Title

ioe

18 %
SIMILARITY INDEX

10 %
ACADEMIC

9 %
INTERNET

Date:	2022-04-28 19:11:25(+00:00 UTC)
Report ID:	626ae6e0f703c42c6
Word count:	3505
Character count:	17904

Similar sources

1	● An Alternative Arduino Sonar Using HCSR-04 - Sensors ● https://forum.arduino.cc/t/an-alternative-arduino-sonar-using-hcsr-04/504446 Internet	**5.4%**
2	● Design of Unmanned Aerial Systems ● Mohammad H. Sadraey Others	**0.6%**
3	● Processing 3.1 Windows64 | PDF | Integer (Computer Science ... ● https://www.scribd.com/document/342902570/processing-3-1-windows64-txt Internet	**0.5%**
4	● Microcontroller ● Design of Unmanned Aerial Systems,2020 Others	**0.4%**
5	● Робот BURN-E из бумаги и "радар" на ультразвуковом ... ● https://robototehnika.ru/content/article/robot-burn-e-iz-bumagi-i-radar-na-ultrazvukovom-dalnomere/ Internet	**0.4%**
6	● A smart IoT based system for monitoring and controlling the sub-station equipment ● Md. Sanwar Hossain,Mostafizur Rahman,Md. Tuhin Sarker,Md. Ershadul Haque,Abu Jahid ● Internet of Things,2019 Academic	**0.3%**
7	● CONSTRUCTION AND TESTING OF THE MEASUREMENT SYSTEM FOR EXCAVATOR PRODUCTIVITY ● Mario Klanfar,Vjekoslav Herceg,Dalibor Kuhinek,Kristijan Sekulić ● Rudarsko-geološko-naftni zbornik,2019 Academic	**0.3%**
8	● Obstacle-avoiding robot with IR and PIR motion sensors ● R Ismail,Z Omar,S Suaibun ● IOP Conference Series: Materials Science and Engineering,2016 Academic	**0.3%**
9	● Developing Educational Printable Robots to Motivate University Students Using Open Source Technologies ● Raúl Pérula-Martínez,Juan Miguel García-Haro,Carlos Balaguer,Miguel A. Salichs ● Journal of Intelligent & Robotic Systems,2015 Academic	**0.3%**
10	● Book reviews [8 books reviewed] ● John Shea ● IEEE Electrical Insulation Magazine,2014 Academic	**0.3%**
11	● Beginning Arduino ● Michael McRoberts Others	**0.3%**
12	● Enhancing an introductory programming course with physical computing modules ● Miguel Angel Rubio,Rocio Romero-Zaliz,Carolina Manoso,Angel P. de Madrid ● 2014 IEEE Frontiers in Education Conference (FIE) Proceedings,2014 Others	**0.3%**
13	● Sign language translator and gesture recognition ● Mohammed Elmahgiubi,Mohamed Ennajar,Nabil Drawil,Mohamed Samir Elbuni ● 2015 Global Summit on Computer & Information Technology (GSCIT),2015 Others	**0.3%**

14	• Web-Based Design and Implementation of Smart Home Appliances Control System • Samuel Bimenyimana,Aimable Ishimwe,Godwin Norense Osarumwense Asemota,Cecilia Messa Kemunt... • IOP Conference Series: Earth and Environmental Science,2018 Academic	0.3%
15	• An educational Arduino robot for visual Deep Learning experiments • He Hu • International Journal of Intelligent Robotics and Applications,2019 Academic	0.3%
16	• Smart Automatic Control and Monitor Water Purification Using Wireless Sensor System • Bhupesh B. Lonkar,Rajesh T. Nakhate,Manoj R. Sayankar • 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC),2018 Others	0.3%
17	• Exploring Arduino® • Jeremy Blum Others	0.3%
18	• A Reliable Wireless Communication System for Hazardous Environments • Sebastian Rosca,Simona Riurean,Monica Leba,Andreea Ionica • Advances in Intelligent Systems and Computing Digital Science,2018 Others	0.3%
19	• Design and Implementation of Unauthorized Object and Living Entity Detector with PROTEUS and Arduino Uno • Samridhi Sajwan,Shabana Urooj,Manoj Kumar Singh • Advances in Intelligent Systems and Computing Information Systems Design and Intelligent Applications,2018 Others	0.3%
20	• An Arduino board with ultrasonic sensor investigation of simple harmonic motion • A Buachoom,A Thedsakhulwong,S Wuttiprom • Journal of Physics: Conference Series,2019 Academic	0.3%
21	• Lights! speed! action! • Erik Brunvand • ACM SIGGRAPH 2013 Courses on - SIGGRAPH '13,2013 Others	0.3%
22	• An Intelligent Walking Stick for the Visually-Impaired People • Nadia Nowshin,Sakib Shadman,Sakib Shadman,Saha Joy,Saha Joy,Sarker Aninda,Sarker Aninda,Is... • International Journal of Online Engineering (iJOE),2017 Academic	0.3%
23	• SPPU-MECH-EEE: Introduction to Arduino IDE - WikiNote • https://www.wikinote.org/mod/page/view.php?id=3650 Internet	0.3%
24	• Arduino Integrated Development Environment (IDE) v1 • https://www.arduino.cc/en/Guide/Environment Internet	0.3%
25	• An Arduino Investigation of Simple Harmonic Motion • Calin Galeriu,Scott Edwards,Geoffrey Esper • The Physics Teacher,2014 Academic	0.2%

