WESTERN NEW ENGLAND UNIVERSITY
SPRINGFIELD, MASSACHUSETTS

DEPARTMENT OF MECHANICAL ENGINEERING

ME 455-41 APPLIED MECHATRONIC SYSTEMS
ME 656-41 ADVANCED MECHATRONICS SYSTEMS

Instructor: Dr. Vedang Chauhan

Project 4 Wall Following and Obstacle Avoidance

Objective: Building on the previous Smart Mobile Robot work, the objective of this project is to implement a wall following robot that can avoid obstacles and continue following its path.

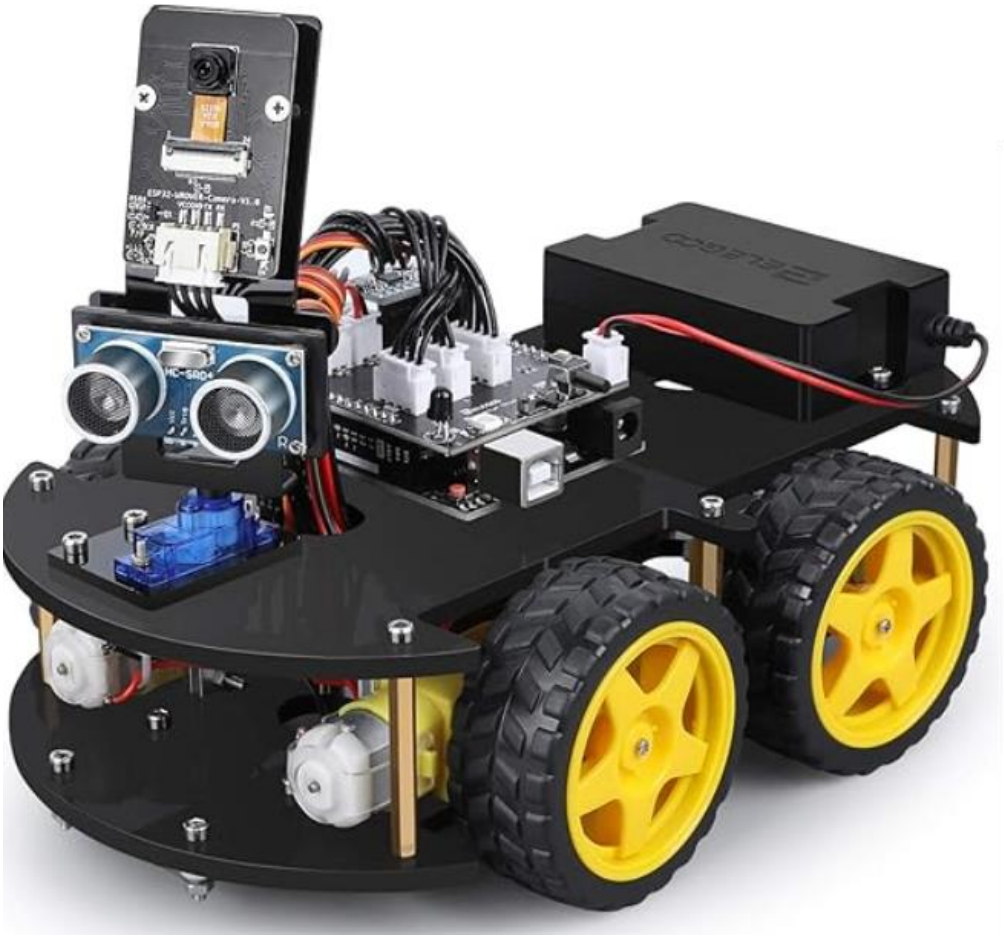

Figure 1 Smart Mobile Robot (image course: *ELEGOO UNO R3 Smart Robot Car Kit*

Amazon.com)

1. Wall Following

For this part, you will program your robot to stay within a given limit from the wall while moving forward as shown in Figure 2. Use a PID controller for wall following. Write your own PID controller logic using the following steps. If the robot is nudged while moving forward, it should return to its wall following range automatically.

