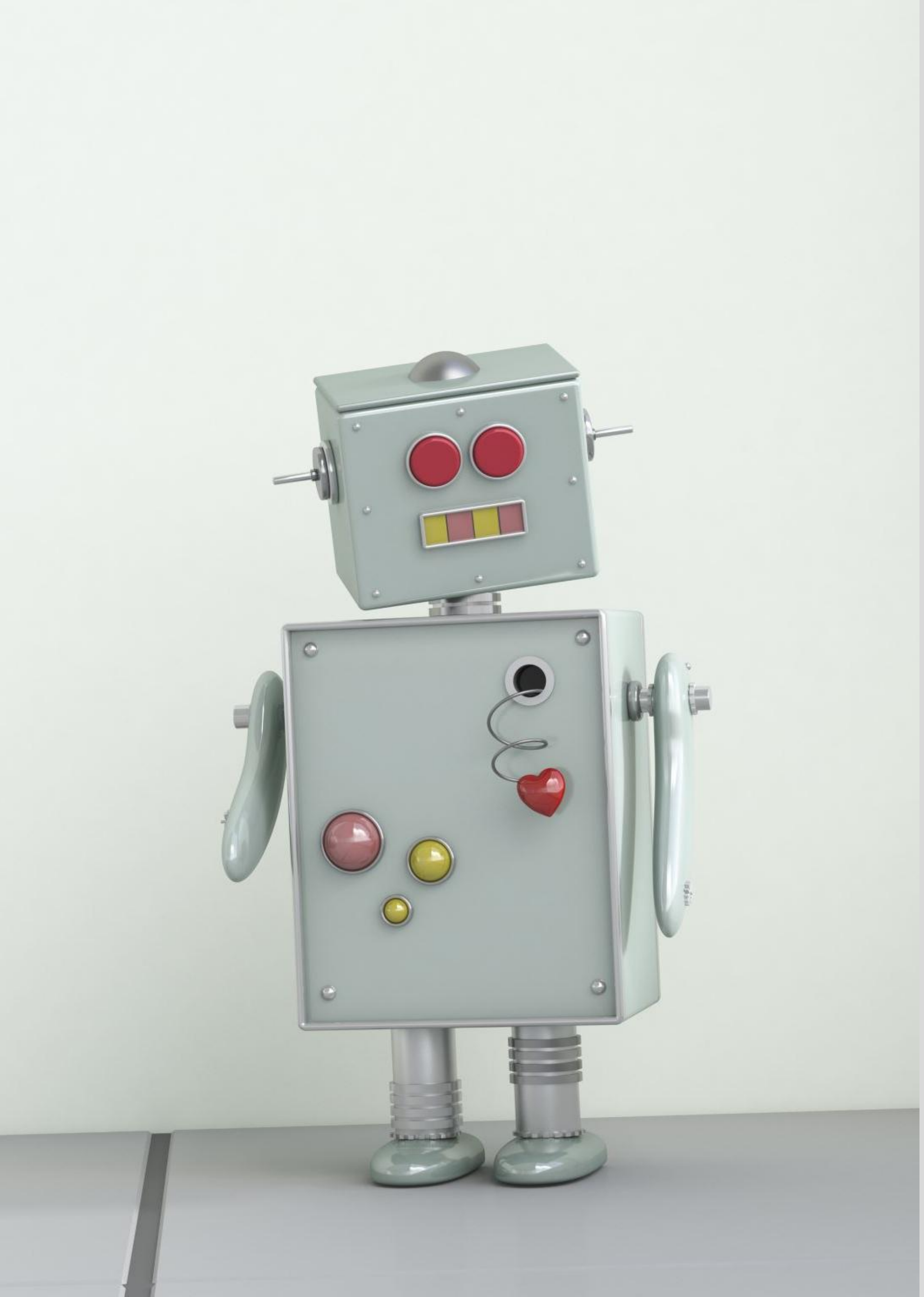


Obstacle Avoiding Robot Car Project

GROUP 3

ABDULRAHMAN QUTAH (2241580)

TURKI ALGHAMDI (2240517)

YOUSEF BASODAN (2243729)

NAWAF ALSHEHRI (2240403)

What is an Obstacle Avoiding Car?