26	• Measurement of Wind Parameters • Methods of Measuring Environmental Parameters,2014 Others	0.2%
27	• A Survey on Odometry for Autonomous Navigation Systems • Sherif A. S. Mohamed,Mohammad-Hashem Haghbayan,Tomi Westerlund,Jukka Heikkonen,Hannu Tenhu... • IEEE Access,2019 Academic	0.2%
28	• IoT-Enabled Microcontroller-Based System • Nasrin Afsarimanesh,Subhas Chandra Mukhopadhyay,Marlena Kruger • Electrochemical Biosensor: Point-of-Care for Early Detection of Bone Loss Smart Sensors, Measurement and Instrumentation,2018 Others	0.2%
29	• Smart Stick for Elderly • Lakshmi Boppana,Vishal Jain,Ravi Kishore • 2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData),2019 Others	0.2%
30	• bin: The intelligent and entertaining bin • Mohammad Wasiul Haq,Nolok Noor Prem,Md. Khalilur Rhaman • 2014 17th International Conference on Computer and Information Technology (ICCIT),2014 Others	0.2%
31	• An enhancement in conventional potential field using a fuzzy system for navigation of a sumo robot • Carlos Erlan Olival Lima,Francisco Marcelino Almeida de Araujo,Mario Bibiano da Silva,Anto... • 2013 16th International Conference on Advanced Robotics (ICAR),2013 Others	0.2%
32	• Design and Analysis of a Scale-Sized Electromechanical Actuator for Unsteady Condition Monitoring Applications • Jason R. Kolodziej,William S. Craig • Volume 2: Diagnostics and Detection; Drilling; Dynamics and Control of Wind Energy Systems; Energy Harvesting; Estimation and Identification; Flexible and Smart Structure Control; Fuels Cells/Energy Storage; Human Robot Interaction; HVAC Building Energy Management; Indu...,2015 Others	0.2%
33	• An Automation System for Controlling Streetlights and Monitoring Objects Using Arduino • Zain Mumtaz,Saleem Ullah,Zeeshan Ilyas,Naila Aslam,Shahid Iqbal,Shuo Liu,Jehangir Meo,Hamz... • Sensors,2018 Academic	0.2%
34	• Automatic Parking Vehicle System • Honghong Liu,Gene Liao,Chih-Ping Yeh,Jimmy Chen • 2016 ASEE Annual Conference & Exposition Proceedings Others	0.2%
35	• ATmega8 Based Obstacle Avoiding Helicopter • Suhas Divakar • Advanced Materials Research,2011 Academic	0.2%
36	• Implementation of Ultrasonic Sensor as a Radar for Obstacle ... • http://ajeeet.ft.unand.ac.id/index.php/ajeeet/article/download/7/2/91#:~:text=The%20Arduino%20UNO%2C%20servo%20motor,using%20the%20attached%20radar%20system. Internet	0.2%
37	• Radar System Using Ultrasonic Sensor - Arduino Project Hub	0.2%

	• https://create.arduino.cc/projecthub/rithikkaparthi/radar-system-using-ultrasonic-sensor-6c71fa?ref=part&ref_id=8233&offset=16 Internet	
38	• Servo Motor - Concept Systems Inc • https://conceptsystemsinc.com/servo-motor/ Internet	**0.2%**
39	• Software | Arduino • https://www.arduino.cc/en/software Internet	**0.2%**
40	• Introduction to the Arduino Programming Language - Flavio ... • https://flaviocopes.com/arduino-programming-language/ Internet	**0.1%**

1. Introduction:

RADAR is an object detection and tracking technology that uses radio waves to determine the direction, range, altitude and speed of an object. Radar frameworks or systems are available in a variety of forms, sizes, and performance levels. At airports, some radars are utilised by aviation authorities, while others are employed by law enforcement. Applications include long-range surveillance and early-warning systems [1].There are a few options. There are a variety of ways to present radar operational data. Radar systems that have been modified and have improved technology for handling the systems are also available. At a higher level, these modified systems are utilised to gather or extract useful or crucial information [2].

The components listed below are relevant to our recommended system's functioning philosophy : an ultrasonic sensor tied to the digital input and output pins of a microcontroller (we chose Arduino). Digital output and input pins are also linked to the servo motor. Our ultrasonic sensor and servo motor linked at the same time, therefore our servo motor rotates near its axis when it rotates from 0 to 180 degrees from extreme right to extreme left [3]. To show the data (distance and angle) on a computer screen, we utilise software called "Processing Development Environment" [1].

