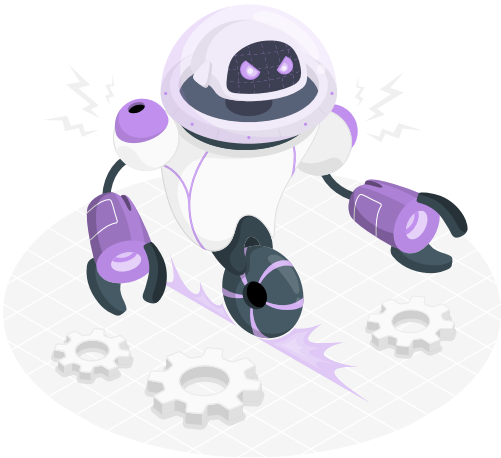


ROBOTICS PROJECT

VOYAGER

Project

-VOYAGER
-Line following bot.

Team

-Group of 5 enthusiast, Team A5.

Institute

IOE, Pashchimanchal Campus.

VOYAGER is the name of our Arduino-based robot designed specifically for line following tasks. The idea behind creating VOYAGER originated from the requirement for a versatile bot that can be employed in industrial settings to transport various objects. It operates using sensors that detect and identify obstacles, allowing the robot to follow specific paths and ensure smooth movement. However, the potential applications of VOYAGER extend beyond just industrial use. It has the capability to transform wild human imaginations into tangible reality.

Concepts

This project VOYAGER is the first step towards robotics. This can be considered the gateway for robotics. It started with imagination, brainstorming, concepts, planning and then executing. From the transformation of theory into practices this project succeeded. Arduino, also known as a microcontroller, is the processor of our bot. It controls our bot to function as user given instructions. This project included Programming , Mechanical design , Electronics and Electrical components, so here we just merged logical , theoretical and practical knowledge.

Who can do this PROJECT?

- A person willing to learn robotics.
- A person with basic knowledge of Arduino, electrical ,electronics devices and programming.

> **Note**
>
> This project demands you to learn about the basics of Arduino and programming. For the resources, you can go to the official website of Arduino
>
> https://docs.arduino.cc/learn

Getting started with project

Every good project always starts with proper planning and brainstorming

Planning

- Firstly, all of the items we need for the project must be available.
- Design the body of the bot as per need.
- Know about the sensors and circuit connection.
- Programming for controlling the bot.
- Circuit connections
- Testing the bot.
- Risks and Error handling

Execution

Required material

-Arduino Nano
-12 volt DC motors.<2 pcs>
-Motor driver L298N.
-Infrared sensor < 3pcs>
-HC-SR04 Ultrasonic sensor< 1 pcs>
-A Castro Wheel<free wheel>
-12 volt battery
-Jumper wires
-Wooden board

Chassis design

We have utilized a wooden board for the chassis design, but alternatively, a lightweight and sturdy material such as carbon fiber can be employed. The chassis serves as the foundational component of the robot, linking all the motors and sensors together. The quality of the chassis directly impacts the performance of the robot. An exceptional chassis is not solely determined by its design but also by its mechanical attributes and dimensions. While our guideline suggests a maximum dimension of 20 cm for the bot, this is not obligatory. Opting for a smaller chassis enables more precise movements compared to a larger one.

