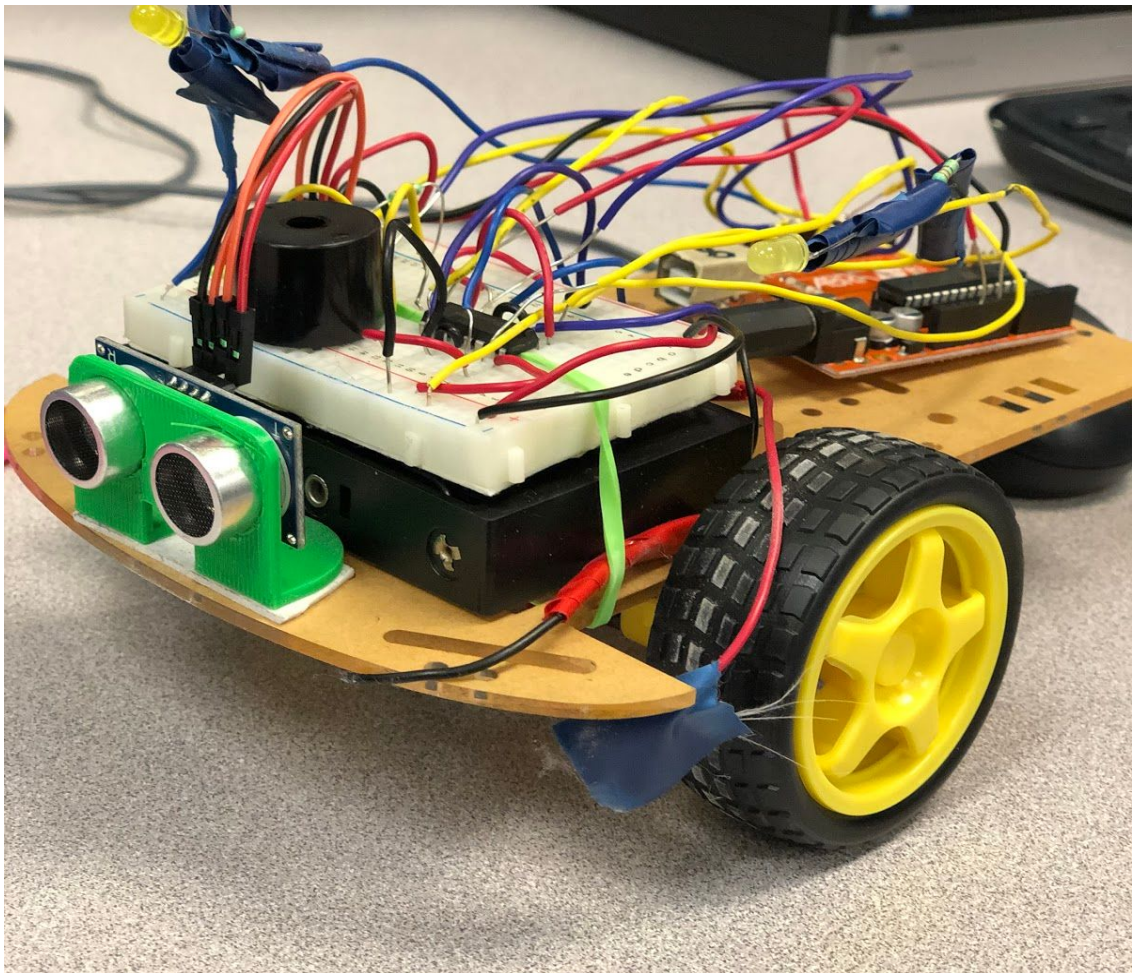


Autonomous Vehicle

01.22.2020

Sanya Sharma
Computer Engineering (TEJ3M0)
Teacher: Mr. Di Iorio

Table of Contents:

- Situation/Scenario Pg 2
- Problems/Possibilities Pg 2
- Investigation/ideas Pg 2-4
- Choose/Create Pg 4-5
- Evaluate Pg 5-6
- Designs Options Pg 7-8
- Final Design Pg 9
- Activity Log Pg 10
- Construction Log Pg 11-12
- Breadboard Diagram Pg 13
- Breadboard Schematic Pg 14
- Block Diagram Pg 15
- Operating Instruction Pg 16
- Original VS My code Pg 17-20
- Blank Page Pg 21

Situation/Scenario:

Write a programme to run an autonomous powered vehicle as a prototype for a new electric-powered vehicle. This is a green company that is looking for small prototype cars to showcase to their clients. The car must autonomously avoid obstacles in the way. The car could also be themed.

Problems/Possibilities:

The list of problems/possibilities:

- Strict deadline of January 20th, 2020 with limited time to work on it.
- Possible school strike
- In-class project
- 10 soldering iron
- Lack of variety of wire colours(only 3 options)
- Lack of knowledge on Ultrasonic Sensors and H-Bridge
- A possibility of getting a jumper wire stuck in one of the digital I\O pins in the Arduino Board.