2. Review of Literature:

Following a review of some of the articles on the use of ultrasonic sensors and ARDUINO, it was discovered that this concept is widely researched and is a mainstream concept that is continuing in development. The techniques used were not only effective and reliable, but also cost-effective [5]. Not only that, but additional extremely beneficial ultrasonic sensor uses were also noticed. This paper discusses a monitoring system that uses an ultrasonic sensor and a microcontroller to measure the speed of waves and the high of a river (Arduino). If the river can't handle the volume of water, the water will overflow and flood the land, causing a phenomena known as a flood or surge. We can avoid flooding by identifying the high of the water and monitoring its speed earlier. If we discover an issue early enough, we can remedy it before it becomes a catastrophe. A basic water level was used to test the system, and it was revealed that ultrasonic had a 96.6 percent accuracy. However, when it is used in rivers, there are numerous inaccuracies due to the various types of water levels caused by heavy waves and water speed, as well as the floating of heavy things. Unlike previous testing results, the author focused this investigation on the rate of water improvement or modification, as well as the amount of water in floods. When the

Arduino was employed as the application's controller, the test was completed. For further study, data on the depth level and water speed of this system will be uploaded to an onlinee database server, which will be analysed on a regular basis [8]. The goal of the study is to offer a system for intelligent driver monitoring and vehicle control. The author discusses some of the most common causes of accidents nowadays. These include drunkenness, carelessness, exhaustion, or medical illness on the part of the driver. The framework's different components are checked and determined to be in excellent working condition, including motors, relays, the power unit, and the ESP8299 module. When another vehicle approaches his automobile, an ultrasonic sensor alerts the driver.The status of the driver can be determined with the help of sensors installed in the vehicle, and the proprietor can be updated on the subtle elements. This technology solves all of the factors that have caused earlier technologies created for this purpose to fail, making it more helpful, efficient, cost-effective, and time-consuming [7].

The authors of this research study describe how to detect radio waves and track or range them using a radar set made up of components such as an ultrasonic sensor, a servo motor, and an Arduino. The author explains how the creation of an ultrasonic dist measurer solved the lineear measurement problem, which made it difficult to quantify dist between specific objects. It enables noncontact measurements to be taken. This system is a highly helpful radar system, according to the author, since it can read or track the distance and angle of an obstruction and show it on a monitor screen. The ultrasonic sensor was put on top of the servomotor to detect impediments in the right and left directions from 0 to 180 degrees. This research demonstrates a method for spotting barriers in a familiar area. This system is powered by a mobile camera that runs on Android. Visually impaired people have a hard time detecting obstacles and navigating while walking. They employ sticks to solve this problem, however this method, or the Arduino-based radar system strategy, is not the best way to go about it. Object indication or detectors can help individuals avoid accidents or collisions, or they can help with correct map reading. Indoor mapping is the goal of the algorithm suggested in this study. All distinctive floors are considered in the indoor environment, and a single picture is retained or saved for each distinctive level. These floor images will serve as a reference. This algorithm, according to its designer, is 96 percent accurate and functions in real time. To overcome the challenges, we use a SONAR sensor and a laser camera. This study [9] proposed a method for detecting hindrance in a known condition utilising an android-based flexible camera that scans a pre-selected region for impediment location before capturing the image.

3. Report on Present Investigation :

The development life cycle of the Radar project is depicted in the diagram below, which includes individual component design, testing, installation, and overall system testing are all tasks that must be completed. The phases indicated below can be used to break down these steps :

a) Hardware System Design.
b) Hardware Circuit Design.
c) Hardware System implementation.
d) Hardware unit testing.
e) GUI System Design.
f) GUI System Implementation.
g) GUI unit testing.
h) Entire system integration.
i) Entire system testing.

Hardware description :

Ultrasonic Sensor :

Fig 1. Ultrasonic Sensor

Ultrasonic and sonar sensors function the same way It works by measuring the distance between objects using sound waves. It operates by using sound waves to measure the distance between things. The sender sends sound waves in a given frequency and direction, and the receiver waits for them to return. By measuring the time it takes for a sound wave to return, we may determine how far away something is.

Servo motor :

Fig 2. Servo Motor

A servomotor is a whirling motor with precision control over angular position, velocity, and acceleration. It is built using a suitable motor and a position feedback sensor. It also necessitates the use of a complex controller, which is usually a separate module dedicated to servomotors. Servomotors are not a distinct type of motor, but rather use servomechanism to achieve closed loop control with a standard open loop motor, according to its essential operating concept. Robotics, CNC machines, and automated manufacturing are just a few examples of where servomotors are used.

Arduino :

Fig 3. Arduino Uno

The Arduino platform is a free of charge and open source hardware and software electronics platform. Thus making it simple to write code and upload it to the board. Windows, Mac OS X, and Linux are all supported. The environment is written in Java and uses open-source tools such as Processing as a foundation. This application may be run on any Arduino board. The Arduino software IDE includes a text editor for writing code, a message box, a text terminal, and a toolbar with basic function buttons. By uploading and downloading code, it links to Arduino and Genuine devices. Sketches are programmes developed in the Arduino programming language.

Bread board:

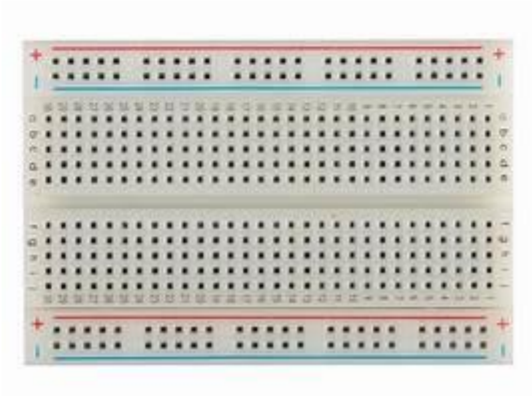


Fig 4. Bread Board

When learning how to create circuits, breadboards are one of the most important components. This lesson will teach you all you need to know about breadboards, including what they are, why they're called that, and how to use one. You should now be able to design a basic circuit on a breadboard and understand how they function.

