Color Programmable Control Board

By

Anthony Shvets

Zhe Tang

Final Report for ECE 445, Senior Design, Spring 2018

TA: Zipeng Wang

May 2018

Project No. 63

Abstract

This report is about the designing of a control board circuit and color sensor that would help kids as young as 5 and 6 years old develop programming skills alongside mechanical engineering skills. The control board and color sensor would interface with an existing building platform that has its own building blocks, gears, axles and motors. Users would be able to program designs built with the control board using color codes as a low barrier entry to developing programming skills. Final build models would be able to accomodate the use of two DC motors and a Servo motor from the building platform. The work done did not result in an reliable system that could integrate into many building platforms, although it did result in a good proof of concept for future work.

Table of Contents

1 Introduction

1.1 Objective

Code is everywhere in the world and coding has become an essential skill for not only the engineers but also everyone who wants to dig in big data. Nowadays, many kids are exposed to coding at a very young age. Their brain is capable of handling different coding problems. According to VentureBeat, many good programmers started programming since 5 or 6 years old [1]. However the steep learning curve of coding can intimidate many kids before they really develop interests in it. Young kids sometimes find out that it is hard to start by typing code into the computer. Thus, it is necessary to have a good platform which could serve as a fun and easy coding environment. This platform should consist of both physical and virtual parts, where the virtual part is a coding software which should be simple to learn and the physical part should be responsive to execution instructions.

The work done in this report is towards developing a system that would provide kids at younger ages (from 4 to 8) a system to start learning coding principles. The system would be composed of a control board and color sensors. These would integrate with two DC motors, a servo motor and other building materials for kids to be able to build varieties of robotic structures and code them with color codes. In order to provide the kids a feasible way of learning the codes, the instructions will be executed as simple motions such as moving forward, backwards and setting the servo motor to different angles of rotation. The system will be operating under a line following setting where the robots built would primarily follow a black line and color code words would be detected along the line.

1.2 Background

STEM education is a growing trend in the United States. With rising interest creates a demand for STEM education services and products that help people learn principles of engineering and programming at earlier ages. There are already popular products provided by companies like LEGO and Rokenbok education that allow for young students to learn engineering and programming engineering principles for students as young as 6 years old through building robots. What they do is provide a low learning curve platform for students to create different build systems structurally, mechanically and electronically. Learning how to program through programming the robots has also been implemented by many of the same education providers, although there are some companies like Ozobot and Robomation that have made learning programming principles even easier to reach even younger students. They do so by having the students learn how to encode actions through colors. The companies provide a robot that will follow a path naturally and perform different actions as it notices color codes along the path.

Students can then create their own paths and encoding to “program” the system, although they are very limited in what they can do to the robots shape and function. What we would like to do is provide a similar color encoding system for young students to learn to code onto an existing building platform. This would allow students to develop basic engineering skills building structural and mechanical systems as well as to explore basic principles of programming. Our responsibilities to the system are to provide sensors for color detection and a electronic control board that would control two DC motors and a Servo motor. The building platform that will be used to create the structure around the electronics can be almost anything; however, for the purposes of this project everything will be done with Robotori pieces [2].

1.3. High-level Overview

The system would be powered by four AA batteries. The power supply would provide an unregulated power source to the DC motor control and provide a regulated 3.3 volts to the servo motor, color sensors and microcontroller. The color sensors would see the colors and send red, blue and green color readings to the microcontroller via an I2C communication interface. The microcontroller would interpret the data to distinguish what color has been seen and send appropriate signals to the servo motor and DC motor control to manipulate the robot. The DC motor module is composed of two H bridge circuits that control the direction of the DC motors.

