PROJECT REPORT

On

"Arduino Based Radar System"

This project & Research Report has been independently developed by me and is not submitted to any college or university for academic evaluation

Developed by

Name:-Suzen Kumar Mohanty (**Reg. No.** 2421331063)

SKMP PROJECT MANAGEMENT

DEPARTMENT OF IOT

POLISHLINE.BALASORE,ODISHA, INDIA

JUNE -2026

“Dedicated to my beloved parents, and teachers who have worked hard throughout my education”

SKMP PROJECT MANAGEMENT

CERTIFICATE

This is to certify that the project entitled " Arduino Based Radar System "
has been independently developed and completed by Suzen Kumar Mohanty for the purpose of enhancing technical knowledge, skills, and practical understanding.

This project is a self-initiated work carried out with personal interest and research. It has not been submitted to any college, university, or institution for the award of any degree or diploma.

The work presented in this project is genuine and has been completed by the developer himself.

Suzen Kumar Mohanty
(Project Developer)

Date: 21 / 05 / 2026
Place:Balasore,Odisha,India

SELF ATTESTATION

I hereby declare that I have personally worked on the project entitled " **Arduino Based Radar System** ".

All the data, analysis, and implementation presented in this project are the result of my own efforts and genuine work. Any external sources of information, if used, have been properly acknowledged.

This project is created solely for learning, skill development, and professional improvement, and has not been submitted to any institution for academic purposes.

SKMP PROJECT MANAGEMENT

Suzen Kumar Mohanty
(Project Developer)

Date: 21/ 05 / 2026
Place:Balasore,Odisha,India

DECLARATION

I hereby declare that the project titled "Arduino Based Radar System" is an original work carried out by me based on my own idea, research, and implementation.

This project has not been submitted previously, in part or in full, to any college, university, or institution for the award of any degree, diploma, or certification.

I confirm that all the information presented in this project is true to the best of my knowledge. Any references or sources used have been properly acknowledged.

I take full responsibility for the authenticity and originality of this work.

SKMP PROJECT MANAGEMENT

Name: Mr. Suzen Kumar Mohanty
Branch: Computer Science And Engineering
Institution : [Srinix Collage Of Engineering , Independent]
Date: [21/05/2026]
My PortPolio Link:- https://suzen-kumar-mohanty-8fdv.vercel.app/

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to everyone who directly or indirectly supported me in the successful completion of my personal project titled “IoT-Based Project”.

This project is a result of my own idea, research, and continuous effort to enhance my technical knowledge and practical skills in the field of Internet of Things (IoT). Working on this project has provided me with valuable learning experiences and improved my understanding of real-world applications.

I would also like to thank my family and friends for their constant encouragement, motivation, and support throughout the development of this project.

Finally, I am thankful to all the online resources, tutorials, and communities that helped me gain knowledge and solve challenges during the project development.

Developed By:
Suzen Kumar Mohanty

ABOUT MY SELF

I am **Mr. Suzen Kumar Mohanty**, a passionate and dedicated **Web Developer** from the field of **Computer Science and Engineering**, with strong expertise in HTML, CSS, JavaScript I specialize in building scalable, responsive, and high-performance web applications that deliver seamless user experiences across all devices and screen sizes.

With a keen eye for detail, I focus on writing clean, maintainable, and well-structured code. My development approach is centered around creating intuitive user interfaces, ensuring both functionality and aesthetic quality. I follow modern front-end development practices such as component-based architecture, responsive design principles, and performance optimization techniques to build efficient and user-friendly applications.

Beyond web development, I actively work on **IoT-based solutions**, where I integrate hardware and software to create smart, real-world systems. I have hands-on experience with Arduino Uno and various sensors, enabling me to develop projects such as environmental monitoring systems, smart automation solutions, and real-time data tracking applications.

One of my key strengths lies in connecting IoT devices with web technologies. I build interactive dashboards that allow real-time data visualization, giving users the ability to monitor, analyze, and control devices remotely. My work focuses on bridging the gap between the physical and digital world by transforming raw sensor data into meaningful insights.

I am deeply interested in developing innovative, data-driven solutions that combine web technologies with embedded systems. My goal is to create impactful applications that solve real-world problems while improving efficiency, automation, and user experience.

With a strong passion for continuous learning and innovation, I consistently explore new tools, frameworks, and technologies to enhance my skills and deliver high-quality, future-ready solutions.

SKMP PROJECT MANAGEMENT

Name: Mr. Suzen Kumar Mohanty
Branch: Computer Science And Engineering
My PortPolio Link:- https://suzen-kumar-mohanty-8fdv.vercel.app/

ABSTRACT

As we know that Radar is an electronic device which consists or contains radio waves to detect the object means to find the distance, angle of the object and range as we know the radar is radio detection and ranging. So, In this project we use ultrasonic sensors in place of radar because both work the same but the main difference is that the ultrasonic sensor uses sound waves to find, detect the distance, angle of velocity and range of the object. So, In this project we will use an ultrasonic sensor, servo motor and Arduino. Now servo motor is used to rotate the ultrasonic sensor and Arduino used for the processing of our project, all with this we also used two software that is Arduino IDE and other is Processing app. In this we use Arduino IDE to embed the code in Arduino and with the processing app we process our project. So, In this project we will show how our projects work and show, detect the distance and angle of the object and how it is shown on the pc with the help of an Arduino and ultrasonic sensor. There is small kind of the information that we will give on the abstract that in this project we put some object ahead of the ultrasonic sensor but when there is no object it will show only green graph only on the screen and not shown any thing in any position but when we put some object in some position than when the ultrasonic sensor means radar comes ahead of this object it will show a red lines on this graph on this position and also shows angle ,distance of this object which is basically made for local patch areas like our borders.