Jumper wires:

Fig 5. Jumper wires

Jumper wires are cables with connector pins on both ends that are used to connect two sites without soldering. With breadboards and other prototype tools, jumper wires are widely utilised to allow for fast circuit adjustments as needed.

(A) Hardware System Design

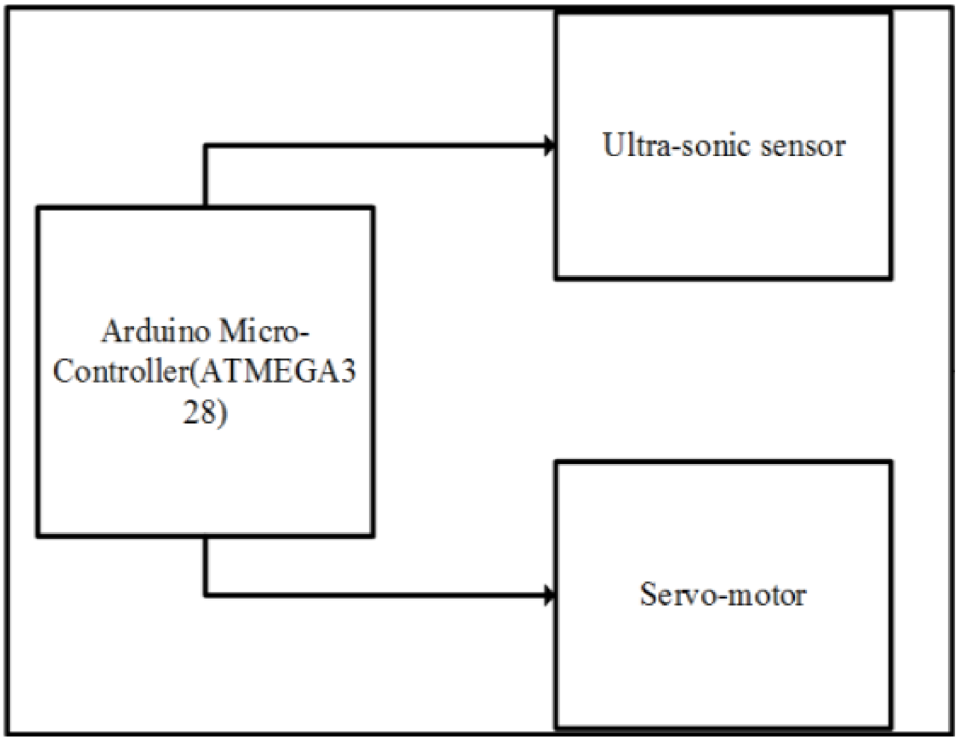


Fig 7. Hardware System Design of Radar System.

80%

The hardware system is made up of three parts: an Arduino, a servo motor, and an ultrasonic sensor. A servo motor is mounded with an ultrasonic sensor, which aids in movement and provides a turning mechanism. 58% Both the ultrasonic sensor and the servo motor are controlled and powered by Arduino. 58% As illustrated in Figure, Arduino powers both the ultrasonic sensor and the servo motor.