2 Design

2.1 Block Diagram for the Entire System

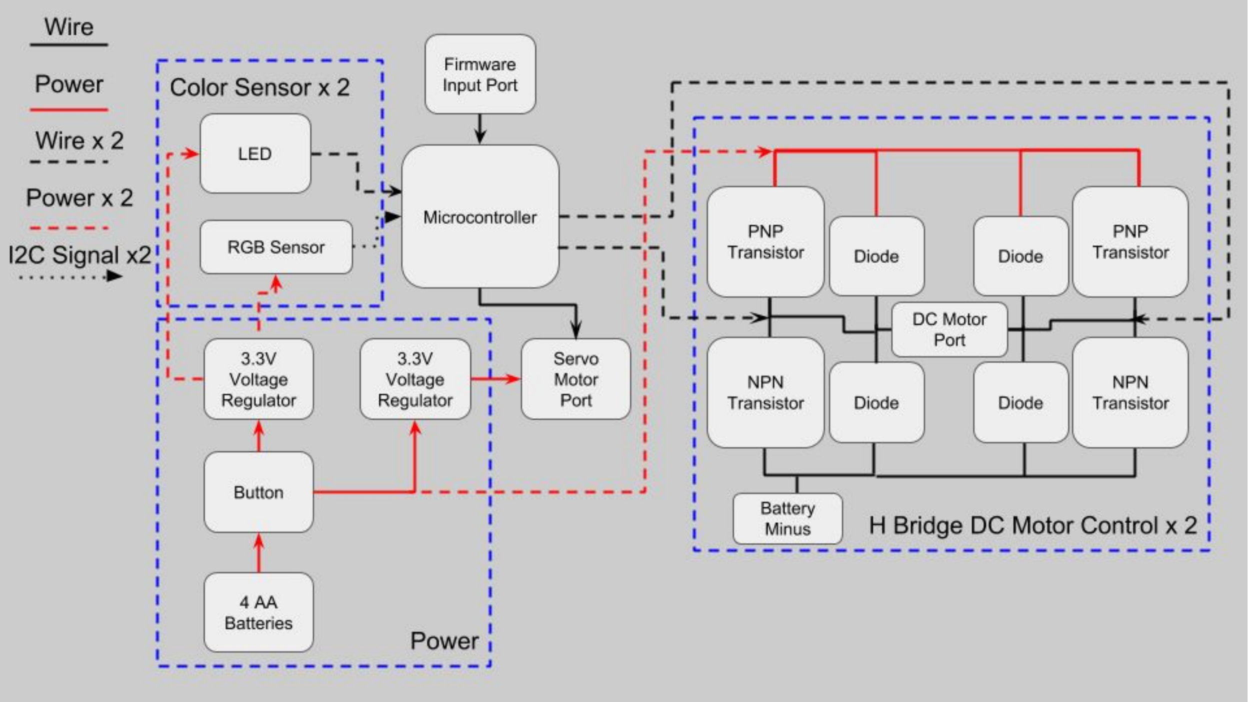


Figure 1. System Block Diagram

2.2 Power Module

The purpose of the power module is to ensure other modules in the system are powered by the necessary voltage requirements for that module. For the DC motor control that power can be unregulated. Having unregulated power for the DC motors could result in slight fluctuations in the speed that the motors operate at. Since the motors only need to be able to move most robot designs in order to follow a line, regulating the exact speed is not necessary, making the power fluctuations acceptable. Having power fluctuations to the servo motor, color sensors and microcontroller are not acceptable though. Aside from power fluctuations from a steadily weakening battery source, there would be non significant fluctuations caused by the changing motions of the DC motors. These fluctuations could cause color sensor readings to be inconsistent and cause the microcontroller to misinterpret information sent by the sensors. Fluctuations in the servo motor could result in slight twitching. In order to prevent these fluctuations from happening, there are two voltage regulators that ensure a steady 3.3 volts is provided to the color sensors, microcontroller and servo motor. Since the servo motor movements could cause fluctuations in the power it draws, a separate voltage regulator has been allocated towards it. Effectively separating the power between the servo motor from the color sensor and microcontroller.

In the circuit below, Battery-1 is the positive lead of the battery source, Battery-2 is the negative lead of the battery, Out-1 is the power lead for the microcontroller and sensors, and Out-2 is the power lead for the servo motor.

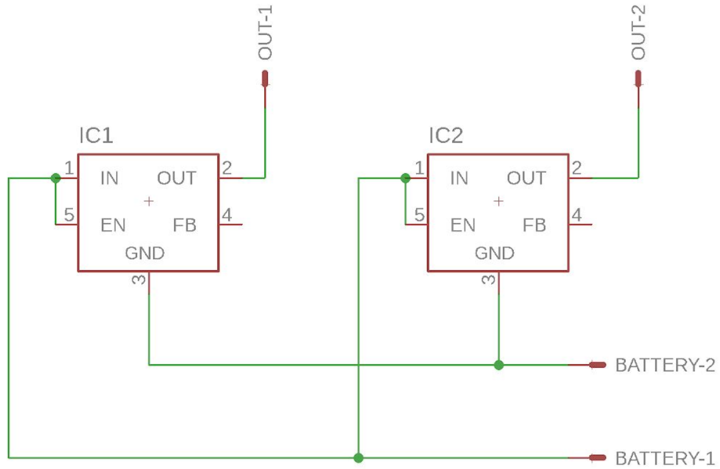


Figure 2. Power supply schematic

2.3 DC Motor Control

The DC motor control module is composed of two H bridge circuits. The purpose of the module is to appropriately redirect current drawn from the power supply to the motors depending on input signals from the microcontroller. The microcontroller would provide 3.3 volts to the bias inputs of the H bridge circuits as active high and near 0 volts as an active low. Control of the motor flow is done via the input combinations shown in Table 1.

IN-1	IN-2	Motor Flow
Low	Low	Stop
High	Low	Clockwise
Low	High	Counter Clockwise

Table 1. Motor Control Truth Table

In the circuit below, Battery-1 is the positive lead of the battery source, Battery-2 is the negative lead of the battery, IN-1 and IN-2 are signals from the microcontroller.