- An obstacle avoiding car is an autonomous robot that uses an ultrasonic sensor to detect nearby objects and avoid crashing into them. It can stop or change direction based on sensor readings without remote control.

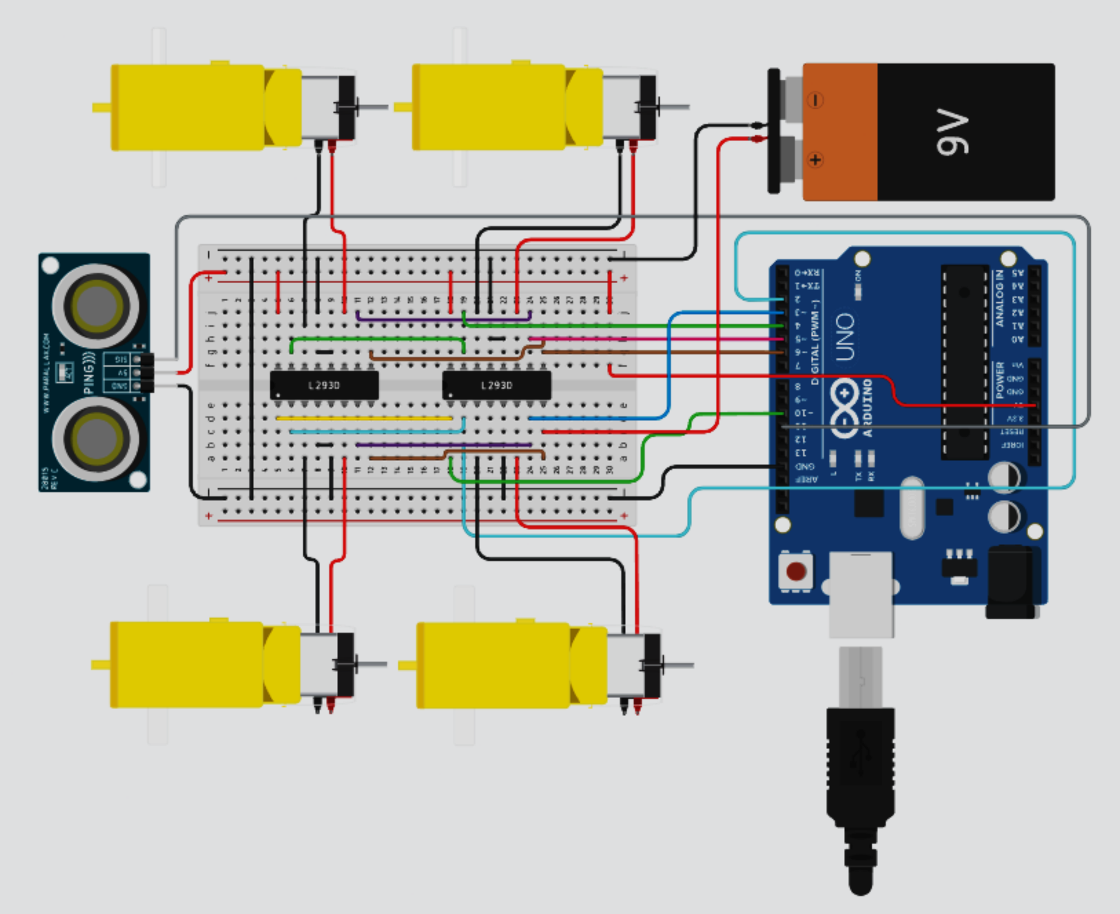
Hardware Components

- Arduino Uno
- 4× DC Motors
- L298N Motor Driver
- Ultrasonic Sensor (HC-SR04)
- Flame Sensor
- Jumper Wires
- Battery Pack
- Chassis + Wheels

NOTE: Same equipment's, but different body!