Investigation/Idea:

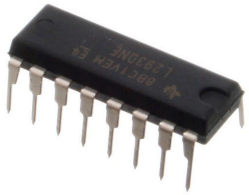
Idea

My idea is to make an autonomous car that avoids obstacles in the way. When the car faces any obstacles it will stop and turn back while playing a beep sound using the buzzer in the circuit. Along with that, I am going to be using LEDs to make headlights at the front of the car which will be turned on while driving forward, left or right.
In terms of the designs, I have two designs in mind which basically only differ from each other in terms of the placement of the breadboard, Arduino UNO and Ultrasonic Sensor.

Designs attached at the end

Research on components

H-Bridge



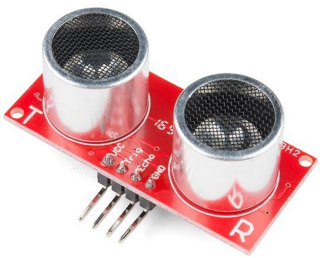
It helps the motor to turn in any direction. in 1 and pin 9, for being able to drive the motor, the pin 1 and 9 need to be high. For driving the motor with left H-bridge you need to enable pin 1 to high. And for right H-Bridge you need to make the pin 9 to high. If anyone of either pin1 or pin9 goes low then the motor in the corresponding section will suspend working. It's like a switch.

There are 4 input pins for l293d, pin 2,7 on the left and pin 15,10 on the right as shown on the pin diagram. Left input pins will regulate the rotation of the motor connected across the left side and right input for the motor on the right-hand side.

- **Pin 2** = **Logic 1** and **Pin 7** = **Logic 0** | Clockwise Direction
- **Pin 2** = **Logic 0** and **Pin 7** = **Logic 1** | Anticlockwise Direction
- **Pin 2** = **Logic 0** and **Pin 7** = **Logic 0** | Idle [No rotation] [Hi-Impedance state]
- **Pin 2** = **Logic 1** and **Pin 7** = **Logic 1** | Idle [No rotation]

VCC is the voltage that it needs for its own internal operation 5v; L293D will not use this voltage for driving the motor. For driving the motors it has a separate provision to provide motor supply VSS (V supply). L293d will use this to drive the motor. It means if you want to operate a motor at 9V then you need to provide a Supply of 9V across VSS Motor supply. I used the website RakeshRon. "L293D Motor Driver IC." *RON ROBOTICS*, 20 Apr. 2018, www.rakeshmondal.info/L293D-Motor-Driver for the information of the h-bridge.

Ultrasonic Sensor



There are four pins that you would use to interface with the sensor: VCC, Trig (signal output pin), Echo (signal input pin), and GND. Each of the four pins is connected to the Arduino: *VCC* to 5v, *Trig* to a digital pin, *Echo* to a digital pin, and *GND* to GND (ground). No resistors are needed, just pin to port.

I have used the codes from Pendergast, Robert L., et al. “Complete Guide for Ultrasonic Sensor HC-SR04 with Arduino.” *Random Nerd Tutorials*, 2 Apr. 2019, randomnerdtutorials.com/complete-guide-for-ultrasonic-sensor-hc-sr04/ to activate and get data from my ultrasonic sensor. The sensor sends waves sound to check how far an object is from it and uses an equation to measure the distance. Using that data I would prevent my vehicle from bumping into an object/person.

Choose/Create:

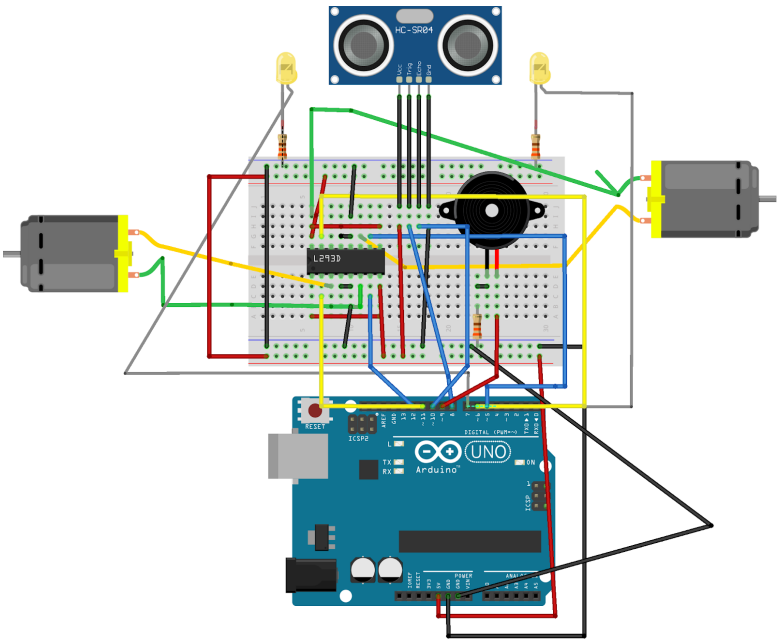
I chose to do the design with the breadboard on top of the flipped battery pack since it is less space-consuming while giving me space for the wires and the switch placement. This design doesn’t require long wires and is easy to fix or improve on anything later in the future.