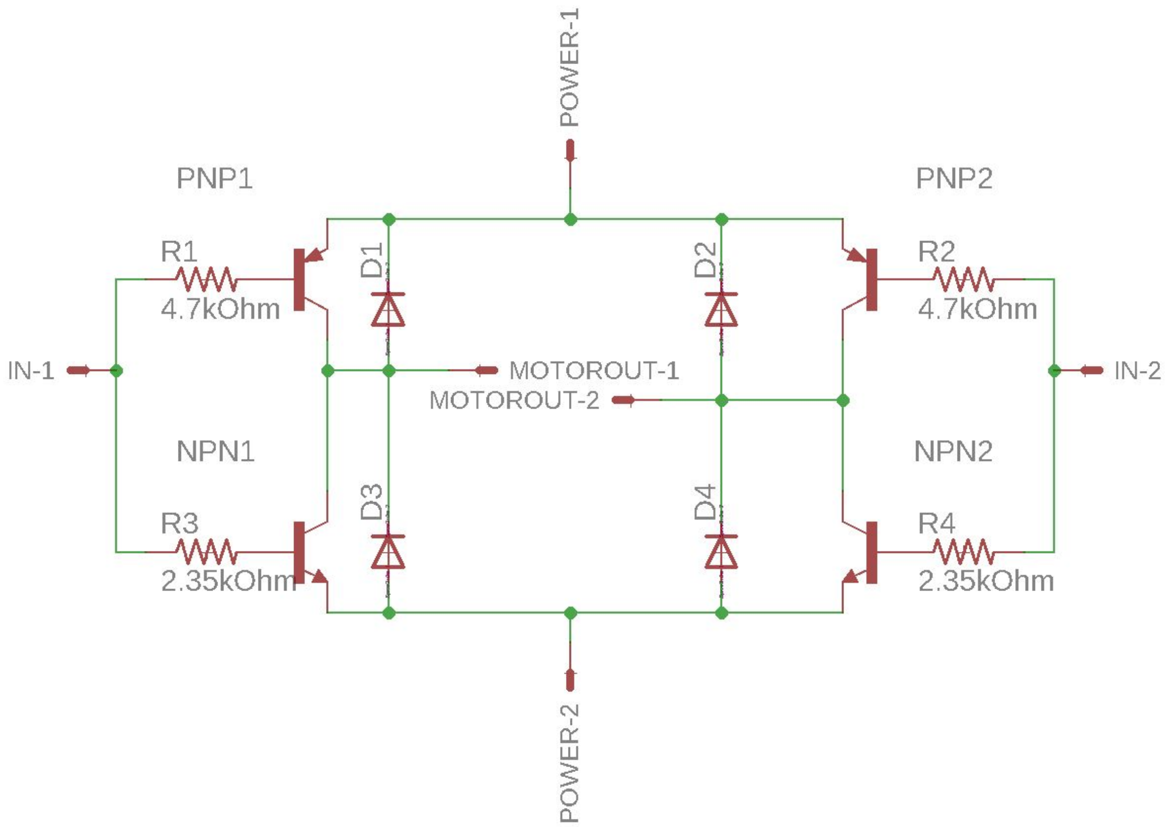


Figure 3. H-bridge schematic

2.4 Color Sensors

The color sensor consists of a single LED and a RGB sensor. RGB sensor is able to detect color illuminated by the single LED. The desired range for color detection is 3 mm vertically and 1 cm horizontally. RGB sensor converts the color into digital values and stores them in the internal R, G and B registers. Each additive primary color has two registers. For instance, red has RDATAL and RDATAH registers, which stand for red data low value and red data high value. Data stored in high register will be passed into low register after the low register's data gets accessed by master device, which is our microcontroller.

Our color sensors use inter-integrated circuit(I2C) communication interface to interact with our microcontroller. I2C bus has two bus lines, SCL and SDA. SCL is the synchronous clock and SDA is the data. At each pulse of in SCL line, SDA is read as color data or instruction in binary. The color sensor is operated as a slave device and our microcontroller as a master device.

Color sensor is mounted at the bottom of our car and the position is fixed so that we can get a consistent color reading during car's movement. Before differentiating colors, in the tuning phase, we use color sensor to detect and record the values for seven different colors. With these values, we can setup correct threshold in our main program to differentiate between each color. Also the existence of shadow affects the color reading by a lot and in order to overcome this problem, we record different values in different ambient environment so that we can find the best range for distinguishing colors.