Wiring Diagram

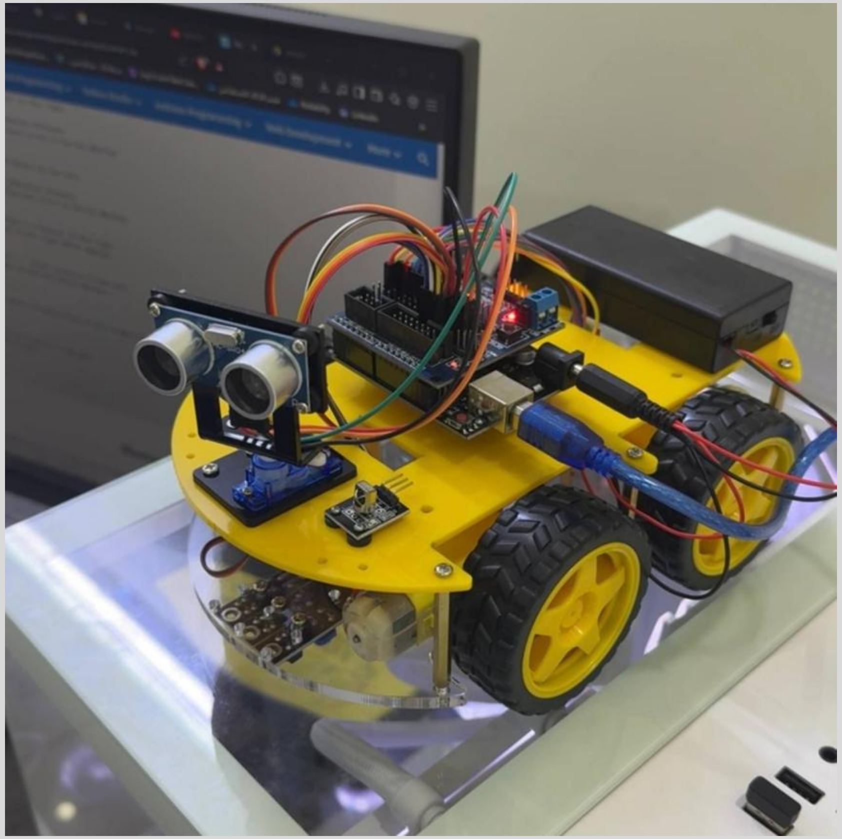
- DC Motors connected to L298N (OUT1-OUT4)
- L298N connected to Arduino (IN1-IN4 + ENA/ENB to PWM pins)
- Ultrasonic Sensor connected to Arduino (Trigger and Echo pins)
- Battery powers L298N, which drives the motors



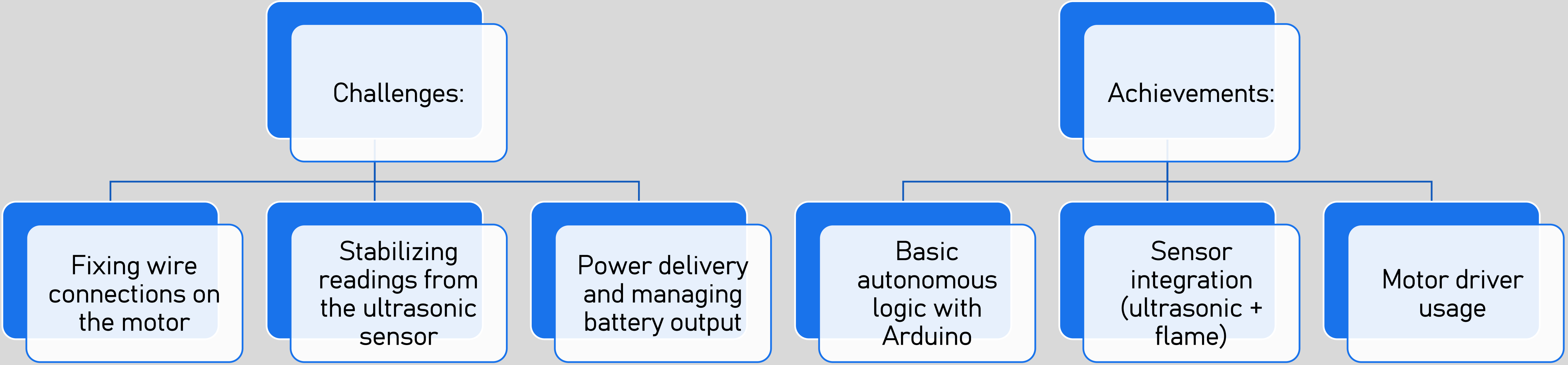
Working Logic

- Robot waits briefly after starting
- Ultrasonic sensor constantly checks for obstacles