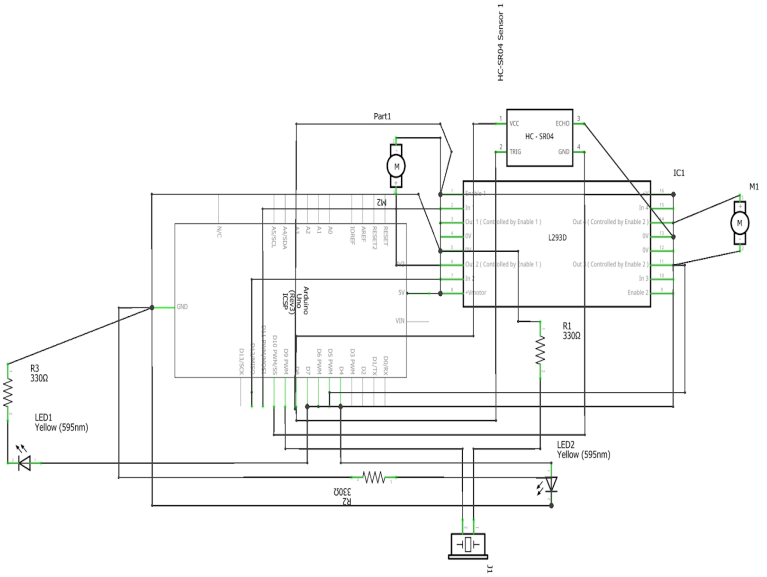
List of parts and components:

- 1x BreadBoard (mini)
- 1x Arduino UNO
- 2x Motor
- 1x Buzzer
- 1x H-Bridge(L293D)
- 1x Ultrasonic Sensor(4 Pin)
- 2x Yellow LEDs
- 2x 330Ω resistor
- 1x 100Ω resistor
- Jumper Wires (approx 20 and more of different length/ Must have red, black and other colour wires)
- 2x wheels + 1 360° wheel
- 1x Car Frame
- 1x Battery Pack with DC Barrel Jack(Supporting 4 AA batteries)
- 1x Switch
- 2x non-polarized Capacitor\
- 1x USB cable to upload the code from the computer
  1x Computer
- 1x Ultrasonic Sensor Holder

The final design will be available at the end of the report

Below is the schematic and the breadboard design of the vehicle:





The full photo will be available at the end of the report
Activity and construction log will be available at the end of the report

Evaluate:

First when I tested my car nothing was working except the LEDs. I check my circuit and code multiple times but failed then I got a brand new h-bridge from the teacher and wired it. After the replacement, the left motor/wheel started to spin. Since my right motor/wheel wasn't spinning I knew there was something wrong with my writing and it was. The wires of the right motor were not connected with the h-bridge in the correct position. After fixing the wiring I placed my batteries in the battery pack and uploaded my code. As I did it the car worked but not as expected. The car was only able to go left but not right, forward and backwards. I knew there was something wrong with my wiring so I checked the placement of my wires and found out that my right motor is not getting enough power to spin and because my left wheel is spinning faster than the

right one it is only able to turn right. To correct it I fixed the placement of my red wires coming from the positive side of the breadboard. While typing this I am at home and didn't get a chance to try my modified code with the buzzer working and modified wiring so I don't know if it worked out. **Update:** The right motor started to spin but just a little slower than the left one. Despite the fact that the right motor is spinning slow the car is able to go forward, backward, right, left and most importantly it senses obstacles and avoids them.

Reflection:

This project was some up of our whole TEJ3M0 course. The knowledge from our hardware, electronics labs, Logic Gates and Arduino labs were used throughout the project. During this project I got a better understanding of how to make functions, use h-bridge for different movements of a motor and use distance data from the ultrasonic sensor to avoid bumping into obstacles.

The two strengths of my design is that firstly the placement of my components such as Battery Pack, breadboard and Ardunio was space efficient which made it easy to access the USB port on the Arduino to upload the code and battery pack to have the batteries hidden which was presented in such a way that it just looks like a stand for the breadboard and secondly my program is easy to understand. My code only consists of basic Arduino functionality with well-written comments for each step making it easy to modify in the future.

The two weaknesses in my circuit is firstly my wires are too long making which could frequently cause a wire to be pulled from its position and hard to see which wires is connected to what. Secondly, there is a problem with my right motor which is causing it to spin slower than the left motor making it to gradually turn right while going forward. This was caused due to an error during solder the wires to the motor.

Communication: I would give myself a 9/10. My S.P.I.C.E report in the proper format, well detailed and no spelling/grammar mistake. I don't have various pictures to show my vehicle's progress.

Sketches: I would give myself a 5/5 since my diagrams are easy to follow and well organized.

Wire/Cable Management: I would like to give myself a 3/5 since I have issues with the right colour and the organization of my cables.

