SONAR BASED RADAR SYSTEM USING ARDUINO

Report for Mini Project/Electronic Design Workshop (EC-681)

B. Tech in Electronics and Communication Engineering

B. P. Poddar Institute of Management & Technology

Under

Maulana Abul Kalam Azad University of Technology

Under the supervision of
Dr. Madhumita sarkar & Mr. Pushpendu Kanjilal
Group No. : 01

Submitted by

Name	University Roll No	Registration. No	Section
Sayan Bhattacharjee	11500322043	221150110196	A
Ritam Chakraborty	11500322042	221150110185	A
Abhishek Kumar Paswan	11500322024	221150110135	A
Debprasad Mallik	11500322007	221150110159	A

Academic Year: 2024-25

Department of Electronics& Communication Engineering

B. P. Poddar Institute of Management & Technology

137, V. I. P. Road. Poddar Vihar
Kolkata – 700 052

Table of content

Department of Electronics & Communication Engineering,

B. P. Poddar Institute of Management& Technology

137, V. I. P. Road, Poddar Vihar. Kolkata – 700052

CERTIFICATE

This is to certify that the project work, entitled "SONAR BASED RADAR SYSTEM USING ARDUINO" submitted by Sayan Bhattacharjee , Ritam Chakraborty, Abhishek Kumar Paswan and Debprasad Mallik have been prepared according to the regulation of the degree B. Tech in Electronics & Communication Engineering of the Maulana Abul Kalam Azad University of Technology, West Bengal. The candidate(s) have partially fulfilled the requirements for the submission of the project work.

--	--
(Name of HOD)	**(Name of the Supervisor)**
Dept. of Electronics & Comm. Engg.	**Dept. of Electronics & Comm. Engg**

ABSTRACT :

The mini-project titled **“Sonar-Based Radar System Using Arduino”** presents a functional prototype that mimics radar-like object detection by utilizing ultrasonic sensing. This system is built around the **HC-SR04 ultrasonic sensor**, which detects obstacles by emitting high-frequency sound pulses and measuring the time it takes for the echo to return. Using this **time-of-flight principle**, the system calculates the distance of objects in its path with considerable accuracy.

The distance data is processed by an **Arduino Uno**, which also controls a **servo motor** to rotate the sensor across an angular range from **0° to 180°**. This rotation enables the sensor to scan the environment in a sweeping motion, similar to how traditional radar systems operate. The Arduino continuously collects angle-distance pairs and sends them via **serial communication** to a computer for further processing and visualization.

On the computer side, the system uses the **Processing IDE** to create a **graphical radar interface** that displays the real-time position of detected objects on a **2D polar grid**. This visual output simulates a classic radar screen, showing how objects appear at various angles and distances relative to the sensor's position.

This project effectively combines several core concepts in electronics and embedded systems, including **ultrasonic sensing**, **PWM-based servo control**, **serial data communication**, and **GUI development**. It serves as a hands-on demonstration of how hardware and software can work together to replicate a real-world technology in a compact, modular format.

Potential applications for this sonar-based radar system include **autonomous navigation in robotics**, **collision detection in smart vehicles**, **intrusion monitoring in security systems**,. Its modular and upgradable design also makes it suitable for enhancements such as **360° scanning using stepper motors**, **wireless data transfer via Bluetooth/Wi-Fi**, **cloud integration through IoT platforms**, and **multi-sensor fusion for mapping larger areas**.

In conclusion, the project offers a rich learning experience by bridging concepts in sensor technology, embedded programming, and visual data representation.

ACKNOWLEDGEMENTS

It is a great pleasure for me/us to express our earnest and great appreciation Dr. Madhumita sarkar & Mr. Pushpendu Kanjilal my project guide. We are very much grateful to her for his kind guidance, encouragement, valuable suggestions, innovative ideas, and supervision throughout this project work, without which the completion of the project work would have been difficult one.

We would like to express our thanks to the Head of the Department, Dr. Ivy Majumdar for her active support.