- If obstacle is far: move forward
- If obstacle is close: stop or turn



Challenges & Results



Challenges:

Fixing wire connections on the motor

Stabilizing readings from the ultrasonic sensor

Power delivery and managing battery output

Achievements:

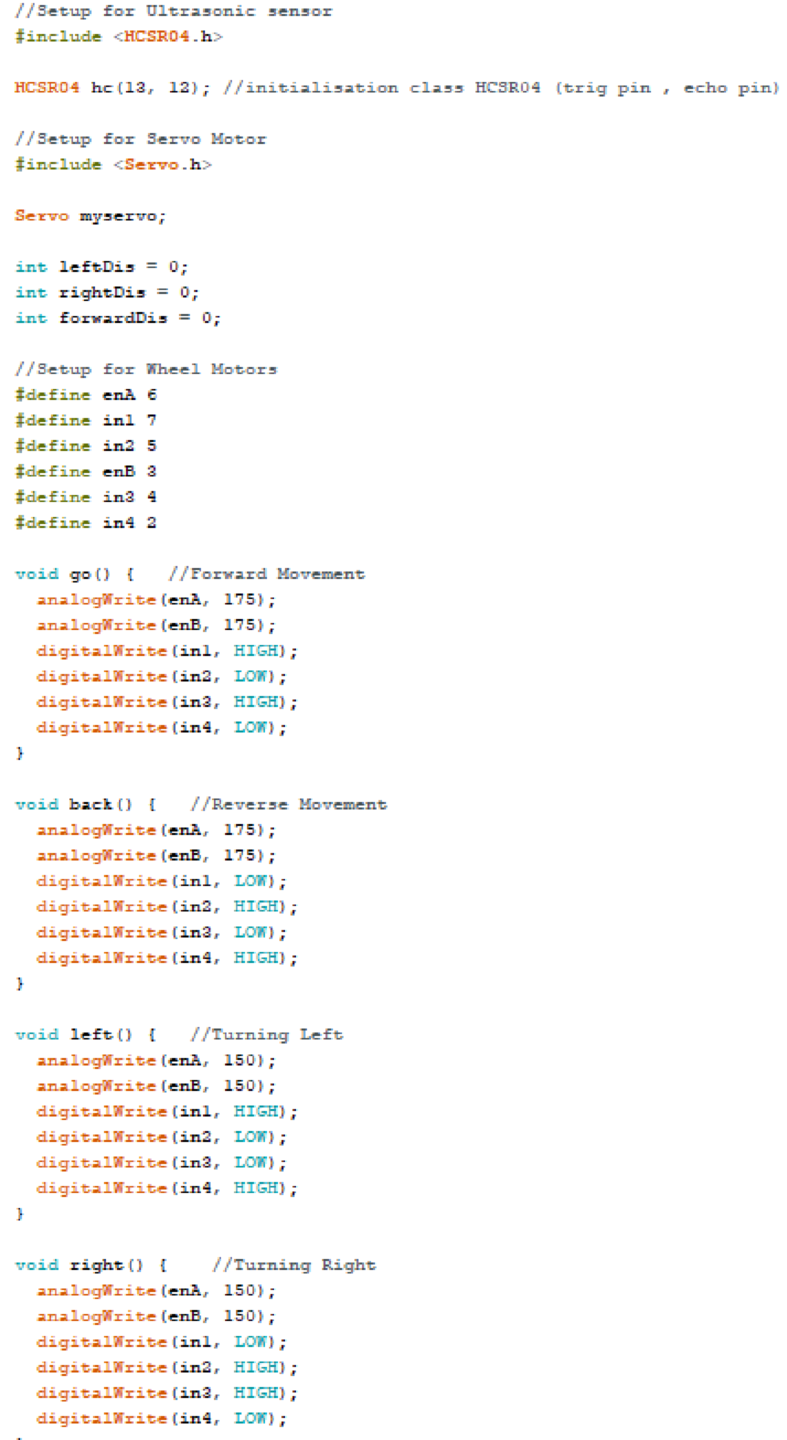
Basic autonomous logic with Arduino

Sensor integration (ultrasonic + flame)