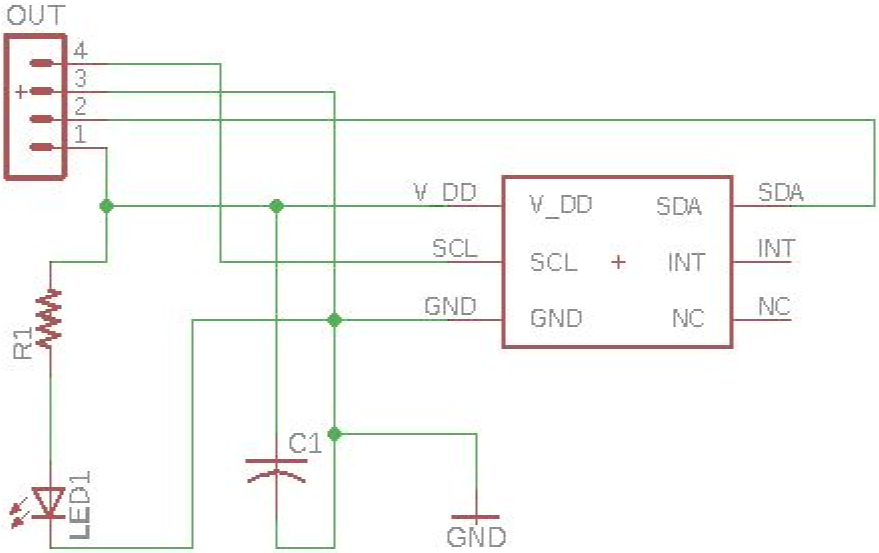


Figure 4. Overall color sensor schematic

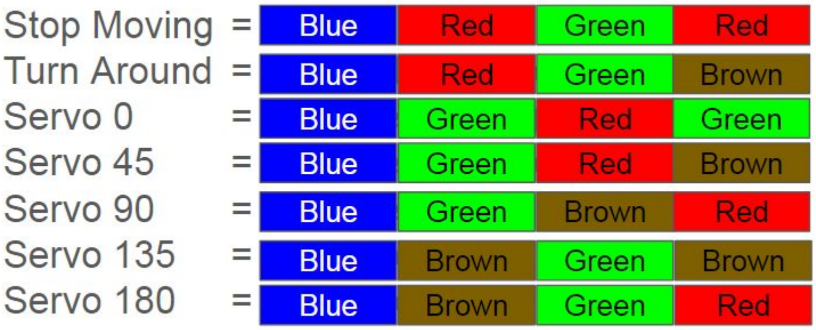
2.5 Microcontroller

The microcontroller is responsible for receiving the color data from color sensor and control motors accordingly. We use five bidirectional I/O pins to control the motors, four of them for H-bridge and the last one for servo motor. The microcontroller utilizes its internal oscillator, FRC, and operates at a frequency of 7.37 MHz.

The microcontroller communicates with color sensor through I2C bus line. It first sends out start condition and 7-bit slave address with a writing bit to pair up with our color sensor. After that, the microcontroller specifies the register needed to be accessed in the color sensor by sending out the register address with command register address to color sensor. Then, microcontroller will ask the color sensor to enter reading mode by sending out restart condition and 7-bit slave address with a reading bit. It can start to read the data from color sensor and the data transmitted back is one byte per transmission. When no data transmission is needed anymore, microcontroller will send out the stop condition to exit the reading mode.

Whenever the motor controls are needed, microcontroller will set the I/O pins according to the truth table discussed in DC motor control section. Also the servo motor needs extra delay function to send our correct pulse and operate in different angles.

The color white has been established as a background color and the color black is established as a standard line to follow. The color code words are all four colors in length and start with a blue bit. The remaining three colors are combinations of red, green and brown. Since the robot may pass a color multiple times before seeing the next color in the code word, no back to back colors can be reliably used in any code word. All possible color code combinations and their actions are shown in Figure 5.



Stop Moving	=	Blue	Red	Green	Red
Turn Around	=	Blue	Red	Green	Brown
Servo 0	=	Blue	Green	Red	Green
Servo 45	=	Blue	Green	Red	Brown
Servo 90	=	Blue	Green	Brown	Red
Servo 135	=	Blue	Brown	Green	Brown
Servo 180	=	Blue	Brown	Green	Red

Figure 5. Color Code Word Combinations

The general control flow of the microcontroller is shown in Figure 6. At first the microcontroller would need to establish an I2C connection with the color sensors. Then the robot would need to be positioned to the right of the line, so first it the microcontroller would induce a rotation in the robot until a black line is found and then position itself slightly to the right of the line. The main sequence that starts after is a repetition of moving back and forth past the line in an S shape movement, checking the color of the line every time the robot passes it by. Only when the color isn't black does the microcontroller make extra decisions. If the the code word is empty then the first color should be blue. If not then a check is done to see if the color detected has just been added to the code word. Since the robot may pass by a color multiple times before reaching the next color, this helps ensure there are no back to back colors added to the code word. If the code word has not been completed yet then the robot resumes normal line following operations until another color has been detected. If the code word has been completed, then the microcontroller checks is it matches one of the code words in Figure 5. Lastly if there is a match then the appropriate instruction is executed.

