COLLEGE OF TECHNOLOGY

COIMBATORE-641004

EMBEDDED SYSTEMS LABORATORY-(15Z610)

ARDUINO RADAR SYSTEM

COMPUTER SCIENCE AND ENGINEERING

TEAM MEMBERS :

S.V.UDHAYAN (17Z356)

S.ASHOK KUMAR (18Z463)

R.S.KAMALAKKANNAN (18Z467)

S.PRAVIN PRABHU(18Z471)

INTRODUCTION:

Radar is a detection system that uses radio waves to determine the range, angle, or velocity of objects. It can be used to detect aircraft, ships, spacecraft, guided missiles, motor vehicles, weather formations, and terrain. A radar system consists of a transmitter producing electromagnetic waves in the radio or microwaves domain, a transmitting antenna, a receiving antenna (often the same antenna is used for transmitting and receiving) and a receiver and processor to determine properties of the object(s). Radio waves (pulsed or continuous) from the transmitter reflect off the object and return to the receiver, giving information about the object's location and speed.

This Arduino Radar Project is implemented with the help of Processing Application.

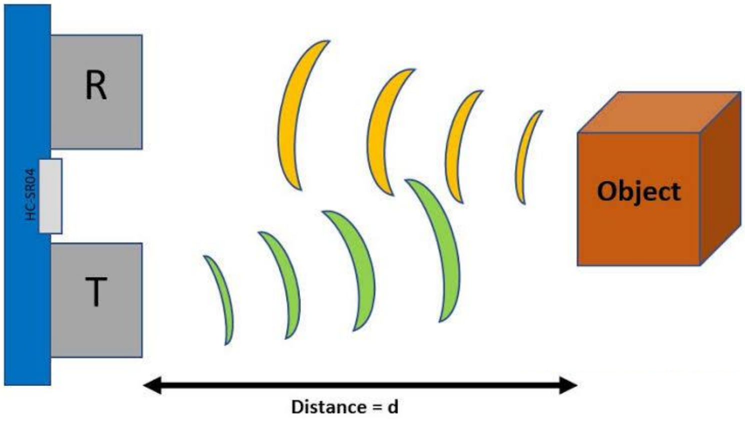
PROBLEM STATEMENT:

Radar is a long-range object detection system that uses radio waves to establish certain parameters of an object like its range, speed and position. Radar technology is used in aircrafts, missiles, marine, weather predictions and automobiles.

Ultrasonic Sensor to determine the presence of any object in a particular range.There is an Ultrasonic Sensor for detecting the objects and a Servo Motor for rotating the sensor and an Arduino Board for controlling them.

A Graphical representation of the data from the Ultrasonic Sensor is represented in a Radar type display. If the Ultrasonic Sensor detects any object within its range, the same will be displayed graphically on the screen.

WORKING OF ULTRASONIC SENSOR:



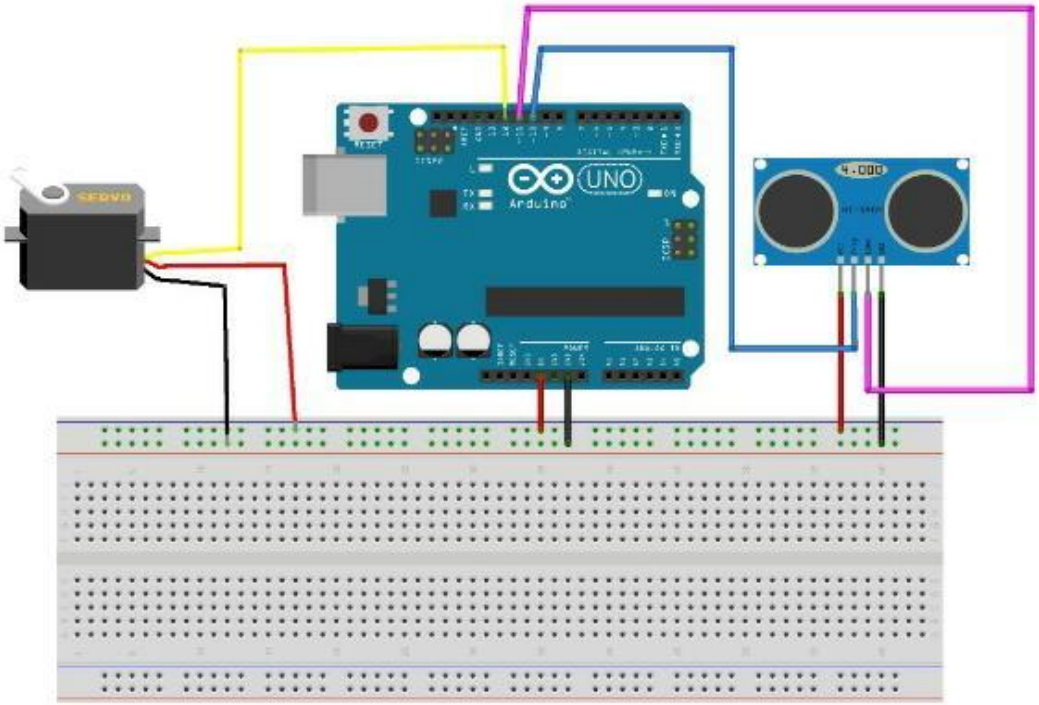
When an electrical pulse of high voltage in applied to the ultrasonic transducer it vibrates across a specific spectrum of frequencies and generate a burst of sound waves Whenever any obstacle comes ahead of the ultrasonic sensor the sound waves will reflect buck in the form of echo and generate an electric pulse. It calculates the time taken between sending sound waves and receiving echo. The echo pattern will be compared with the patterns of sound waves to determine detected signal conditioner.

