Final Report

Brendan Blacklock, Undergraduate – Computer Engineering (Primary)

Jacob Holyoak, Undergraduate – Computer Engineering

Josiah Ramirez, Undergraduate – Electrical Engineering

Texas Tech University

May 2021

Abstract

This design implements the L298 H-bridge and Basys 3 FPGA for control of the Rover 5 motors. The motors move both directions with variable speed and shut off when current is over 1 A. The Rover 5 avoids walls using the Sharp GPZY0A60SZLF, an infrared distance measuring sensor, and the JXADC PMOD of the Basys 3; using the TCS3200 color sensor, the rover detects and visually indicates red, green, and blue. Command communication with the Rover 5 is accomplished with a strobe beacon flashlight. The beacon flashes the morse code commands go, pause, resume, and stop to an OP593 phototransistor.

Table of Contents

List of Figures

List of Tables

1. Design Introduction

The Rover 5 is currently utilizing the programmable Basys 3 FPGA for motor control, overcurrent protection, wall avoidance, color sensing, and morse code command communication. Motor control is accomplished using the L298 H-bridge; the motors move both directions and deploy pulse width modulation to achieve variable speed. Overcurrent protection uses a LM339 comparator circuit to send a high signal to the Basys 3 when current exceeds 1 A through the motors. For wall avoidance, the Sharp GPZY0A60SZLF infrared sensor and Basys 3 JXADC PMOD port operate together to monitor the distance of the Rover 5 from the wall. This distance measuring sensor outputs varying voltages based on the calculated distance; by scaling the various output voltages down to between 0 – 1 V, the distance is monitored, and a threshold is set for the Rover 5. Color sensing deploys the TCS3200 color sensor; this sensor outputs different frequencies based on what color it is currently sensing. These varying output frequencies determine whether the sensor is currently reading red, blue, or green. Morse code communication is achieved using a strobe beacon flashlight and an OP593 phototransistor. The strobe beacon is powered by an Arduino Uno and implements four pushbuttons to flash the commands go, pause, resume, and stop. The command from the strobe beacon is received by the OP593 phototransistor in a common collector amplifier configuration.

2. Design Aspects

The main aspects of this design include the following: motor control, overcurrent protection, wall avoidance, color sensing, and morse code communication. All aspects of the design exhibit the capabilities of the Basys 3 FPGA and will be discussed further.

2.1 Motor Control and Overcurrent Protection Circuit

Motor control of the Rover 5 is accomplished using the L298 H-bridge circuit and the Basys 3 FPGA. The L298 H-bridge circuit utilizes several key components. The first of these components being the L298 dual H-bridge, capable of driving two different motors. The H-bridge has two enables and four inputs. For instance, to drive Motor A in the forward direction, enable A and input 1 are driven high while input 2 is driven low. To reverse direction, the previous low inputs are driven high, and the previous high inputs are driven low. The same process applies for both motors. The motor enables and inputs are wired to the Basys 3 PMOD port.

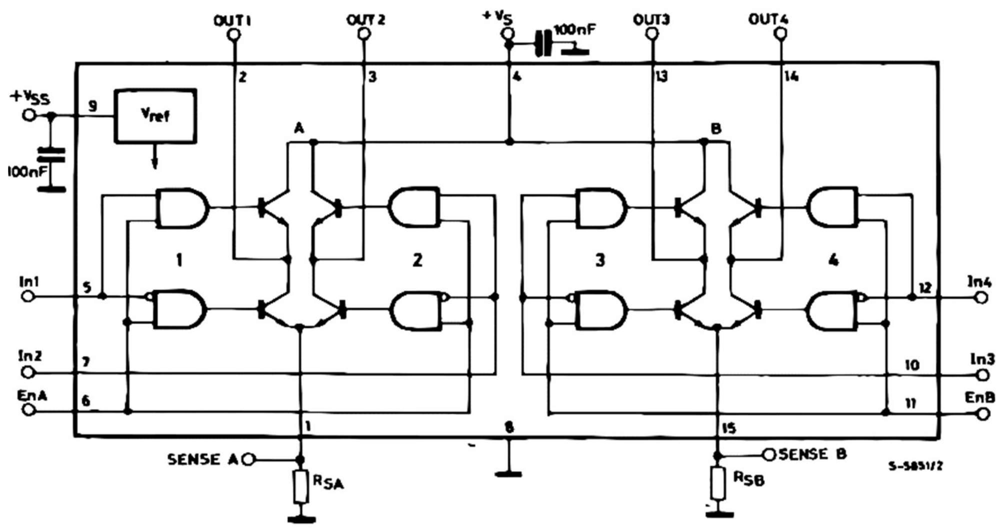

Figure 1: L298 Block Diagram [3]

Another key component of the H-bridge circuit is the 5 V linear voltage regulator. The system uses a 9.6 V battery; therefore, to ensure the L298 functions properly at a logic voltage of 5 V, the voltage regulator is implemented. Two shunt resistors are employed in series with sense A and B; a shunt resistor, also known as a current sensing resistor, measures the voltage drop through the resistor.

Overcurrent protection implements the LM339 comparator and the output pins from sense A and sense B of the L298 H-bridge circuit. The LM339 comparator is supplied 3.3 V from the Basys 3 PMOD port. Two voltage dividers each with a 1 kΩ and a 2.2 kΩ resistor, scale down the 3.3 V to our desired reference voltage of 1.031 V. This equation shows the resulting voltage value that is implemented:

$$V_{ref} = \frac{1}{1+2.2} \times 3.3\,V = 1.031\,V. \tag{1}$$

This resulting voltage is fed into two inverting terminals of separate op-amps in the LM339. Wires are connected from sense A and sense B pins on the H-bridge circuit board to the non-inverting terminals of the op-amps the constant 1.031 V is wired to. The output from each op-amp is wired to the Basys 3. The outputs are a low voltage, or logic 0, when the non-inverting terminal is less than the inverting terminal. When the non-inverting terminal is greater than the inverting terminal, the outputs are a high voltage, or logic 1. The outputs employ separate 10 kΩ pull-up resistors to ensure a current is supplied.

