Rajiv Gandhi College of Engineering, Research and Technology, Chandrapur

Department of Electronics and Communication Engineering

A

Synopsis of

Mini Project on

"Obstacle Detecting Robot"

Name of Students
Prachi Arun Tamgadge

Project Guide

Prof. (Meena Shende Mam)

Dr. Babasaheb Ambedkar Technological University, Lonere, Raigad

Academic Year (2023-2024)

Vision and Mission of Institute

Vision

To be on forefront to impart quality education to address societal and industrial needs and imbibe career skills through perseverance and practice.

Mission

- To adapt innovative student centric learning methods based on understanding and practice.
- To enhance professional and entrepreneurial skills.
- To motivate students to meet dynamic needs of the society with novelty and creativity.
- To promote research and continuing education to keep country ahead.
- To promote the mindset to acquire local solutions to local problems(LS2LP).

Vision and Mission of Department

Vision

Emerge as a Centre of excellence for Electronics and Communication Engineering, imparting value based education and conducting quality research in cutting edge technologies, thus contributing in socio-economic growth of the country.

Mission

Mission no.	Mission Statement
M1	Foster effective teaching-learning process to ensure proper diffusion of knowledge to the students
M2	Create an environment to encourage and motivate students to apply their mind to the engineering problems
M3	Provide platform to students to hone their technical and soft skills necessary to succeed in their professional life

Abstract:

This report presents the design and development of an autonomous obstacle avoiding robot, capable of navigating through complex environments with ease. The robot utilizes a combination of sensors and navigation algorithms to detect and avoid obstacles, ensuring efficient and safe navigation. The robot's components include an Arduino Uno microcontroller, HC-SR04 ultrasonic sensor, L298 motor driver, lithium iron battery, 12V 100RPM gear motor, robotics wheels, and a servo motor. The robot's programming is based on a simple yet effective algorithm that enables it to adapt to changing environments and obstacles.

The robot's key features include:

- Autonomous navigation
- Obstacle detection and avoidance
- Efficient navigation algorithms
- Adaptability to changing environments
- Robust and reliable hardware components

This robot has numerous applications in various fields, including:

- Warehouse management
- Surveillance
- Search and rescue operations
- Healthcare

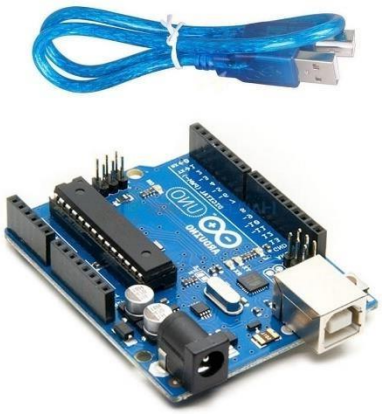
The report concludes with a demonstration of the robot's capabilities and potential for further development and improvement.

•Components Required

1. *Arduino Uno*
2. *HC-SR04 Ultrasonic Sensor & Sensor Socket*
3. *LM298N Motor Driver Module*
4. *Battery Holder for Lithium-ion*
5. *Lithium-ion Battery 1200mAh*
6. *Robotic wheel 112 mm*
7. *DC Geared Motor 12volt*
8. *Jumper Wire (Male Female)*
9. *Robot / Car Component keeping stand*

“Basics About The Electronics Component Which Use”

• Arduino Uno



→ Arduino is an open source electronics creation platform , which is based on free, flexible and easy to use hardware and software for creators and developers. This platform allows you to create different types of single-board microcomputers to which the community of creators can give different types of use.

→ This is a microcontroller board used for creating electronic projects, including the obstacle-avoiding robot. It has 14 digital input/output pins, 6 analog input pins, a USB connection, power jack, reset button, and more.

→ In order to understand this concept, first you need to know about free hardware and free software concepts. Free hardware are devices whose specifications and diagrams are publicly accessible, so anyone can replicate them. This means that Arduino offers the base so that any other person or company can create their own boards **,** being able to be different from each other but equally functional when starting from the same base.

→ Free software is a computer program whose code is accessible by anyone so that whoever wants to use can use and modify it. Arduino offers the Arduino IDE (Integrated Development Environment) platform, which is a programming environment with which anyone can create applications for Arduino boards, so that they can be given all kinds of utilities.