The ultrasonic receiver shall detect signal from the utmovie transmitter while the transmit waves hit on the object. The combination of these two sensors will allow the robot to detect the object in its path. The ultrasonic Sensor is attached in fromt of the robot and tu sensor will also help the robot navigate through the hall of any boil

COMPONENTS REQUIRED:

- Arduino UNO
- Ultrasonic sensor(HC-SR04)
- Servo motor
- Jumpers
- Bread Board

SCHEMATIC DIAGRAM:



CODE:

ARDUINO CODE:

```
#include <Servo.h>
const int triggpin = 10;
const int echoopin = 11;
long durationn;
int dis;
Servo Servo1;
void setup() {
pinMode(triggpin, OUTPUT);
pinMode(echoopin, INPUT);
Serial.begin(9600);
Servo1.attach(12);
}
void loop() {
for(int i=15;i<=165;i++){
Servo1.write(i);
delay(30);
dis = calculatedis();
Serial.print(i);
Serial.print(",");
Serial.print(dis);
Serial.print(".");
}
for(int i=165;i>15;i--){
Servo1.write(i);
delay(30);
dis = calculatedis();
Serial.print(i);
Serial.print(",");
Serial.print(dis);
Serial.print(".");
}
}
int calculatedis(){
digitalWrite(triggpin, LOW);
delayMicroseconds(2);
digitalWrite(triggpin, HIGH);
delayMicroseconds(10);
digitalWrite(triggpin, LOW);
durationn = pulseIn(echoopin, HIGH);
dis= durationn*0.034/2;
return dis;
}
```