SKMP PROJECT MANAGEMENT

LIST OF FIGURES

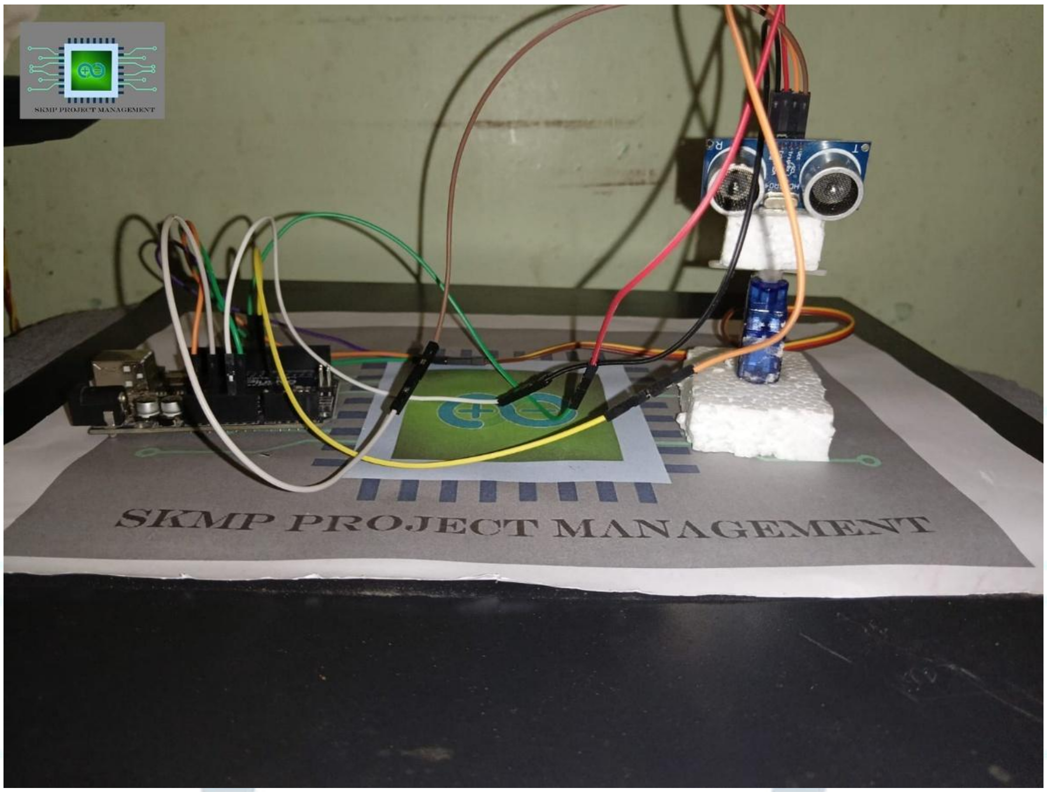
SKMP PROJECT MANAGEMENT

TABLE OF CONTENTS

SKMP PROJECT MANAGEMENT

Chapter 1

Introduction



FIG(1,1) Radar Real Image

This report presents our project, **Arduino-Based Ultrasonic Radar**, which demonstrates how ultrasonic technology can be used for object detection and tracking. The report explains the major aspects of the project, including its hardware components, software requirements, working principle, and applications.

Radar is an electronic system that uses radio waves to detect objects and determine their distance, range, and direction. However, in this project, an ultrasonic sensor is used instead of a conventional radar system. Although both systems perform similar functions, they differ in the type of waves they use. Traditional radar employs radio waves, whereas ultrasonic radar uses high-frequency sound waves to detect nearby objects and calculate their distance and angular position.

Two software platforms are used in this project. The first is the **Arduino IDE**, which is used to write, compile, and upload the program to the Arduino microcontroller. The second is the **Processing application**, which provides a graphical radar interface and displays the detected objects in real time.

The hardware setup consists of an Arduino board, an HC-SR04 ultrasonic sensor, and a servo motor. The ultrasonic sensor has four pins: Trigger (Trig), Echo, VCC, and Ground. The servo motor has three terminals: Signal, VCC, and Ground. The servo motor rotates the ultrasonic sensor through a predefined angle, allowing the system to scan the surrounding area.

During operation, the ultrasonic sensor rotates continuously with the help of the servo motor. If no object is detected within the scanning range, the Processing interface displays only a green scanning line. When an object enters the detection range, the system identifies its position and displays it as a red indication on the screen, along with its corresponding distance, angle, and range information.

This project provides a simple, low-cost, and efficient method for short-range object detection and monitoring.

Objective

The primary objective of this project is to design and develop an Arduino-Based Ultrasonic Radar System capable of detecting objects and determining their distance and angular position.
The objectives of this project are:

- To build an ultrasonic radar system using Arduino.
- To detect both stationary and moving objects.
- To measure the distance between the sensor and the detected object accurately.
- To determine the angular position of the detected object.
- To rotate the ultrasonic sensor using a servo motor through a scanning range of approximately 0° to 150°.
- To display the object's distance, angle, and detection status in real time using the Processing application.
- To detect objects up to 400 cm away from the ultrasonic sensor under suitable operating conditions.
- To reduce human effort and improve monitoring efficiency.

This system can be used for local surveillance, obstacle detection, and security applications. It may also be adapted for use in border villages, residential security systems, and military bunkers. Since ultrasonic sensors are less affected by lighting conditions, they can operate effectively in both day and night environments, making them suitable for patrolling and

SKM PROJECT MANAGEMENT

restricted-area monitoring

Problem Statement

Electronic components used to build practical circuits often require careful testing, troubleshooting, and optimization to achieve the desired performance. During the development of this Arduino-Based Ultrasonic Radar System, I encountered several challenges related to sensor calibration, servo motor synchronization, wiring connections, and software integration. Ensuring accurate object detection and smooth communication between the Arduino IDE and the Processing application was essential for achieving reliable system performance. Addressing these issues successfully contributed to the effective implementation of the project.

Scope

This project has significant potential for future development and enhancement. By incorporating wireless ultrasonic sensors, the system could monitor a much larger area and provide complete surrounding coverage. The addition of cameras and image processing techniques could enable visual identification of detected objects. Furthermore, integrating Internet of Things (IoT) technologies, wireless communication, and advanced sensors could transform this prototype into a more intelligent surveillance and monitoring system suitable for real-world applications.

SKMP PROJECT MANAGEMENT

Methodology

To evaluate the performance of the Arduino-Based Ultrasonic Radar System, I assembled the hardware components, completed the programming, and tested the system using objects placed at different distances within the sensor's detection range.

The ultrasonic sensor was mounted on a servo motor, allowing it to rotate and scan the surrounding area. The Processing application displayed the scanning output in real time on the computer screen. When the sensor detected an object, the graphical interface displayed a red indication showing the object's distance and angular position.

For validation, an object was placed at a measured distance of 30.5 cm using a ruler, while the system reported a distance of 32 cm. Similarly, another object positioned at 20 cm was detected at approximately 21 cm. Based on these observations, the system achieved an estimated detection accuracy of about 95%.

The Arduino microcontroller served as the central processing unit of the system. It controlled the rotation of the servo motor, processed the data received from the ultrasonic sensor, calculated the object's distance and angle, and transmitted the information to the Processing application for visualization.

The complete operation of the system depended on two software platforms: Arduino IDE and Processing. The Arduino IDE was used to develop and upload the control program to the Arduino board, while the Processing application generated the radar visualization interface. Through the integration of hardware and software, the system successfully detected and displayed nearby objects in real time.

. Applications

The Arduino-Based Ultrasonic Radar System has several practical applications in safety, surveillance, and monitoring. Some of its important applications include:

SKMP PROJECT MANAGEMENT

Air Force Applications: Radar systems are widely used in aircraft to detect obstacles, provide weather information, and assist pilots during landing operations in low-visibility conditions. The concepts demonstrated in this project reflect the basic principles used in aviation radar systems.

Naval Applications: Marine radar systems help ships determine bearing and distance to avoid collisions, assist in navigation, and identify surrounding objects. Similar detection principles can support maritime safety and monitoring applications.

Military Applications: This system can be adapted for border surveillance, bunker monitoring, and night patrolling. It can assist security personnel in detecting movement and identifying the location of nearby objects within a limited range.

Meteorological Applications: Radar technology is extensively used for monitoring precipitation, wind patterns, and severe weather conditions. It plays an important role in weather forecasting and environmental observation.

Automotive Applications: Modern vehicles use ultrasonic sensors for parking assistance and obstacle detection. These systems alert drivers about nearby objects and improve vehicle safety during maneuvering.

The concepts demonstrated in this project can be further extended and integrated into advanced monitoring and security systems.

Working

In this section, I explain the working principle of my Arduino-Based Ultrasonic Radar System. First, I assembled all the required hardware components and uploaded the necessary programs using the Arduino IDE and Processing application. After connecting the Arduino board to a personal computer or laptop, the system became operational.