→ There hard work resulted an Arduino, a board with all the necessary elements to connect peripherals to the inputs and outputs of a micro controller**,** and which can

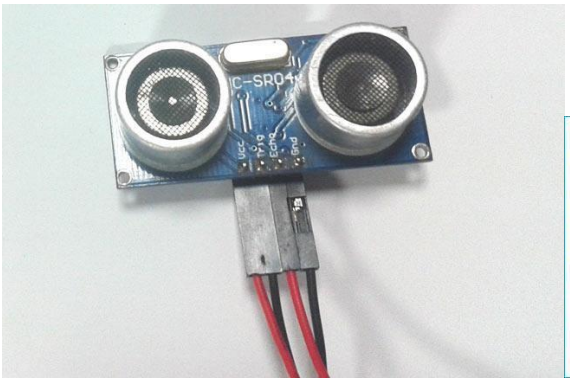
be programmed in Windows as well as macOS and GNU / Linux. A project that promotes the 'learning by doing' philosophy, which means that the best way to learn is by practically twerking the project around.

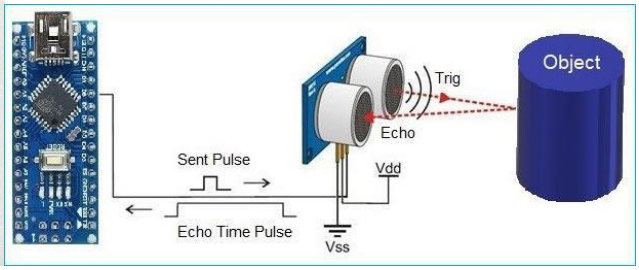
•Ultrasonic Sensor

→Before going to build the robot, it is important to understand how the ultrasonic sensor works because this sensor will have important role in detecting obstacle.

→The basic principle behind the working of ultrasonic sensor is to note down the time taken by sensor to transmit ultrasonic beams and receiving the ultrasonic beams after hitting the surface.

→Then further the distance is calculated using the formula. In this project, the widely available HC-SR04 Ultrasonic Sensor is used. To use this sensor, similar approach will be followed explained above.





→So, the Trig pin of HC-SR04 is made high for at least 10 us. A sonic beam is transmitted with 8 pulses of 40KHz each.

→The signal then hits the surface and return back and captured by the receiver Echo pin of HC-SR04. The Echo pin had already made high at the time sending high.

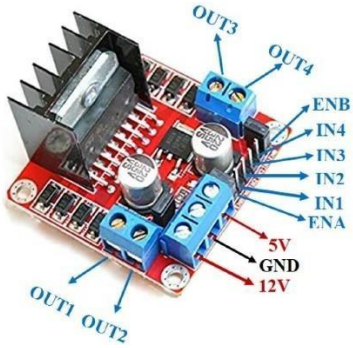
→The time taken by beam to return back is saved in variable and converted to distance using appropriate calculations like below

Distance= (Time x Speed of Sound in Air (343 m/s))/2

→We used ultrasonic sensor in many projects, to learn more about Ultrasonic sensor, check other projects related to Ultrasonic sensor.

→The components for this obstacle avoiding robot can be found easily. In order to make chassis, any toy chassis can be used or can be custom made.

• L298 2A Dual Motor Driver Module



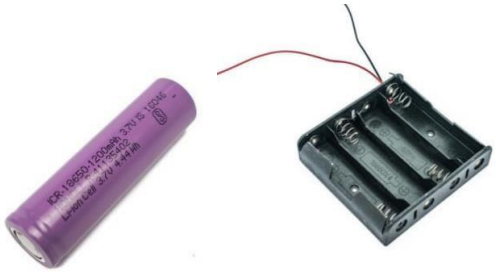
→L298N module is a high voltage, high current dual full-bridge motor driver module for controlling DC motor and stepper motor. It can control both the speed and rotation direction of two DC motors. This module consists of an L298 dual-channel H-Bridge motor driver IC. This module uses two techniques for the control speed and rotation direction of the DC motors. These are PWM – For controlling the speed and H-Bridge – For controlling rotation direction. These modules can control two DC motor or one stepper motor at the same time.