PROCESSING CODE:

```
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;
```

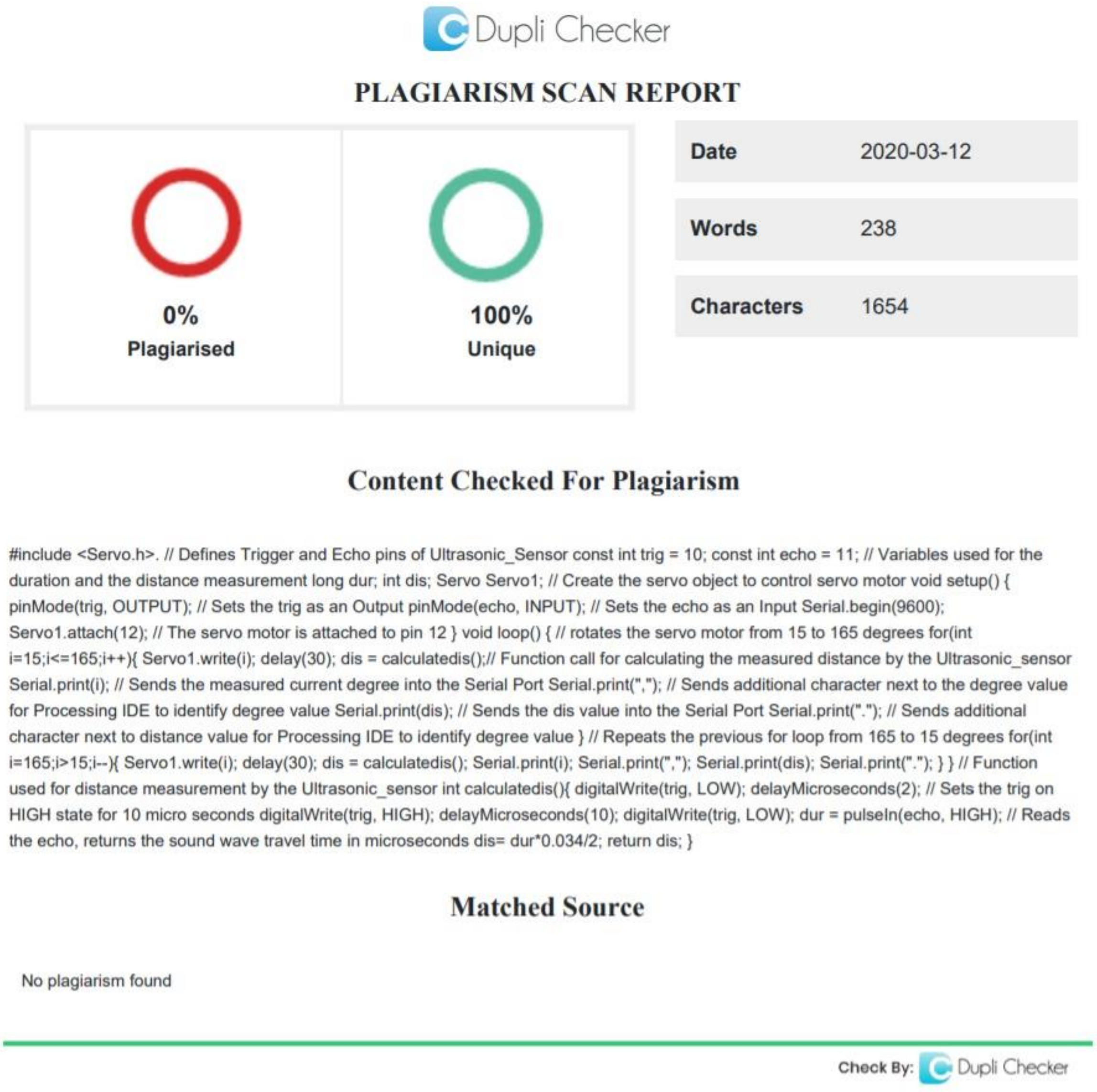
```
void setup() {
size (1200, 700);
smooth();
myPort = new Serial(this,"COM5", 9600);
myPort.bufferUntil('.');
}
void draw() {
fill(98,245,31);
noStroke();
fill(0,4);
rect(0, 0, width, height-height*0.065);
fill(98,245,31); // green color
drawRadar();
drawLine();
drawObject();
drawText();
}
void serialEvent (Serial myPort) {
data = myPort.readStringUntil('.');
data = data.substring(0,data.length()-1);
index1 = data.indexOf(",");
angle= data.substring(0, index1);
distance= data.substring(index1+1, data.length());
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);

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);

strokeWeight(9); iDistance

stroke(255,10,10); // red color

pixsDistance = *((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)));
}
void drawText() {
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(" VIRAL SCIENCE ", 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);
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();}
```

PLAGIARISM(ARDUINO CODE):