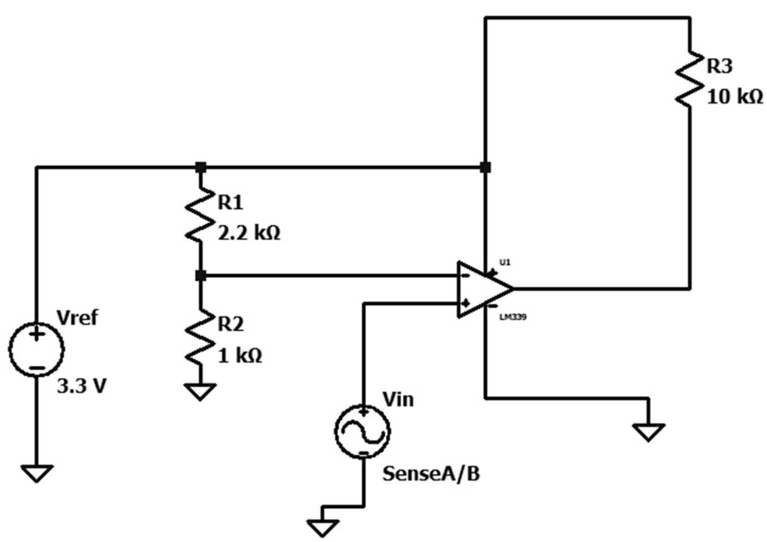

Figure 2: Overcurrent Protection - LM339 Single Op-Amp Circuit Schematic

2.2 Motor Control and Overcurrent Protection Verilog Code

A module is programmed to ensure control of both Rover 5 motors using the L298 H-bridge circuit board. Enables are driven high and low based on the morse code command flashed by the strobe beacon. The direction of the motors is dependent upon the distance of the Rover 5 from the wall. The Basys 3 utilizes a 100 MHz clock signal; a 21-bit counter is implemented to ensure 1,666,666 counts of the 100 MHz clock signal or a rate of 60 Hz. Two more 21-bit counters with decimal values 833,333 and 1,245,000 are used to guarantee the desired roaming duty cycle. The Rover 5 motors are currently running at a duty cycle of 50% when forward and 75% when turning. For overcurrent protection, a register, turnoff, is & operated with PWM, and set to 1 initially, to keep the motors running when in a steady state. A 19-bit counter is implemented to avoid an initial spike in voltage when turning on enables. When over 1 A of current, a counter increments by 1 until it reaches the value of the 19-bit count limit. When the count limit is met, turnoff is set to 0; therefore, turning off the enables of both motors. The enables stay off and can only be reset by turning the Basys 3 off and on again.

2.3 Wall Avoidance Circuit

The Sharp GPZY0A60SZLF infrared distance measuring sensor measures distances ranging from 4 in to 60 in at a rate of 60 Hz. The output voltage varies based on the distance the sensor is measuring. The sensor is supplied with 3.3 V from the Basys 3, and the output voltage ranges from 3.25 V at 4 in to 0.6 V at 60 in. For the JXADC PMOD to properly operate in a unipolar configuration, the varying output voltage of the Sharp GPZY0A60SZLF sensor is scaled down to between 0 – 1 V using a voltage divider. The voltage divider uses a 2.2 kΩ and a 1 kΩ resistor and the resulting 1.031 V is wired to the JXADC PMOD port.

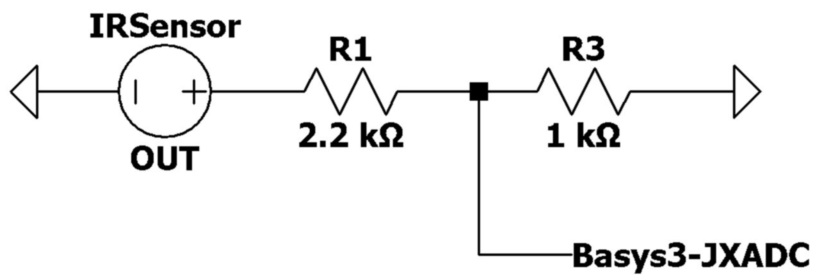

Figure 3: Wall Avoidance – Voltage Divider Circuit Schematic

2.4 Wall Avoidance Verilog Code

A module is programmed to determine the distance of the Rover 5 from the boundary wall and turn when a distance threshold of 20 in is met. For the JXADC to properly operate, a pair of port pins must be used on the JXADC PMOD port. The scaled voltage from the Sharp sensor is wired to the positive pin and the negative pin is grounded. The maximum input for the positive pin is 1 V, which is why the voltage requires scaling with the voltage divider. The analog signal is received into the multiplexer and configured by the JXADC wizard. A digital output, representing the analog voltage, is generated every 26 counts of the positive edge of the clock. The digital output is between 000h and FFFh. A state machine with three states is designed to ensure proper directional change when the Rover 5 is 20 in away from the wall. State zero simply resets all counters and sets the next state to one. State one checks if the distance threshold digital value of 76Ch has been met, if not, it sets the motors direction to forward; if so, it sets the next state to two. State two implements a 33-bit counter to turn the rover right 160°. When the max counter value of 245,000,000 is met and the turn has completed, the state is set back to zero to repeat the process.

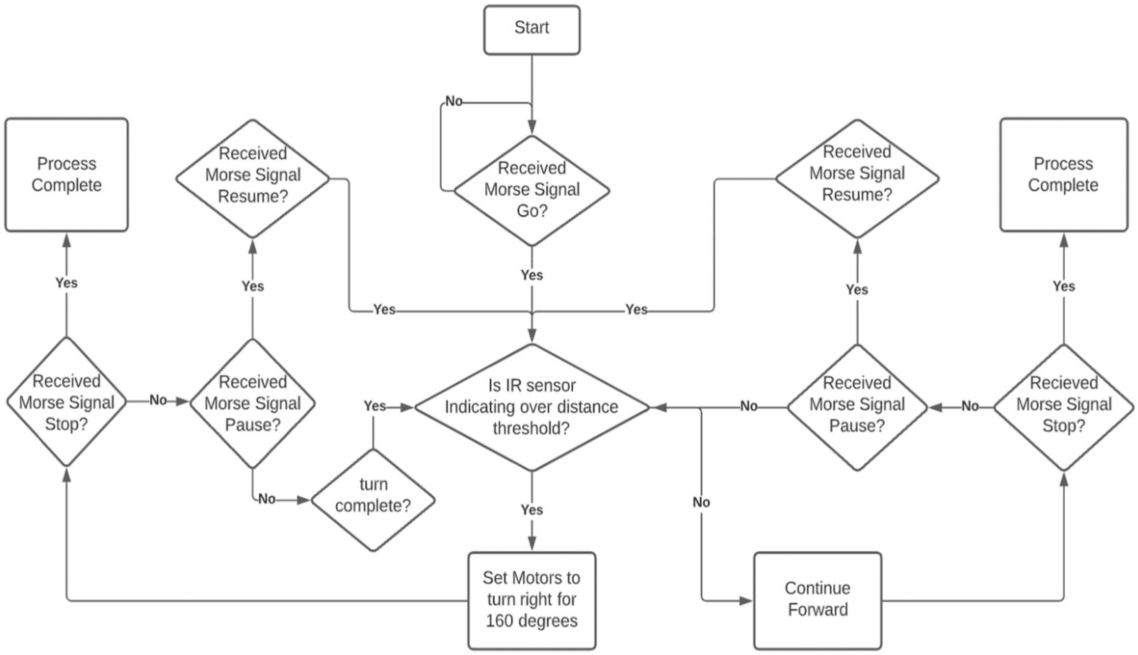

Figure 4: Wall Avoidance State Machine Diagram

2.5 TCS3200 Color Sensor and Visual Indicator

Color sensing is accomplished using the TCS3200 color sensor and the capabilities of the Basys 3. The TCS3200 is a color light-to-frequency convertor where the output frequency is directly proportional to the light intensity. The output frequency is scaled to 20% by driving pin S0 high and pin S1 low. The TCS3200 sensor uses an 8x8 array of photodiodes with an equal number of red, green, blue, and clear filters. The specific photodiode type is selected by driving pins S2 and S3 either high or low based on which type you want to currently read. The TCS3200 is supplied with 3.3 V from the Basys 3 PMOD port and mounted on the underside of the Rover 5. The output signal of the TCS3200 is a square wave with 50% duty cycle and wired to a Basys 3 PMOD port. The frequency of this signal is calculated using a counter and implemented in a state machine to determine the color being read by the sensor. The TCS3200 color sensor and Verilog module correctly identify the colors red, green, and blue with no false readings.