Construction Chart and Technical skills in soldering. Professionalism (attendance, work habits, respect for peers/equipment/ safety): I would like to give myself 15/15 because through the project I maintained my professionalism by my work habits, respects to my peers by sharing limited tools and following the safety precautions.

Programming Requirements: I would like to give myself 10/10 because my program 1met all of the listed expectations for my program to have.

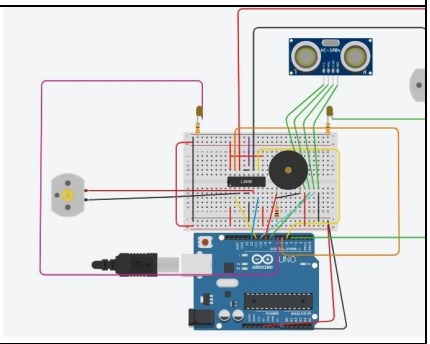
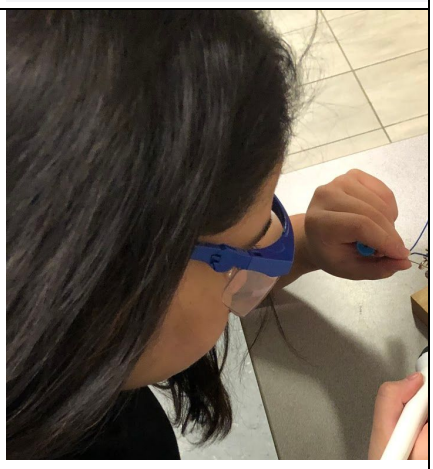
Circuit Summary, Block Diagram and Operating Instructions: I would like to give myself 10/10 because my circuit summary, block diagram and operation instruction were easy to understand.

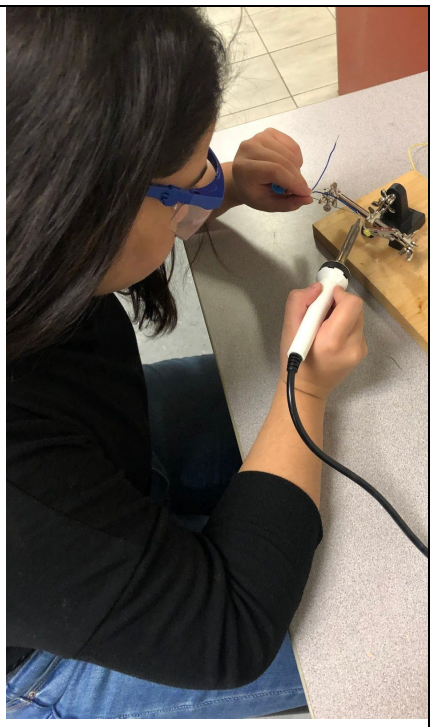
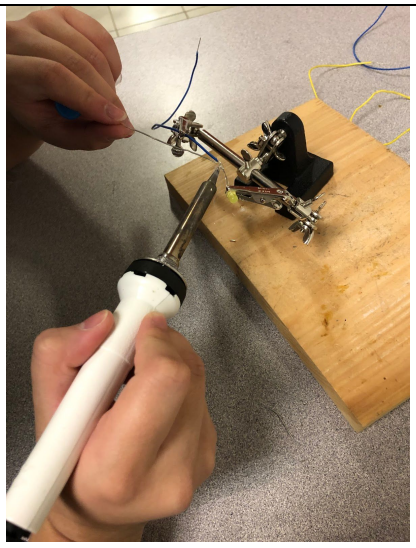
Activity Log