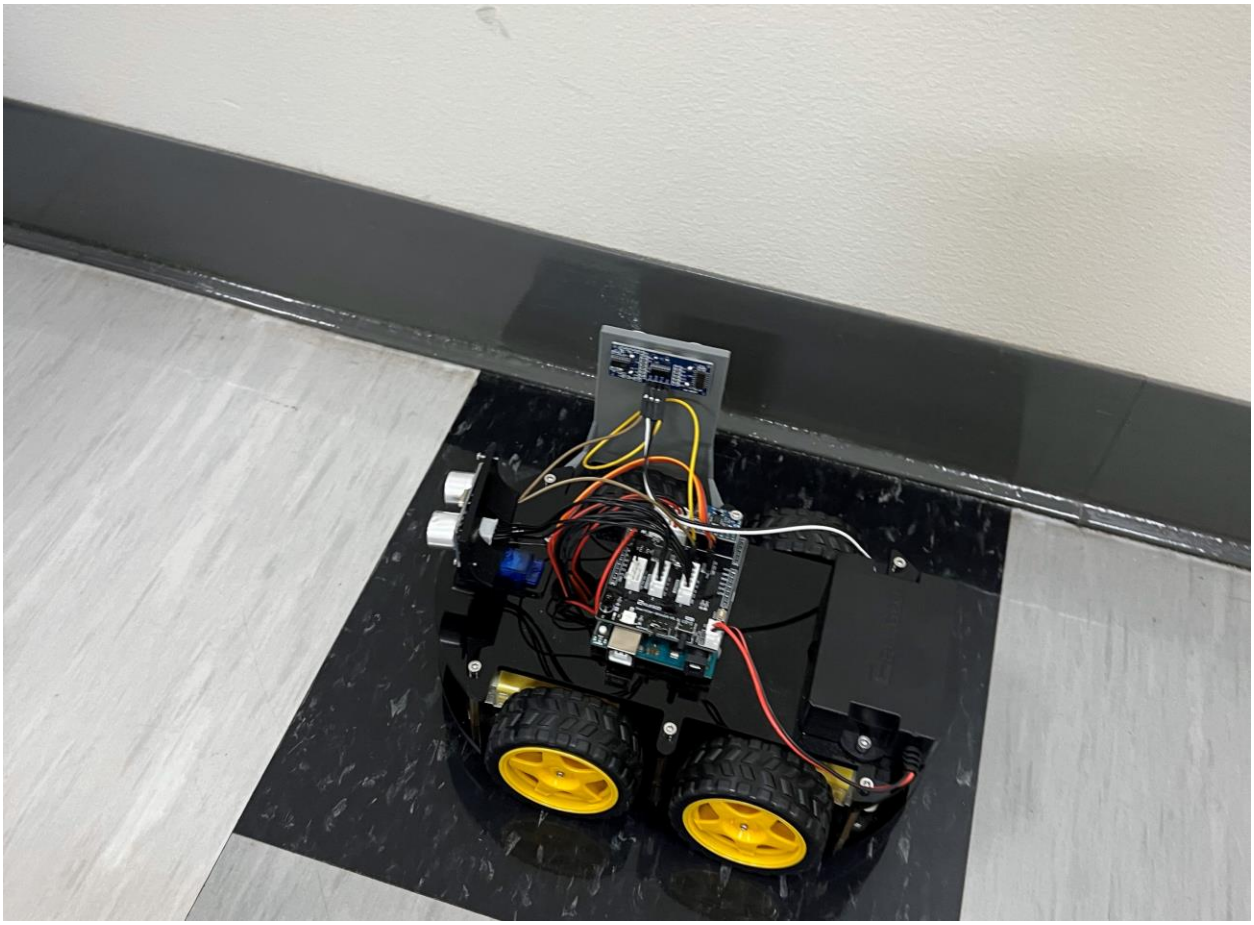

Figure 2 Wall Following Robot

Wall Following Flowchart Steps

- **Include Header file NewPing.h (**https://docs.arduino.cc/libraries/newping/#Releases**)**
- **Define and Initialize variables**
  - setpoint, gains, timeVariables, motorpins, etc.
  - Define all variables as float other than time variables. For time variables define them as unsigned long
- **Void Setup**
  - Setup motor pins as outputs

- Initialize serial communication

Void Loop

- **Check Time Interval**
  - Read currentTime
  - If currentTime - prevTime >= 100 ms, continue; else, go back to main loop
  - PID loop runs at 100 ms rate.

- **Read Distance from Ultrasonic Sensor**
  - Measure distance from wall
- **Calculate PID Components**
  - Calculate error
  - Update integral using the cumulative error and time
  - Constrain integral to (-50,50)
  - Calculate derivative using the difference between error and time
- **Update Previous Error for the next iteration**
  - Update previousError = error
  - Update prevTime = currentTime

- **Compute PID Output**
  - Calculate PID output using proportional, integral, and derivative terms. Adjust gains if required.
- **Constrain Output**
  - Constrain to PWM range (-255, 255)
- **Set Motor Directions and Speed**
  - Define the base speed as 150
  - Adjust left and right wheel speeds by adding or subtracting the motorSpeed from the base speed. The robot moves only in a forward direction.
  - Send the updated left and right wheel speeds to the corresponding motors.
- **Print Debug Information**
  - Print distance, error, output

End of Void Loop

2. Wall Following and Obstacle Detection

For this part, you will program your robot to stay within a given limit from the wall while moving. In case of an obstacle, the robot should stop and wait for the obstacle to pass. Once the obstacle moves out of the robot's path, the robot should continue its wall following. Use your servo-mounted ultrasonic sensor for obstacle detection and the side mounted ultrasonic sensor for wall following.

3. Wall Following and Obstacle Avoidance

You will build this part on the previous part. Program your robot to stay within a given limit from the wall while moving forward. In case of an obstacle, the robot safely navigates around the obstacle and continues its wall following.

Deliverables and assessment rubric:

Submit onto Kodiak.

Final PDF report (file name: Project4_YourGroupNumber.PDF) that contains • Your programs for task 1, task 2, and task 3 • Videos of all the tasks and live demonstrations.	**90**
Quality and professionalism of the report.	**10**

***