S2	S3	PHOTODIODE TYPE
L	L	Red
L	H	Blue
H	L	Clear (no filter)
H	H	Green

Figure 5: TCS3200 - Table of Selectable Photodiode Options [7]

Based on which color the TCS3200 sensor is currently reading, a red, blue, or green LED is driven high for the duration of the reading. To operate the LEDs within there forward voltage and current specifications, a 36 Ω resistor is implemented in series with the red LED and a 12 Ω resistor is implemented in series with the blue LED. A resistor is not required for the green LED. The different LEDs are supplied a 3.3 V signal from their respective PMOD output pins.

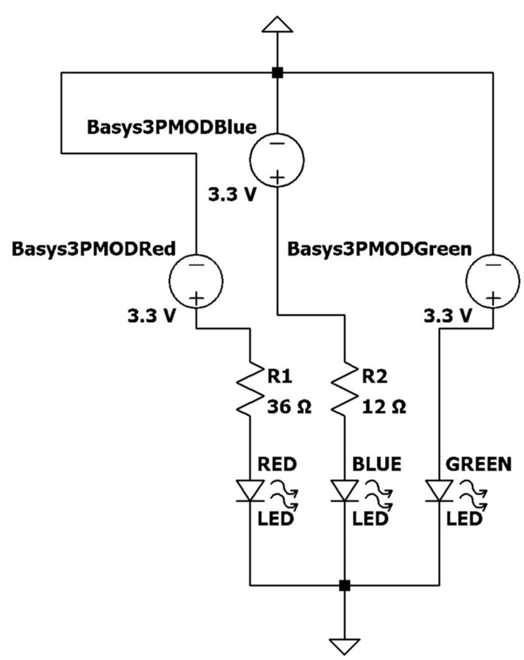

Figure 6: Color Sensing Visual Indicator LED Circuit Schematic

2.6 Color Sensing Verilog Code

Code is designed to read the output frequency of the TCS3200 color sensor to determine whether the color sensor is currently reading red, green, or blue. A state machine is designed and consists of five states. State zero sets all the initial variables to zero; there is also a brief delay implemented with a 21-bit counter to avoid an initial false reading. When the delay has ended, the next state is set to one. State one reads the output frequency of the red photodiode filter. An if statement checks if the color sensor output and a variable called status check are equal to zero. If so, status check is set to one and status is set to zero. If not, another condition checks if the color sensor output is high and status check is still equal to zero; if so, status check is set to one and status is set to zero.

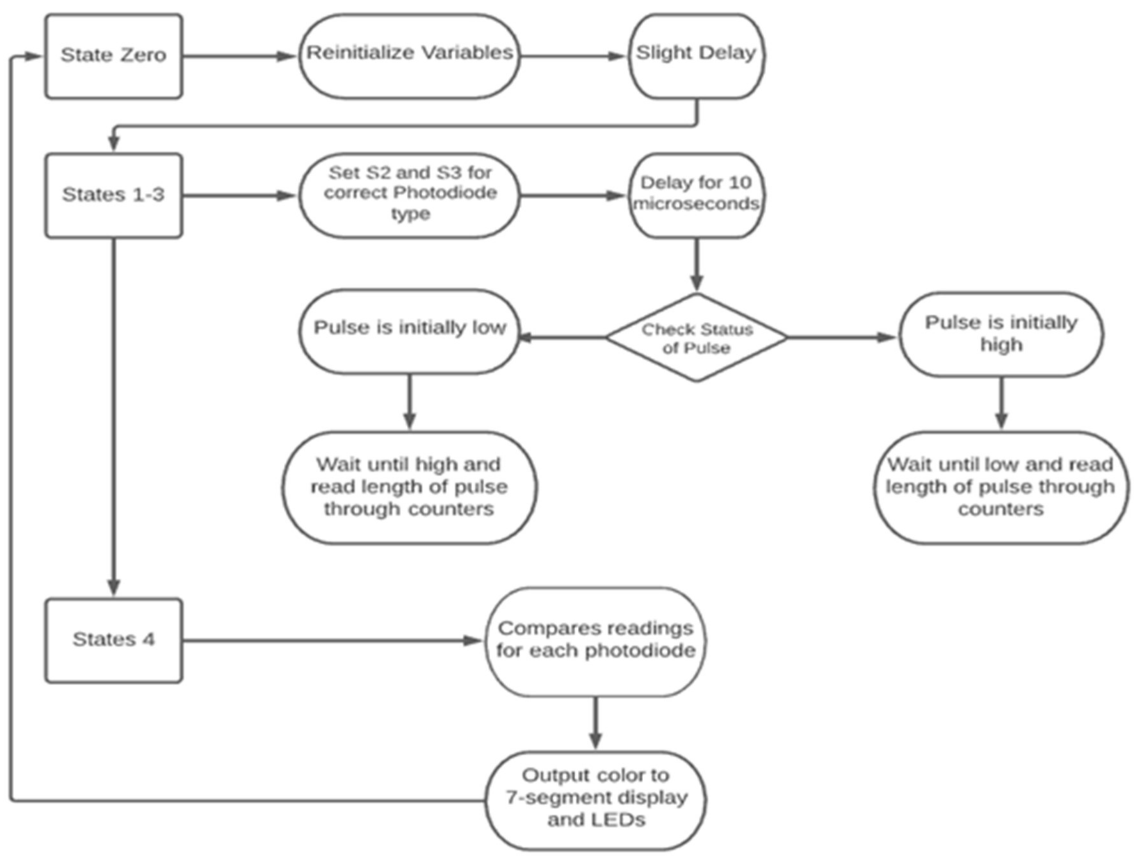

Figure 7: Color Sensing State Machine Diagram [2]

These conditionals are executed to determine if the output signal is currently high or low (square wave output). If the signal is high, a set of conditionals are executed to count the half length of the pulse. The same process applies when the signal is low. When the color sensor output switches from high to low, or low to high, the next state is set to two. State two and three follow the same process as state one for the blue and green photodiode filters. State four compares the length of the pulse for the red, blue, and green photodiode to previously experimented ranges to determine whether the sensor is currently reading red, green, or blue. The appropriate color is displayed on the 7-segment display and on the visual indicating LEDs.

2.7 Morse Code Communication Circuits

Morse code communication utilizes the OP593 phototransistor. The OP593 phototransistor acts as an open circuit when not exposed to light; however, when a light is flashed at the phototransistor, base current is induced and allows current to flow through the emitter. The OP593 has a wider receiving angle of 130°, this lessens the need for exact precision from the strobe beacon. A common collector amplifier circuit is used to ensure a low voltage when light is not flashed, and a high voltage when light is flashed.

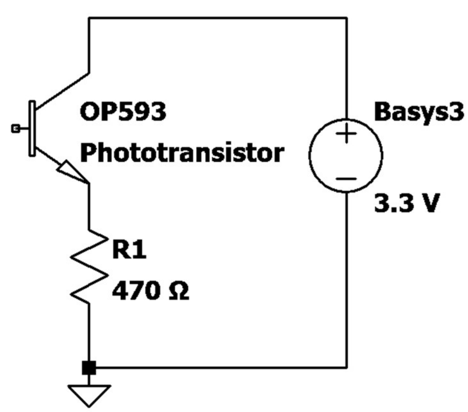

Figure 8: OP593 Common Collector Amplifier Circuit Schematic

The circuit supplies 3.3 V in series with the OP593 phototransistor and a 470 Ω resistor. The output of the amplifier is wired to the Basys 3 PMOD port to send a high signal when a command is being given.

