# ****Motorized Tank Project Report****

**Name:** Hari and Team  
**Class / Section:** 11  
**Date:**

## ****Project Title:****

**Motorized Tank with Obstacle Detection and Distance Display**

## ****Objective:****

To design and construct a motorized tank capable of autonomous movement, detecting obstacles using a servo-mounted ultrasonic sensor, and displaying the distance to obstacles in real time on an OLED display. The project demonstrates motor control, sensor integration, and Arduino-based electronics.

## ****Components Needed:****

* Arduino Uno -(Having)
* L298N motor driver module - 1
* Bo geared motor - 4
* Ultrasonic sensor (HC-SR04) - 1
* Servo motor - 1
* OLED / LCD display (Optional)
* 2S Li-ion battery (7.4V) or suitable powerbank
* Acrylic sheet for basement and body
* Wires, resistors, and switches

## ****Working Principle:****

1. **Motorized Movement:**
   * Four DC motors are connected to the L298N motor driver.
   * Arduino controls speed and direction to move the tank forward, backward, and turn.
2. **Obstacle Detection and Avoidance:**
   * Ultrasonic sensor measures the distance to the nearest obstacle.
   * Servo sweeps the sensor from 40° to 140° to scan multiple directions.
   * If an obstacle is detected within the threshold (15 cm), the tank stops, backs up, and turns to avoid it.
   * After the turn is completed, normal forward movement resumes.
3. **Display Integration:**
   * OLED display shows the current distance in cm.
   * Also displays the tank’s status: “Clear”, “Obstacle!”, or “Turning…”
4. **Power Supply Options:**
   * Primary: 2S Li-ion battery (7.4V)
   * Alternative: Powerbank capable of providing sufficient current for motors and Arduino.
   * Arduino and sensors powered through regulated 5V; L298N ensures proper voltage to the motors.
5. **Connections Overview:**
   * DC motors → L298N motor driver → Arduino control pins
   * Servo → Arduino PWM pin
   * Ultrasonic sensor → Arduino Trigger & Echo pins
   * OLED display → Arduino via I2C
   * Battery → L298N and Arduino (regulated)

## ****Key Features / Notes:****

* Fully motorized and autonomous movement.
* Servo-mounted ultrasonic sensor allows multi-angle obstacle detection.
* Real-time distance display on OLED.
* Modular design allows future upgrades (additional sensors, remote control, etc.).
* Optimized battery usage for motors, sensors, and display.

## ****Arduino Code:****

#include <Servo.h>

#include <Wire.h>

#include <Adafruit\_GFX.h>

#include <Adafruit\_SSD1306.h>

// Motor pins

#define IN1 5

#define IN2 4

#define IN3 8

#define IN4 12

#define ENA 3

#define ENB 11

// Front ultrasonic sensor pins

#define FRONT\_TRIG 6

#define FRONT\_ECHO 7

// Radar sensor + servo pins

#define RADAR\_TRIG 9

#define RADAR\_ECHO 10

#define SERVO\_PIN 11

Servo radarServo;

Adafruit\_SSD1306 display(128, 64, &Wire, -1);

int threshold = 15; // obstacle threshold in cm

bool turning = false;

unsigned long lastTurnTime = 0;

void setup() {

Serial.begin(9600);

// Motor pins

pinMode(IN1, OUTPUT);

pinMode(IN2, OUTPUT);

pinMode(IN3, OUTPUT);

pinMode(IN4, OUTPUT);

pinMode(ENA, OUTPUT);

pinMode(ENB, OUTPUT);

analogWrite(ENA, 180);

analogWrite(ENB, 180);

// Front sensor

pinMode(FRONT\_TRIG, OUTPUT);

pinMode(FRONT\_ECHO, INPUT);

// Radar sensor + servo

pinMode(RADAR\_TRIG, OUTPUT);

pinMode(RADAR\_ECHO, INPUT);

radarServo.attach(SERVO\_PIN);

// OLED display

display.begin(SSD1306\_SWITCHCAPVCC, 0x3C);

display.clearDisplay();

display.setTextSize(1);

display.setTextColor(SSD1306\_WHITE);

}

void loop() {

// Radar sweep

for(int pos = 15; pos <= 190; pos += 5){

radarServo.write(pos);

delay(30);

int radarDist = getDistance(RADAR\_TRIG, RADAR\_ECHO);

Serial.print(pos); Serial.print(","); Serial.print(radarDist); Serial.println(".");

}

for(int pos = 190; pos >= 15; pos -= 5){

radarServo.write(pos);

delay(30);

int radarDist = getDistance(RADAR\_TRIG, RADAR\_ECHO);

Serial.print(pos); Serial.print(","); Serial.print(radarDist); Serial.println(".");

}

// Front obstacle check

int frontDist = getDistance(FRONT\_TRIG, FRONT\_ECHO);

displayDistance(frontDist);

if(frontDist > 0 && frontDist < threshold && !turning){

stopBot();

backUp();

turnRight();

lastTurnTime = millis();

turning = true;

}

if(turning && millis() - lastTurnTime > 800){

turning = false;

}

if(!turning) moveForward();

}

// Function to measure distance

int getDistance(int trigPin, int echoPin){

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

long duration = pulseIn(echoPin, HIGH, 30000);

if(duration == 0) return -1;

return duration \* 0.034 / 2;

}

// Display front distance on OLED

void displayDistance(int d){

display.clearDisplay();

display.setCursor(0,10);

display.print("Front Dist: ");

display.print(d); display.print(" cm");

display.setCursor(0,25);

if(turning) display.print("Turning...");

else if(d<threshold && d>0) display.print("Obstacle!");

else display.print("Clear");

display.display();

