AMERICAN INTERNATIONAL UNIVERSITY-BANGLADESH

Faculty of Engineering

Performance Activity

Title: Controlling speed of DC motor with Ultrasonic sensor

Date of Perform	**9/10/2025**	**Date of Submission**	**9/10/2025**

Course Title:	**MICROPROCESSOR AND EMBEDDED SYSTEMS**

Course Code	**COE3104**	**Section:**	**B**
Semester	**SUMMER (24-25)**	**Degree Program:**	**BSc in CSE**

Course Teacher :	**MD.ALINOOR**
Group:	**04**

SlNo	Name	ID	PROGRAM	
1.	Basudeb Kundu	23-50856-1	CSE	
2.	Prohlad Chandra Das	23-50922-1	CSE	
3.	Debashis Kumar Das	23-50953-1	CSE	
4.	Indronil Dutta Nill	23-50974-1	CSE	
5.	Nafiur Rahman Nirob	23-50991-1	CSE	

Faculty use only		
FACULTY COMMENTS	**Marks Obtained**	
	Total Marks	

Table of Contents

Objectives :

1. **Measure Distance:**
   - Use the ultrasonic sensor (HC-SR04 or similar) to detect the distance of an object within a predefined range.
2. **Signal Processing:**
   - Convert the measured distance into a control signal (PWM) that can vary the motor speed smoothly.
3. **Motor Speed Control:**
   - Implement Pulse Width Modulation (PWM) through a microcontroller (e.g., Arduino) to adjust the DC motor's speed proportionally to the detected distance.
4. **Safety and Automation:**
   - Automatically stop or slow down the motor when an object is too close to prevent damage or unsafe operation.
5. **System Integration:**
   - Combine sensor, controller, and motor driver circuit to create a functional prototype that demonstrates real-time control.
6. **Testing and Validation:**
   - Verify the system by testing with different distances and loads to ensure reliability and accuracy.

Equipment List:

1. Arduino UNO / Nano
2. Ultrasonic Sensor (HC-SR04)
3. DC Motor
4. Motor Driver (L298N or MOSFET + Diode)
5. Power Supply (9–12V for motor, 5V for Arduino)
6. Resistors (220Ω, 10kΩ)
7. Breadboard & Jumper Wires
8. USB Cable

Block Diagram :

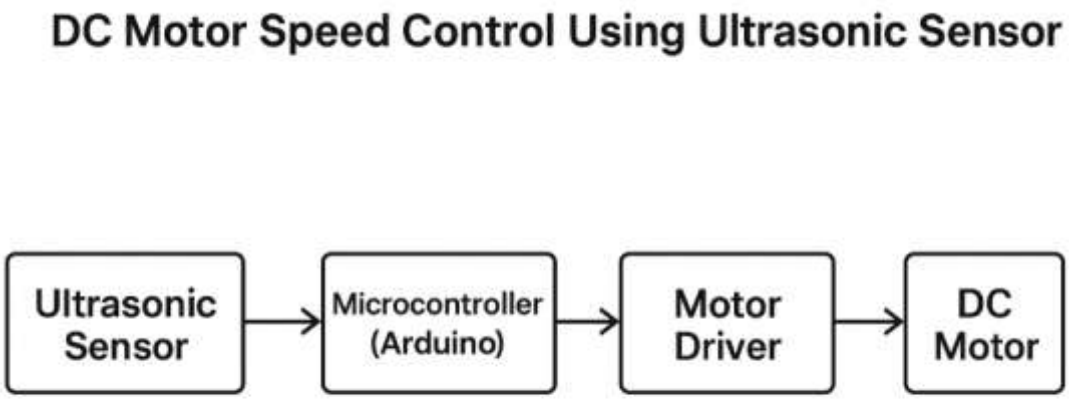
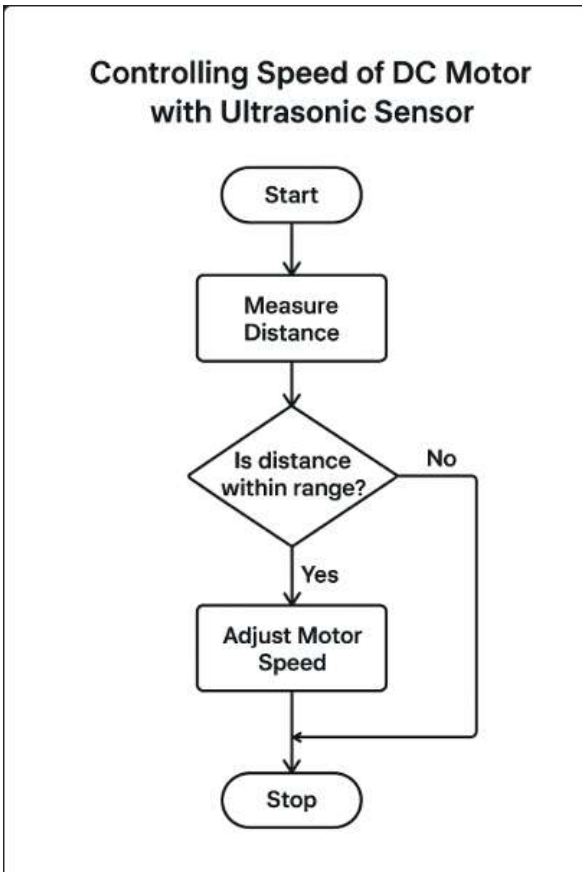