→Power Supply Pins

1VCCVCC pin is used to supply power to the motor. Its input voltage is between 5 to 35V.2GNDGND is a ground pin. It needs to be connected to the power supply ground(negative).3+5V+5V pin supplies power for the switching logic circuitry inside the L298N IC. If the 5V-EN jumper is in place, this pin acts as output and can be used to power up a microcontroller or other circuitry (sensor).
→Control Pins
1IN1These pins are input pins of Motor A. These are used to control the rotating direction of Motor A. When one of them is HIGH and the other is LOW, Motor A will start rotating in a particular direction. If both the inputs are either HIGH or LOW the Motor A will stop.2IN23IN3These pins are input pins of Motor B. These are used to control the rotating direction of Motor A. When one of them is HIGH and the other is LOW, Motor A will start rotating in a particular direction.
→Speed Control Pins
1ENA
ENA pin is used to control the speed of Motor A. If a jumper is present on this pin, so the pin connected to +5 V and the motor will be enabled, then the Motor A rotates maximum speed.

if we remove the jumper, we need to connect this pin to a PWM input of the microcontroller. In that way, we can control the speed of Motor A. If we connect this pin to Ground the Motor A will be disabled.
2ENB
ENB pin is used to control the speed of Motor B. If a jumper is present on this pin, so the pin connected to +5 V and the motor will be enabled, then the Motor B rotates maximum speed.
if we remove the jumper.
→Output Pins
1OUT1 & OUT2This terminal block will provide the output for Motor A.2OUT3 & OUT4This terminal block will provide the output for Motor B.

• Lithium-ion Battery 1200mAh & Battery Holder



→ A lithium-ion or Li-ion battery is a type of rechargeable battery that uses the reversible intercalation of Li^+ ions into electronically conducting solids to store energy. In comparison with other commercial rechargeable batteries, Li-ion batteries are characterized by higher specific energy, higher energy density, higher energy efficiency, a longer cycle life, and a longer calendar life. Also noteworthy is a dramatic improvement in lithium-ion battery properties after their market introduction in 1991: within the next 30 years, their volumetric energy density increased threefold while their cost dropped tenfold.

•Jumper Wire (Male Female)

→The difference between each is in the end point of the wire. Male ends have a pin protruding and can plug into things, while female ends do not and are used to plug things into. Male-to-male jumper wires are the most common and what you likely will use most often. When connecting two ports on a breadboard, a male-to-male wire is what you'll need.

•DC Geared Motor 12volt

→A DC motor is an electrical motor that uses direct current (DC) to produce mechanical force.

→The most common types rely on magnetic forces produced by currents in the coils. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current in part of the motor.

•Robotic wheel 112 mm

→One of the strongest robot wheels that are commonly used in industrial machines and different robots is Mecanum wheels which Bengt Erland ILon created and patented in the 1970s.

→ A Mecanum wheel is a wheel with a series of rubber rollers placed on its circumference. These rubber rollers are located at 45 degrees.

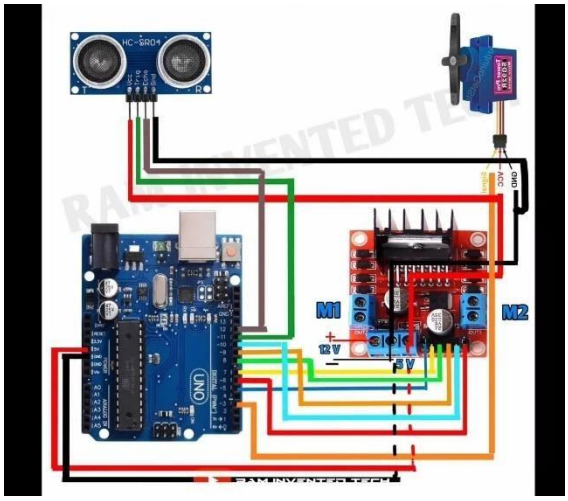
→ The rollers are diagonally mounted on the main part of the wheel at 45 degrees to the main axis. Unlike normal wheels which the driving force is applied in a straight line, the driving force in these wheels is applied at 45 degrees to the vehicle while the wheels move in a straight line. This position is the opposite of the rollers' position in Omni wheels, which are perpendicular (90 degrees) to the body of the wheel.

Servo Motor SG-90

Most of the hobby Servo motors operates from 4.8V to 6.5V, the higher the voltage higher the torque we can achieve, but most commonly they are operated at +5V. Almost all hobby servo motors can rotate only from 0° to 180° due to their gear arrangement so make sure you project can live with the half circle if no, you can prefer for a 0° to 360° motor or modify the motor to make a full circle. The gears in the motors are easily subjected to wear and tear, so if your application requires stronger and long running motors you can go with metal gears or just stick with normal plastic gear.

Next comes the most important parameter, which is the **torque** at which the motor operates. Again there are many choices here but the commonly available one is the 2.5kg/cm torque which comes with the Towerpro SG90 Motor. This 2.5kg/cm torque means that the motor can pull a weight of 2.5kg when it is suspended at a distance of 1cm. So if you suspend the load at 0.5cm then the motor can pull a load of 5kg similarly if you suspend the load at 2cm then can pull only 1.25. Based on the load which you use in the project you can select the motor with proper torque. The below picture will illustrate the same.