(B) System circuit design

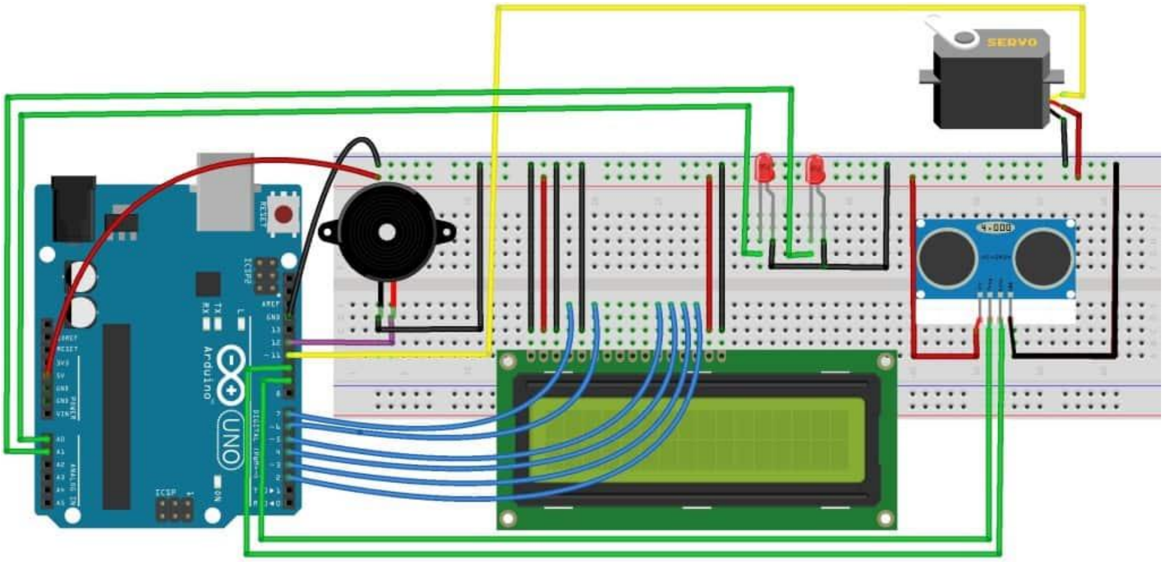


Fig 8. System circuit design

A hardware system design produced in the Fritzing environment is shown in Figure. It shows how several electrical components are linked together. The triggering pins of the ultrasonic sensor are connected to the Arduino's D10 pin, the servo motor's control linee to the Arduino's D8, and the echo pin to the Arduino's D11. The VCC pins of the servo motor and ultrasonic sensor are connected to the Arduino's 5V pin, while the Arduino's ground pin is connected to both the servo motor and ultrasonic sensor's ground pin.

(C) System circuit implementation on bread board –

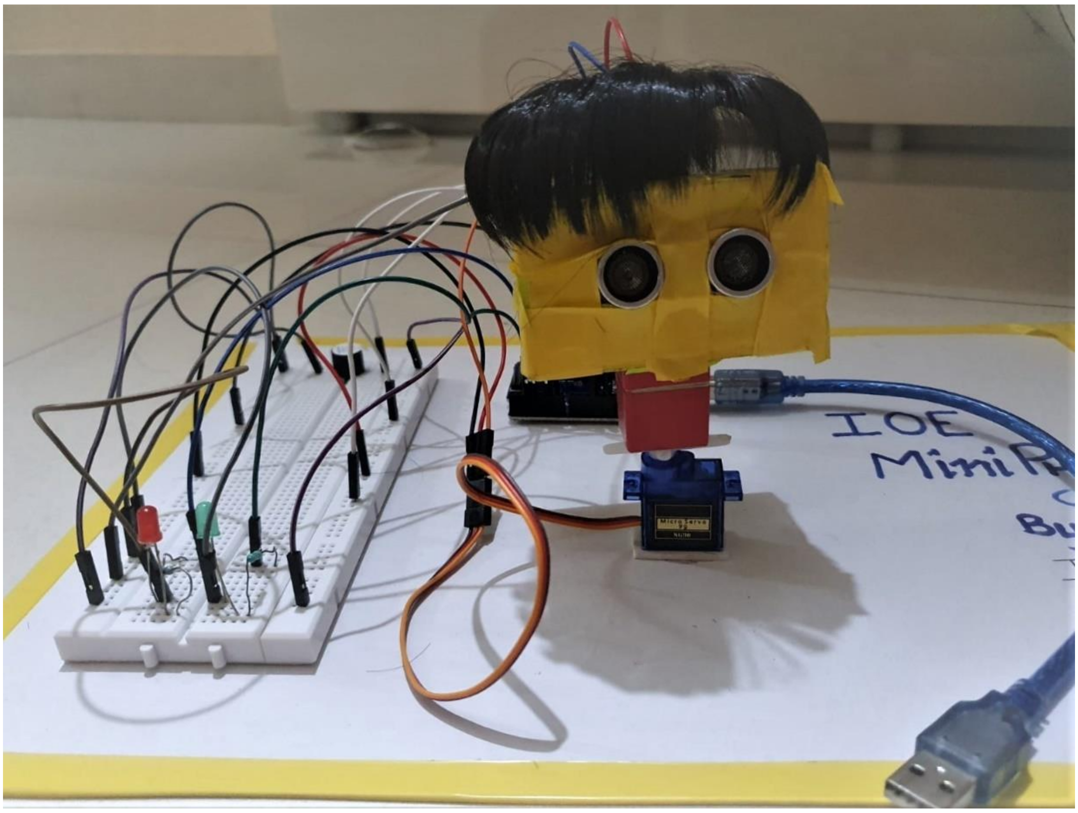


Fig 9. Implementation on bread board

The following diagram depicts the hardware system in its entirety. The ultrasonic servomotor may be seen mounted on a servo motor and positioned above the breadboard. On the other side of the breadboard, Arduino is installed, and the whole connection is formed between them. To keep the Arduino and servo motor from tripping over as the servo motor travels, they are stuck to the breadboard. To create code and upload it to Arduino, the Arduino IDE was utilised. The Arduino code reads the servo motor's location and calculates the dist to the closest item in the route.

The Arduino code is given below:

```
// Includes the Servo library
#include <Servo.h>
// Defines Tirgger and Echo pins of Ultrasonic Sensor
const int trigPin = 10;
const int echoPin = 11;
// Variables for the duration and the dist
```

```
long duration;
int dist;
int angle = 10;
int i = 10;
int buzzer = 13;
Servo myServo; // Creates a servo object for controlling the servo motor
void setup() {
 pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
 pinMode(echoPin, INPUT); // Sets the echoPin as an Input
 Serial.begin(9600);
 myServo.attach(8);
}
void loop() {

//  tone(buzzer,400);
//  delay(300);
//  noTone(buzzer);
//  delay(300);
//  tone(buzzer,400);
//  delay(300);
 for(i = 10; i < 180; i++)  {
   myServo.write(i);
//    delay(30);
   dist = calculateDist();
   if (dist >= 4 && dist <= 40 ){
      tone(buzzer,400);
     }
   else {
      noTone(buzzer);
     }
   Serial.print(i);  // Sends the current degree into the Serial Port
   Serial.print(",");
   Serial.print(dist);  // Sends the dist value into the Serial Port
   Serial.print(".");
  }

 for(i = 180; i > 10; i--)   {
   myServo.write(i);
```

```
    delay(30);
    dist = calculateDist();
    if (dist >= 4 && dist <= 40 ){
       tone(buzzer,400);
      }
    else {
       noTone(buzzer);
      }
    Serial.print(i);
    Serial.print(",");
    Serial.print(dist);
    Serial.print(".");
  }
}
int calculateDist(){

  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  dist= duration*0.034/2;
  return dist;
}
```