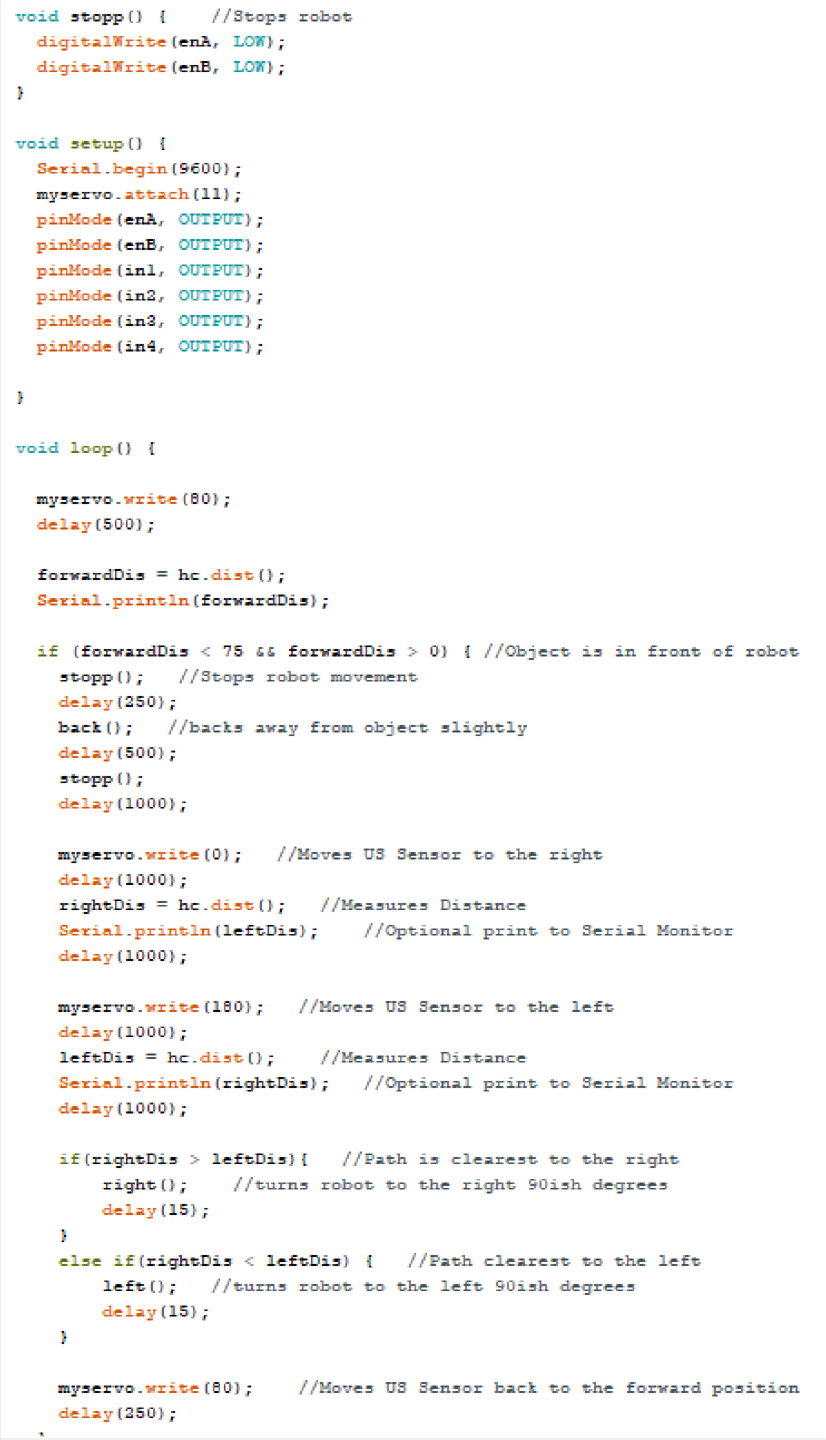
Motor driver usage

Result: The car avoids obstacles effectively using sensors and runs independently.