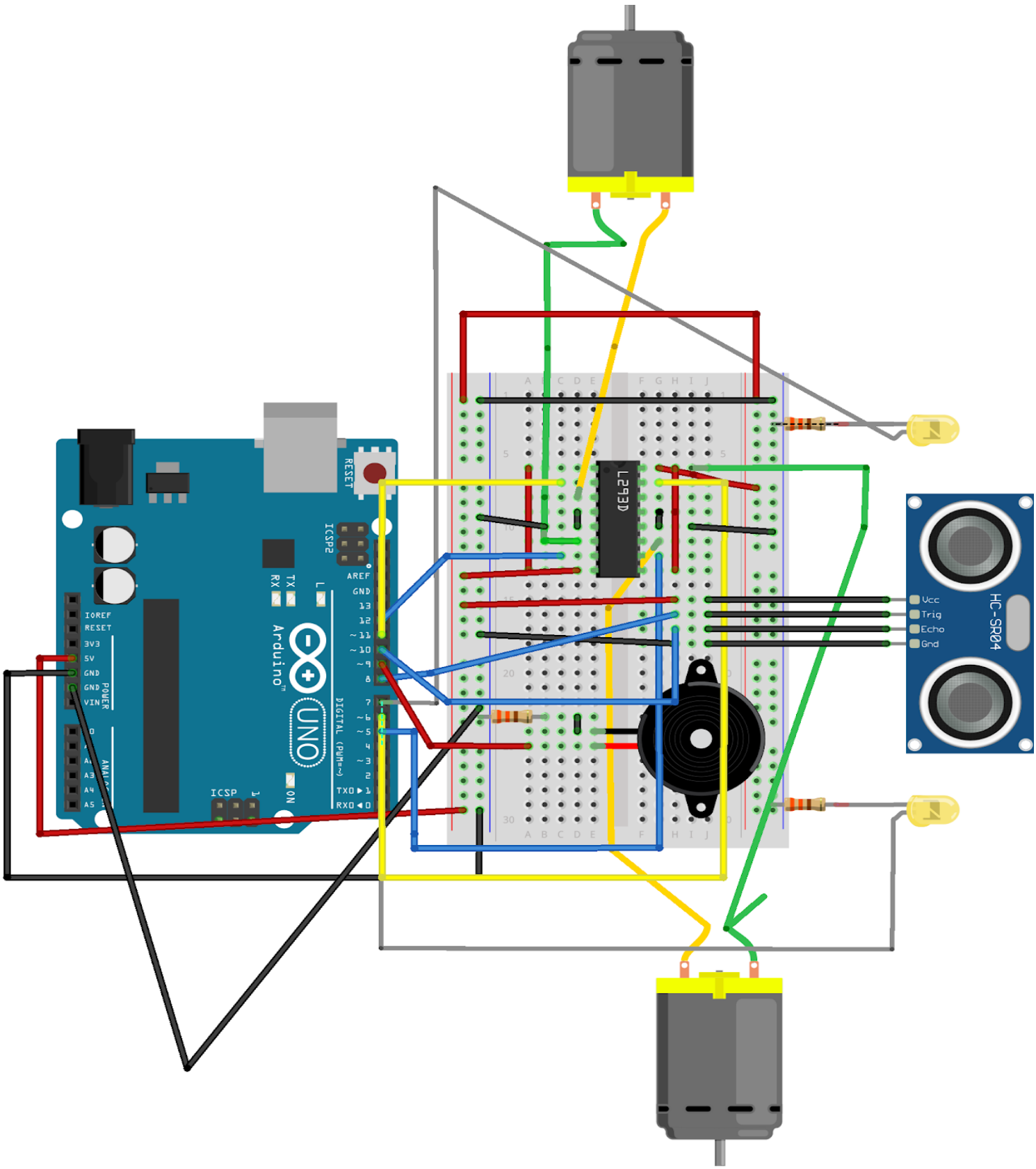
Date	Description of Work Done	Hours
January 7, 2020	Today I chose an idea for the appearance of the car. I started research on the h-bridge and Ultrasonic sensor but couldn't complete it.	30 min
January 9, 2020	I was sick today and was at home so I didn't work on it for a long time. I completed my research on the ultrasonic sensor and the h-bridge.	1 hour
January 10, 2020	I worked on the designing part of the car to know what component will go where. I was constantly forgetting to draw components in my design so I made a list of components for the car along with where they are placed.	2 hour
January 12, 2020	I worked on the placement of my components on tinkercad and it's done. I started wrong on my code too and I was able to finish the code for forward and backward using the ultrasonic sensor readings.	3 hours
January 13 , 2020	Today I soldered my motors and switch. To solder the LEDs and resistors first I checked my 6 LEDs to see if they work.	72 min
January 14, 2020	Today I assembled by car. I screwed in the back wheels, placed Arduino, battery pack, motors and breadboard.	72 min
January 15, 2020	Today I soldered my two led and resistor for the extension part. At home ok completed my code for the car.	2 hours
January 16, 2020	Today I started working on making my circuit. I am 70% done.	72 min
January 17, 2020	Today I finished wiring my circuit and uploaded the code. The motors didn't work.	72 min

Total Hours:	
Student Signature:	
Teacher Signature:	

