

URL of Youtube Video.....	1
Purpose.....	1
Functionality.....	1
How Tested.....	2
Challenges/Lessons Learned.....	3

URL of Youtube Video

<https://youtu.be/w87ad7NRsu8>

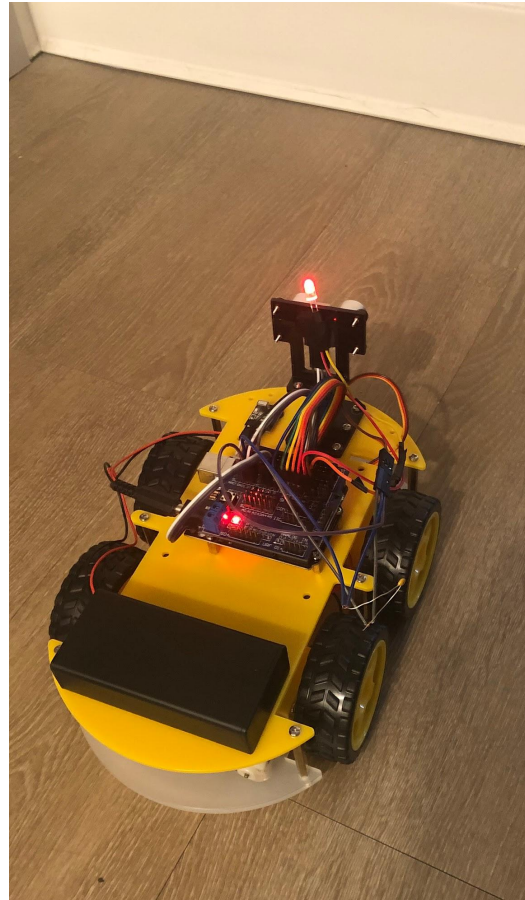
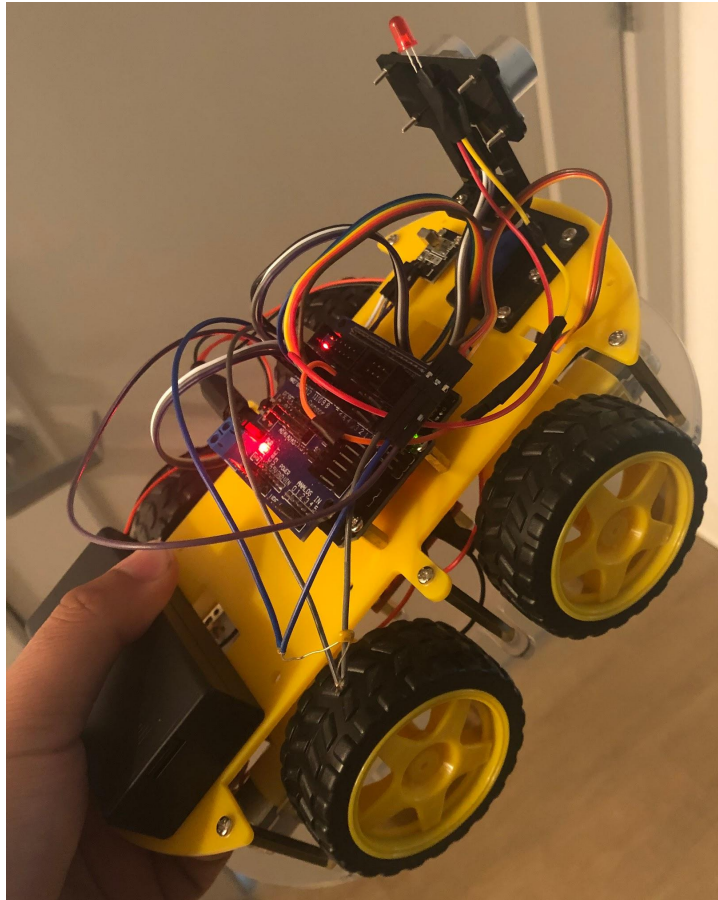
Purpose

Creating an Remote Controlled car that can be ran with a controller that uses IR. As well as implementing a feature that stops the car from hitting an object when moving forward.

Functionality

The car runs on a L298N Motor Driver which is controlled by the inputs of the controller through an IR Receiver. The input on the controller will start the motors to either go forward, backwards, spin left, spin right, or park and the direction can be changed as it's moving also. There's also a set of buttons that can change the speed of the current direction of the RC car, unfortunately the motor has trouble running low speeds so I have to program it to stay in a pretty high speed range. There is also a set of buttons that can turn the Servo Motor left or right, which is attached to the Ultra Sonic sensor. The Ultra Sonic sensor, attached to the front, detects how close an object is in front of the RC car in inches. The Ultra Sonic sensor triggers when an object is detected 15 inches or less. When it triggers, the motor stops running if and only if the motor is going forward and also turns on a red LED at the front to alert the user. When the LED is on, the

car can no longer work to go forward, instead the user can go back, spin left, or spin right to get away from the object without having to move the RC car by hand.



How Tested

I first tested each hardware component to make sure it works before hand and also understand how to use it and what I can do with them. Then I planned a diagram that describes all the functionalities, putting each functionality in it's own state machine. Afterwords I would plan how each state machine would interact with one another. In addition, I also commented any potential issues for each state to keep in mind when writing the code so I can minimize and problems at the end. I then tested the car each time I added a new function by hardcoding them first time to run this functionality. Then tested it again with input from a controller and the rest of

the system for any software or hardware issues. Once I was at the final stage, I ran the RC car with in both a small and large area to see how it runs for both a short and long period. As well as testing all the features of the RC car in different combinations to make sure that everything functions as it should.

Challenges/Lessons Learned

The biggest initial challenge for me was working with so many different hardware components in one Arduino project. This also made me prepare a lot more when it came to having the hardware components working with one another. With so many components working, this was also the first time I ran into an issue where one component was getting interference from another component. I received help from a friend of mine not in the class who is an Electrical Engineering major to understand why and how to fix the interference. Also I had to make some adjustments to my code then originally planned when I ran into some problems with the hardware. Specifically being unable to run the motor at a lower speed as the motor no longer worked as intended. I couldn't figure out what was causing the issue as it was not the code so I just limited the speed in the higher range. I was also hoping to run the RC car with a Bluetooth Module but it seems to work only for Android and I have an Apple so I had to stick with using an IR receiver.