We also express our sincere thanks to all the teachers of the department for their precious help, encouragement, kind cooperation and suggestions throughout the development of the project work.

We would like to express my/our gratitude to the library staff and laboratory staff for providing us with a congenial working environment.

1.

2.

3.

4.

Date: _______________ ______________________________

(Full Signature of the Student(s)

B. Tech in Electronics & Comm. Engg.

Department of Electronics & Comm. Engg.

B P Poddar Institute of Management and Technology

List of Figures. **Page No.**

TITLE : SONAR BASED RADAR SYSTEM USING ARDUINO

OBJECTIVE: : The purpose of this project is to

1. **To develop an ultrasonic radar system using Arduino Uno** for detecting and visualizing objects in real time, ensuring accurate distance measurement and angle detection.
2. **To implement data visualization using Processing software** by effectively displaying detected objects in a graphical radar-like interface for better interpretation and analysis.
3. **To enhance object detection accuracy and response time** by optimizing sensor placement, data filtering, and serial communication between the Arduino and Processing software.
4. **To explore the practical applications of ultrasonic radar technology** in areas such as security surveillance, obstacle detection, and automation, demonstrating its relevance in real-world scenarios.

DEPARTMENTAL MISSION & VISION :

Program Outcomes (POs)

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. **Problem analysis**: Identify, formulate, research literature, and analyse complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. **Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. **Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions

5. **Modern tool usage**: Create, select, and apply 5 appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

6. **The engineer and society**: Apply to reason informed by the contextual knowledge to health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. **Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of and need for sustainable development.

8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. **Individual and teamwork**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

10. **Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

11. **Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

12. **Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSO)

1. Students will acquire knowledge in Advance Communication Engineering, Signal and Image Processing, Embedded and VLSI System Design.

2. Students will qualify in various competitive examinations for successful employment, higher studies and research.

PO& PSO MAPPING:

PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
3	3	3	3	2	3	2	3	3	3	2	1	3	2

JUSTIFICATIONS OF MAPPING:

PO/PSO MAPPED	LEVEL OF MAPPING	JUSTIFICATION
PO1	3	The project applies mathematical concepts (such as distance calculation using the speed of sound), physics (sound wave reflection), and engineering principles (sensor integration and microcontroller programming) to build a functional radar system.
PO2	3	The system involves identifying object detection challenges, formulating distance measurement techniques, analyzing signal reception errors, and refining the ultrasonic sensor's accuracy using mathematical and scientific principles.
PO3	3	The project required designing a system that accurately detects objects and represents them visually, considering factors such as sensor placement, scanning angles, and real-time data processing.
PO4	3	The project involved testing different sensor configurations, analyzing distance errors due to environmental factors, and optimizing data filtering to improve detection accuracy.
PO5	2	Processing software was used for visualization, Arduino IDE for coding, and serial communication tools for data transmission, demonstrating the application of modern engineering tools.
PO6	3	The radar system can be applied in security surveillance, automotive safety, and obstacle detection, addressing societal needs like accident prevention and automated monitoring.
PO7	2	The project utilizes an energy-efficient ultrasonic sensor and microcontroller, demonstrating an understanding of resource-conscious engineering solutions that minimize environmental impact.

PO8	3	The project adheres to ethical engineering practices, ensuring data accuracy and responsible use of technology in areas like security and automation.
PO9	3	The project required collaboration among team members for hardware assembly, programming, and visualization, fostering teamwork in an interdisciplinary environment.
PO10	3	Effective communication was essential for documenting system functionality, presenting findings, and troubleshooting software and hardware issues within the team.
PO11	2	The project involved managing resources like sensors and microcontrollers within budget constraints and ensuring timely completion by following a structured development plan.
PO12	1	The project introduced advanced concepts in sensor technology, data visualization, and microcontroller programming, encouraging continuous learning and skill development.
PSO1	3	The project involves embedded system design (Arduino programming), real-time data processing, and communication between hardware and software (sensor data transfer and visualization).
PSO2	2	The hands-on experience gained through this project prepares students for higher studies, research in radar and automation systems, and employment in embedded systems and communication engineering fields.