Figure: Block Diagram of Controlling speed of DC motor with Ultrasonic sensor

Flowchart :



Circuit Diagram :

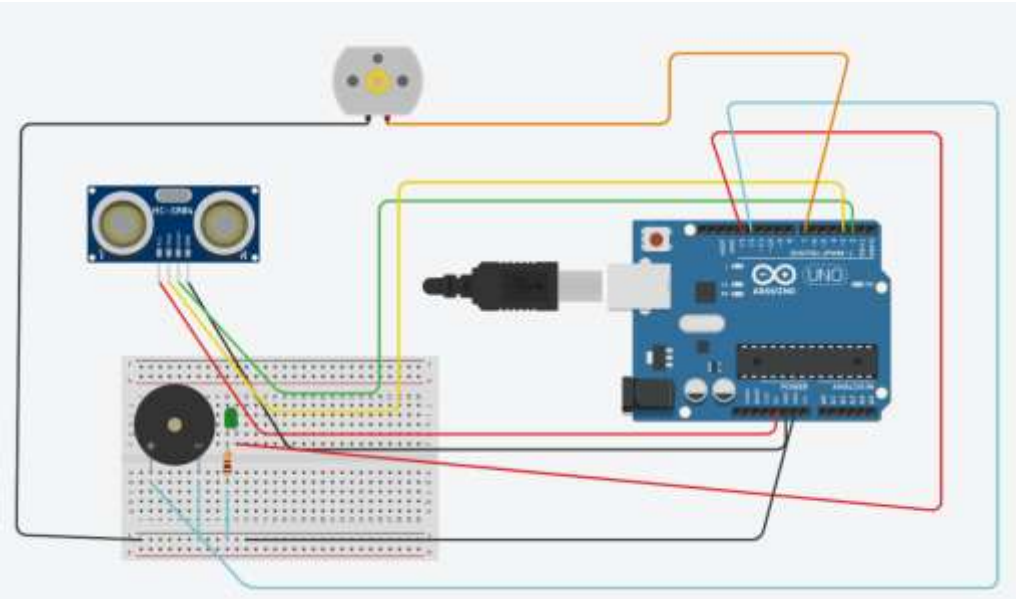


Figure:Hardware circuit diagram of the projectusing Proteus.

Code for simulation :