A strobe beacon circuit consists of four pushbuttons, a 11 kΩ pull-down resistor for each button, a flashlight, a NPN transistor, and an Arduino Uno. The positive contact of the flashlight is wired to the collector of the transistor and the negative contact is wired to the emitter. The four buttons are powered with 5 V and represent the four commands: go, pause, resume, and stop. Arduino code reads the state of the buttons, and when high, sends a high output, in the form of dots and dashes, to the base of the transistor to flash the desired command. A dot is 300 ms and a dash is 650 ms.

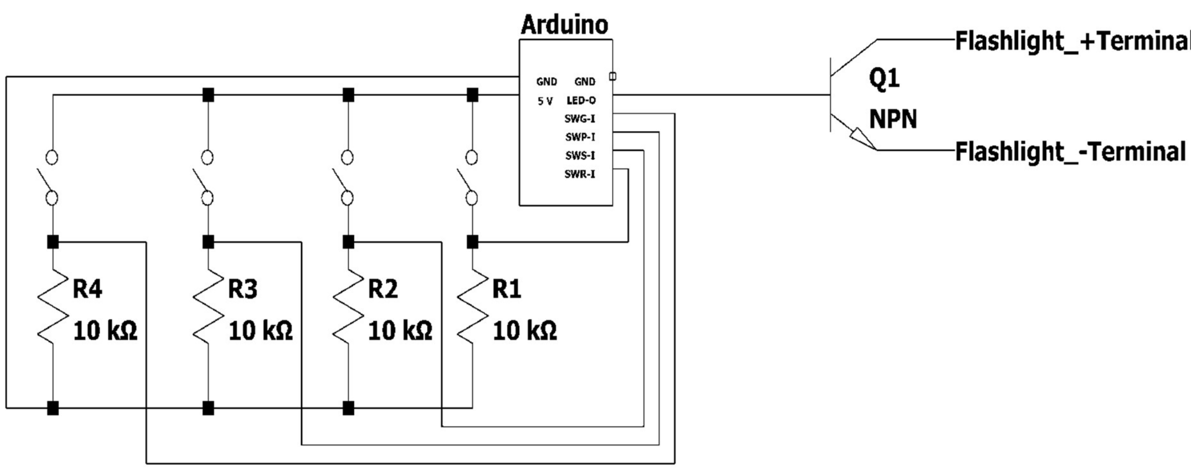

Figure 9: Strobe Beacon Flashlight Circuit Schematic

2.8 Morse Code Communication Verilog Code

A module is developed to read the morse code signal from the strobe beacon flashlight and communicate the correct command to the Rover 5; a state machine with six states assists in doing this. When a signal is flashed, state zero counts the length of the high signal. After the signal goes low again, and a high signal was just counted, the state is set to one. State one checks if the high signal counted was either a dot, 300 ms, or a dash, 650 ms. If so, dot or dash is incremented by 1 and the state is set back to zero. After a dot or dash is registered and the signal is low, a set of conditionals are executed to see if one of the commands has been given. If go is given, state two is entered, variables are reset, and the enables are turned on. When pause is provided, state three is entered, variables are reset, and the enables are turned off. If resume is sent, state four is entered, variables are reset, and the enables turn back on. Finally, when stop is delivered, state four is entered, variables are reset, and the enables turn off and cannot be turned back on without resetting the Basys 3. The Rover 5 only begins with the go command. From go, the rover can be paused or stopped. If paused, it can only be started again with resume. Once resuming, the rover can either be paused again, or stopped completely.

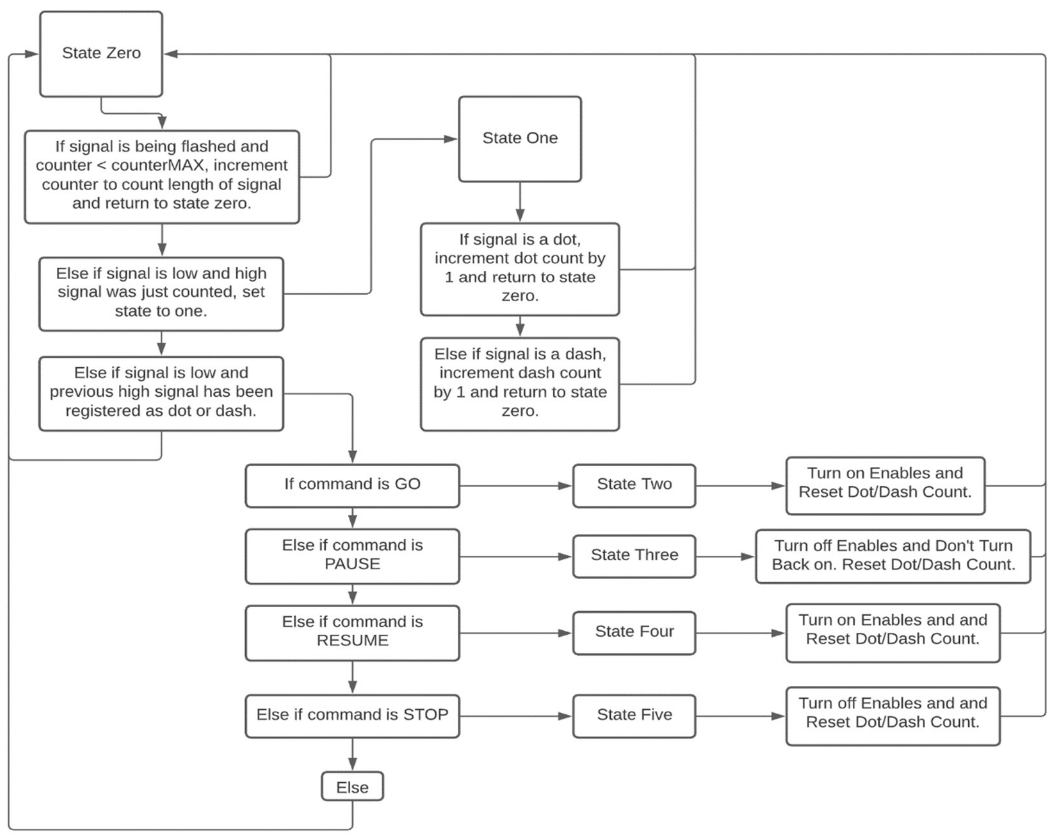

Figure 10: Morse Code State Machine Diagram

3. Design Evaluation

Each aspect of the design is tested throughout the process of implementation. Most software was tested by writing testbench modules and simulating waveforms in Xilinx Vivado; color sensing was tested by adjusting pulse length ranges and sensing different shades of red, blue, and green. The hardware components were tested using tools such as LTSpice, the oscilloscope, the function generator, and the DC power supply.