The ultrasonic sensor was mounted on a servo motor, which continuously rotated to scan the surrounding area. As the sensor swept across its scanning range, it transmitted ultrasonic waves and received the reflected signals from nearby objects. The Arduino microcontroller processed these signals to calculate the distance and angular position of the detected objects.

If no object was present within the detection range, the Processing interface displayed only a green scanning line and graph on the computer screen. However, when an object entered the scanning area, the interface displayed a red indication showing the object's position, along with its distance, angle, and range information.

The accuracy and smooth operation of the system depended on proper wiring, correct coding, and the use of reliable hardware components. Even minor errors in the program or loose connections could affect the measured values. Therefore, special attention was given to the circuit connections and software implementation to ensure accurate object detection and real-time visualization.

Thus, my project successfully demonstrated the working of an ultrasonic radar system capable of detecting and displaying nearby objects in real time.

Advantages of the Project

- The system is simple, cost-effective, and easy to construct.
- Ultrasonic sensors are capable of operating effectively in both bright and dark environments.
- Object detection is not significantly affected by the color or transparency of the target.
- The system provides real-time information about the distance and angular position of detected objects.
- It offers high sensitivity and accuracy for short-range object detection.
- The system is resistant to environmental conditions such as dust, rain, and low-light situations.
- It can be used in applications such as obstacle detection, security monitoring, robotic navigation, and parking assistance systems.
- The project can be further enhanced and adapted for advanced surveillance and patrolling applications.

Limitations of the Project

- Soft materials that absorb sound waves may reduce the detection capability of the ultrasonic sensor.
- Since ultrasonic sensors rely on sound propagation through air, the system cannot operate in a vacuum.
- The detection range is limited and depends on the quality and specifications of the ultrasonic sensor used.
- Irregularly shaped surfaces may affect the accuracy of distance measurements.
- The system is designed primarily for short-range applications and cannot replace conventional long-range radar systems.
- The scanning speed is restricted by the rotational speed of the servo motor.
- Environmental noise and improper calibration may occasionally influence sensor performance

SKMP PROJECT MANAGEMENT

Chapter 2

OVERVIEW OF THE PROJECT

Theory of Operation

The operation of this Arduino-Based Ultrasonic Radar System is based on ultrasonic sound waves. Conventional radar systems use radio waves to detect the distance, range, and direction of objects. In contrast, my project uses an ultrasonic sensor that emits high-frequency sound waves to detect nearby objects and determine their distance and angular position.

The primary objective of this system is to monitor nearby areas and provide real-time information about the presence and location of objects. The ultrasonic sensor continuously transmits sound waves and receives the reflected signals from surrounding objects. By measuring the time taken for the echo to return, the Arduino microcontroller calculates the distance of the object from the sensor.

This project can be useful for short-range surveillance and night patrolling applications because ultrasonic sensors are highly sensitive and capable of detecting nearby obstacles effectively. The system can also be adapted for use in robotic navigation, vehicle parking assistance, obstacle avoidance, and security monitoring.
Advanced versions of this technology can incorporate wireless communication, cameras, and IoT capabilities to enhance functionality and extend its range and applications.

The Idea

SKMP PROJECT MANAGEMENT

The inspiration for this project came from the need for simple and affordable surveillance systems used for safety and monitoring purposes. Modern defense systems use advanced radar technologies to detect and track objects. I aimed to demonstrate similar object-detection principles using easily available components such as Arduino, ultrasonic sensors, and servo motors.

In addition to security applications, similar technologies are widely used in automobiles for parking assistance and obstacle detection. Through this project, I sought to develop a low-cost prototype that could be further enhanced using wireless communication, mobile applications, and IoT technologies. Future improvements could enable remote monitoring and control through smartphones, making the system more practical and versatile.

Historical Background of Radar

The term RADAR stands for **Radio Detection and Ranging**. The development of radar technology began in the nineteenth century with experiments demonstrating that radio waves could be reflected by metallic objects. These discoveries were based on the principles of electromagnetism and laid the foundation for modern radar systems.

During the early twentieth century, radar technology advanced significantly and became widely used for military applications, navigation, and aircraft guidance. Over time, radar systems evolved into sophisticated technologies capable of long-range detection and tracking.

Inspired by these principles, my project uses ultrasonic sensing technology to perform similar object-detection tasks at a short range. Although ultrasonic radar cannot replace conventional radar systems, it provides an effective and economical solution for educational purposes and local monitoring applications.

Introduction

The main objective of my project is to use an ultrasonic sensor connected to an Arduino Uno board to detect nearby objects and display their positions in real time. The ultrasonic sensor sends signals to the Arduino, which processes the information and transfers it to the Processing application running on a computer.

The graphical interface displays the detected object's distance, angle, and range, making it easier to identify the location of obstacles within the scanning area. This system demonstrates the practical implementation of ultrasonic sensing technology for object detection and visualization.

Technological Overview

This project combines both hardware and software components to achieve real-time object detection and visualization.

The software components used in this project are **Arduino IDE** and **Processing IDE**. Arduino IDE is used to write, compile, and upload the control program to the Arduino Uno microcontroller. Processing IDE is used to develop the graphical radar interface that displays the detected objects on the computer screen.

The hardware components include the Arduino Uno board, HC-SR04 ultrasonic sensor, servo motor, connecting wires, and a personal computer. The ultrasonic sensor detects the distance of nearby objects, while the servo motor rotates the sensor to scan the surrounding area. The Arduino processes the sensor data and communicates with the Processing application to generate a real-time radar display.

The integration of these hardware and software components enables the system to accurately determine the distance and angular position of nearby objects, thereby demonstrating the functionality of an Arduino-Based Ultrasonic Radar System.

SoftwareUsed

2. 6.2 ArduinoIDE

The arduino IDE is a kind of software that is used to embed the code in arduino. Arduino IDE software is onlythe software that is used for the processing in arduino based projects. So, in our project we use this software. In this software we write our programs in two languages that are c and c++. This is also known as open source because as we know that arduino is an open source hardware and software where we work on both platform hardware and software. We can also say that the Arduino integrated development environment is a type of cross platform application written in java language and the name integrated development environment is for the processing of the arduino.