Processing code to display on laptop:

```
import processing.serial.*; // imports library for serial communication
import java.awt.event.KeyEvent;
import java.io.IOException;
Serial myPort; // defines Object Serial
// defubes variables
String angle="";
String dist="";
String data="";
String noObject;
float pixsDist;
int iAngle, iDist;
int index1=0;
```

```
int index2=0;
PFont orcFont;
void setup() {

 size (1200, 700); // ***CHANGE THIS TO YOUR SCREEN
RESOLUTION***
 smooth();
 myPort = new Serial(this,"COM5", 9600); // starts the serial communication
 myPort.bufferUntil('.'); // So actually it reads this: angle,dist.
}
void draw() {

  fill(98,245,31);
  fill(0,4);
  rect(0, 0, wid, high-high*0.065);

  fill(98,245,31);  // green color
  // calls the functions for drawing the radar
  drawRadar();
  drawLinee();
  drawObject();
  drawText();
}
void serialEvent (Serial myPort) {
  data = myPort.readStringUntil('.');
  data = data.substring(0,data.length()-1);

  index1 = data.indexOf(",");
  angle= data.substring(0, index1);
  dist= data.substring(index1+1, data.length());
  // converts the String variables into Integer
  iAngle = int(angle);
  iDist = int(dist);
}
void drawRadar() {
  pushMat();
  translate(wid/2,high-high*0.074);  // moves the starting coordinats to new
location
  noFill();
  strokeWeight(2);
  stroke(98,245,31);
  // draws the arc linees
  arc(0,0,(wid-wid*0.0625),(wid-wid*0.0625),PI,TWO_PI);
```

```
  arc(0,0,(wid-wid*0.27),(wid-wid*0.27),PI,TWO_PI);
  arc(0,0,(wid-wid*0.479),(wid-wid*0.479),PI,TWO_PI);
  arc(0,0,(wid-wid*0.687),(wid-wid*0.687),PI,TWO_PI);
  // draws the angle linees
  linee(-wid/2,0,wid/2,0);
  linee(0,0,(-wid/2)*cos(rad(30)),(-wid/2)*sin(rad(30)));
  linee(0,0,(-wid/2)*cos(rad(60)),(-wid/2)*sin(rad(60)));
  linee(0,0,(-wid/2)*cos(rad(90)),(-wid/2)*sin(rad(90)));
  linee(0,0,(-wid/2)*cos(rad(120)),(-wid/2)*sin(rad(120)));
  linee(0,0,(-wid/2)*cos(rad(150)),(-wid/2)*sin(rad(150)));
  linee((-wid/2)*cos(rad(30)),0,wid/2,0);
  popMat();
}
void drawObject() {
  pushMat();
  translate(wid/2,high-high*0.074);  // moves the starting coordinats to new
location
  strokeWeight(9);
  stroke(255,10,10); // red color
  pixsDist = iDist*((high-high*0.1666)*0.025);  // covers the dist from the sensor
from cm to pixels
  // limiting the range to 40 cms
  if(iDist<40){
    // draws the object according to the angle and the dist
  linee(pixsDist*cos(rad(iAngle)),-pixsDist*sin(rad(iAngle)),(wid-
wid*0.505)*cos(rad(iAngle)),-(wid-wid*0.505)*sin(rad(iAngle)));
  }
  popMat();
}
void drawLinee() {
  pushMat();
  strokeWeight(9);
  stroke(30,250,60);
  translate(wid/2,high-high*0.074);  // moves the starting coordinats to new
location
  linee(0,0,(high-high*0.12)*cos(rad(iAngle)),-(high-
high*0.12)*sin(rad(iAngle)));  // draws the linee according to the angle
  popMat();
}
void drawText() { // draws the texts on the screen

  pushMat();
  if(iDist>40)  {
```

```
noObject = "Out of Range";
}
else {
noObject = "In Range";
}
fill(0,0,0);
noStroke();
rect(0, high-high*0.0648,  wid, high);
fill(98,245,31);
textSize(25);

text("10cm",wid-wid*0.3854,high-high*0.0833);
text("20cm",wid-wid*0.281,high-high*0.0833);
text("30cm",wid-wid*0.177,high-high*0.0833);
text("40cm",wid-wid*0.0729,high-high*0.0833);
textSize(40);
text("Indian  Life ", wid-wid*0.875,  high-high*0.0277);
text("Angle:  " + iAngle  +"  °", wid-wid*0.48,  high-high*0.0277);
text("Dist:  ", wid-wid*0.26,  high-high*0.0277);
if(iDist<40)  {
textSize(25);
fill(98,245,60);
translate((wid-wid*0.4994)+wid/2*cos(rad(30)),(high-high*0.0907)-
wid/2*sin(rad(30)));
rotate(-rad(-60));
text("30 °",0,0);
resetMat();
translate((wid-wid*0.503)+wid/2*cos(rad(60)),(high-high*0.0888)-
wid/2*sin(rad(60)));
rotate(-rad(-30));
text("60 °",0,0);
resetMat();
translate((wid-wid*0.507)+wid/2*cos(rad(90)),(high-high*0.0833)-
wid/2*sin(rad(90)));
rotate(rad(0));
text("90 °",0,0);
resetMat();
translate(wid-wid*0.513+wid/2*cos(rad(120)),(high-high*0.07129)-
wid/2*sin(rad(120)));
rotate(rad(-30));
text("120 °",0,0);
resetMat();
```

```
 translate((wid-wid*0.5104)+wid/2*cos(rad(150)),(high-high*0.0574)-
wid/2*sin(rad(150)));
 rotate(rad(-60));
 text("150 °",0,0);
 popMat();
}
```