3.1 Motor Control, Overcurrent Protection, and Wall Avoidance Evaluation

After development of motor code, overcurrent protection, and wall avoidance modules, testbenches are developed to ensure inputs and outputs are acting appropriately. The motors operate as intended, capable of moving both directions at a duty cycle of 50% and 75%. When a wall is within 20 in of the Rover 5, it turns 160°, then begins going forward again. If current over 1 A is exceeded, the rover motors turn off. The simulation waveform (refer to Appendix A) accurately exhibits the designed function of the Rover 5.

The LM339 comparator circuit for overcurrent protection is tested on LTSpice and in the lab. The circuit is tested in the lab by connecting the function generator to the non-inverting terminal with an AC voltage, connecting the DC power supply to the breadboard with 3.3 V, and connecting the Oscilloscope probes to the non-inverting terminal and output of the LM339 op-amp. The output of the overcurrent protection op-amp comparator is high when the voltage exceeds 1.031 V, and low when less than 1.031 V. The overcurrent protection circuit is soldered on a PCB and inside the junction box mounted on the Rover

5. The wall avoidance voltage divider is on a solderable breadboard attached to the top of the junction box, with the distance measuring sensor mounted on the front of the rover.

3.2 Color Sensing Evaluation

The TCS3200 outputs a different frequency based on the current color it is sensing. Arduino code is used to test the different output frequencies for a red, blue, and green paint swatch. When a card is placed, the code samples the frequencies for each photodiode color 100 times and the min and max frequency in μS for each photodiode is printed. After repeating this step for all three colors, these values were placed into the comparison state in the color sensing FPGA state machine.

Table I: Experimental Analysis of TCS32000 Output Frequencies

Color Tested	**Photodiode Type**	**Minimum Value (μS)**	**Maximum Value (μS)**
Red	Red	66	84
	Blue	152	172
	Green	199	223
Blue	Red	125	281
	Blue	43	247
	Green	100	283
Green	Red	220	366
	Blue	157	342
	Green	115	399

After implementing these values and realizing the Arduino and Basys 3 run on different clocks, the ranges in the FPGA state machine were adjusted. The color sensor correctly identifies the colors red, blue, and green with no false readings.

3.3 Morse Code Communication Evaluation

The OP593 common collector amplifier is supplied a voltage of 3.3 V and tested with a flashlight and a multimeter at the output of the amplifier. With no light flashed, the amplifier outputs a voltage around 0.5 V, due to ambient light. With light flashed, the amplifier outputs a voltage around 3.2 V. The common collector amplifier circuit functions appropriately. Also, the strobe beacon circuit operates properly; when a pushbutton is pressed, the Arduino reads which button was pressed, and sends the correct number of dots and dashes for the command to the base of the transistor. By doing this, current is induced, and the flashlight turns on. The Verilog code is tested with test benches and the simulation waveforms behave as expected (refer to Appendix A).

4. Design Conclusion

This design implements important aspects of electrical automation through hardware and FPGA design. The Basys 3 FPGA, consisting of 33,280 logic cells, and circuits work together to ensure the Rover 5 can avoid boundary walls, correctly identify, and visually indicate the colors red, green, and blue, and be communicated with through morse code. All aspects of design have been deliberately engineered to ensure proper function.

References

[1] Basys 3 FPGA Board Reference Manual. Digilent, Pullman, WA, United States, 2019, pp. 2. [Online]. Available: basys3_rm.pdf (digilentinc.com)

[2] Holyoak, Jacob. "Color Sensing State Machine Diagram", March 27th, 2021.

[3] L298 Dual Full-Bridge Driver Datasheet. STMicroelectronics, Geneva, Switzerland, 2000. [Online]. Available: Dual full-bridge driver (st.com)

[4] LM339, LM239, LM139, LM2901 Quad Differential Comparators Datasheet. Texas Instruments, Dallas, TX, United States, 2020. [Online]. Available: LM339, LM239, LM139, LM2901 Quad Differential Comparators datasheet (Rev. U)

[5] NPN Plastic Silicon Phototransistor Datasheet. TT Electronics, Woking, United Kingdom, 2016. [Online]: Available: OP593-598-798.pdf (ttelectronics.com)

[6] Sharp GP2Y0A60SZ0F/GP2Y0A60SZLF Datasheet. Sharp, Montvale, NJ, United States, 2020. [Online]: Available: pc123xnnsz_j (socle-tech.com).

[7] TCS3200, TCS3210 PROGRAMMABLE COLOR LIGHT-TO-FREQUENCY CONVERTER. TAOS, Plano, TX, United States, 2009. [Online]: Available: Document: (mouser.com)

Appendix A

This appendix displays experimental results throughout design process.