ACTIVITY CHART:

JOB	**15th -30th January**	**1st -30th February**	**1st -30th March**	**1st-15th April**	**16th-30st April**	**1st-15th May**	**16th-25th May**
Literature Review	↔						
0th Review		↔					
Problem definition and requirement analysis			↔				
Midterm report and presentation				↔			
Design and Implementation					↔		
Optimization and Results						↔	
Report writing and project presentation							↔

Chapter 1

INTRODUCTION:

The *"Sonar-Based Radar System Using Arduino"* is a compact and cost-effective mini-project that simulates radar functionality using ultrasonic sensing. At its core, the system uses an **HC-SR04 ultrasonic sensor** mounted on a **servo motor**, both controlled by an **Arduino Uno** microcontroller. The sensor scans the environment by rotating within a defined angular range (typically 0° to 180°), emitting ultrasonic pulses and calculating the time taken for the echo to return from nearby objects. This time-of-flight data is used to compute distances.

The collected distance and angle data are sent in real time to a computer via serial communication, where a radar-style graphical interface is built using **Processing IDE**. This interface visually represents object positions in a 2D polar format, mimicking a real radar screen.

This project demonstrates essential concepts in **embedded systems, real-time data processing, servo control, and sensor integration**, and has valuable applications in **robotics, security systems, obstacle avoidance, and automation**. The design is modular and scalable, offering potential for future enhancements such as 360° scanning, wireless data transmission, and IoT connectivity. It serves as an excellent learning platform for understanding how hardware and software work together in intelligent sensing systems.

Chapter 2

THEORY

The increasing demand for automated surveillance, obstacle detection, and intelligent monitoring systems in robotics, security, and industrial automation has led to the development of compact, efficient, and cost-effective radar alternatives. Traditional radar systems, although highly effective, are often expensive and complex. This project aims to design and implement a simplified **ultrasonic radar system** using an **Arduino Uno** microcontroller and an **HC-SR04 ultrasonic sensor** to detect and track nearby objects in real time.

The HC-SR04 sensor functions by emitting ultrasonic pulses at 40 kHz and measuring the time taken for the echoes to reflect off obstacles and return to the sensor. By applying the **time-of-flight** principle and knowing the speed of sound, the system calculates the distance between the sensor and the object. The sensor is mounted on a **servo motor**, which rotates through a specified angular range (typically 0° to 180°), allowing the system to scan its surroundings.

The Arduino collects distance and angle data, which is then sent to a computer through **serial communication**. A graphical user interface (GUI), developed using **Processing IDE**, receives this data and visually represents object positions on a simulated radar screen. This provides a real-time, radar-like sweeping display of the scanned environment.

The project explores several core engineering principles, including **sensor interfacing, servo control using PWM (Pulse Width Modulation), real-time signal acquisition, data transmission, and graphical data visualization**. The system has been tested in various conditions to analyze accuracy, response time, and adaptability to environmental factors.

Overall, the project demonstrates how ultrasonic technology can serve as a reliable and economical alternative to conventional short-range radar systems. It shows great potential for application in **autonomous robotics, proximity sensing, and smart automation systems**, especially where budget and simplicity are important constraints.