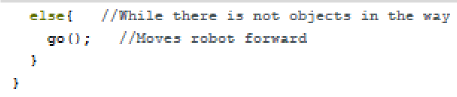
Code



```
//Setup for Ultrasonic sensor
#include <HCSR04.h>

HCSR04 hc(13, 12); //initialisation class HCSR04 (trig pin , echo pin)

//Setup for Servo Motor
#include <Servo.h>

Servo myservo;

int leftDis = 0;
int rightDis = 0;
int forwardDis = 0;

//Setup for Wheel Motors
#define enA 6
#define in1 7
#define in2 5
#define enB 3
#define in3 4
#define in4 2

void go() {   //Forward Movement
  analogWrite(enA, 175);
  analogWrite(enB, 175);
  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
}

void back() {   //Reverse Movement
  analogWrite(enA, 175);
  analogWrite(enB, 175);
  digitalWrite(in1, LOW);
  digitalWrite(in2, HIGH);
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
}

void left() {   //Turning Left
  analogWrite(enA, 150);
  analogWrite(enB, 150);
  digitalWrite(in1, HIGH);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
}

void right() {    //Turning Right
  analogWrite(enA, 150);
  analogWrite(enB, 150);
  digitalWrite(in1, LOW);
  digitalWrite(in2, HIGH);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
}
```

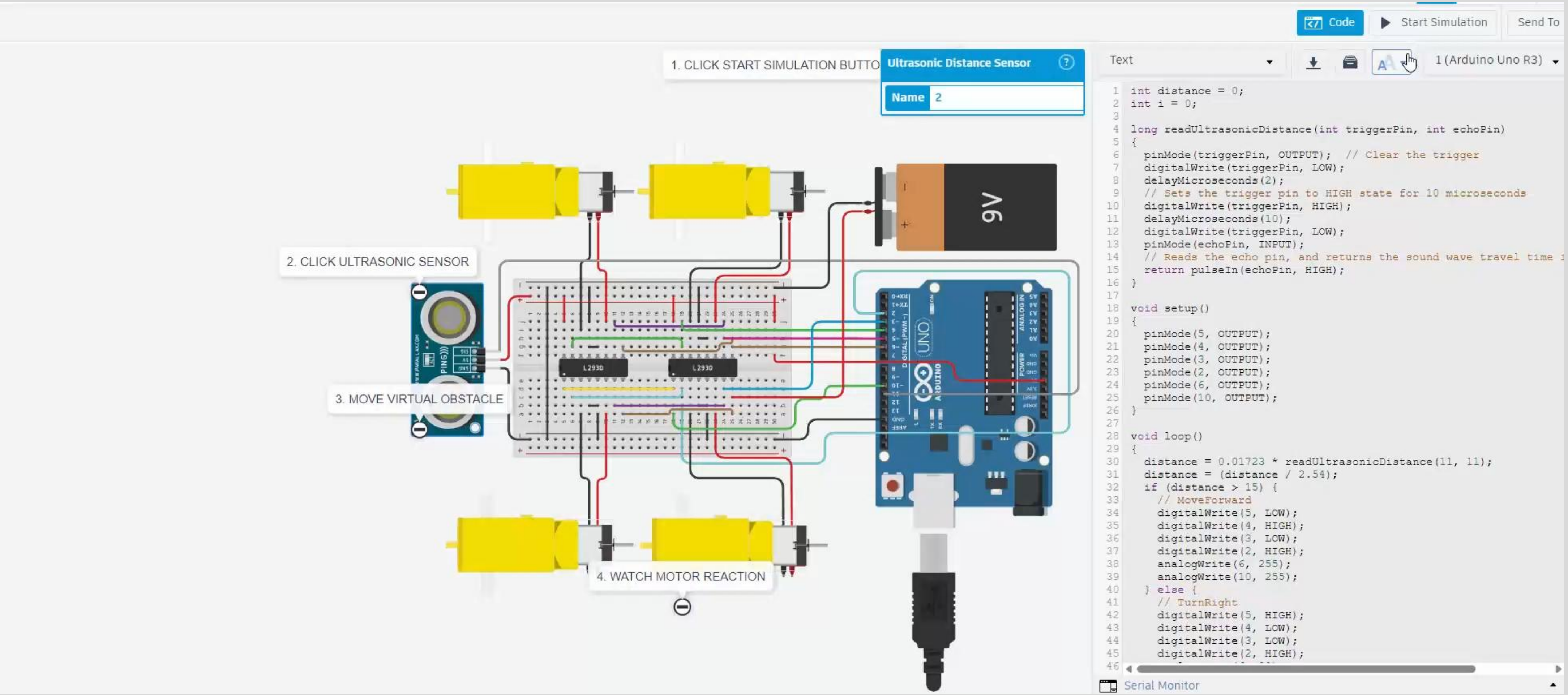


```
void stopp() {    //Stops robot
  digitalWrite(enA, LOW);
  digitalWrite(enB, LOW);
}

void setup() {
  Serial.begin(9600);
  myservo.attach(11);
  pinMode(enA, OUTPUT);
  pinMode(enB, OUTPUT);
  pinMode(in1, OUTPUT);
  pinMode(in2, OUTPUT);
  pinMode(in3, OUTPUT);
  pinMode(in4, OUTPUT);

}

void loop() {

  myservo.write(80);
  delay(500);

  forwardDis = hc.dist();
  Serial.println(forwardDis);

  if (forwardDis < 75 && forwardDis > 0) { //Object is in front of robot
    stopp();   //Stops robot movement
    delay(250);
    back();   //backs away from object slightly
    delay(500);
    stopp();
    delay(1000);

    myservo.write(0);   //Moves US Sensor to the right
    delay(1000);
    rightDis = hc.dist();   //Measures Distance
    Serial.println(leftDis);    //Optional print to Serial Monitor
    delay(1000);

    myservo.write(180);   //Moves US Sensor to the left
    delay(1000);
    leftDis = hc.dist();    //Measures Distance
    Serial.println(rightDis);   //Optional print to Serial Monitor
    delay(1000);

    if(rightDis > leftDis){   //Path is clearest to the right
        right();    //turns robot to the right 90ish degrees
        delay(15);
    }
    else if(rightDis < leftDis) {    //Path clearest to the left
        left();   //turns robot to the left 90ish degrees
        delay(15);
    }

    myservo.write(80);    //Moves US Sensor back to the forward position
    delay(250);
  }
```



```
  else{   //While there is not objects in the way
    go();   //Moves robot forward
  }
}
```

