We tested each possible input and got the expected output shown in the figure above. We have included a couple images below to show examples of some of our tests. Figure 19 is the binary input of $10 + 10$ which gives an output of 100, figure 20 is the binary input of $11 + 11$ which gives an output of 110, and figure 21 is the binary input of $00 + 11$ which gives an output of 011.

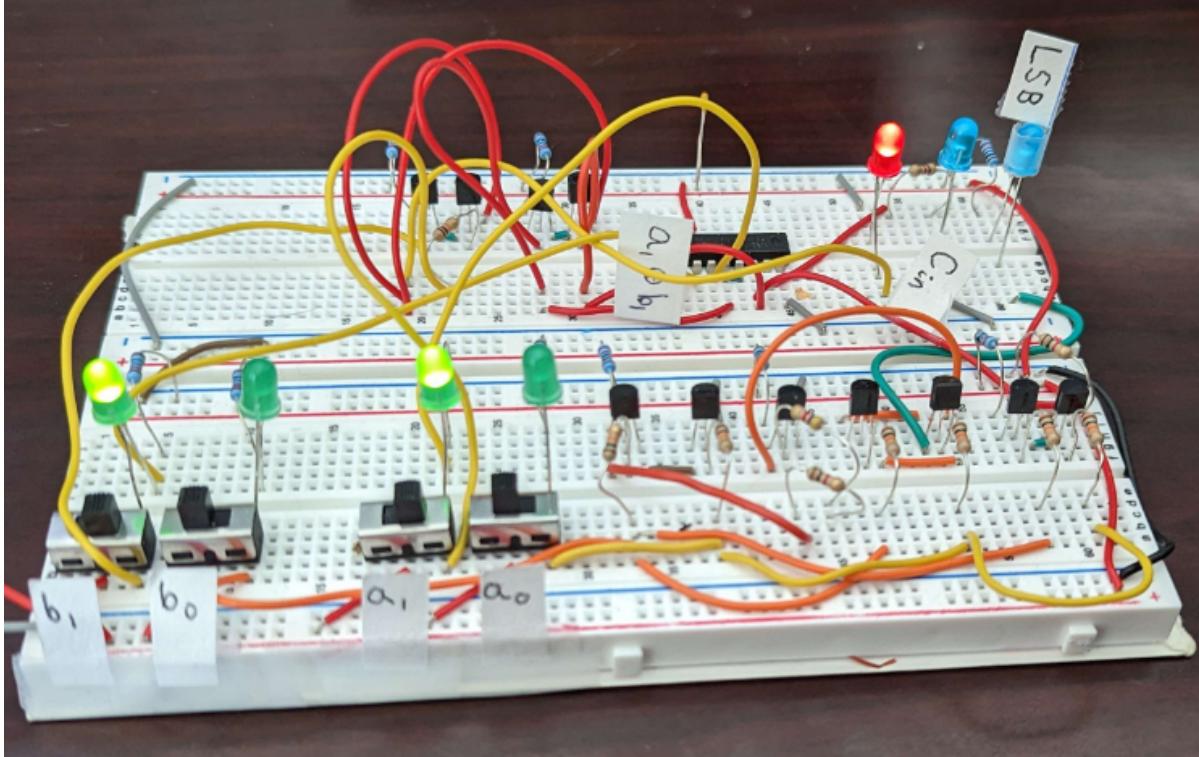


Figure 19 - Circuit result: $10 + 10 = 100$

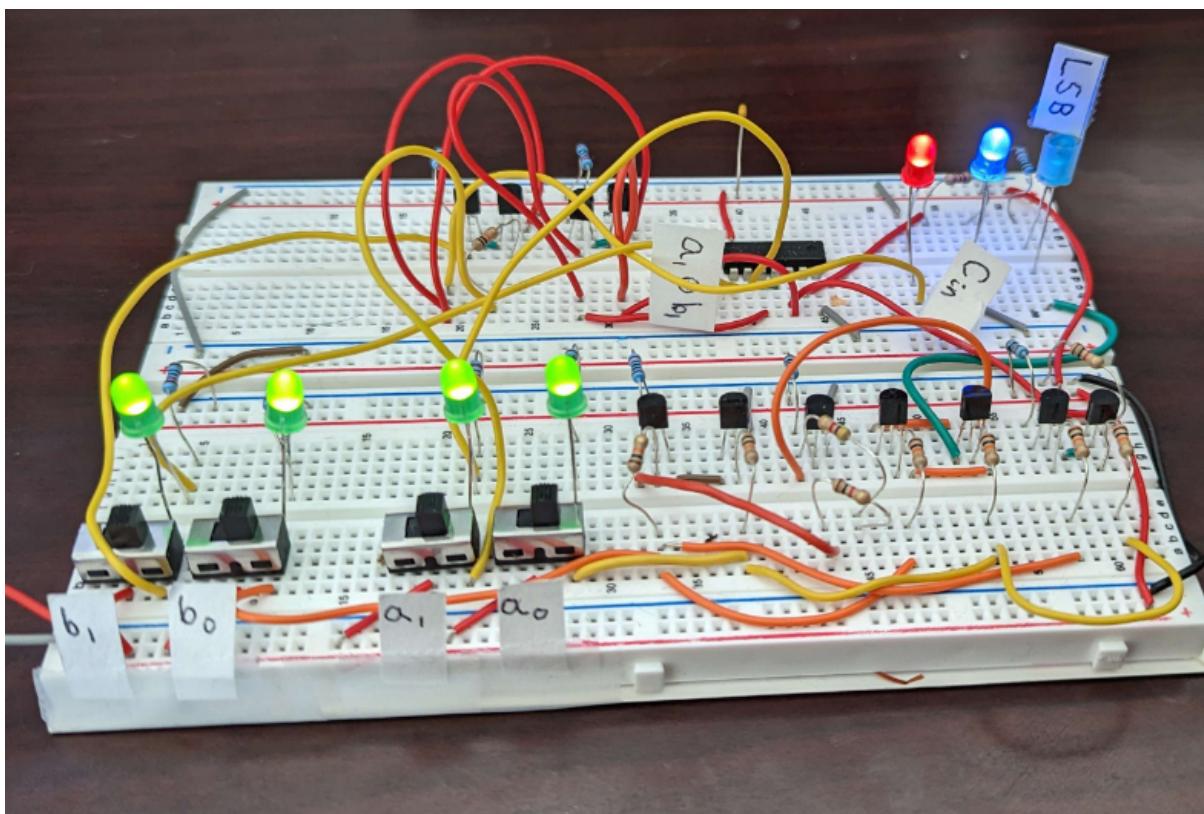


Figure 20 - Circuit result: $11 + 11 = 110$

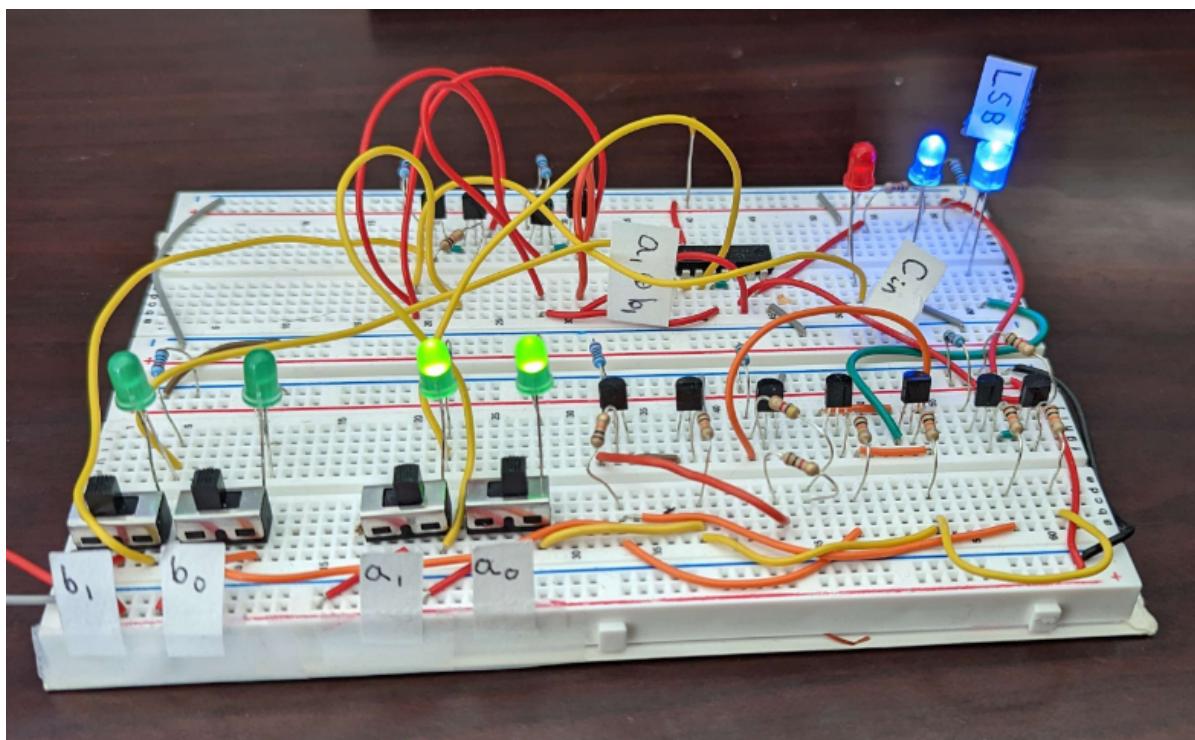


Figure 21 - Circuit result: $00 + 11 = 011$

Analysis of Results

Based on our results, the circuit seemed to be successful. The simulation results were the same as the experimental results except for a couple differences as the simulation had the wrong output for a couple of the inputs. Our experimental results also coincide with our circuit theory since our