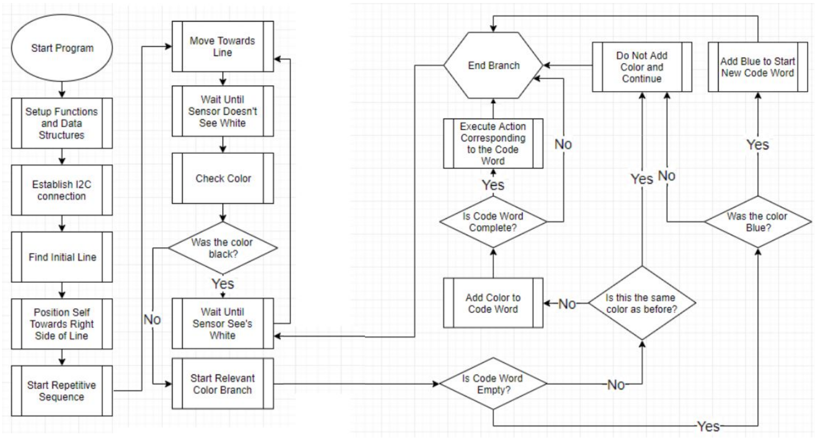

Start Program
Setup Functions and Data Structures
Establish I2C connection
Find Initial Line
Position Self Towards Right Side of Line
Start Repetitive Sequence
Move Towards Line
Wait Until Sensor Doesn't See White
Check Color
Was the color black?
Yes
No
Wait Until Sensor See's White
Start Relevant Color Branch
End Branch
Execute Action Corresponding to the Code Word
Yes
No
Is Code Word Complete?
Add Color to Code Word
No
Is this the same color as before?
Yes
No
Do Not Add Color and Continue
Add Blue to Start New Code Word
Yes
Was the color Blue?
Is Code Word Empty?
No
Yes

Figure 6. Microcontroller Flow Chart

2.6 Servo Motor

The servo motor would receive pulses from the microcontroller to control the angle that the motor turns too. Pulses are sent every 20ms and vary between 0.6ms to 2.05ms in length as shown in Table 2.

Angle	0 degrees	45 degrees	90 degrees	135 degrees	180 degrees
Pulse Width	0.60ms	1.00ms	1.40ms	1.75ms	2.05ms

Table 2. Servo Motor Control Pulses

2.7 Physical Design

The physical shape of the robot used for testing the system had the two DC motors near the back with the control board on top. The color sensor is at the front of the robot near the surface. The servo motor has an arm attachment build onto it so that the robot can grip objects.

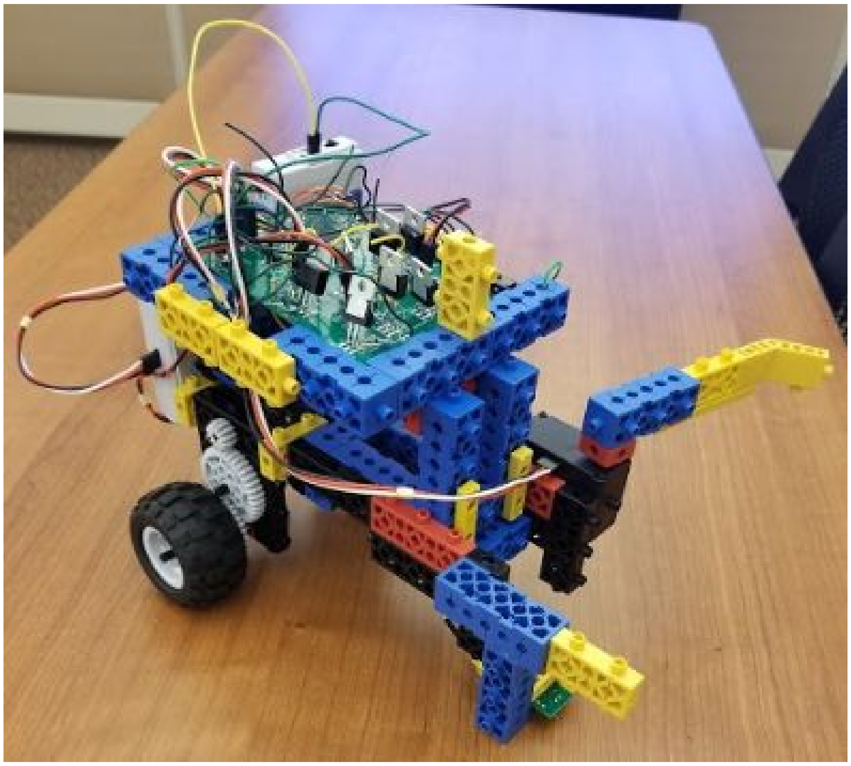


Figure 7. Physical Build of Robot

3 Design Verification

3.1 Power Module

The voltage regulators in the power module needed to output a steady 3.3 volts over the range of the battery voltages. The voltages ranges from 1.2 volts to 1.5 volts per battery, which is 4.8 volts to 6 volts in total. Testing was done between 4.5 volts to 7 volts as shown in Table 3.

Input (Volts)	Output from Power Module (Volts)
4.5	3.279
5	3.282
5.5	3.285
6	3.288
6.5	3.292
7	3.295

Table 3. Voltage Regulator Tests

3.2 DC Motor Control

The current that flows into the H-bridges is around 232 mA and the motors are strong enough to move the car. Also direction change is successfully done using four bidirectional outputs. An example code of the control function is shown below:

```
void turnRight() {//this would have
//the left motor go forwards and the
//right motor should stop
      LATAbits.LATA0 = 0;
      LATAbits.LATA1 = 0;
      LATBbits.LATB0 = 0;
      LATBbits.LATB1 = 1;
}
```

3.3 Color Sensors

The color sensor can successfully communicate with microcontroller by sending the device ID (0x44) in ID register to the microcontroller. Also it can differentiate different color and values can be read through oscilloscope. The following example shows the reading 0010010(36 in decimal) of R register for color blue in the oscilloscope.