Dupli Checker

PLAGIARISM SCAN REPORT

0%
Plagiarised

100%
Unique

Date	2020-03-12
Words	238
Characters	1654

Content Checked For Plagiarism

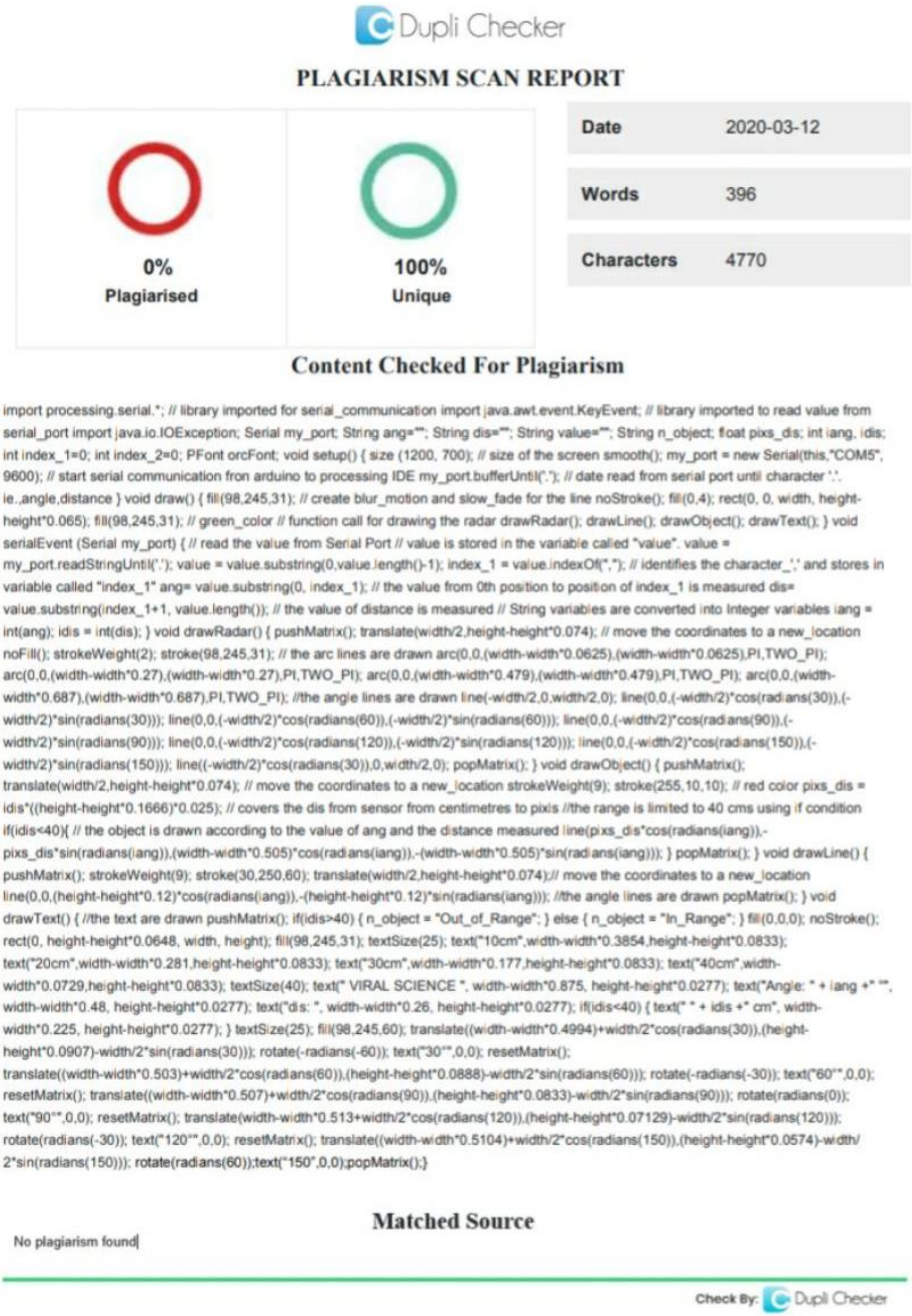
```
#include <Servo.h>. // Defines Trigger and Echo pins of Ultrasonic_Sensor const int trig = 10; const int echo = 11; // Variables used for the
duration and the distance measurement long dur; int dis; Servo Servo1; // Create the servo object to control servo motor void setup() {
pinMode(trig, OUTPUT); // Sets the trig as an Output pinMode(echo, INPUT); // Sets the echo as an Input Serial.begin(9600);
Servo1.attach(12); // The servo motor is attached to pin 12 } void loop() { // rotates the servo motor from 15 to 165 degrees for(int
i=15;i<=165;i++){ Servo1.write(i); delay(30); dis = calculatedis();// Function call for calculating the measured distance by the Ultrasonic_sensor
Serial.print(i); // Sends the measured current degree into the Serial Port Serial.print(","); // Sends additional character next to the degree value
for Processing IDE to identify degree value Serial.print(dis); // Sends the dis value into the Serial Port Serial.print("."); // Sends additional
character next to distance value for Processing IDE to identify degree value } // Repeats the previous for loop from 165 to 15 degrees for(int
i=165;i>15;i--){ Servo1.write(i); delay(30); dis = calculatedis(); Serial.print(i); Serial.print(","); Serial.print(dis); Serial.print("."); } } // Function
used for distance measurement by the Ultrasonic_sensor int calculatedis(){ digitalWrite(trig, LOW); delayMicroseconds(2); // Sets the trig on
HIGH state for 10 micro seconds digitalWrite(trig, HIGH); delayMicroseconds(10); digitalWrite(trig, LOW); dur = pulseIn(echo, HIGH); // Reads
the echo, returns the sound wave travel time in microseconds dis= dur*0.034/2; return dis; }
```