Chapter 3

PROPOSED SYSTEM

Hardware Part:

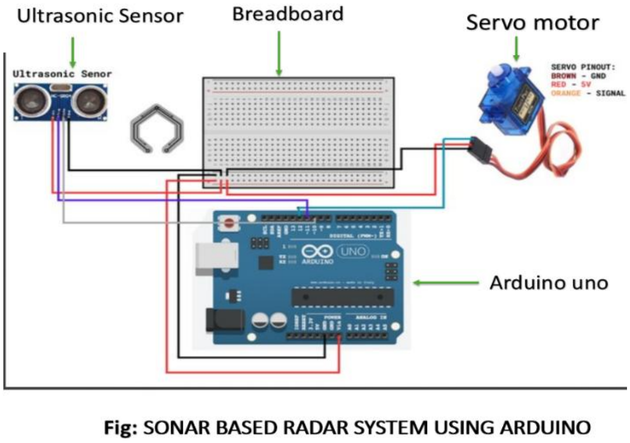


Fig.1. Circuit Diagram

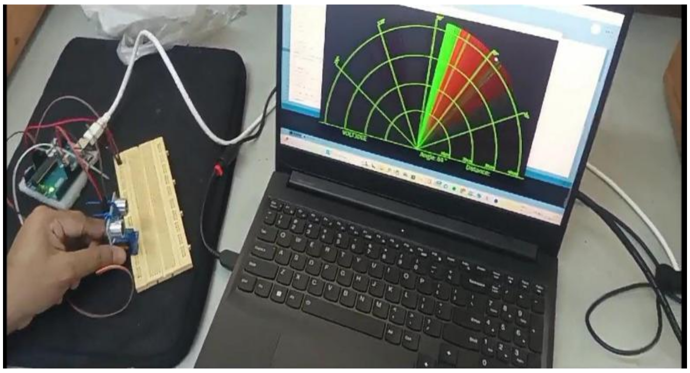


Fig.2. Proposed system in Real life

- **Arduino UNO:**
  Acts as the main controller of the system. It sends control signals to the servo motor and reads distance data from the ultrasonic sensor.
- **Ultrasonic Sensor (HC-SR04)**:
  Used for measuring distance by emitting ultrasonic waves and receiving the reflected echo from objects.
- **Servo Motor (SG90)**:
  Rotates the ultrasonic sensor from 0° to 180° to scan the surroundings like a radar.

- **Vero board:**

  Helps in distributing power (5V and GND) from the Arduino to both the ultrasonic sensor and the servo motor.

Connection Description:

The ultrasonic sensor's **VCC** and **GND** pins are connected to the **5V and GND** lines on the breadboard, which are in turn powered by the **Arduino UNO**. The **TRIG** pin of the sensor is connected to **digital pin 9** of the Arduino, and the **ECHO** pin is connected to **digital pin 10**. The servo motor has three wires: the **brown wire (GND)** is connected to the Arduino's GND via the breadboard, the **red wire (VCC)** is connected to 5V, and the **orange wire (signal)** is connected to **digital pin 11** on the Arduino. The breadboard acts as a power distribution unit to simplify and organize the connections.

Software Part:

Arduino code:

Radar_code_new.ino

```
#include <Servo.h>

const int trigPin = 8;
const int echoPin = 9;

long duration;
int distance;
Servo myServo;

void setup() {
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  Serial.begin(9600);
  myServo.attach(10);
}

void loop() {
  // Try slightly overdriving the servo (watch for strain!)
  for(int angle = -5; angle <= 185; angle++) {
    int safeAngle = constrain(angle, 0, 180); // Clamp angle safely
    myServo.write(safeAngle);
    delay(30);
    distance = calculateDistance();

    Serial.print(safeAngle);
    Serial.print(",");
    Serial.print(distance);
    Serial.print(".");
  }

  for(int angle = 185; angle >= -5; angle--) {
    int safeAngle = constrain(angle, 0, 180); // Clamp angle safely
    myServo.write(safeAngle);
    delay(30);
    distance = calculateDistance();

    Serial.print(safeAngle);
    Serial.print(",");
    Serial.print(distance);
    Serial.print(".");
  }
}

int calculateDistance() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  duration = pulseIn(echoPin, HIGH);
  distance = duration * 0.034 / 2;

  return distance;
```

Processing Code:

Radar processing

```
import processing.serial.*;
import java.awt.event.KeyEvent;
import java.io.IOException;

Serial myPort;
String angle = "";
String distance = "";
String data = "";
String noObject;
float pixsDistance;
int iAngle, iDistance;
int index1 = 0;
int index2 = 0;
PFont orcFont;

void setup() {
  size(1366, 768);
  smooth();
  myPort = new Serial(this, "COM4", 9600); //
  myPort.bufferUntil('.');
}

void draw() {
  fill(98, 245, 31);
  noStroke();
  fill(0, 4);
  rect(0, 0, width, height - height * 0.065);

  fill(98, 245, 31);
  drawRadar();
  drawLine();
  drawObject();
  drawText();
}

void serialEvent(Serial myPort) {
  data = myPort.readStringUntil('.');
  if (data != null) {
    data = data.trim();
    index1 = data.indexOf(",");
    if (index1 > 0) {
      angle = data.substring(0, index1);
      distance = data.substring(index1 + 1);
      iAngle = int(angle);
      iDistance = int(distance);
    }
  }
}

void drawRadar() {
  pushMatrix();
  translate(width / 2, height - height * 0.074);
  noFill();
  strokeWeight(2);
  stroke(98, 245, 31);
  arc(0, 0, (width - width * 0.0625), (width - width * 0.0625), PI, TWO_PI);
  arc(0, 0, (width - width * 0.27), (width - width * 0.27), PI, TWO_PI);
  arc(0, 0, (width - width * 0.479), (width - width * 0.479), PI, TWO_PI);
  arc(0, 0, (width - width * 0.687), (width - width * 0.687), PI, TWO_PI);
  line(-width / 2, 0, width / 2, 0);
  for (int angle = 30; angle <= 150; angle += 30) {
    line(0, 0, (-width / 2) * cos(radians(angle)), (-width / 2) * sin(radians(angle)));
  }
  popMatrix();
}

void drawObject() {
  pushMatrix();
  translate(width / 2, height - height * 0.074);
```

```
  strokeWeight(9);
  stroke(255, 10, 10);
  pixsDistance = iDistance * ((height - height * 0.1666) * 0.025);
  if (iDistance < 40) {
    line(pixsDistance * cos(radians(iAngle)), -pixsDistance * sin(radians(iAngle)),
         (width - width * 0.505) * cos(radians(iAngle)), -(width - width * 0.505) * sin(radians(iAngle)));
  }
  popMatrix();
}

void drawLine() {
  pushMatrix();
  strokeWeight(9);
  stroke(30, 250, 60);
  translate(width / 2, height - height * 0.074);
  line(0, 0, (height - height * 0.12) * cos(radians(iAngle)), -(height - height * 0.12) * sin(radians(iAngle)));
  popMatrix();
}

void drawText() {
  pushMatrix();
  noObject = (iDistance > 40) ? "Out of Range" : "In Range";

  fill(0, 0, 0);
  noStroke();
  rect(0, height - height * 0.0648, width, height);
  fill(98, 245, 31);
  textSize(25);

  text("10cm", width - width * 0.3854, height - height * 0.0833);
  text("20cm", width - width * 0.281, height - height * 0.0833);
  text("30cm", width - width * 0.177, height - height * 0.0833);
  text("40cm", width - width * 0.0729, height - height * 0.0833);

  textSize(40);
  text("GROUP-1", width - width * 0.875, height - height * 0.0277);
  text("Angle: " + iAngle + " °", width - width * 0.48, height - height * 0.0277);
  text("Distance:", width - width * 0.26, height - height * 0.0277);

  if (iDistance < 40) {
    // Moved distance value further right to avoid overlap
    text(iDistance + " cm", width - width * 0.145, height - height * 0.0277);
  }

  textSize(25);
  fill(98, 245, 60);

  int[] angles = {30, 60, 90, 120, 150};
  for (int angle : angles) {
    float x = (width / 2) * cos(radians(angle));
    float y = -(width / 2) * sin(radians(angle));
    resetMatrix();
    translate(width / 2 + x, height - height * 0.074 + y);
    rotate(radians(-angle + 90));
    text(angle + "°", 0, 0);
  }
  popMatrix();
}
```

Chapter 4

MATHEMATICAL FORMULATION

Mathematical Formulation of Sonar-Based Radar System

A sonar-based radar system operates by measuring the time taken for an ultrasonic pulse to travel to an object and reflect back. This principle is known as Time of Flight (ToF) and is fundamental to distance measurement in ultrasonic systems.