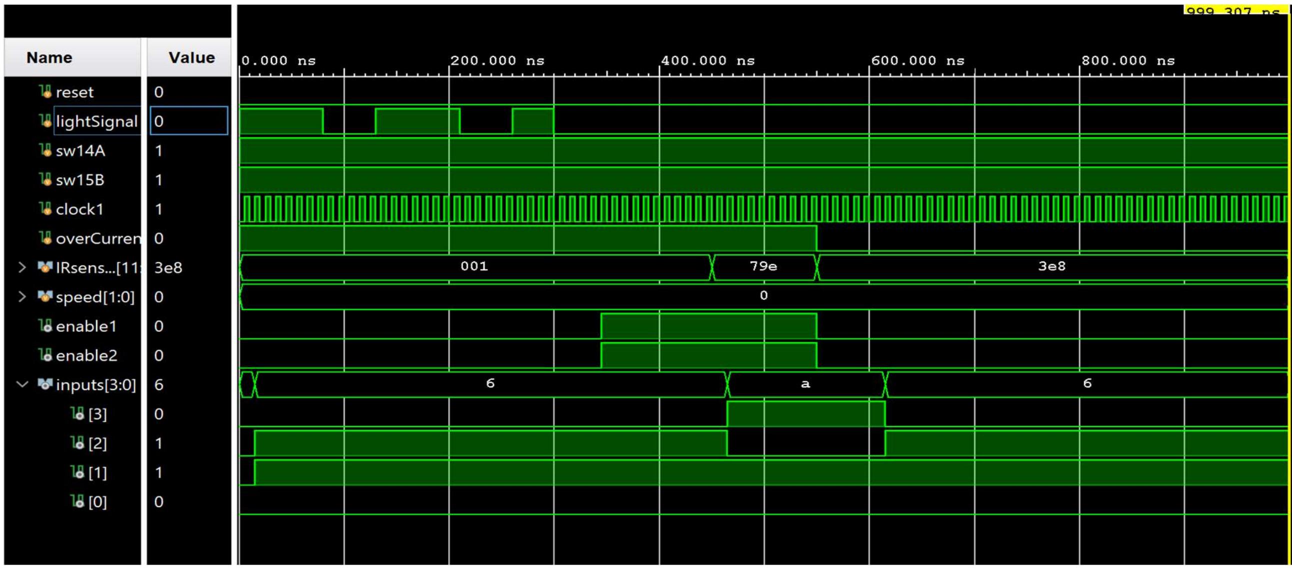

Figure 11: Wall Avoidance and Overcurrent Protection – Simulation Waveform

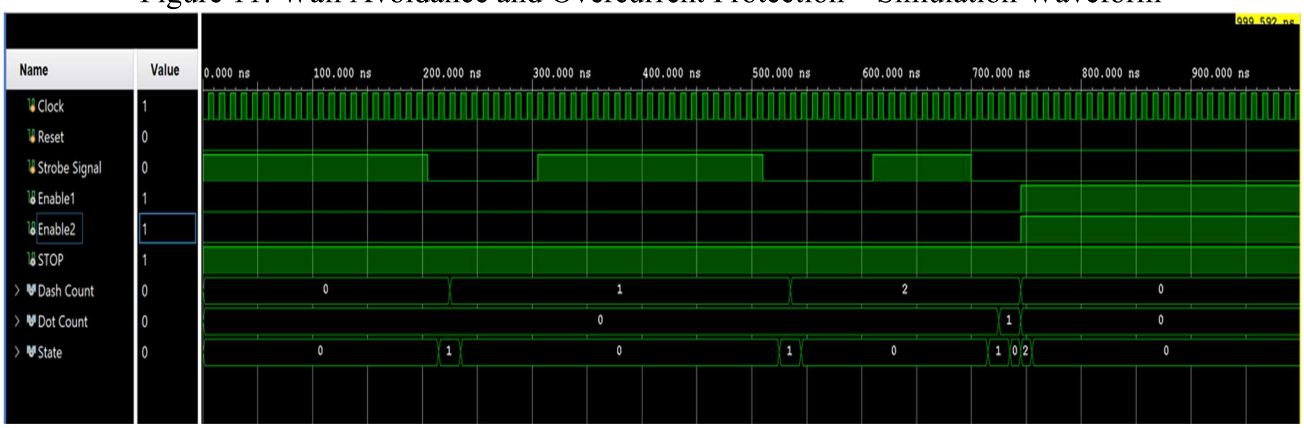

Figure 12: Morse Code GO – Simulation Waveform

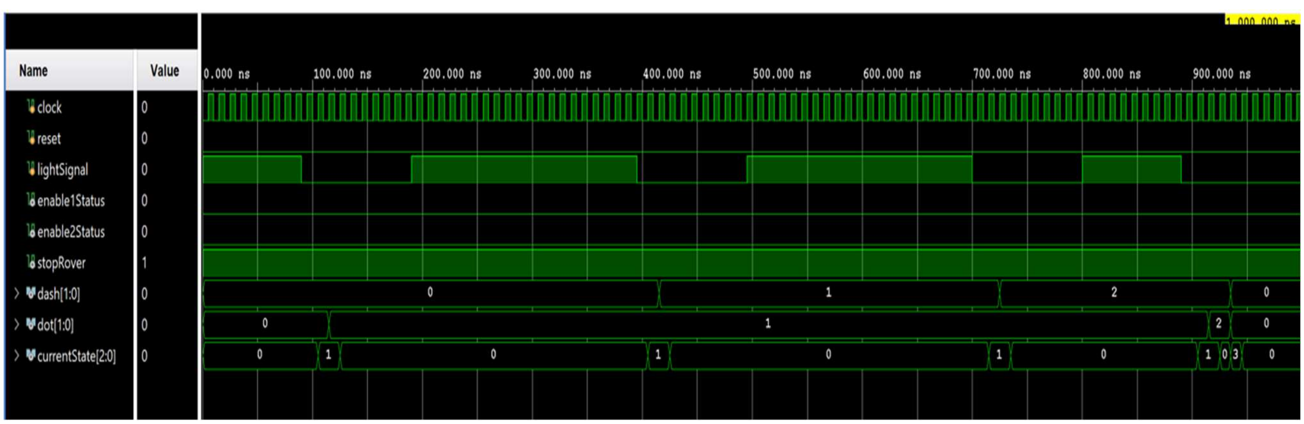

Figure 13: Morse Code PAUSE – Simulation Waveform

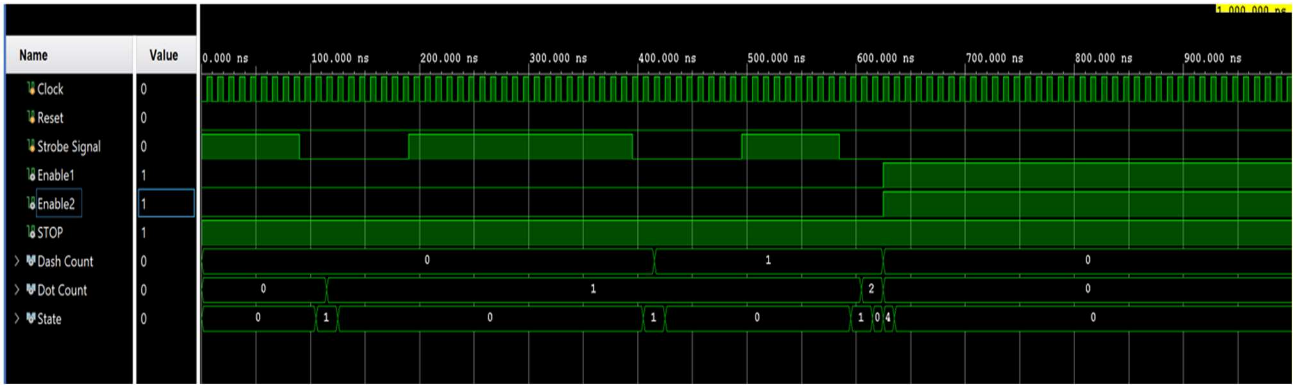

Figure 14: Morse Code RESUME – Simulation Waveform

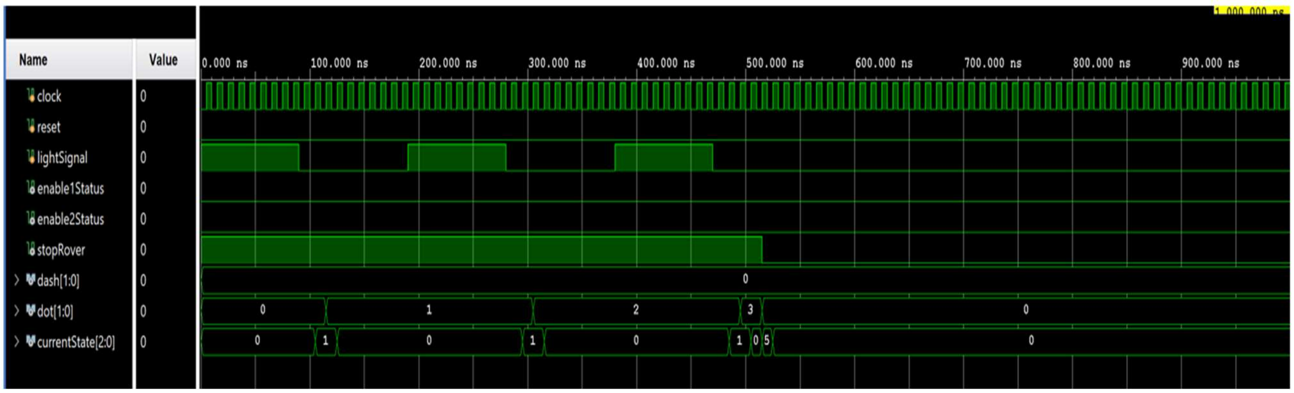

Figure 15: Morse Code STOP – Simulation Waveform