Code Segment

```
#include<Servo.h>

#define trigPin 8
#define echoPin 9

long duration;
int distance ;

Servo myservo;

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

SKMP PROJECT MANAGEMENT

```
  digitalWrite(trigPin,LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = duration*0.034/2;
  return distance;
}

void setup()
{
  pinMode(trigPin , OUTPUT);
  pinMode(echoPin, INPUT);
  myservo.attach(11);
  Serial.begin(9600);
}

void loop()
{
 int i ;
 for (i=15; i<=165; i++)
 {
  myservo.write(i);
  delay(15);
  calculateDistance();
  Serial.print(i);
  Serial.print(",");
  Serial.print(distance);
  Serial.print(".");
 }
 for(i=165; i>=15; i--)
 {
  myservo.write(i);
  delay(15);
  calculateDistance();
  Serial.print(i);
  Serial.print(",");
  Serial.print(distance);
  Serial.print(".");

 }


}
```

SKMP PROJECT MANAGEMENT

2. 6.2 Processing Software

FIG(1.2) Processing software

Processing is an open-source software development environment used to create visual applications and graphical interfaces. In this project, the **Processing IDE** plays a vital role in displaying the radar output generated by the Arduino-based ultrasonic system. It is used to write, compile, and execute programs written in the Java programming language.

The Processing software receives data from the Arduino through serial communication and converts it into a real-time graphical radar display. This interface shows the scanning motion of the ultrasonic sensor and indicates the presence of detected objects by displaying their distance and angular position on the computer screen.

Processing was originally developed to support visual arts, interactive media, and educational projects. Today, it is widely used in data visualization, animation, simulation, multimedia applications, and interactive system development. In this project, Processing provides an effective and user-friendly way to visualize the operation of the ultrasonic radar system, making it easier to monitor and analyze object detection in real time.

SKMP PROJECT MANAGEMENT

Code Segment

```
import processing.serial.*; // imports library for serial communication
import java.awt.event.KeyEvent; // imports library for reading the data from the serial
port
import java.io.IOException;
Serial myPort; // defines Object Serial
// defubes variables
String angle="";
```

```
String distance="";
String data="";
String noObject;
float pixsDistance;
int iAngle, iDistance;
int index1=0;
int index2=0;
PFont orcFont;
void setup() {

  size (1200, 700); // ***CHANGE THIS TO YOUR SCREEN RESOLUTION***
  smooth();
  myPort = new Serial(this, "COM7", 9600); // starts the serial communication
  myPort.bufferUntil('.'); // reads the data from the serial port up to the character '.'.
So actually it reads this: angle,distance.
}
void draw() {

  fill(98, 245, 31);
  // simulating motion blur and slow fade of the moving line
  noStroke();
  fill(0, 4);
  rect(0, 0, width, height-height*0.065);

  fill(98, 245, 31); // green color
  // calls the functions for drawing the radar
  drawRadar();
  drawLine();
  drawObject();
  drawText();
}
void serialEvent (Serial myPort) { // starts reading data from the Serial Port
  // reads the data from the Serial Port up to the character '.' and puts it into the String
variable "data".
  data = myPort.readStringUntil('.');
  data = data.substring(0, data.length()-1);

  index1 = data.indexOf(","); // find the character ',' and puts it into the variable
"index1"
  angle= data.substring(0, index1); // read the data from position "0" to position of the
variable index1 or thats the value of the angle the Arduino Board sent into the Serial
```

SKMP PROJECT MANAGEMENT

```
Port
  distance= data.substring(index1+1, data.length()); // read the data from position
"index1" to the end of the data pr thats the value of the distance

  // converts the String variables into Integer
  iAngle = int(angle);
  iDistance = int(distance);
}
void drawRadar() {
  pushMatrix();
  translate(width/2, height-height*0.074); // moves the starting coordinats to new
location
  noFill();
  strokeWeight(2);
  stroke(98, 245, 31);
  // draws the arc lines
  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);
  // draws the angle lines
  line(-width/2, 0, width/2, 0);
  line(0, 0, (-width/2)*cos(radians(30)), (-width/2)*sin(radians(30)));
  line(0, 0, (-width/2)*cos(radians(60)), (-width/2)*sin(radians(60)));
  line(0, 0, (-width/2)*cos(radians(90)), (-width/2)*sin(radians(90)));
  line(0, 0, (-width/2)*cos(radians(120)), (-width/2)*sin(radians(120)));
  line(0, 0, (-width/2)*cos(radians(150)), (-width/2)*sin(radians(150)));
  line((-width/2)*cos(radians(30)), 0, width/2, 0);
  popMatrix();
}
void drawObject() {
  pushMatrix();
  translate(width/2, height-height*0.074); // moves the starting coordinats to new
location
  strokeWeight(9);
  stroke(255, 10, 10); // red color
  pixsDistance = iDistance*((height-height*0.1666)*0.025); // covers the distance from
the sensor from cm to pixels
  // limiting the range to 40 cms
  if (iDistance<40) {
    // draws the object according to the angle and the distance
```

SKMP PROJECT MANAGEMENT

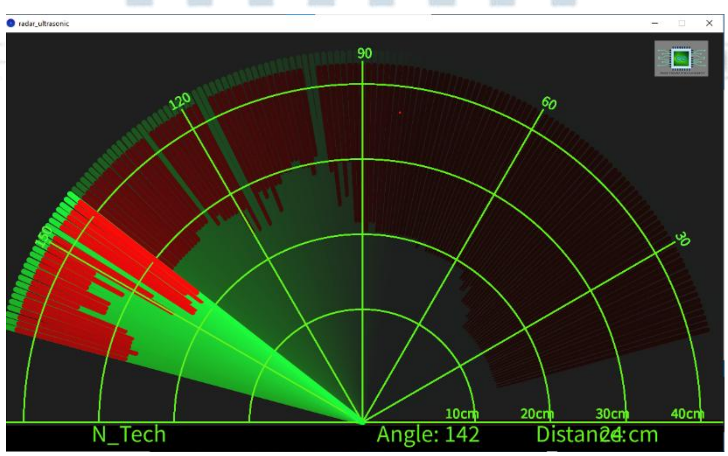
```
    line(pixsDistance*cos(radians(iAngle)), -pixsDistance*sin(radians(iAngle)), (width-width*0.505)*cos(radians(iAngle)), -(width-width*0.505)*sin(radians(iAngle)));
  }
  popMatrix();
}
void drawLine() {
  pushMatrix();
  strokeWeight(9);
  stroke(30, 250, 60);
  translate(width/2, height-height*0.074); // moves the starting coordinats to new location
  line(0, 0, (height-height*0.12)*cos(radians(iAngle)), -(height-height*0.12)*sin(radians(iAngle))); // draws the line according to the angle
  popMatrix();
}
void drawText() { // draws the texts on the screen

  pushMatrix();
  if (iDistance>40) {
    noObject = "Out of Range";
  } else {
    noObject = "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("N_Tech ", 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) {
    text("        " + iDistance +" cm", width-width*0.225, height-height*0.0277);
  }
  textSize(25);
```

SKMP PROJECT MANAGEMENT