1. Distance Calculation

Formula:

$$d = (v \times t) / 2$$

Where:

d = Distance from the sensor to the object

v = Speed of sound in air (approx. 343 m/s at room temperature)

t = Total time for the ultrasonic pulse to travel to the object and return

Note: The division by 2 accounts for the round trip of the wave (sensor → object → sensor).

2. Angular Position

θ = Current angle of the servo motor (in degrees)

This angle is used to determine the direction in which the object is located during the scan.

3. Cartesian Coordinates for Display

To display the object's position on a 2D radar screen, convert the polar coordinates (d, θ) to Cartesian: $x = d \times \cos(\theta)$, $y = d \times \sin(\theta)$

Note: Convert θ from degrees to radians before applying trigonometric functions:

$$\theta\_rad = \theta\_deg \times \pi / 180$$

Example Calculation:

Given: $t = 0.01$ seconds

$v = 343$ m/s

Then:

$d = (343 \times 0.01) / 2 = 1.715$ meters

If $\theta = 90°$:

$x = 1.715 \times \cos(90°) = 0$

$y = 1.715 \times \sin(90°) = 1.715$

Result: The object is directly in front of the sensor, 1.715 meters away.

Chapter 5

RESULTS & DISCUSSIONS

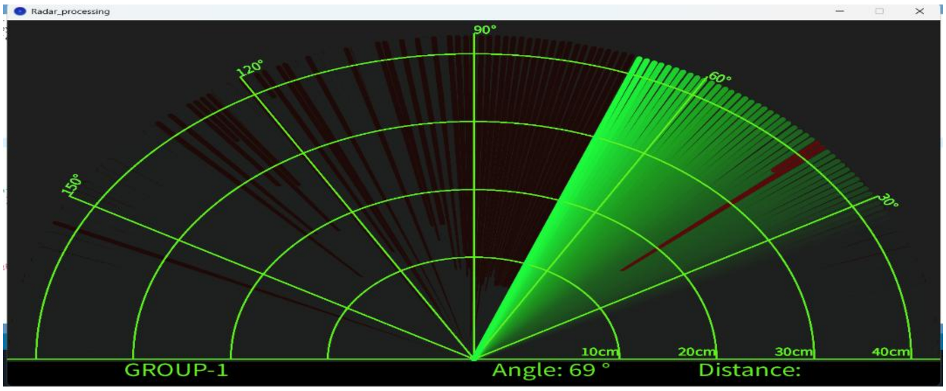


Fig.3 : Radar detection at normal case When no object is detected

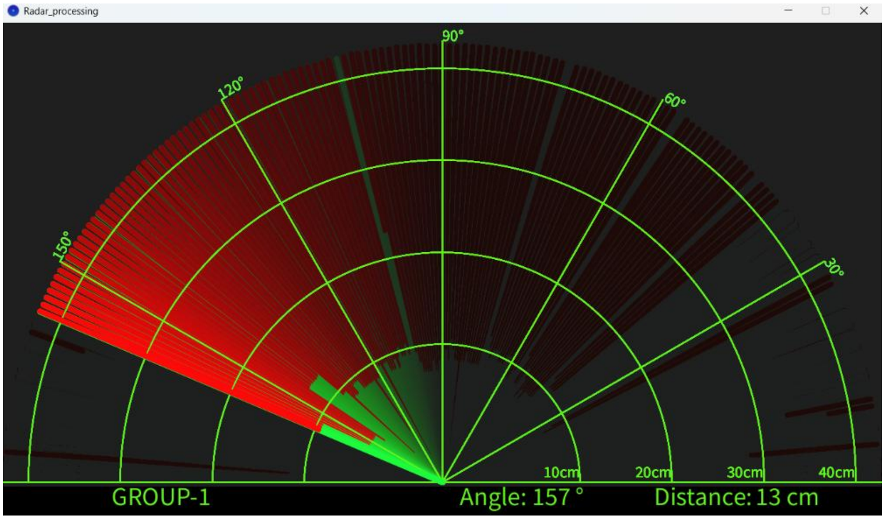


Fig.4: Radar detection when an object is detected (with distance)

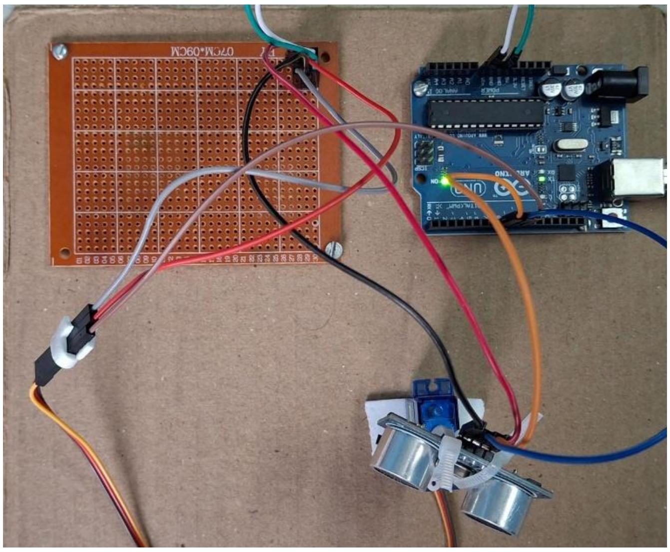
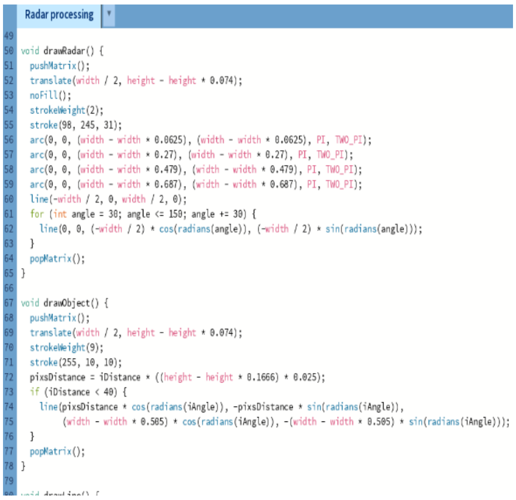


Fig.5: Radar detection hardware model



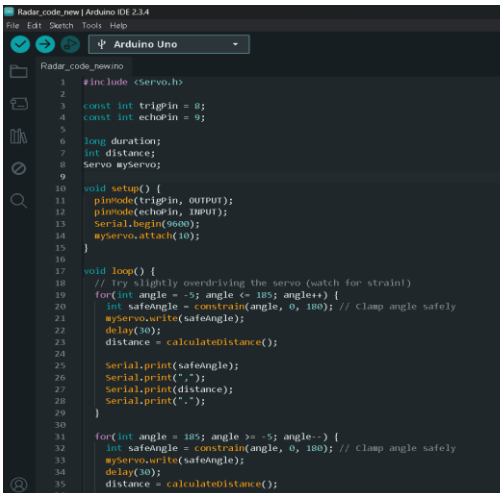


Fig.6 : Radar detection software model

Discussion

The **"Sonar-Based Radar System Using Arduino"** successfully demonstrated how **ultrasonic sensing technology** can be leveraged to detect and visualize objects in real time. By integrating the **HC-SR04 ultrasonic sensor**, a **servo motor**, and an **Arduino Uno**, the system was able to perform continuous scanning of the environment through precise **angular sweeps** and **distance measurements** based on the **time-of-flight** principle. The resulting data was transmitted to a computer and visualized through a radar-style interface developed using the **Processing software**, offering a clear and interactive display of detected objects.