The Process of Assemble the Project



Here's a basic outline to get you started on your obstacle-avoiding robot:

Circuit Connections:

1. Connect the Ultrasonic Sensor to the Arduino UNO:

- VCC of the sensor to 5V on Arduino
- GND of the sensor to GND on Arduino
- Trig pin of the sensor to digital pin 9 on Arduino
- Echo pin of the sensor to digital pin 10 on Arduino

2. Connect the L298N Motor Driver Module to the Arduino UNO:

- IN1, IN2, IN3, IN4 of the L298N to digital pins 2, 3, 4, 5 on Arduino respectively
- OUT1, OUT2, OUT3, OUT4 of the L298N to the respective motors

3. Connect the Servo Motor:

- Connect the signal pin of the servo motor to digital pin 6 on Arduino
- Connect the VCC pin of the servo motor to 5V on Arduino - Connect the GND pin of the servo motor to GND on Arduino

4. Connect the Lithium-ion Battery to the Arduino UNO:

- Connect the positive terminal of the battery to VIN on Arduino
- Connect the negative terminal of the battery to GND on Arduino

Programming for Arduino UNO

```
#include <AFMotor.h>
#include <NewPing.h>
#include <Servo.h>

#define TRIG_PIN A0
#define ECHO_PIN A1
#define MAX_DISTANCE 200
#define MAX_SPEED 190 // sets speed
of DC  motors
#define MAX_SPEED_OFFSET 20

NewPing sonar(TRIG_PIN, ECHO_PIN,
MAX_DISTANCE);

AF_DCMotor motor1(1, MOTOR12_1KHZ);
AF_DCMotor motor2(2, MOTOR12_1KHZ);
AF_DCMotor motor3(3, MOTOR34_1KHZ);
AF_DCMotor motor4(4, MOTOR34_1KHZ);
Servo myservo;
 boolean
goesForward=false; int
distance = 100; int
speedSet = 0;
 void setup()
{
   myservo.attach(10);
myservo.write(115);
delay(2000);   distance
= readPing();
delay(100);   distance
= readPing();
delay(100);   distance
= readPing();
delay(100);   distance
= readPing();
delay(100);
}
void loop() {  int
distanceR = 0;  int
distanceL =  0;
delay(40);
if(distance<=15)
 {
  moveStop();
delay(100);
moveBackward();
delay(300);   moveStop();
delay(200);   distanceR =
lookRight();
delay(200);   distanceL =
lookLeft();   delay(200);

if(distanceR>=distanceL)
  {
turnRight();
moveStop();
  }else   {
turnLeft();
moveStop();
  }
 }else  {
moveForward();
 }  distance =
readPing();
}  int
lookRight()
{
    myservo.write(50);
delay(500);
    int distance = readPing();
delay(100);
myservo.write(115);
return distance;
}
int lookLeft()
{myservo.write(170);
delay(500);     int distance
= readPing();     delay(100);
myservo.write(115);
return distance;
delay(100);
}
```

```
int readPing() {
delay(70);   int cm =
sonar.ping_cm();
if(cm==0)
  {
    cm = 250;
  }
  return cm;
}  void moveStop() {
motor1.run(RELEASE);
motor2.run(RELEASE);
motor3.run(RELEASE);
motor4.run(RELEASE);
  }     void
moveForward() {

if(!goesForward)
  {     goesForward=true;
motor1.run(FORWARD);
motor2.run(FORWARD);
motor3.run(FORWARD);
motor4.run(FORWARD);          for
(speedSet = 0; speedSet <
MAX_SPEED; speedSet +=2) // slowly
bring the speed up to avoid
loading down the batteries too
quickly
{motor1.setSpeed(speedSet);
motor2.setSpeed(speedSet);
motor3.setSpeed(speedSet);
motor4.setSpeed(speedSet);
delay(5);
   }
   {
motor1.setSpeed(speedSet);
motor2.setSpeed(speedSet);
motor3.setSpeed(speedSet);
motor4.setSpeed(speedSet);
delay(5);
  }
}    void turnRight()
{
motor1.run(FORWARD);
motor2.run(FORWARD);
motor3.run(BACKWARD);
motor4.run(BACKWARD);
delay(500);
motor1.run(FORWARD);
motor2.run(FORWARD);
motor3.run(FORWARD);
motor4.run(FORWARD);
}    void turnLeft()
{
motor1.run(BACKWARD);
motor2.run(BACKWARD);
motor3.run(FORWARD);
motor4.run(FORWARD);
delay(500);
motor1.run(FORWARD);
motor2.run(FORWARD);
motor3.run(FORWARD);
motor4.run(FORWARD);
}
```

```
  }
}
void moveBackward() {
goesForward=false;
motor1.run(BACKWARD);
motor2.run(BACKWARD);
motor3.run(BACKWARD);
motor4.run(BACKWARD);     for
(speedSet = 0; speedSet <
MAX_SPEED; speedSet +=2) // slowly
bring the speed up to avoid
loading down the batteries too
quickly
```