```
  fill(98, 245, 60);
  translate((width-width*0.4994)+width/2*cos(radians(30)), (height-height*0.0907)-
width/2*sin(radians(30)));
  rotate(-radians(-60));
  text("30", 0, 0);
  resetMatrix();
  translate((width-width*0.503)+width/2*cos(radians(60)), (height-height*0.0888)-
width/2*sin(radians(60)));
  rotate(-radians(-30));
  text("60", 0, 0);
  resetMatrix();
  translate((width-width*0.507)+width/2*cos(radians(90)), (height-height*0.0833)-
width/2*sin(radians(90)));
  rotate(radians(0));
  text("90", 0, 0);
  resetMatrix();
  translate(width-width*0.513+width/2*cos(radians(120)), (height-height*0.07129)-
width/2*sin(radians(120)));
  rotate(radians(-30));
  text("120", 0, 0);
  resetMatrix();
  translate((width-width*0.5104)+width/2*cos(radians(150)), (height-height*0.0574)-
width/2*sin(radians(150)));
  rotate(radians(-60));
  text("150", 0, 0);
  popMatrix();
}
```



FIG(1,3) Radar Software

Chapter 3

METHODOLOGY

SYSTEM ARCHITECTURE

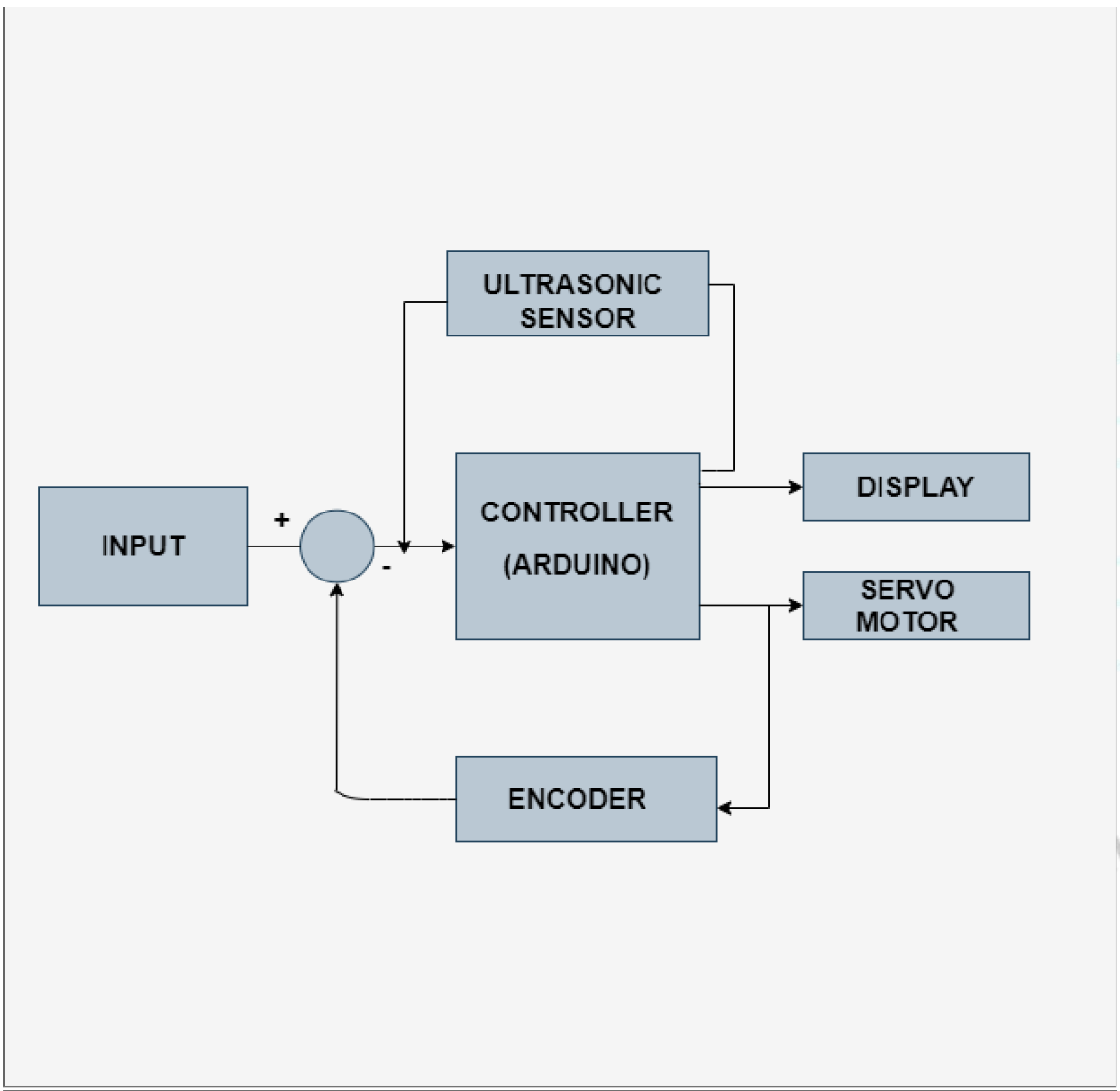


Fig-1.4 Components of Radar Project

2.4 Block Diagram Description

The block diagram of the **Short Range Radar System** is shown in Fig. 1. The system is based on the **Arduino Uno** microcontroller, which serves as the main control unit. The Arduino uses the **ATmega328P** microcontroller to coordinate all system operations.
The Arduino sends a **10-microsecond trigger pulse** to the ultrasonic sensor. The ultrasonic transmitter emits sound waves, and the reflected echo signal from nearby obstacles is received by the sensor. The microcontroller measures the time taken for the echo to return and calculates the distance of the object.
An ultrasonic sensor is mounted on a **servo motor**, allowing it to rotate through an angle of **180 degrees**. As the servo motor sweeps across the area, the sensor continuously detects obstacles within a range of approximately **250 cm**. The measured distance and angular position are transmitted from the Arduino to the computer through **serial communication at a baud rate of 9600 bps** using the **ASCII protocol**.
The received data is processed in the **Processing IDE**, where a radar-like graphical interface is generated. Obstacles detected within the scanning range are displayed on the screen as **red spots**, providing a visual representation similar to a real radar system.

2.5 Arduino Uno

The **Arduino Uno** is an open-source electronics platform consisting of both hardware and software. It is based on the **ATmega328P microcontroller** and is widely used in embedded system applications due to its simplicity and flexibility.
The Arduino Uno provides:

- **14 digital input/output pins**, of which 6 can be used as PWM outputs.
- **6 analog input pins** for reading analog signals.
- **3 ground (GND) pins** for circuit grounding.
- A **RESET pin** to restart the microcontroller when required.
- An **AREF (Analog Reference) pin** for analog voltage reference.
- **3.3 V and 5 V output pins** to power external devices.
- A **DC barrel jack** for external power supply input.
- A **USB port** used for programming and serial communication with a computer.
- **Power and status indicator LEDs** to show the operating condition of the board.

SKMP PROJECT MANAGEMENT

Fig-2.2 Arduino uno

2.6 Ultrasonic Sensor

The **ultrasonic sensor** is one of the most important components used in this project. Instead of using an actual radar system, an ultrasonic sensor is employed because it can detect objects using **sound waves**, making it cost-effective and suitable for short-range applications. Similar t determine the distance and position of object frequency sound waves to measure the **dista** obstacles.

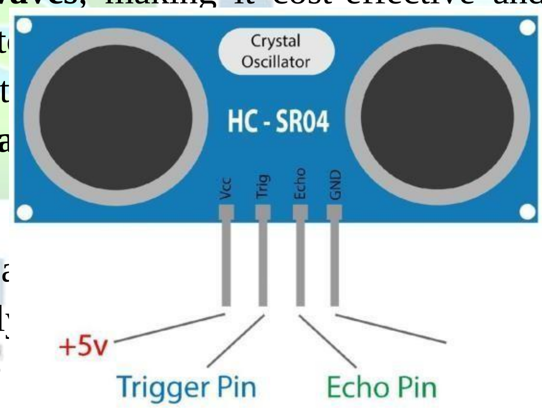


Fig-2.3 Ultrasonic Sensor

The ultrasonic sensor consists of two main pa The transmitter emits ultrasonic waves, usuall travel through the air. When these waves strike

and received by the receiver section of the sensor. The microcontroller calculates the time taken for the echo to return and determines the distance of the object.

Ultrasonic sensors are widely used because they provide accurate measurements, are inexpensive, and operate effectively under various lighting conditions. In this project, the sensor continuously scans the surroundings while mounted on a servo motor, enabling the radar system to detect obstacles within its operating range.

Features of the Ultrasonic Sensor

- Operates using high-frequency sound waves (typically 40 kHz).
- Consists of a transmitter and a receiver.
- Measures the distance of objects using the echo principle.
- Provides non-contact object detection.
- Offers accurate and reliable distance measurement.