(D)Hardware system testing – The Arduino was connected to the development machine through a wire. Using the Arduino IDE, we were able to get a result in the serial monitor.

(E) Design and implementation of a graphical user interface system The GUI was created using the JAVA programming language and consists of two classes. The radar project's object class represents the objects it meets, such as dist, target/range, and angle/direction of position. The dist () method, angle () method, and location () method all accept needed values like dist and angle and display them on a GUI for simulation. Figures depict a linee sweep from one direction to the other, as well as a smudge in the GUI where ultrasonic sensors detect impediments.

WORKING :

Our design's main goal is to determine the dist, location, and speed of a barrier placed at a specific dist from the sensor. The ultrasonic sensor rotates with the assistance of servo motors to deliver the ultrasonic wave in various directions. This wave travels through the air and is reflected back after colliding with an item. This wave is detected by the sensor once again, and its characteristics are examined, with the results shown on the screen as metrics such as object dist and location. The Arduino IDE is used to write and transfer code in Arduino, allowing us to detect the position or angle of a servo motor, which is conveyed over the serial port together with the dist covered by the nearest item in its path. The outcome of all of this labour is displayed in the processing programme, which displays the object's input/output and range [4]. The sensors are implemented in such a manner that an ultrasonic sensor is coupled to the servo motor in order to detect the item and its dist. The ultrasonic sensor and servo motor will be controlled by an Arduino (microcontroller), and both will be powered by the microcontroller [3].

It displays a green light all of the time, indicating that object detection is active, and a buzzer and red light begin as soon as an item approaches it (0 degrees to 180 degrees) across a dist of 4cm to 40cm.

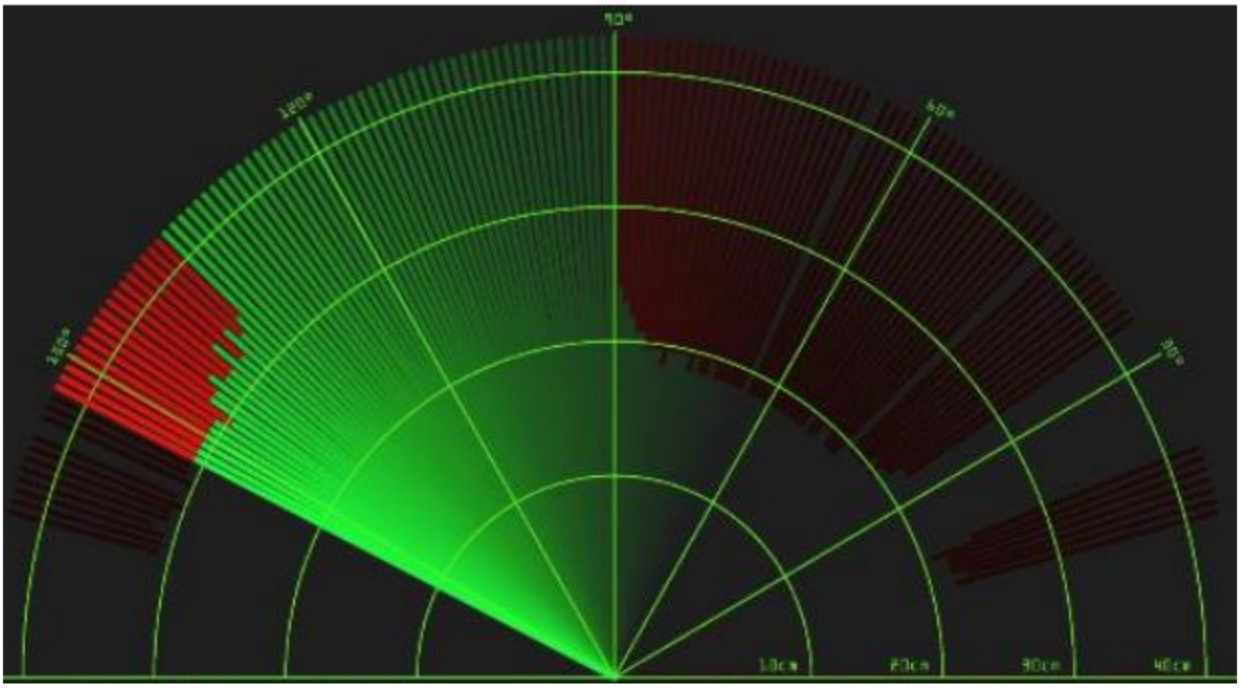
4. Result and Discussions :