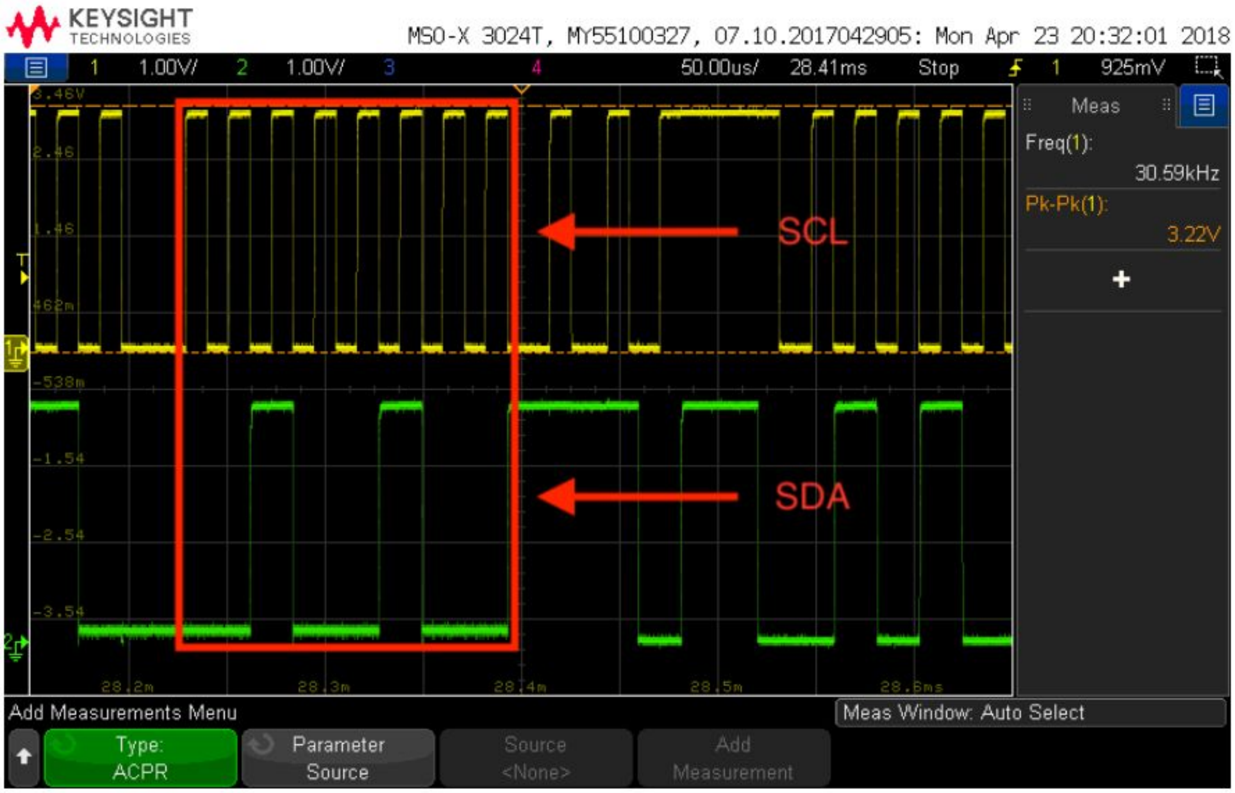


Figure 8. Oscilloscope Reading of Red Register for Color Blue

We recorded the values for seven different colors. The recorded values are shown in the following figure. One problem to notice is that the RGB values do not follow the real values in theory. One possible reason is that we use a blue LED to illuminate the color. Also we only use the lower 8 bits in RGB registers because they are sufficient to distinguish colors. However the real data should be in 16 bits.

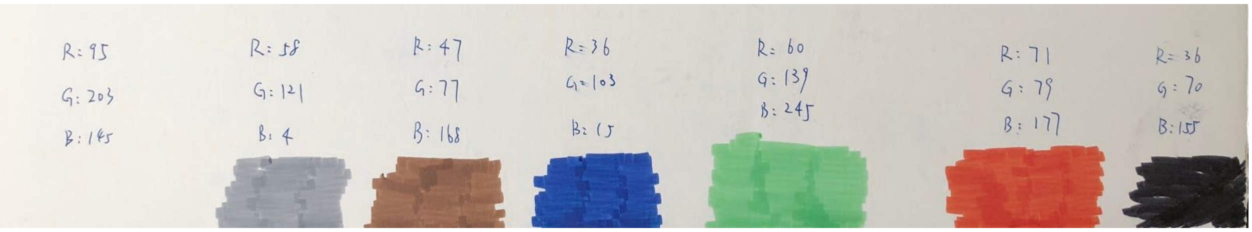


Figure 9. RGB Readings for All Colors

	White	Gray	Brown	Blue	Green	Red	Black
R	95	58	47	36	60	71	36
G	203	121	77	103	139	79	70
B	145	4	168	15	245	177	155

Table 4. RGB Readings for All Colors

3.4 Microcontroller

We designed our PCB poorly so that our microcontroller didn't actually fit into our PCB. Specifically, our PCB was designed to accommodate MCU in QFN package but the MCU we actually ordered is in TQFN package. What we were able to do is solder a corner of the MCU to the PCB, specifically the pins outline in red in Figure 10 and had to forcefully solder a wire to the pin outlined in blue in Figure 10. The system was able to work on a breadboard setup shown in Figure 11. We tested everything on breadboard because most parts are in fixed places but when we moved everything onto the pcb and tried to solder them, the Vcap pin broke several times and is sometimes disconnected. The Vcap pin is the CPU logic filter pin which is essential for the MCU to operate. So our MCU could not function properly without a stable Vcap connection.