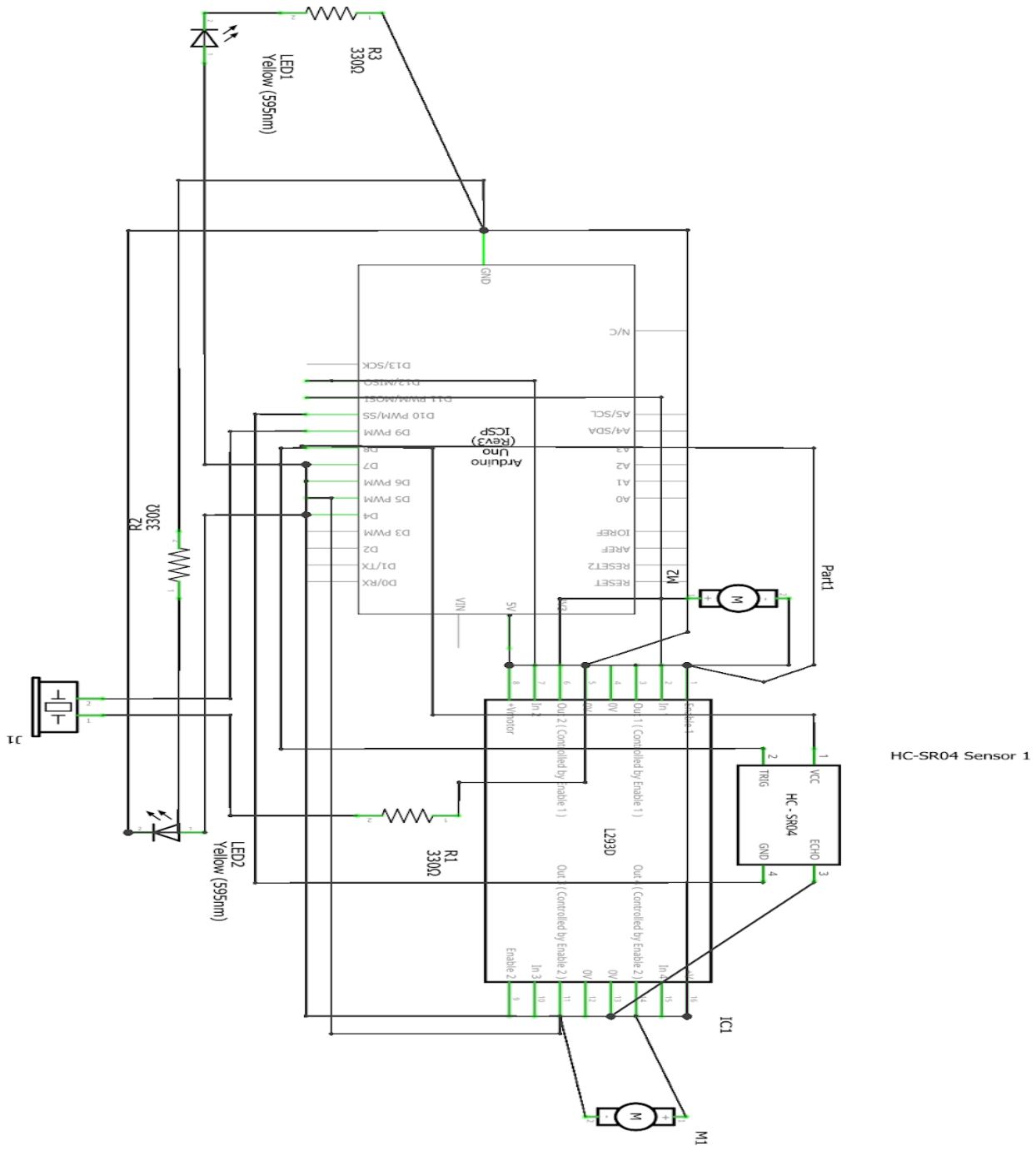
Construction Chart

Step #	Operation	Tips / Details	Equipment / Tools	Safety	Pictures (if applicable)
1	Research	Get knowledge of the components being used.	Computer Internet	N/A	
2	Design	Plan the design of your circuit on paper to know the placement on each component on the car frame	Pencil Ruler Eraser Sharper Components for measurements	N/A	
3	Create	Use tinkercad/fritzing to make a circuit online and code to make it work before you start building anything	Computer Internet	N/A	
4	Solder	Solder 2 jumper wires to the motors. After, solder the legs a non= polarized capacitor on the same solder as the wires. Repeat these steps with the second motor.	Solder Iron Safety Glasses Helping hands 2x jumper wires 2x motors 2x capacitor	Well ventilated room Wear safety glasses Don't inhale the smoke from the solder Tin the tip of the iron before use Put a bit of solder on the iron and clean it before leaving it Plug off the iron if not in use Always assume that the iron is hot and never touch the tip. If burnt report the burn to the teacher. Do not burn/melt the motor	

5	Solder	Solar the switch to the red wires of the battery pack which is connected to the DC barrel jack using two jumper wires. Keep the length of the jumper wires in mind. It should be able to reach the Arduino	Solder Iron Safety Glasses Helping hands 2x jumper wires Switch Battery Pack	Well ventilated room Wear safety glasses Don't inhale the smoke from the solder Tin the tip of the iron before use Put a bit of solder on the iron and clean it before leaving it Plug off the iron if not in use Always assume that the iron is hot and never touch the tip. If burnt report the burn to the teacher.	
6	Solder	Solar a 330Ω resistor to the negative leg of a yellow led. Then solder a jumper wire to the other leg of the resistor and led. The jumper wire should be able to reach the digital I/O pin in the Arduino. Repeat these again with the second resistor and led.			