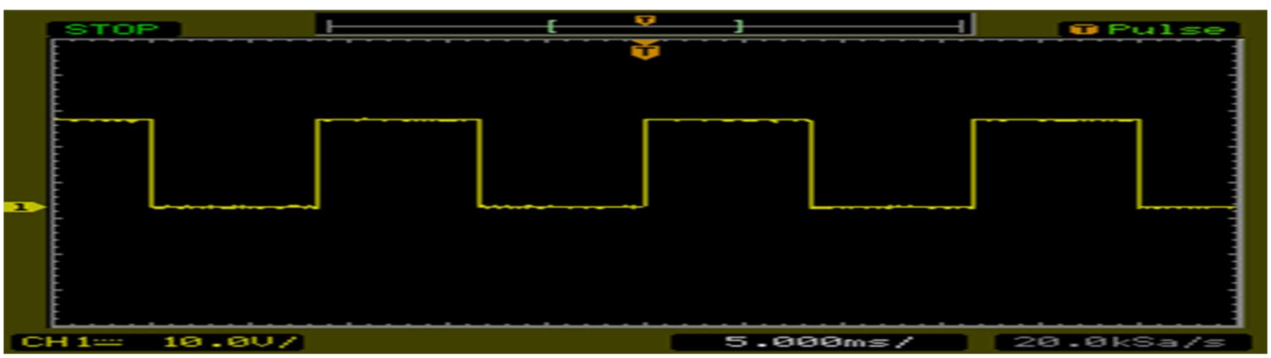

Figure 16: PWM 50% - Oscilloscope Images

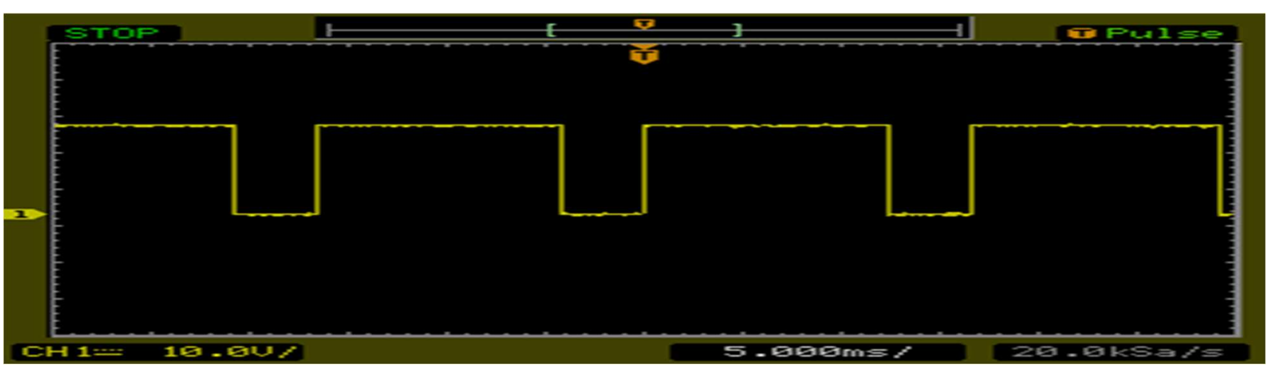

Figure 17: PWM 75% - Oscilloscope Images

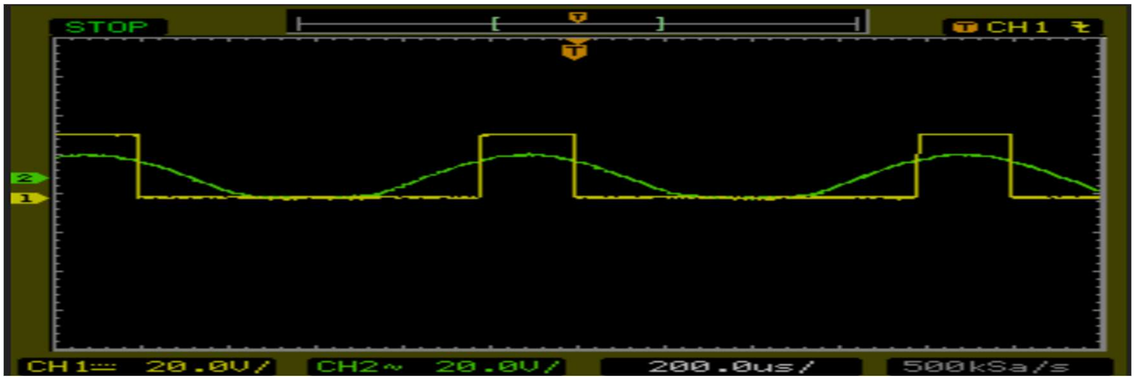

Figure 18: Overcurrent Protection Circuit – Oscilloscope Images

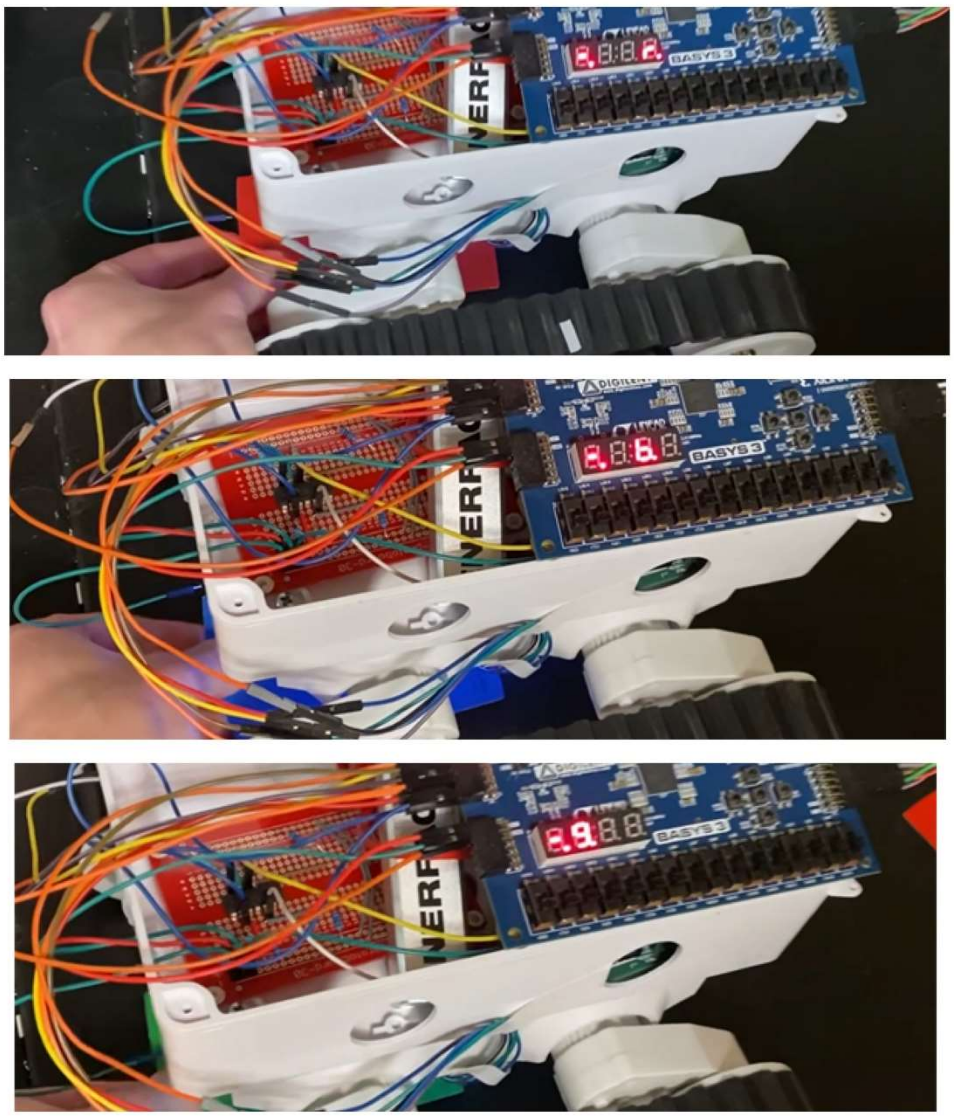

Figure 19: Color Sensor Reading and Displaying RGB

Appendix B

These figures show the finalized Gantt chart and Budget for the project.