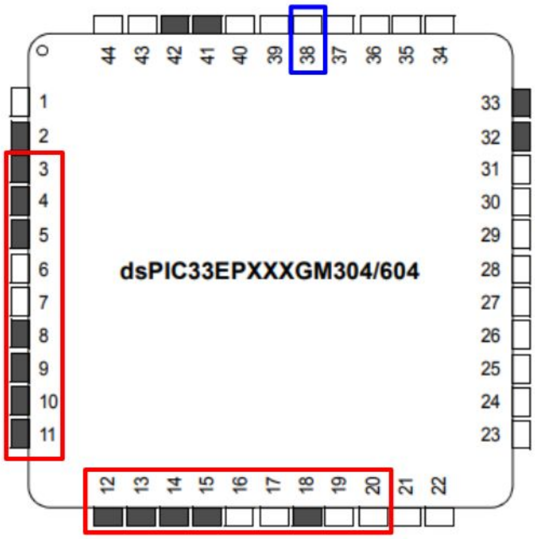


Figure 10. Microcontroller Pin Reference

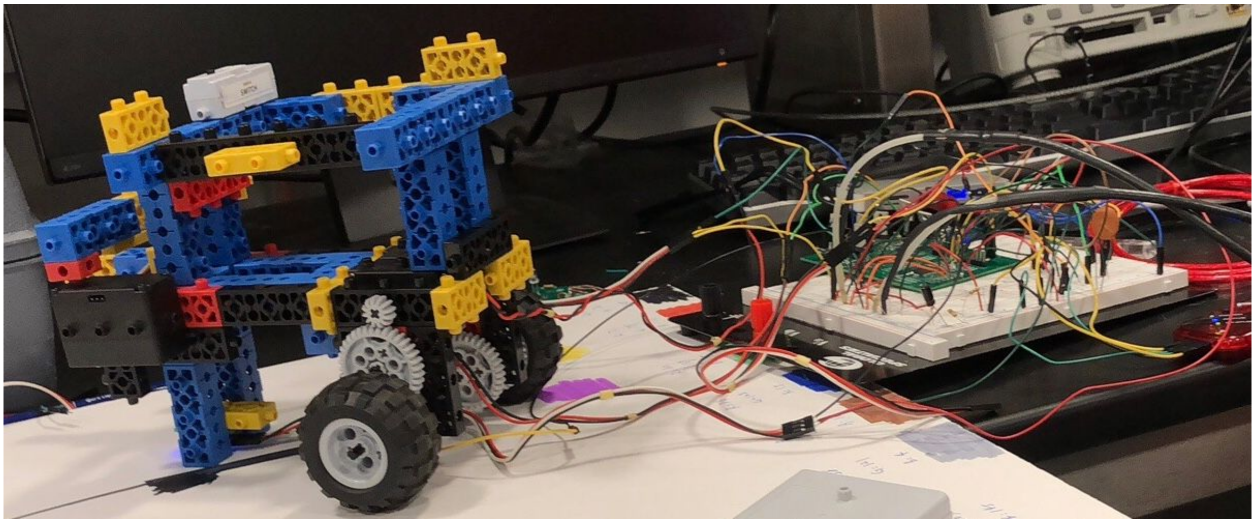


Figure 11. Breadboard Testing Setup

3.5 Servo Motor

Since the smallest pulse length that could be formed using the microcontrollers build in delay function was 1.45ms, a custom pulse forming function needed to be made in order to form the pulses mentioned in Table 2. Forming the pulses this way took advantage of the microcontrollers natural delay in execution times.

4 Costs

4.1 Parts

Items	Quantity	Unit Cost ($)	Total Cost ($)
Voltage Regulator	2	0.69	1.38
Diode	4	0.97	3.88
Transistor	6	0.57	3.42
Microcontroller	2	5.89	11.78
Servo Motor	1	15	15
Motor	2	8	16
Color Converter	1	1.71	1.71
LED	1	0.36	0.36
Button	1	0.5	0.5
PCB	20	1	20
Total			**74.03**

4.2 Labor

Name	Weekly Hours (hrs)	Hourly Rate ($)	Weeks	Cost ($)
Anthony	25	20	8	4000
Danson	25	20	8	4000
Totals	**50**	**20**	**8**	**8000**

4.3 Grand Total

Total Cost = Labor Cost x 2.5 + Part Cost = **$20074.03**

5 Conclusion

5.1 Accomplishments

Doing a live demonstration of the system was not accomplished due to a fault in integrating the microcontroller onto a PCB that resulted in a stop in the program when an I2C connection attempted to be established. What was able to be accomplished is effective movement of the robot with DC motors controlled by the microcontroller. The microcontroller was able to control the servo motor accurately to angles of 45 degree increments between 0 degrees and 180 degrees. The colors white, black, red, green, blue, brown and gray were differentiable by the color sensor. Lastly the power module was able to consistently provide a 3.3 volt power source to the color sensors, the microcontroller and the servo motor.