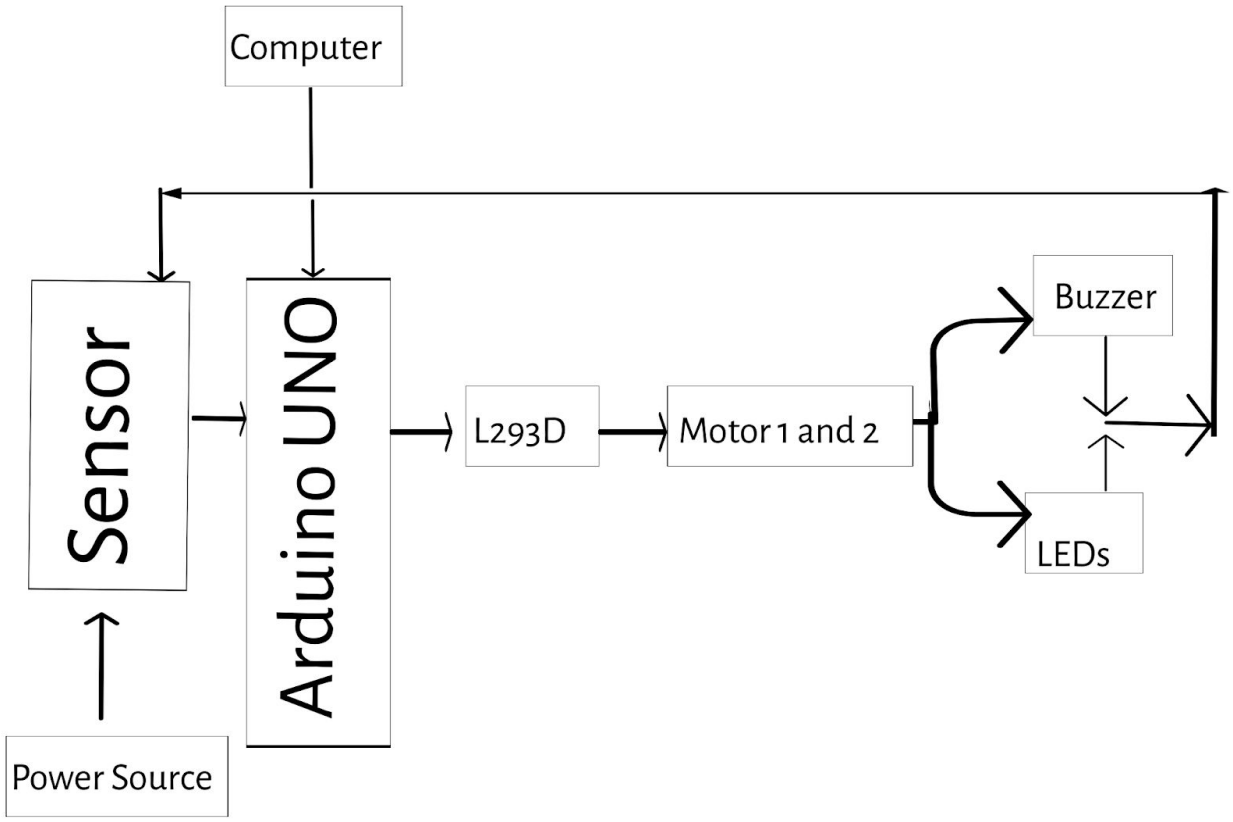

L293D
HC-SR04
Vcc
Trig
Echo
Gnd
Arduino
UNO
RESET
ICSP2
AREF
GND
13
12
~11
~10
~9
8
7
~6
~5
4
~3
2
TX
RX
L
IOREF
RESET
3V3
5V
GND
GND
VIN
POWER
ANALOG IN
A0
A1
A2
A3
A4
A5
ICSP
ON
DIGITAL (PWM~)
TXD 1
RXD 0

fritzing


LED1
Yellow (595nm)
R3
330Ω
GND
N/C
Arduino
Uno
(Rev3)
ICSP
R2
330Ω
Part1
M2
J1
HC-SR04 Sensor 1
HC-SR04
VCC
TRIG
ECHO
GND
L293D
R1
330Ω
LED2
Yellow (595nm)
IC1
M1

fritzing

Block Diagram



Operating Instruction

1. Upload the code from the computer to the Arduino UNO
2. Switch on the switch button
3. Step in front of the sensor to watch it avoid the obstacle

What is actually happening?

The circuit is getting all of its power from the 6 AA battery pack. When the code is uploaded the car start to follow the loop code on the program. In the program the Ultrasonic Sensor is continuously sending in hearable sound waves and uses a calculation to measure the distance between an obstacle. If the distance is 30 or less the car will stop. Earlier in the code we have used a function to produce a random number from 0 or 1. When the distance 30 or less is measured the code produces a random number and according to the number it either turns right or left and before turning it makes a beeping sound. If the sensor detects an object 30 or more centimeters away the vehicle will continue to work forward with LED headlights on.