TinkerCAD Design



1. CLICK START SIMULATION BUTTO

2. CLICK ULTRASONIC SENSOR

3. MOVE VIRTUAL OBSTACLE

4. WATCH MOTOR REACTION

Name 2

```
int distance = 0;
int i = 0;

long readUltrasonicDistance(int triggerPin, int echoPin)
{
  pinMode(triggerPin, OUTPUT);  // Clear the trigger
  digitalWrite(triggerPin, LOW);
  delayMicroseconds(2);
  // Sets the trigger pin to HIGH state for 10 microseconds
  digitalWrite(triggerPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(triggerPin, LOW);
  pinMode(echoPin, INPUT);
  // Reads the echo pin, and returns the sound wave travel time i
  return pulseIn(echoPin, HIGH);
}

void setup()
{
  pinMode(5, OUTPUT);
  pinMode(4, OUTPUT);
  pinMode(3, OUTPUT);
  pinMode(2, OUTPUT);
  pinMode(6, OUTPUT);
  pinMode(10, OUTPUT);
}

void loop()
{
  distance = 0.01723 * readUltrasonicDistance(11, 11);
  distance = (distance / 2.54);
  if (distance > 15) {
    // MoveForward
    digitalWrite(5, LOW);
    digitalWrite(4, HIGH);
    digitalWrite(3, LOW);
    digitalWrite(2, HIGH);
    analogWrite(6, 255);
    analogWrite(10, 255);
  } else {
    // TurnRight
    digitalWrite(5, HIGH);
    digitalWrite(4, LOW);
    digitalWrite(3, LOW);
    digitalWrite(2, HIGH);
```

Demo Robot