Main-Project

Texas Tech ECE

Gantt Chart Template © 2006-2018 by Vertex42.com.

Project Start Date	2/22/2021 (Monday)	Display Week	8
Project Demo Date	4/26/2021 (Monday)		

Week 8 — 12 Apr 2021; Week 9 — 19 Apr 2021; Week 10 — 26 Apr 2021

WBS	TASK	MEMBER	START	END	DAYS	% DONE	LOCATION
1	**Wall Avoidance Circuit**		Mon 2/22/21	Sat 4/17/21	54	100%	
1.1	Research and Acquire Sensors	Josiah/Brendan	Mon 2/22/21	Wed 3/03/21	10	100%	Internet
1.2	Construct and Test Virtual Circuit w/ LTSpice	Josiah/Brendan	Fri 2/26/21	Tue 3/09/21	12	100%	Internet/Lab
1.3	Construct and Test Circuit in Lab	Josiah/Brendan	Fri 3/05/21	Tue 3/16/21	12	100%	Lab
1.4	Finalize Circuit Design/Troubleshoot	Josiah/Brendan	Mon 4/05/21	Sat 4/17/21	13	100%	Lab
2	**Color Sensing Circuit**		Sun 2/28/21	Sat 4/17/21	7	100%	
2.1	Research and Acquire Color Sensors	Josiah	Sun 2/28/21	Sun 3/07/21	8	100%	Internet
2.2	Construct and Test Circuit in Lab	Josiah	Mon 3/15/21	Fri 3/26/21	12	100%	Lab
2.3	Implement Visual Indicators for Condition	Josiah	Sun 4/11/21	Thu 4/22/21	12	100%	Internet/Lab
2.4	Finalize Circuit Design/Troubleshoot	Josiah	Sun 4/11/21	Sat 4/17/21	7	100%	Lab
3	**Command Communication**		Sat 2/27/21	Fri 4/16/21	7	100%	
3.1	Research and Acquire Phototransistors and Strobe Beacons	Josiah/Brendan	Sat 2/27/21	Fri 3/05/21	7	100%	Internet
3.2	Construct and Test Virtual Circuit w/ LTSpice	Josiah/Brendan	Sat 3/06/21	Fri 3/12/21	7	100%	Internet/Lab
3.3	Construct and Test Circuit in Lab	Josiah/Brendan	Wed 3/10/21	Sun 3/21/21	12	100%	Lab
3.4	Assemble Components to Rover and Test with Commands (Morse-Code)	Josiah/Brendan	Thu 3/25/21	Sat 4/03/21	10	100%	Lab
3.5	Finalize Design/Troubleshoot	Josiah/Brendan	Mon 4/05/21	Fri 4/16/21	12	100%	Lab
4	**Verilog Modules**		Wed 2/24/21	Tue 4/20/21	7	100%	
4.1	Research and Develop Module for Wall Avoidance/Simulate Waveform	Jacob/Brendan	Wed 2/24/21	Fri 3/12/21	17	100%	Internet/Lab
4.2	Research and Develop Module for Color Sensing/Simulate Waveform	Jacob/Brendan	Fri 3/12/21	Sun 3/28/21	17	100%	Internet/Lab
4.3	Research and Develop Module for Command Comm./Simulate Waveform	Jacob/Brendan	Wed 3/24/21	Fri 4/09/21	17	100%	Internet/Lab
4.4	Finalize Modules/Troubleshoot	Jacob	Wed 4/14/21	Tue 4/20/21	7	100%	Internet/Lab
5	**Other Important Tasks/Finalize**		Thu 3/18/21	Mon 5/03/21	7	100%	
5.1	Interim Report (3/29/21)	Team	Mon 3/22/21	Sun 3/28/21	7	100%	Internet
5.2	Printed Circuit Board (4/5/21)	Team	Mon 3/15/21	Mon 4/05/21	7	100%	Lab
4.3	Troubleshoot and Finalize Design (4/26/21)	Team	Mon 4/05/21	Mon 4/26/21	7	100%	Lab
4.4	Compile Final Report and Notebook (5/3/21)	Team	Mon 4/26/21	Mon 5/03/21	7	100%	Internet

Figure 20: Final Main-Project Gantt Chart

Main-Project Budget	Running Cost			Cost Estimate			Start Date	2/22/2021
Direct Labor:	**Rate/Hr**	**Hrs**	**Total**	**Rate/Hr**	**Hrs**	**Total**	**Today**	4/30/2021
Brendan Blacklock	$ 15.00	118	$ 1,770.00	$ 15.00	200	$ 3,000.00		
Josiah Ramirez	$ 15.00	100	$ 1,500.00	$ 15.00	200	$ 3,000.00	**End Date**	5/3/2021
Jacob Holyoak	$ 15.00	111	$ 1,665.00	$ 15.00	200	$ 3,000.00		
Direct Labor Subtotal			$ 4,935.00			$ 9,000.00		
Labor Overhead (Rate: 100%)			$ 4,935.00			$ 9,000.00		
Direct Labor Total			$ 9,870.00			$ 18,000.00		
Contract Labor:	**Rate/Hr**	**Hrs**	**Total**	**Rate/Hr**	**Hrs**	**Total**		
Lab Tutor	$ 40.00	0	$ -	$ 40.00	5	$ 200.00		
Dr. Clark	$ 200.00	0	$ -	$ 200.00	2	$ 400.00		
Contract Labor Total			$ -			$ 600.00		
Direct Material Costs:			**Total**			**Total**		
(Refer to Material Costs Sheet)								
Direct Material Total			$ 290.61			$ 450.00		
Equipment Rental Costs:	**Value**	**Rental Rate**	**Total**	**Value**	**Rental Rate**	**Total**		
Oscilloscope	$ 5,300.00	0.20%	$ 710.20	$ 5,300.00	0.20%	$ 742.00		
Function Generator	$ 1,600.00	0.20%	$ 214.40	$ 1,600.00	0.20%	$ 224.00		
Power Supply	$ 958.00	0.20%	$ 128.37	$ 958.00	0.20%	$ 134.12		
DMM	$ 1,700.00	0.20%	$ 227.80	$ 1,700.00	0.20%	$ 238.00		
Equipment Rental Total			$ 1,280.77			$ 1,338.12		
Subtotal Cost:			$ 11,441.38			$ 20,388.12		
Business Overhead (Rate: 55%)			$ 6,292.76			$ 11,213.47		
Total Cost:			$ 17,734.14			$ 31,601.59		

Figure 21: Final Main-Project Budget