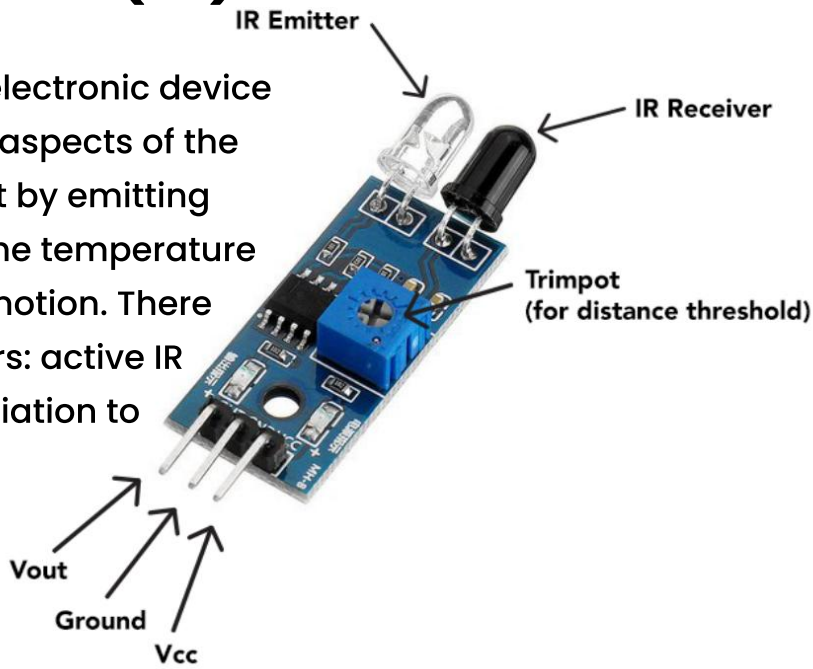
Don’t ignore

This project demands you to learn about the basics of Arduino and programming. For the resources, you can go to the official website of Arduino

https://docs.arduino.cc/learn

Learning about sensor

Infrared Sensor (IR)



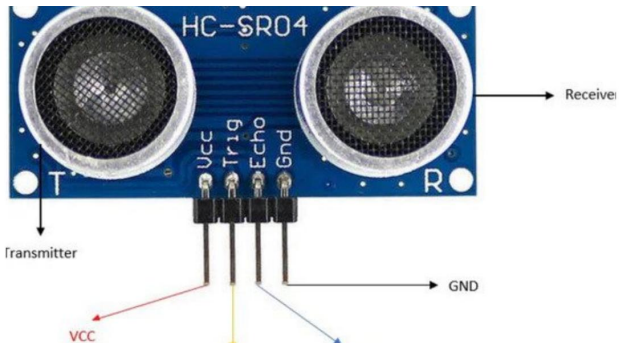
An infrared sensor is an electronic device utilized to detect various aspects of the surrounding environment by emitting signals. It can measure the temperature of an object and detect motion. There are two types of IR sensors: active IR sensors emit infrared radiation to measure, while passive IR sensors only measure existing infrared radiation.

Typically, objects emit thermal radiation within the infrared spectrum. In the case of an active IR sensor, when infrared rays emitted from a white surface are reflected and captured by a black surface, the sensor produces a high output signal (Vout). Conversely, when no signal is received, the output signal (Vout) becomes low. IR sensors are commonly used for line following purposes.

Ultrasonic Sensor

The HC-SR04 is an ultrasonic sensor that utilizes sonar technology to determine the distance between the sensor and an object. It offers a remarkable range for contactless detection, providing highly accurate and stable readings. The sensor consists of two modules: an ultrasonic transmitter and a receiver.

This versatile sensor finds applications in various fields such as directional and speed measurement, burglar alarms, medical devices, sonar systems, humidifiers, wireless charging, non-destructive testing, and ultrasonography.



To function effectively, an ultrasonic sensor requires both a transmitter and a receiver. In the standard setup, these components are positioned closely side by side. Placing the receiver near the transmitter allows sound to travel in a more direct path from the transmitter to the object being detected and back to the receiver. Ultrasonic sensors are commonly used for wall detection and collision avoidance purposes.

CODING

Arduino coding is accomplished using the Arduino Integrated Development Environment (IDE), which employs a syntax similar to that of C/C++. Those familiar with the C/C++ language will find it easy to write code for Arduino. It is essential to grasp the fundamental pin definitions and program structure. If you wish to learn Arduino programming from the beginning or are a beginner, the official Arduino website is a valuable resource where you can find comprehensive guidance on Arduino code.