- Suitable for short-range applications.
- Works efficiently in both bright and dark environments.
- Easy to interface with Arduino and other microcontrollers.
- Low power consumption and low cost.

Commonly used in robotics, obstacle avoidance, parking assistance, and radar visualization system

2.7 SERVOMOTOR

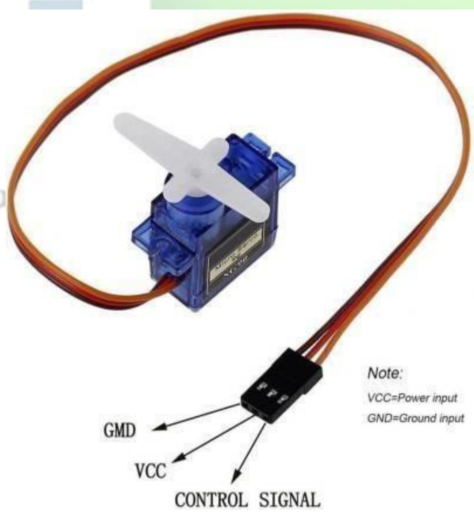


Fig-2.4 ServoMotor

The servo motor is an important component used in this short-range radar system. It is a rotary actuator that allows precise control of angular position. In this project, the ultrasonic sensor is mounted on the servo motor so that it can rotate and scan the surrounding area.
The servo motor receives control signals from the Arduino Uno, which determines the angle of rotation. Based on these signals, the servo motor rotates the ultrasonic sensor from 0° to 180°

and then back again, enabling the sensor to detect obstacles over a wide area. As the sensor moves, distance measurements are continuously taken and transmitted to the Processing software, where they are displayed on the radar interface.
A servo motor contains a DC motor, a gear mechanism, a position sensor (potentiometer), and a control circuit enclosed within a compact housing. The built-in feedback mechanism ensures accurate positioning and stable operation, making it suitable for applications requiring controlled movement.
Features of the Servo Motor

- Provides precise angular position control.
- Operates within a specific range, typically 0° to 180°.
- Contains an internal feedback mechanism for accurate positioning.
- Offers smooth and controlled movement.
- Easy to interface with Arduino using PWM signals.
- Compact in size and lightweight.
- Low power consumption. s
- High reliability and good holding torque.
- Suitable for scanning and tracking applications.
- Widely used in robotics, automation, antenna positioning, and radar systems.

2.8 JUMPER WIRE

Fig-2.4 Jumper Wire

A **jumper wire** is an insulated electrical wire with connector pins at one or both ends. It is used to establish temporary electrical connections between electronic components during circuit prototyping and testing. Jumper wires eliminate the need for soldering, making them ideal for use with breadboards, Arduino boards, sensors, and other development modules.
In this project, jumper wires are used to connect the **Arduino Uno**, **HC-SR04 Ultrasonic Sensor**, and **Servo Motor**. They provide reliable transmission of power and signal connections between the components, ensuring proper communication and operation of the radar system.

Types of Jumper Wires

- **Male-to-Male (M-M):** Used to connect two female headers or a breadboard to another breadboard.
- **Male-to-Female (M-F):** Used to connect an Arduino board to modules or sensors with male header pins.
- **Female-to-Female (F-F):** Used to connect two male header pins or modules together.

Advantages

- Easy to use and reusable.
- No soldering required.
- Ideal for rapid prototyping and testing.
- Provides flexible and reliable electrical connections.
- Available in different lengths and colors for easy identification.

2.9 BREAD BOARD

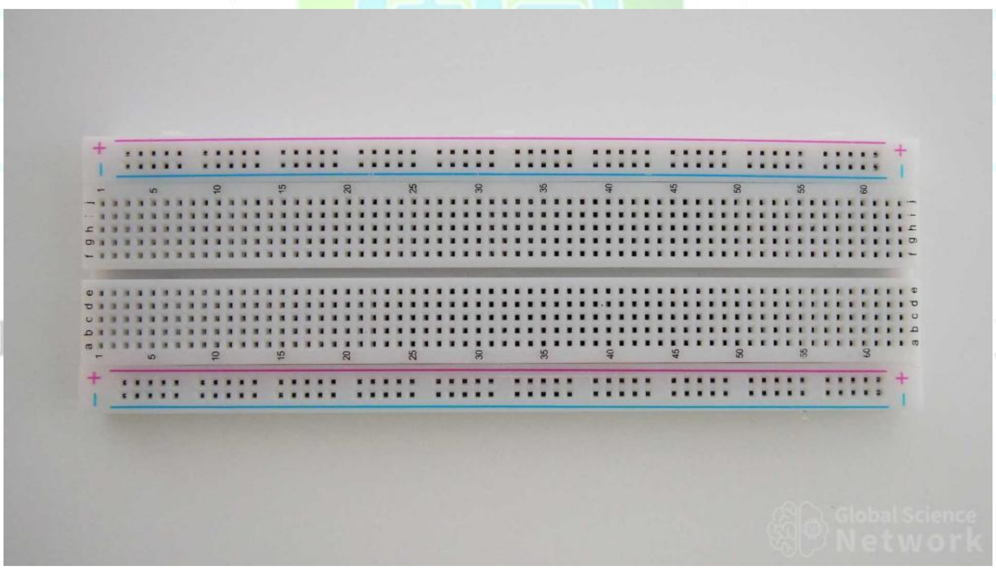


Fig-2.5 Bread Board

A **breadboard** is a reusable prototyping board used to build and test electronic circuits without soldering. It contains a network of interconnected metal strips beneath the surface, allowing electronic components and jumper wires to be easily inserted, connected, removed, and modified. Breadboards are widely used for designing, testing, and debugging electronic circuits before creating a permanent PCB.

In this project, the breadboard is used to make temporary connections between the **Arduino Uno**, **HC-SR04 Ultrasonic Sensor**, **Servo Motor**, and jumper wires. It simplifies circuit assembly, enables quick modifications, and helps verify the circuit before final implementation.

Features

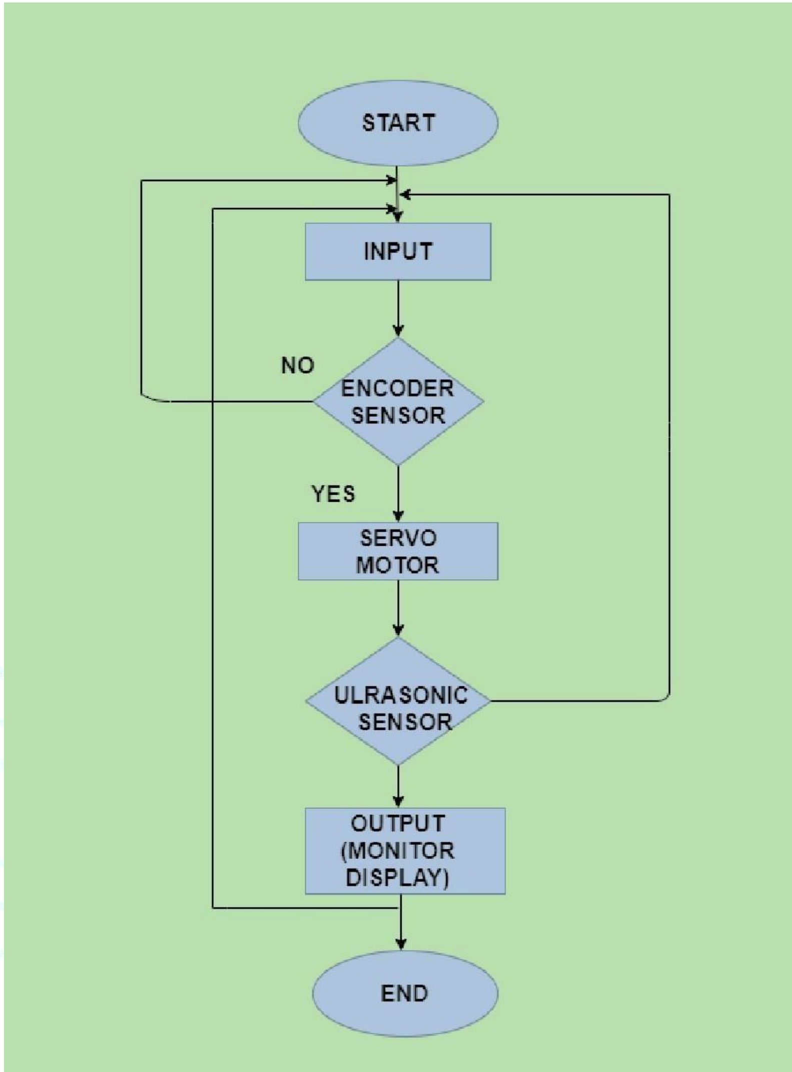
- Reusable and solderless design.
- Easy insertion and removal of components.
- Suitable for rapid circuit prototyping.
- Compatible with Arduino, sensors, ICs, LEDs, resistors, and other electronic components.
- Available in various sizes depending on project requirements.
-

Advantages

- Eliminates the need for soldering.
- Saves time during circuit development and testing.
- Allows easy troubleshooting and circuit modifications.
- Reduces the risk of damaging electronic components.
- Cost-effective and ideal for educational, research, and prototype development projects.

SKMP PROJECT MANAGEMENT

FLOWCHART



SKMP PROJECT MANAGEMENT

Fig-3.3 Flow Chart of Radar M

