

Avoider from Obstacles Project

1. Introduction

The **Avoider from Obstacles** project is a robotic system that uses an **ultrasonic sensor** to detect obstacles in its path and automatically avoids them. The robot is built with an **Arduino microcontroller**, a **servo motor**, **four wheels**, and **motors**, powered by **two batteries**. The robot will move autonomously, adjusting its movement whenever it detects an obstacle in front of it.

2. Components Used

2.1. Arduino (Microcontroller)

- **Model:** Arduino Uno (or any compatible board)
- **Function:** The Arduino board serves as the brain of the robot. It receives inputs from the ultrasonic sensor and controls the motors and servo to navigate the robot.

2.2. Ultrasonic Sensor

- **Model:** HC-SR04 (common ultrasonic sensor)
- **Function:** Measures the distance between the robot and any object in front of it. The sensor sends out sound waves and measures how long it takes for the waves to return, thus calculating the distance.

2.3. Servo Motor

- **Model:** Standard servo motor (e.g., SG90)
- **Function:** Controls the robot's direction. It can be used to steer or rotate the robot left or right, depending on obstacle detection.

2.4. Motors & Wheels

- **Motors:** Two DC motors
- **Wheels:** Four wheels (2 powered wheels + 2 caster wheels)
- **Function:** The DC motors drive the robot forward or backward. The four wheels help the robot to move and maintain stability.

2.5. Batteries

- **Model:** 2x 6V or 7.4V batteries
 - **Function:** Provides the necessary power to the motors and Arduino board.
-

3. Circuit Diagram

- **Arduino Connections:**
 - The ultrasonic sensor's **VCC** is connected to the **5V** pin on the Arduino.
 - The **Trig** pin of the ultrasonic sensor is connected to a digital output pin (e.g., pin 9) on Arduino.
 - The **Echo** pin of the ultrasonic sensor is connected to a digital input pin (e.g., pin 10) on Arduino.
 - The **servo motor** is connected to a PWM pin (e.g., pin 11) on the Arduino.
 - The **motor driver** (such as L298N) is connected to the motors, with control pins connected to Arduino digital pins (pins 5, 6 for motor direction, and 3, 4 for motor speed).
-

4. Working Principle

4.1. Obstacle Detection

The ultrasonic sensor continuously measures the distance to objects in front of the robot. When an obstacle is detected within a specified range (e.g., less than 30 cm), the robot will take action to avoid the obstacle.

4.2. Movement and Steering

The robot moves forward until an obstacle is detected. When an obstacle is detected, the Arduino instructs the servo motor to turn the robot either left or right to avoid the obstacle. After a successful turn, the robot resumes moving forward. The motors control the forward and backward movement of the robot, while the servo controls the steering.

4.3. Power Management

The robot is powered by two batteries connected to the motor driver and Arduino. The battery life depends on the motor's usage, and the robot can be manually turned off to conserve power when not in use.

5. Code Implementation

Here is a basic example of Arduino code to control the movement of the robot based on obstacle detection:

```
#include <Servo.h>

const int trigPin = 9; const int echoPin = 10; const int Motor = 11; const
int buzzer = 13; long duration; int distance; int safetyDistance; void
setup() { pinMode(buzzer, OUTPUT); pinMode(Motor, OUTPUT);
pinMode(trigPin, OUTPUT); pinMode(echoPin, INPUT);
Serial.begin(9600); } void loop() { dos(); } void dos() { digitalWrite(trigPin,
LOW); delayMicroseconds(2); digitalWrite(trigPin, HIGH);
delayMicroseconds(10); digitalWrite(trigPin, LOW); duration =
pulseIn(echoPin, HIGH); distance = duration * 0.034 / 2; safetyDistance
= distance; if (safetyDistance <= 5) { digitalWrite(buzzer, HIGH);
digitalWrite(Motor, HIGH); } else { digitalWrite(buzzer, LOW);
digitalWrite(Motor, LOW); } Serial.print("Distance: ");
Serial.println(distance); }
```

6. Testing and Calibration

6.1. Obstacle Detection

- Test the ultrasonic sensor by placing objects at various distances in front of the robot to ensure accurate distance measurement.

6.2. Servo Motor Steering

- Adjust the servo angles in the code to ensure that the robot turns correctly when an obstacle is detected.

6.3. Motor Speed and Control

- Test the forward and backward movements of the robot to ensure the motors respond correctly. If needed, adjust motor control logic to optimize movement

- **7. Conclusion**

- This **Avoider from Obstacles** project demonstrates how an autonomous robot can be built using basic electronic components like an Arduino, ultrasonic sensor, motors, and servo motor. The robot is capable of detecting obstacles and avoiding them using simple logic and control.