experimental results need to have the corresponding paths shown in the figures for the circuit theory section in order for the circuit to function properly. The images above also prove functionality of the circuit which is concrete evidence that our logic is correct.

Conclusion

Overall, our binary adder which would take two 2 bit inputs and produce a 3 bit output was successful. Every possible binary input would produce the correct binary output. Our hope for this circuit is to help show high school or college students what binary is in the terms of electronics and how it is used like in our calculators as it is useful general knowledge for any student and for any student interested in electronics it's important to know. This project also allowed us to interpret and apply information from data sheets on different components that were used in the circuit. We also had a side goal of gaining experience with creating/working with logic gates using transistors which will be useful professional experience for when we graduate.

Appendix

Components Used	# of them	Operating Voltage/Current
SN74LS86N (XOR IC)	2	Min 5V (2V for High)
2N2222A TO-92 NPN Transistor	11	Max 800 mA base (600 mA collector)
CYT1107 Mechanical Switch	4	Max 2A
Arduino LEDs (red, blue, and green)	7	Max 23mA
0.1uF capacitor	1	-
220Ω resistors	10	-
1000Ω resistors	2	-
4700Ω resistors	2	-
10kΩ resistors	7	-
Arduino UNO	1	5V DC
AUSTOR 560 jumper wire kit	1	-

ECE304 Component operation guide				
		Components Used	# of them	Operating Voltage/Current
	SN74LS86N (XOR IC)		2	Min 5V (2V for High)
	SN74HC32N (OR IC)		4	Min 5V (1.5V for High)
	2N2222A TO-92 NPN Transistor		11	Max 800 mA base (600 mA collector)
	CYT1107 Mechanical		4	Max 2A

		Switch		
		Arduino LEDs (red, blue, and green)	7	Max 23mA
		0.1uF capacitor	1	-
		220Ω resistors	10	-
		1000Ω resistors	2	-
		4700Ω resistors	2	-
		10kΩ resistors	7	-
		Arduino UNO	1	5V DC
		AUSTOR 560 jumper wire kit	1	-