CODE

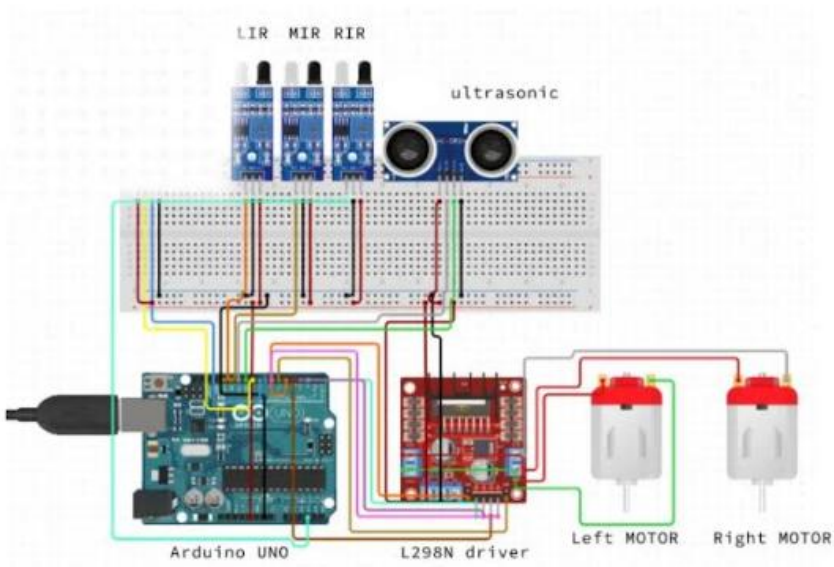
```
#define lm 5//l motor
#define rm 6//r motor
int obstacleThreshold = 10;
int trigPin = 7;
int echoPin = 8;
void setup() {
  pinMode(A0,INPUT);//left IR sensor
  pinMode(A1,INPUT);//middle IR sensor
  pinMode(A2,INPUT);//right IR sensor
  pinMode(5,OUTPUT);//left motor
  pinMode(6,OUTPUT);//right motor
  pinMode(9,OUTPUT);//left motor
  pinMode(10,OUTPUT);//right motor
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  analogWrite(9,220);
  analogWrite(10,220);
   Serial.begin(9600);
}
void loop() {
  int LIR,MIR,RIR;
   LIR=digitalRead(A0);//left IR sensor
   MIR=digitalRead(A1); //middle IR sensor
   RIR=digitalRead(A2);//right IR sensor

           digitalWrite(trigPin, LOW);
           delayMicroseconds(2);
           digitalWrite(trigPin, HIGH);
           delayMicroseconds(10);
           digitalWrite(trigPin, LOW);
           int duration = pulseIn(echoPin, HIGH);
           int distance = (duration *0.034) / 2;
```

```
    if (distance < obstacleThreshold) {
            // Stop the bot
            digitalWrite(lm, LOW);
            digitalWrite(rm, LOW);
           Serial.println("stop");
          }
   else{
  if((LIR==1 && MIR==0 && RIR==1)){
   Serial.println(" forward ");
   digitalWrite(lm,1);
   digitalWrite(rm,1);
   }

   else if((LIR==0 && MIR==1 && RIR==1) || (LIR==0 && MIR==0 && RIR==1) ){
   Serial.println("left");
   digitalWrite(lm,0);
   digitalWrite(rm,1);
   }
   else if((LIR==1 && MIR==1 && RIR==0) ||(LIR==1 && MIR==0 && RIR==0)){
   Serial.println("right");
   digitalWrite(lm,1);
   digitalWrite(rm,0);
   }
}}
```

Combined Circuit diagram



Challenges faced and precaution

Sometimes our bot may not produce the expected results, which can be attributed to various factors outlined below:

- The circuit must align precisely with the code.
- Sensors need to be properly calibrated prior to usage.
- Frequent connecting and disconnecting of the circuit can loosen the ports.
- Each sensor requires an accurate power supply.
- Occasionally, adjustments to the code may be necessary based on the specific path.
- It is important to ensure that the width does not exceed the breadth.
- Circuit connections should be clean and easily comprehensible.
- Correct polarity is crucial when connecting the battery

Final Look of our bot

Team members

Prashant Bhandari (Leader)

Electronics Engineering 1st year. Github: **prashantbhandary**

Sabina Poudel

Electronics Engineering 1st year.

Rajan Lamichhane

Mechanical Engineering 1st year.

Kshitiz Dawadi

Mechanical Engineering 1st year.

Samit Poudel

Computer Engineering 1st year.

Siddartha Gupta

MENTOR

Electronic Engineering 2nd year.

Reference

-Planning illustration <source undrwa.co>

-circuit diagram < tinkercard.com