ORIGINAL CODE PROVIDED:

```
const int motorPin1= 2;
const int motorPin2= 3;
void setup() {
// put your setup code here, to run once:
pinMode(motorPin1, OUTPUT);
pinMode(motorPin2, OUTPUT);
}
void loop() {
// put your main code here, to run repeatedly:
digitalWrite(motorPin1, LOW); //Motor 1 Forward
digitalWrite(motorPin2, HIGH);
delay(1000); //waits for one second or 1000 milliseconds
digitalWrite(motorPin1, HIGH); //Motor 1 backwards
digitalWrite(motorPin2, LOW); //Motor 1 backwards

delay(1000);
}
```

My CODE:

```
//Autonomous Vehicle
//Author: Sanya Sharma
//Date: January 22, 2020
//Description: The code below drives a car forward and when an obstacle is encountered it choose to
//turn left or right. While backing up before turning it makes beeping sounds.



const int rightFor = 12; //When the right wheel is turning clockwise it is on pin 12
const int rightBack = 11;//When the right wheel is turning anticlockwise it is on pin 11
const int leftFor = 6;//When the left wheel is turning clockwise it is on pin 6
const int leftBack = 5;//When the left wheel is turning anticlockwise it is on pin 5
const int trigPin = 8; // Trig pin on the Ultrasonic Sensor is on pin 9
const int echoPin = 10;// echo pin on the Ultrasonic Sensor is on pin 10
const int headA = 4;// ledA is on pin 4
const int headB = 7;// ledB is on pin 7
const int piezo = 9; // buzzer at pin 9
int nextNo = 2500;
int restart = 3000;
int beep = 2000;
long duration; //variable for the ultrasonic readings
int distance;//variable for the ultrasonic readings
long randNumber;//variable for the random number function

void setup()
{
  pinMode(rightFor, OUTPUT);//rightForward wheel is set as output
  pinMode(rightBack, OUTPUT);//rightBack wheel is set as output
```

```
  pinMode(leftFor, OUTPUT);// leftFor wheel is set as output
  pinMode(leftBack, OUTPUT);// wheel is set as output
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input
  pinMode(headA, OUTPUT);// ledA is set as output
  pinMode(headB, OUTPUT);// ledB is set as output
  pinMode(piezo, OUTPUT); // setting speaker as Output
  pinMode(headA, HIGH);// default status of led on
  pinMode(headB, HIGH);// default status of led on

  Serial.begin(9600); // Starts the serial communication
}

void stop() //Function to bring the speed of the motor 0
{
  analogWrite(rightFor, 0);
  analogWrite(rightBack, 0);
  analogWrite(leftFor, 0);
  analogWrite(leftBack, 0);
  delay(1500);
}

void goBackwards()// Function to move backwards
{
  digitalWrite(rightFor, LOW);
  digitalWrite(rightBack, HIGH);
  digitalWrite(leftFor, LOW);
  digitalWrite(leftBack, HIGH);
  digitalWrite(headA, LOW);
  digitalWrite(headB, LOW);
  tone(piezo, beep, 1000); // Code for the dash
  delay(1500);
  tone(piezo, beep, 1000); // Code for the dash
  delay(1500);
  tone(piezo, beep, 1000); // Code for the dash
  delay(1500);
}

void goForward()// Function to go forward
{
  digitalWrite(rightFor, HIGH);
  digitalWrite(rightBack, LOW);
  digitalWrite(leftFor, HIGH);
  digitalWrite(leftBack, LOW);
  digitalWrite(headA, HIGH);
  digitalWrite(headB, HIGH);
  delay(1000);
}

void turnRight()// Function to turn right
{
  digitalWrite(rightFor, LOW);
  digitalWrite(rightBack, HIGH);
  analogWrite(rightBack, 90);
  digitalWrite(leftFor, HIGH);
  digitalWrite(leftBack, LOW);
```

```
  analogWrite(leftBack, 90);
  digitalWrite(headA, LOW);
  digitalWrite(headB, LOW);
  delay(1000);
}

void turnLeft()// Function to turn left
{
  digitalWrite(rightFor, HIGH);
  analogWrite(rightFor, 90);
  digitalWrite(rightBack, LOW);
  digitalWrite(leftFor, LOW);
  digitalWrite(leftBack, HIGH);
  analogWrite(leftBack, 90);
  digitalWrite(headA, LOW);
  digitalWrite(headB, LOW);
  delay(1000);
}

void loop()
{
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  // Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(echoPin, HIGH);
  // Calculating the distance
  distance = duration * 0.034 / 2;
  // Prints the distance on the Serial Monitor
  Serial.print("   Distance: ");
  Serial.println(distance);
  randNumber = random(2);// Generate a random number from 0 to 2
  Serial.print("Random Number: ");// Print the number on the serial monitor
  Serial.println(randNumber);



  if (randNumber == 0 && distance <= 30)//if random number is 0 and distance between the object and
//the car is less than 20 following the following functions
  {
    stop;
    goBackwards();
    turnLeft();
    goForward();
    Serial.print("Left"); // Print the direction on the Serial Monitor
  }

  else if (randNumber == 1 && distance <= 30)// if random number is 1 and the distance between the
//object and the car is less than 20 following the following functions
  {
    stop;
    goBackwards();
    delay(1000);
```

```
    turnRight();
    goForward();
    Serial.print("Right"); // Print the direction on the Serial Monitor
   }

   else// if any of the above conditions not met just drive forward
   {
    goForward();
   }
}
```