Throughout the implementation, several core concepts were effectively applied, including **sensor interfacing**, **pulse control of servo motors**, **serial data communication**, and **2D graphical visualization**. The real-time responsiveness of the system highlighted the reliability of the components when used within their optimal range. Despite being a low-cost prototype, the radar system achieved reasonably accurate object detection and angular mapping for short-range applications.

However, the project also faced a few limitations. The **detection range** of the HC-SR04 sensor is limited to approximately **2–400 cm**, and its performance can be affected by **sound-absorbing surfaces**, irregular shapes, or angled objects. The servo motor's speed and mechanical precision also influenced the smoothness and accuracy of scanning. Additionally, the absence of data filtering meant that occasional noise or false readings could occur.

Nonetheless, the system demonstrates significant potential for low-cost applications in fields such as **surveillance**, **robotics**, **automation**, and **obstacle avoidance**. Overall, this project not only fulfilled its intended purpose but also provided valuable hands-on experience in **embedded systems**, **sensor networks**, and **interactive visualization**, laying the foundation for more advanced developments in real-world automation and monitoring solutions.

FUTURE PLAN:

To enhance the capabilities and practical applications of the SONAR-based radar system, several promising upgrades can be planned. A major improvement would be the integration of a graphical user interface using software like Processing or Python, allowing real-time visualization of detected objects on a radar-like screen, which makes the system more interactive and user-friendly. Implementing wireless communication through modules like Bluetooth (HC-05) or Wi-Fi (ESP8266) would enable remote monitoring of radar data on smartphones or computers, eliminating the need for a wired connection. The current scanning range can be extended from 180° to a full 360° by using a continuous rotation servo or a stepper motor, allowing the system to cover a complete circular area for surveillance. An additional feature such as an obstacle alert mechanism using buzzers or LEDs can provide real-time warnings when objects are detected within a predefined critical distance, enhancing safety applications. To ensure long-term analysis and performance tracking, data logging can be introduced through an SD card module or cloud platforms like ThingSpeak, enabling users to store and review historical data. Implementing simple object tracking algorithms would allow the system to not only detect but also follow moving objects within its range. Structurally, designing a durable and portable casing using 3D printing or acrylic sheets would protect internal components and improve the aesthetics and usability of the system. In the long term, this radar module can be mounted on robotic platforms, enabling autonomous navigation and intelligent obstacle avoidance, making it suitable for real-world applications in security, automation, and robotics.

REFERENCE:

- O. Kela, V. Deshmane, V. Dange, and S. Bhorge, "Ultrasonic Radar System using Arduino," *Quest Journals Journal of Electronics and Communication Engineering Research*, vol. 10, no. 11, pp. 09-12, 2024.

[Quest Journals]

- H. Hatem and A. Al-Fahham, "Design and Implementation of Ultrasonic Radar System for Distance Measurements Using Arduino," *International Journal of Engineering and Technology*, vol. 7, no. 4, pp. 123-129, 2018.

ResearchGate

- N. N. Mishra, H. Chaudhary, and A. Singhal, "Making of a Short-Range Radar System Using Arduino and Ultrasonic Sensor," *International Journal of Novel Research and Development*, vol. 8, no. 5, pp. 117-124, 2023.

IJNRD

- S. Margaj and A. Ghadigaonkar, "Arduino Radar Model using Ultrasonic Sensor," *International Research Journal of Engineering and Technology*, vol. 8, no. 5, pp. 2435-2441, 2021.

IRJET

- Y. Srivastava, S. Singh, and S. P. S. Ibrahim, "Autonomous Bot with ML-Based Reactive Navigation for Indoor Environment," *arXiv preprint arXiv:2111.12542*, 2021.