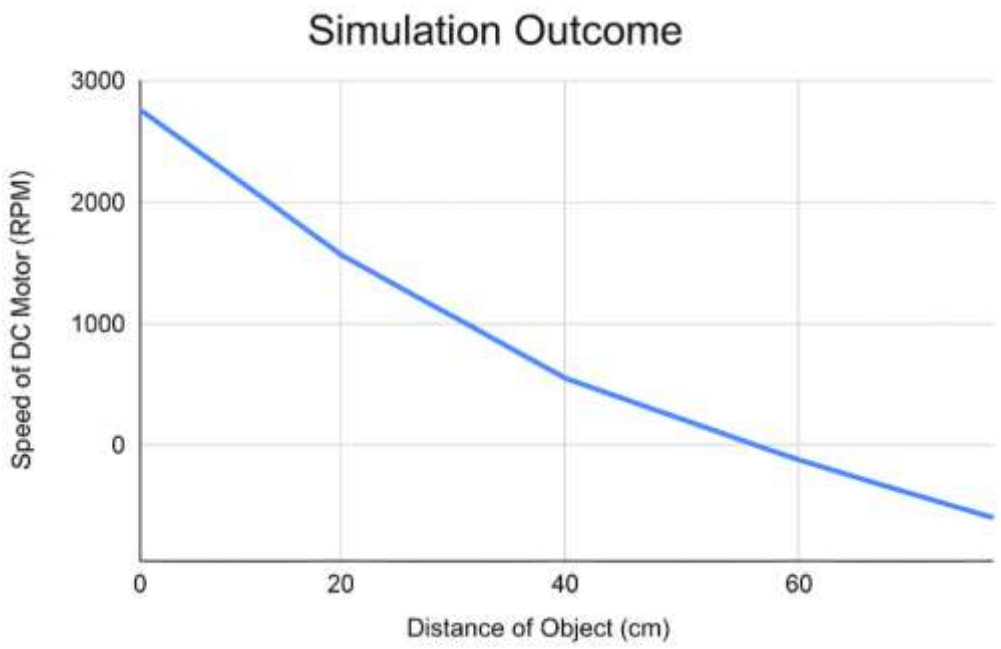
```
// Motor control with ultrasonic sensor (HC-SR04)
    // Arduino Uno example
    // PWM output to MOSFET gate (or ENA of motor driver)
    const int trigPin = 9;
    const int echoPin = 8;
    const int pwmPin  = 6;   // PWM capable pin
    const int dirPin1 = 7;   // optional: direction pin if using H-bridge (unused for
    single direction)
    const int NUM_SAMPLES = 5;
    const unsigned int minD = 2;    // cm (closest)
    const unsigned int maxD = 200;  // cm (farthest)
    const unsigned int stopDistance = 4; // cm -> treat as stop
    unsigned long readings[NUM_SAMPLES];
    int readIndex = 0;
    unsigned long total = 0;
    unsigned long averageDistance = 0;
    void setup() {
      Serial.begin(9600);
      pinMode(trigPin, OUTPUT);
      pinMode(echoPin, INPUT);
      pinMode(pwmPin, OUTPUT);
      pinMode(dirPin1, OUTPUT);
      digitalWrite(dirPin1, LOW); // set direction if needed
      // initialize readings array
      for (int i = 0; i < NUM_SAMPLES; i++) readings[i] = maxD;
    }
    unsigned int getDistanceCM() {
      // send pulse
      digitalWrite(trigPin, LOW);
      delayMicroseconds(2);
      digitalWrite(trigPin, HIGH);
      delayMicroseconds(10);
      digitalWrite(trigPin, LOW);
      unsigned long duration = pulseIn(echoPin, HIGH, 30000UL); // timeout 30ms
      if (duration == 0) return maxD; // no echo - return max distance
      // speed of sound 343 m/s => 29.1 microseconds per cm for round trip
      // distance cm = duration / 58.0  (common approx)
      unsigned int distance = (unsigned int)(duration / 58UL);
      return distance;
```

```
}
unsigned int getSmoothedDistance() {
  // moving average
  unsigned int d = getDistanceCM();
  // subtract oldest
  total = total - readings[readIndex];
  // add new
  readings[readIndex] = d;
  total = total + readings[readIndex];
  readIndex = (readIndex + 1) % NUM_SAMPLES;
  averageDistance = total / NUM_SAMPLES;
  return (unsigned int)averageDistance;
}
void loop() {
  unsigned int dist = getSmoothedDistance();
  // clamp
  if (dist < minD) dist = minD;
  if (dist > maxD) dist = maxD;
  int pwmVal;
  if (dist <= stopDistance) {
    pwmVal = 0; // safety stop
  } else {
    // map distance to PWM: close (minD) => 0, far (maxD) => 255
    pwmVal = map(dist, minD, maxD, 0, 255);

    // optionally invert mapping: uncomment next line to make close => fast
    // pwmVal = map(dist, minD, maxD, 255, 0);
  }
  // apply low-pass on PWM to make gradual change (simple)
  static int lastPwm = 0;
  int filtered = lastPwm + (pwmVal - lastPwm) / 3; // smooth step, factor = 3
  lastPwm = filtered;
  analogWrite(pwmPin, filtered);

  // debug
  Serial.print("dist(cm): ");
  Serial.print(dist);
  Serial.print("  pwm: ");
  Serial.println(filtered);
  delay(60); // adjust update rate (ms)
```

Simulation Output :



Simulation Outcome:

The simulation successfully demonstrated the control of DC motor speed using an ultrasonic sensor. As the distance of an object decreased, the motor speed reduced, and when the object moved farther away, the motor speed increased. The PWM signals generated by the microcontroller responded accurately to the sensor input, showing smooth variation in motor speed. The results confirm that the system can effectively regulate motor speed based on distance sensing.

Conclusion:

The project successfully demonstrates how an ultrasonic sensor can be used to control the speed of a DC motor through a microcontroller. By measuring the distance of an object, the system automatically adjusts the motor speed using PWM signals, ensuring smooth and efficient operation. This approach not only improves automation but also enhances safety by stopping or slowing the motor when objects come too close. The integration of sensor, controller, and driver proves effective in creating a low-cost, reliable, and scalable solution for real-world applications such as smart fans, conveyor systems, and robotic control.

Discussion:

This project demonstrates how an ultrasonic sensor can be used to control the speed of a DC motor. The sensor measures distance and sends it to the microcontroller, which adjusts the motor speed using PWM signals. When an object is closer, the motor slows down; when it is farther, the motor speeds up. This method is simple, cost-effective, and useful in automation systems like conveyor belts or smart fans. However, accuracy can be affected by environmental conditions and surface types. Overall, the system works effectively, though future improvements like PID control and better sensors could enhance performance.

References:

1. Arduino. "Arduino Uno Rev3." [Online]. Available: https://www.arduino.cc/en/Main/arduinoBoardUno. [Accessed: 04-Jul-2024].
2. https://microdigisoft.com/traffic-light-using-arduino-in-proteus/