Assembly:

1. Mount the motors on the chassis, ensuring the wheels can move freely.
2. Mount the Arduino UNO, motor driver module, ultrasonic sensor, and servo motor on the chassis securely.
3. Connect the motors to the motor driver outputs.
4. Connect the battery to the Arduino and motor driver.
5. Upload the code to the Arduino UNO using the Arduino IDE.
6. Power on the robot and test its functionality.

This is a basic outline, and you can expand upon it by adding features like PID control, obstacle avoidance algorithms, etc., depending on your requirements and expertise level.

•Expected Behavior And The Result of the project

Initialization:

- Upon powering on, the Arduino UNO initializes and starts the setup process.

2. **Ultrasonic Sensor Operation:**
   - The ultrasonic sensor continuously measures the distance to the nearest obstacle in front of the robot.
3. **Obstacle Detection:**
   - If the measured distance is greater than a predefined threshold (e.g., 20cm), the robot considers the path clear and proceeds to move forward.
4. **Forward Movement:**
   - The robot moves forward with all four DC geared motors running at an appropriate speed to propel the robot.
5. **Obstacle Avoidance:**
   - If the measured distance falls below the threshold, indicating the presence of an obstacle, the robot stops moving forward.
   - The servo motor rotates the ultrasonic sensor to scan the surroundings.
   - After scanning, the robot determines a direction with the most available space and rotates towards it.
6. **Rotation and Forward Movement:**
   - After rotation, the robot resumes moving forward in the newly determined direction, avoiding the obstacle.
7. **Continuous Operation:**
   - The robot continues this cycle of scanning, obstacle detection, rotation, and forward movement to navigate through its environment, avoiding obstacles along the way.

Result:

- Upon successful implementation, you should have a functioning obstacleavoiding robot that autonomously navigates its surroundings, avoiding obstacles in its path.
- The robot should demonstrate smooth movement, accurate obstacle detection, and effective avoidance maneuvers.
- You can further refine and optimize the robot's behavior by adjusting parameters such as the distance threshold for obstacle detection, motor speeds, and turning angles.

Make sure to test your robot in different environments to observe its performance under various conditions. If you encounter any issues or unexpected behavior, debugging and fine-tuning may be necessary to achieve the desired result.

In conclusion, the obstacle-avoiding robot project utilizing Arduino UNO, Ultrasonic Sensor, L298N Motor Driver Module, servo motor, and other mentioned components offers an excellent opportunity to delve into robotics and learn about sensor integration, motor control, and autonomous navigation.

•Conclusion For The Whole Project

Through this project, you have gained practical experience in:

1. Hardware Integration: Understanding how to connect various electronic components, including sensors, motors, motor drivers, and microcontrollers, to build a functional robotic system.

2. **Sensor-Based Control:** Implementing code to interact with the ultrasonic sensor to detect obstacles and make decisions based on environmental inputs.

3. **Motor Control:** Utilizing the L298N motor driver module to control the speed and direction of DC geared motors, enabling forward movement and rotation for obstacle avoidance.

4. **Autonomous Behavior:** Developing algorithms to autonomously navigate the robot, avoid obstacles, and adjust its trajectory based on real-time sensor feedback.

5. **Interfacing with Servo Motor:** Integrating a servo motor to enable the rotation of the ultrasonic sensor for scanning the environment and detecting obstacles in different directions.

By successfully completing this project, you have not only built a functional robot but also acquired valuable skills in electronics, programming, and robotics. This project can serve as a foundation for further exploration into advanced robotics concepts such as path planning, PID control, wireless communication, and more.

Overall, the obstacle-avoiding robot project provides a hands-on learning experience that fosters creativity, problem-solving, and innovation in the field of robotics and automation. It opens doors to endless possibilities for future projects and endeavors in the exciting world of robotics.