}

// Motor functions

void moveForward(){

digitalWrite(IN1,HIGH); digitalWrite(IN2,LOW);

digitalWrite(IN3,HIGH); digitalWrite(IN4,LOW);

}

void stopBot(){

digitalWrite(IN1,LOW); digitalWrite(IN2,LOW);

digitalWrite(IN3,LOW); digitalWrite(IN4,LOW);

}

void backUp(){

digitalWrite(IN1,LOW); digitalWrite(IN2,HIGH);

digitalWrite(IN3,LOW); digitalWrite(IN4,HIGH);

delay(300);

stopBot();

}

void turnRight(){

digitalWrite(IN1,LOW); digitalWrite(IN2,HIGH);

digitalWrite(IN3,HIGH); digitalWrite(IN4,LOW);

}

**Project Display & Visualization**

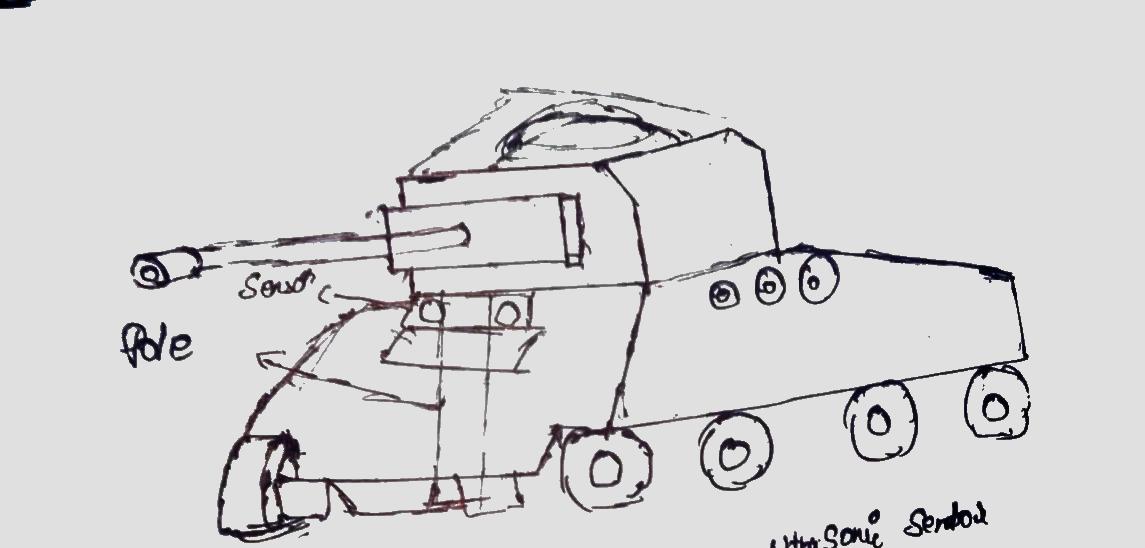
1. Front Sensor + OLED (On the Tank)

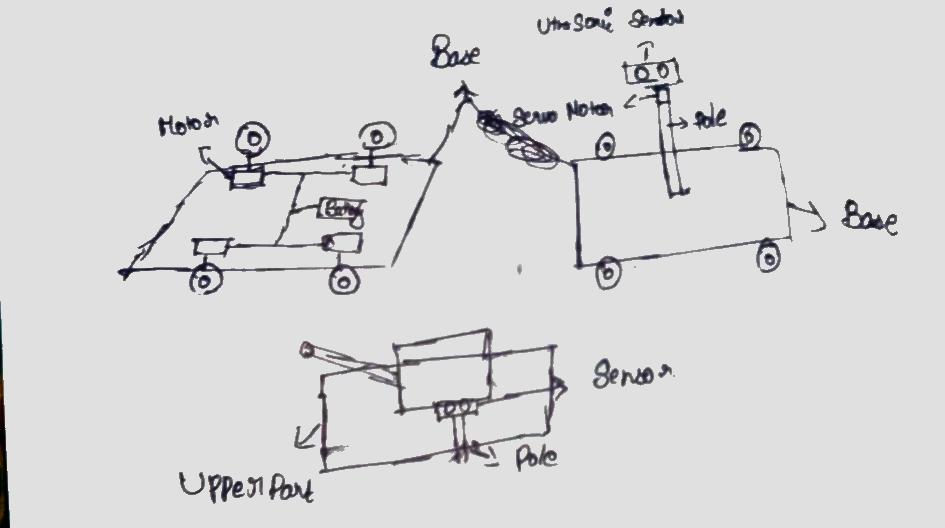
* The front ultrasonic sensor is mounted fixed on the tank.
* It constantly measures distance directly ahead.
* The OLED display on the tank shows:
* Distance in cm
* Status: “Clear”, “Obstacle!”, or “Turning…”
* This works fully independently, no PC needed.

1. Radar Sensor + Processing Software (PC Visualization)

* The radar ultrasonic sensor is mounted on a servo and sweeps from 15° → 190°.
* Arduino sends angle + distance data over the USB Serial connection.
* Processing software on a PC reads this data and draws a radar visualization:
* Sweep line shows servo angle
* Red marks show detected objects and distances
* Without a PC, this radar visualization cannot appear on the OLED — OLED only shows front distance and status.

Sample design





## ****Conclusion:****

The motorized tank successfully navigates its environment using DC motors and a servo-mounted ultrasonic sensor. It detects obstacles and displays distance data on the OLED screen in real time. This project demonstrates practical applications of Arduino programming, motor control, and sensor integration in a mobile robotics platform.