**Workbench Project**

Arduino based line following and obstacle avoiding robot

**MEMBERS:**

Muhammad Saad (BSCE22021)

Maham Batool (BECE22028)

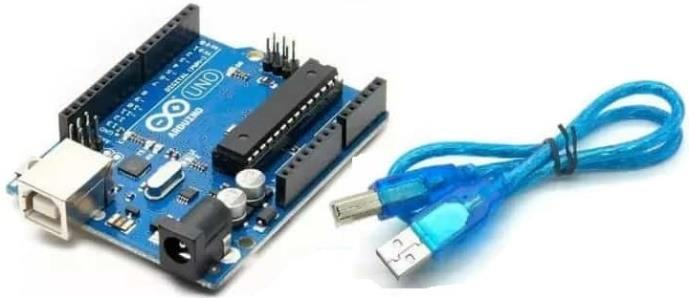
Muhammad Miqdad Ahmad (BSCE22001)

**Introduction**

We built a line following and obstacle avoiding robot. We used two colour sensors and an infrared sensor attached to the Arduino. The main purpose of the bot is to follow a line, that has been laid down. On the press of a button, it transforms from line following to obstacle avoiding. Then it moves in a straight line until it encounters some obstacle that is detected by the infrared sensor. Then it changes its direction.

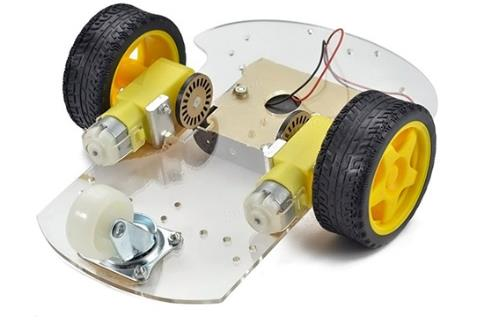
**Components**

**Arduino UNO:**



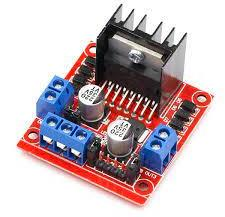
Arduino is the control center of the robot. All the functionality of the robot is controlled through here. We provide the power to the Arduino (12 volt). All the inputs from the sensors are sent to the Arduino in response to which it sends signals to the motor in accordance with the code fed to it.

**Robot Car Chassis:**



It is the main body of the robot. All the other components are attached on it.

**L298 Motor Driver:**



It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors.

**Pin to Hole Jumper wire Set:**



Simple set of wires for conductance.

**18650 Lithium Ion cells:**



Rechargeable cells used to drive the robot.

**3 x 18650 Cell holder:**



A cell holder that can be used to hold the cells.

**SPST On-Off Switch:**



On and off switches used to open or close a circuit.

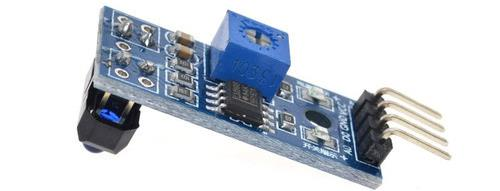
**Ultrasonic Sensor module:**

A picture containing electronics

Description automatically generated

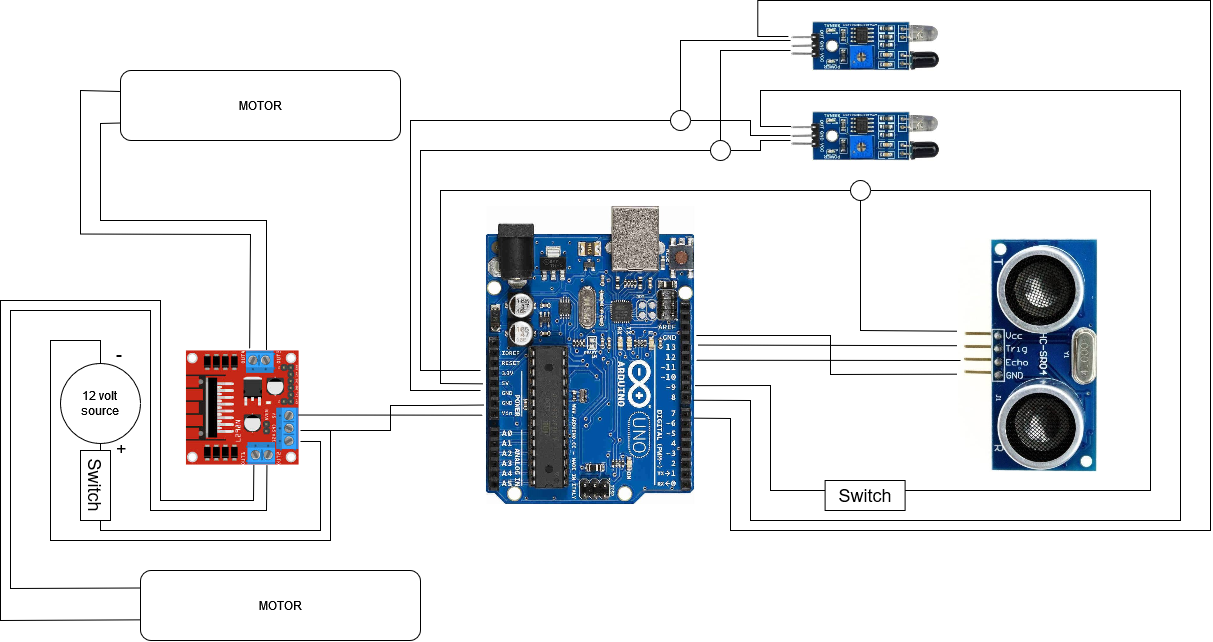
An ultrasonic sensor. It sends a pulse that when returns as echo allows the sensor to calculate the distance between itself and the obstacle in front.