As we know that flow chart is used to know or explain the processor better understand anything. So, In this flowchart we try to explain the process or entire process of our project. This help us to better understand means firstly we start that comes to its input process means which is given by the ultrasonic sensor to detect the object and gives supply to our Arduino, means gives input to the processing software. It detects the object by rotating with the help of a servomotor. Then next it comes to the output means to the monitor display it shows us the graphical representation of the output which also shows the distance and the range or angle of the object that is detected by our radar. So, this is the flow chart that helps us to better understand the process.

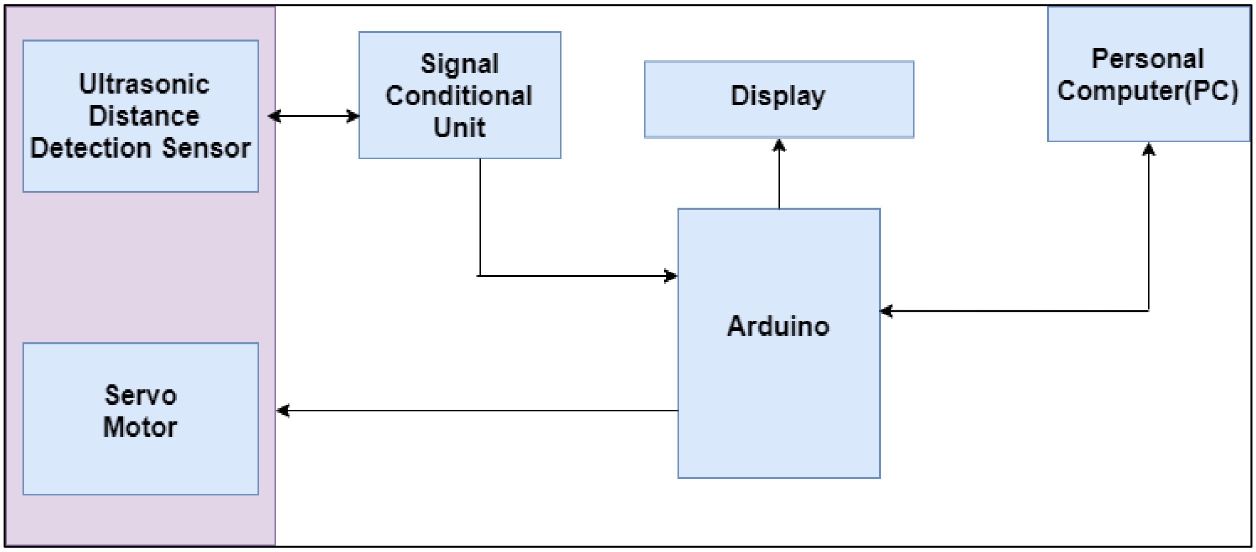


Fig-3.4 Hardware system design of radar system

Hardware system design:

In this project we are busily using three hardware's or our project mainly depends on three projects one is Arduino, Servo motor and Ultrasonic sensor. All these hardware's are most important and also other hardware's like jumper wire and breadboard e used in this project. On the previous page we make the hardware system design that is easily understandable to each of us. So, in this diagram we have given every step of hardware design that is used in our project.

System circuit design:

In the previous figure we also explain the circuit design of our project in which we connect the digital pin of the Arduino with the trigger and the digital pin eight with the echo and the Vcc with the 3.3 volt pin and ground with the ground pin. Because in ultrasonic sensors there are four connections one is echo, trigger, Vcc and Ground. So in the previous page there is a circuit diagram or circuit design for our project.

SKMP PROJECT MANAGEMENT

Chapter 4

Literature Review

A review of previous research shows that Arduino-based radar systems have become a widely accepted and cost-effective solution for various monitoring and detection applications. These systems are not only efficient and reliable but also economically feasible. In addition to radar applications, ultrasonic sensors have been successfully used in several other practical fields.

One study presents a monitoring system designed to measure the speed of water flow and the water level of rivers using an ultrasonic sensor connected to an Arduino microcontroller. When a river cannot accommodate the increasing volume of water, flooding may occur. Early detection of rising water levels and flow speed can help prevent disasters by providing timely warnings. Experimental results showed that the ultrasonic sensor provides high accuracy under controlled conditions. However, when deployed in real rivers, measurement errors may occur due to strong waves, varying water levels, water flow speed, and floating objects. To improve performance, calibration techniques were introduced. The Arduino acts as the main controller, and the collected data can be transmitted to a database server for continuous monitoring.

Another study proposes a smart driver monitoring and vehicle control system aimed at reducing road accidents by monitoring the driver's behavior. The system addresses major causes of accidents such as alcohol consumption, driver negligence, drowsiness, and medical emergencies. It incorporates sensors, motors, relays, and an ultrasonic sensor to detect nearby vehicles and warn the driver of potential collisions. The system updates relevant information for the vehicle owner and overcomes many limitations of previous technologies, making it more efficient, reliable, cost-effective, and practical.

Several researchers have also developed Arduino-based radar systems using ultrasonic sensors, servo motors, and Arduino boards. These systems detect and track nearby objects by measuring their distance and angle. Earlier methods faced challenges in accurately measuring distances between certain objects, but the introduction of ultrasonic distance measurement enabled reliable non-contact detection. Such radar systems significantly reduce power consumption while providing accurate obstacle detection. The ultrasonic sensor is mounted on a servo motor, allowing it to scan from left to right and display the detected obstacles on a screen.

Another research paper focuses on obstacle detection in known environments using an Android-based mobile camera. This system is particularly helpful for visually impaired individuals who often rely on walking sticks for navigation. The proposed solution improves safety by detecting obstacles and assisting with navigation. The system uses indoor mapping techniques in which reference images of different floors are stored for comparison. The developed algorithm performs real-time obstacle detection with high accuracy. Various obstacle detection methods, including SONAR sensors and laser cameras, are discussed. The proposed Android-based camera system continuously scans the selected area in front of the user to identify obstacles and provide navigation assistance.