59%

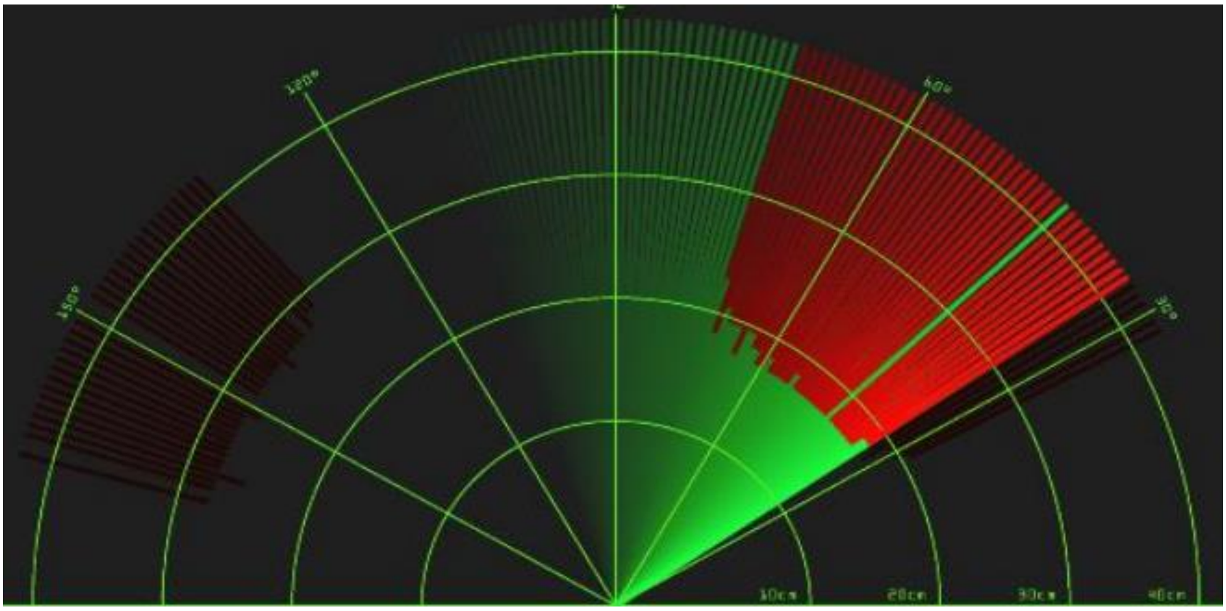
We said in this study report that our system is made up of the following components: a servo-motor, an ultrasonic sensor, and a microcontroller (Arduino). The goal of the system is to track the object's dist and angle and to visually display this information. This implies that the system's output should be in graphical form, which will be rendered using processing software. We may get an estimate of the radar's effectiveness by testing items at various highs and seeing how quickly or smoothly it identifies an object and gives us a predicted range of the obstacle [3]. The outcomes of our design's monitor screen when the sensor rotates across the region and finds an impediment are shown in the following image. The existence of an obstruction is shown by the red region, and the angle of incident and dist are presented below.

Testing of the system

a) Object A is kept at 20.5 far from the radar, radar gives the dist 32 cm, so: o error =(32-20.5)/20.5)*100= 3.918% o efficiency 1 = 100-error =94.08%



b) object 2 placed at a dist of 19.3 cm, radar gives the dist 21 cm so: o error = ((21-19.3)/20.3)*100 = 2.44% o efficiency 2 =100-error= 95.55%



After observation and calculation we conclude that this system is 94.815% efficient.

5. Conclusion and future scope of project:

In this work, an Arduino, servomotor, and ultrasonic sensor were used to create a system radar system that can detect the location and dist of an obstacle in its path and transform it into a graphically representable form. This method may be utilised in robotics for object identification and avoidance, as well as intrusion detection for various sizes of locations. The system's range is determined by the type of ultrasonic sensor employed. The sensor we utilised was the HC-SR04, which has a range of 2 to 40 cm. Designers now have more control over a variety of sophisticated applications thanks to a variety of improved control mechanisms. The proposed whole-system mapping approach is evaluated in our study on modest principles or scale [9]. The field of "Radar System," which we have chosen for our design, is quite broad, and the technology's future potential is enormous. Radar systems have been deployed or employed in a variety of applications [3]. Because of its security capabilities, this design has a lot of future potential. It has a wide range of uses[4]. Our investigation was restricted and constrained since we constructed a short-range radar. Because the servo motor we used can only rotate to this range, our system can only detect objects from 0 to 180 degrees. As a result of this limitation, our method cannot be utilised to identify larger-scale obstacles in locations or areas. The system's efficiency can be improved by using a 360-degree rotating servo motor. By altering this system and adding a fully 360-degree rotating servo and a higher-ranged ultrasonic sensor, we intend to better our research. We may enhance this system by making it transportable and adding an alert system that activates when an impediment is identified. Additional improvements might include an obstacle-avoiding robot with a surveillance system.