**Line Sensor Module:**



A sensor that reads the reflected light and gives some value.

**Diagram**



**CODE**

int left\_forward = 3;                 // pin that moves the left motor forward

int left\_backward = 5;                // pin that moves the right motor forward

int right\_forward = 6;                // pin that moves the left motor backward

int right\_backward = 9;               // pin that moves the right motor backward

int sensor\_2 = 8;                     // this is a pin for the right sensor

int sensor\_1 = 7;                     // this is a pin for the left sensor

int send = 13;                        // this pin is the one attached to the trigpin of the ultrasound sensor

int recieve = 12;                     // this pin is attached to the echo pin of the ultra sound sensor

long distance;

int button = 11;                      // this is the pin of the switch

int forward\_speed = 63;

int backward\_speed = 63;

void forward(int speed)                        // function to run both the motors forward

{

  analogWrite(right\_forward, speed);

  analogWrite(left\_forward, speed);

  analogWrite(left\_backward, 0);

  analogWrite(right\_backward, 0);

}

void backward(int speed)                       // to run both the motors backward

{

  analogWrite(right\_forward, 0);

  analogWrite(left\_forward, 0);

  analogWrite(left\_backward, speed);

  analogWrite(right\_backward, speed);

}

void right\_for()                      // to run the right forward at low speed

{                                     // and the left at full speed

  analogWrite(right\_forward, 100);

  analogWrite(left\_forward, 255);

  analogWrite(left\_backward, 0);

  analogWrite(right\_backward, 0);

}

void right(int speed)                          // to turn the bot right

{

  analogWrite(right\_forward, 0);

  analogWrite(left\_forward, speed);

  analogWrite(left\_backward, 0);

  analogWrite(right\_backward, 0);

}

void left\_for()                       // to run the left forward at low speed

{                                     // and the right at full speed

  analogWrite(right\_forward, 255);

  analogWrite(left\_forward, 100);

  analogWrite(left\_backward, 0);

  analogWrite(right\_backward, 0);

}

void left (int speed)                          // to turn the bot left

{

  analogWrite(right\_forward, speed);

  analogWrite(left\_forward, 0);

  analogWrite(left\_backward, 0);

  analogWrite(right\_backward, 0);

}

void halt()                            // to stop the bot

{

  analogWrite(right\_forward, 0);

  analogWrite(left\_forward, 0);

  analogWrite(left\_backward, 0);

  analogWrite(right\_backward, 0);

}

long measure\_distance()               // to measure the distance from the ultrasound sensor

{

  digitalWrite(send, LOW);

  delay(2);

  digitalWrite(send, HIGH);

  delay(5);

  digitalWrite(send, LOW);

  long duration = pulseIn (recieve, HIGH);

  return duration / 29 / 2;

}

void line\_follow()                     // function for the line following bot

{

  if (digitalRead(sensor\_1) && digitalRead(sensor\_2))

  {

    forward(forward\_speed);

  }

  else if (!digitalRead(sensor\_1) && digitalRead(sensor\_2))

  {

    left(63);

  }

  else if (digitalRead(sensor\_1) && !digitalRead(sensor\_2))

  {

    right(63);

  }

  else if (!digitalRead(sensor\_1) && !digitalRead(sensor\_2))

  {

    halt();

  }

}

void ostracle\_avoiding()                  // function for the obstacle avoiding bot

{

  Serial.println(measure\_distance());

  if (measure\_distance() <= 40)

  {

    halt();

    delay(2000);

    backward(backward\_speed);

    delay(1000);

    halt();

    delay(500);

    right(127);

    delay(250);

  }

  else

  {

    forward(127);

  }

}

void setup()

{

  pinMode(left\_forward, OUTPUT);

  pinMode(left\_backward, OUTPUT);

  pinMode(right\_forward, OUTPUT);

  pinMode(right\_backward, OUTPUT);

  pinMode(sensor\_1, INPUT);

  pinMode(sensor\_2, INPUT);

  pinMode(recieve, INPUT);

  pinMode(send, OUTPUT);

  pinMode(button, INPUT);

  Serial.begin(9600);

}

void loop() {

  if (digitalRead(button))

  {

    line\_follow();

  }

  else

  {

    ostracle\_avoiding();

  }

}

**WORKING**

It is a line following and obstacle avoiding robot that can change its modes when the switch is pressed.

In the line following mode, the infrared sensors pick up colour from the ground and give either 1 or 0 value. If its black then they give 1, if it’s any other colour then they give zero. So, we program the Arduino to run only when it detects black, i.e., it gives a value 1. Other wise if both sensors give white then it stops. If the left sensor gives a zero, then the bot turns right and vice versa.

In the obstacle avoiding mode, the bot goes forward until the ultrasonic sensor detects an obstacle that is less than 40 cm away. In that case the bot comes to a stop and then backs up. Then it turns to the left and starts going forward again until it encounters some obstacle.

**CONCLUSION**

We learned that the infrared sensor detects the intensity of the reflected light. If there is no light reflected, (i.e., the surface below is dull black) then it will give us 1. If some light is reflected, then the value we get is 0.

From the ultrasonic sensor, a pulse is sent out and then when it is reflected from the obstacle in front, the sensor calculates the distance by calculating the time it took for the pulse to return.

**VIDEO AND PICTURE LINK**

https://drive.google.com/drive/folders/1t3iE44P6pIUydIpa8W9JN8tlE96fiAHP