5.2 Uncertainties

There are still some uncertainties in the system that haven't been solved. First is the overall lifespan of the components and how long it can be used in a single period of time. Our tests with the system have only had it on for roughly 3 hours at a maximum, which is a sufficient amount of time for most users although there may be cases of someone wanting to use it for 4-8 hours at a time. Even if a user only played with the system for an hour a day, it is unknown for how many days it can do this until replacement or maintenance is necessary.

Another uncertainty is the reason for the color sensor RGB readings. For example, the color blue had an abnormally small blue register reading while the color green had an abnormally large blue register reading. The readings were consistent so differentiating colors was still possible, although this may be an issue when using different types of markers with the system.

5.3 Safety and Ethical considerations

This project is meant to be used by children around the ages of 4-8 years old so we do have an obligation to make sure that everything is safe to use. This includes making sure there are no dangerous amounts of current or voltage present that might cause harm to the child as well as making sure that temperatures of all devices stay at a comfortably safe level. Current around 20-100mA can be lethal to the human body although our devices must not only prevent death but also ensure no children are even slightly hurt by using them [3]. We will ensure that there is never a situation where even 5mA has a chance to flow through the user. We will have cases to prevent users from getting touch with the circuit and also short circuit protection. The two areas where current flows the most would be directly from the batteries and from the DC motor ports. To safely protect users from these areas the ports must be encased such that the leads cannot be touched while the casing is on.

IEEE code of ethics states that, “to improve the understanding by individuals and society of the capabilities and societal implications of conventional and emerging technologies, including intelligent systems.” [5] Our project aims at help kids to explore their capability of coding when they did not start to realize and understand their ability in this field. We strive to develop a straightforward and obstacle-free platform for these kids. In the foreseen future, skill of coding would be essential for new technologies in science. We hope our project can serve as a good introductory product for kids. IEEE code of ethics also states that, “to hold paramount the safety, health, and welfare of the public, to strive to comply with ethical design and sustainable development practices, and to disclose promptly factors that might endanger the public or the environment.” [5]

5.4 Future Work

There is a lot of room for improvement off of the work from this project. A good starting point for future improvements involve fixing technical issues that were mentioned earlier in the paper. There is also a need for scaling down the size of the control board. The control board is currently 9.5 cm by 7.5 cm in area, which is a large amount of space to take up on a simple machine or robot. Scaling down the size to be 5 cm by 5 cm in area would allow the control board to integrate into a larger variety of build designs. The work done in this project integrated well with the Robotori building platform, although it would be more appropriate to have the system integrate with a variety of building platforms such as LEGO, K’Nex and VEX. Creating a casing for the control board and sensors that could seamlessly connect with the building blocks of other platforms as well as creating easy connections or adaptors for the motors on those platforms would significantly expand base of people who could apply this system. Lastly, to expand on the programming element, there could be more color code words to allow users more flexibility in what their builds could do as well as a creating tiers of color code work complexity to create a better learning experience for users.

References

[1] Farr. ‘What’s the right age for kids to learn to code?’. 2013. [Online]. Available: https://venturebeat.com/2013/06/25/whats-the-right-age-for-kids-to-learn-to-code-debate/

[2] ‘Education by Robotori’. [Online]. Available: http://www.robotori.com/web_eng/main/main.asp

[3] ‘Electrical Safety: The Fatal Current’. 2014. [Online]. Available: https://www.physics.ohio-state.edu/~p616/safety/fatal_current.html

[4] Ouimette. ‘What Size Is Considered to Be a Choking Hazard?’. 2017. [Online]. Available: https://www.livestrong.com/article/532717-what-size-is-considered-to-be-a-choking-hazard/

[5] IEEE. ‘IEEE IEEE Code of Ethics.’ [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html

Appendix A Requirement and Verification Table

Module Name	High Level Requirement
Microcontroller	● The microcontroller must be able to appropriately differentiate between red, blue, green, black, gray, and brown from an I2C signal sent by the Color Sensors. ● The microcontroller must be able to interpret color codes of up to four colors long and make decisions on how to control the motors as a result. ● The microcontroller should be able to control a servo motor with a PWM signal. ● The microcontroller must be able to control DC motors by sending signals to the DC motor control module
Color Sensor	● The Color sensor module will be composed of color sensors that must be able to differentiate between red, blue, green, black, gray, and brown colors and send an appropriate I2C signal to the microcontroller.
DC Motor Control	● The DC motor control module must be able to direct sufficient current to two DC motors individually and be able to control their rotation direction.
Power Module	● The power module must provide sufficient power to the microcontroller, sensors, servo motor, and DC motor control.