SKMP PROJECT MANAGEMENT

Last Chapter

CONCLUSION & DISCUSSION

CONCLUSION

Various advanced control techniques provide designers with greater control over modern applications. In this project, the proposed design and implementation of the Arduino-based Radar System were successfully developed and tested on a small scale. The study demonstrates that radar systems have significant potential for future applications due to their reliability, cost-effectiveness, and security features.

The developed system can be applied in various fields, including obstacle detection, surveillance, robotics, and automation. Moreover, the system can be further enhanced to meet future technological requirements and user demands. Since this project focuses on a short-range radar system, the scope of the research is limited. The current system detects objects only within a range of **0° to 180°** because the servo motor used can rotate only within this angle. As a result, the system is not suitable for full-area monitoring or large-scale obstacle detection.

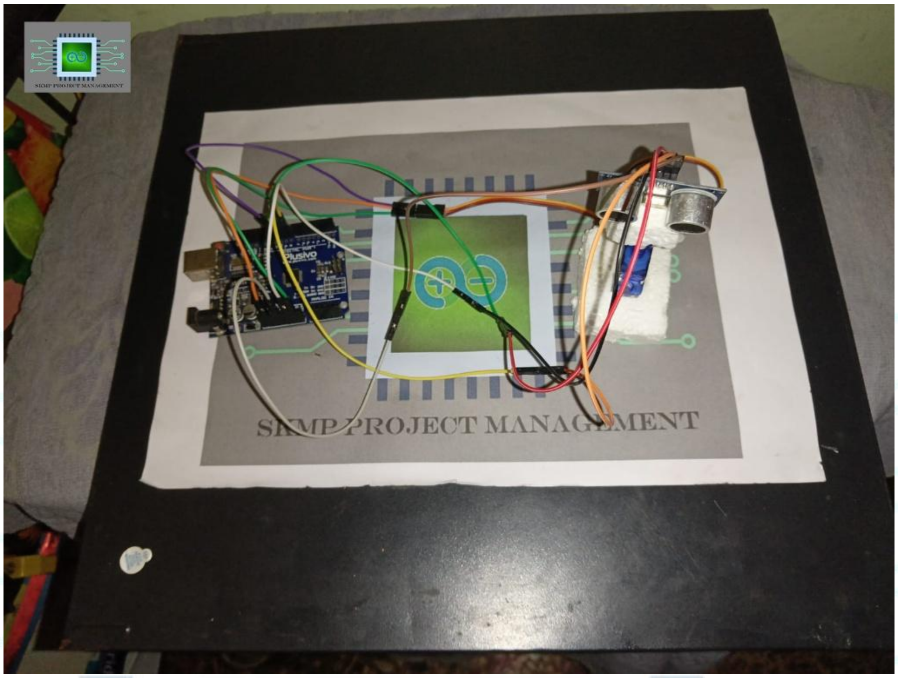
Future improvements include replacing the existing servo motor with a **360° rotating servo motor** to achieve complete environmental scanning. A higher-range ultrasonic sensor can also be integrated to increase the detection distance. Additional features such as portability, wireless communication, automatic alert systems, and obstacle-avoiding robotic platforms with surveillance capabilities can further improve the system's performance and practical applications.

SKMP PROJECT MANAGEMENT

DISCUSSION

In this paper, an ultrasonic sensor was successfully used to implement an Arduino-based radar system, and the obtained results met the expected objectives. The developed prototype effectively detects obstacles and displays their position in real time while reducing the need for human intervention. The system is simple, low-cost, and suitable for educational, industrial, and automation applications.

Although the current prototype has limitations in terms of scanning angle and detection range, it demonstrates the feasibility of using ultrasonic sensing technology for radar-based object detection. With further improvements, such as a 360° scanning mechanism, long-range sensors, wireless connectivity, and enhanced signal processing, the system can be adapted for real-time monitoring, smart surveillance, autonomous robots, and other advanced applications.



Fig(3.5) Radar Original image 2

SKMP PROJECT MANAGEMENT

Last Chapter

REFERENCES

[1] A. E. Oja, A. Myam, O. Olwadailol, and L. Adwale, **"Embedded System-Based Radio Detection and Ranging (RADAR) System Using Arduino and Ultrasonic Sensor,"** *Embedded Systems and Applications*, 2017.

[2] S. Mehta and S. Tiwari, **"Radar System Using Arduino and Ultrasonic Sensor,"** *International Journal of Novel Research and Development (IJNRD)*, vol. 3, no. 4, April 2018.

[3] A. Tedeschi, S. Calatera, and F. Benedetto, **"Ultrasonic Radar System Using Arduino,"** *IEEE Sensors Journal*, vol. 17, no. 14, July 15, 2017.

[4] G. Kiruthikamani, B. Saranya, and P. Pandyan, **"Intelligent Driver Monitoring and Vehicle Control System,"** *International Journal for Scientific Research and Development (IJSRD)*, vol. 5, no. 9, 2017.

[5] M. M. Abdulkaem, Q. A. Mohamed, and M. M. Shakir, **"Short Range Radar System."**

[6] S. Dubey, S. Tiwari, and S. Roy, **"Implementation of Radar Using Ultrasonic Sensor,"** *Indian Journal of Scientific Research*, 2017.

[7] A. Dutt, **"Arduino-Based Radar System,"** 2014.

SKMT PROJECT MANAGEMENT

APPENDIX

- The main hardware components used in this project are:
  - Arduino Uno
  - Ultrasonic Sensor (HC-SR04)
  - Servo Motor (SG90)
- The Arduino program was developed and uploaded using the **Arduino IDE**.
- The radar visualization and graphical interface were developed using the **Processing IDE** software.
- The Arduino Uno controls the ultrasonic sensor and servo motor to detect obstacles and measure their distance.
- The Processing IDE receives data from the Arduino through serial communication and displays the detected objects on a radar-like graphical interface in real time.

SKMP PROJECT MANAGEMENT

PROJECT LINKS

Project Video Link:- link
Project Folder Link:- LINK

SKMP PROJECT MANAGEMENT

==

SKMP PROJECT MANAGEMENT

DEPARTMENT OF IOT