Matched Source

No plagiarism found

Check By: Dupli Checker

PLAGIARISM (PROCESSING CODE):



Dupli Checker

PLAGIARISM SCAN REPORT

Date	2020-03-12
Words	396
Characters	4770

0%
Plagiarised

100%
Unique

Content Checked For Plagiarism

```
import processing.serial.*; // library imported for serial_communication import java.awt.event.KeyEvent; // library imported to read value from
serial_port import java.io.IOException; Serial my_port; String ang=""; String dis=""; String value=""; String n_object; float pixs_dis; int iang, idis;
int index_1=0; int index_2=0; PFont orcFont; void setup() { size (1200, 700); // size of the screen smooth(); my_port = new Serial(this,"COM5",
9600); // start serial communication fron arduino to processing IDE my_port.bufferUntil('.'); // date read from serial port until character '.'.
ie.,angle,distance } void draw() { fill(98,245,31); // create blur_motion and slow_fade for the line noStroke(); fill(0,4); rect(0, 0, width, height-
height*0.065); fill(98,245,31); // green_color // function call for drawing the radar drawRadar(); drawLine(); drawObject(); drawText(); } void
serialEvent (Serial my_port) { // read the value from Serial Port // value is stored in the variable called "value". value =
my_port.readStringUntil('.'); value = value.substring(0,value.length()-1); index_1 = value.indexOf(","); // identifies the character_',' and stores in
variable called "index_1" ang= value.substring(0, index_1); // the value from 0th position to position of index_1 is measured dis=
value.substring(index_1+1, value.length()); // the value of distance is measured // String variables are converted into Integer variables iang =
int(ang); idis = int(dis); } void drawRadar() { pushMatrix(); translate(width/2,height-height*0.074); // move the coordinates to a new_location
noFill(); strokeWeight(2); stroke(98,245,31); // the arc lines are drawn 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); //the angle lines are drawn 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); // move the coordinates to a new_location strokeWeight(9); stroke(255,10,10); // red color pixs_dis =
idis*((height-height*0.1666)*0.025); // covers the dis from sensor from centimetres to pixls //the range is limited to 40 cms using if condition
if(idis<40){ // the object is drawn according to the value of ang and the distance measured line(pixs_dis*cos(radians(iang)),-
pixs_dis*sin(radians(iang)),(width-width*0.505)*cos(radians(iang)),-(width-width*0.505)*sin(radians(iang))); } popMatrix(); } void drawLine() {
pushMatrix(); strokeWeight(9); stroke(30,250,60); translate(width/2,height-height*0.074);// move the coordinates to a new_location
line(0,0,(height-height*0.12)*cos(radians(iang)),-(height-height*0.12)*sin(radians(iang))); //the angle lines are drawn popMatrix(); } void
drawText() { //the text are drawn pushMatrix(); if(idis>40) { n_object = "Out_of_Range"; } else { n_object = "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(" VIRAL SCIENCE ", width-width*0.875, height-height*0.0277); text("Angle: " + iang +" "",
width-width*0.48, height-height*0.0277); text("dis: ", width-width*0.26, height-height*0.0277); if(idis<40) { text(" " + idis +" cm", width-
width*0.225, height-height*0.0277); } textSize(25); 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();}
```

Matched Source

No plagiarism found

Check By: Dupli Checker

CHALLENGES FACED:

1. Understanding the Working of Ultrasonic Sensor
2. Connecting Arduino to processing IDE for serial communication
3. Display the presence of objects graphically using processing IDE

CONTRIBUTION OF TEAM MEMBERS:

Udhayan -Processing IDE code implementation

Ashok Kumar -Arduino IDE code implementation

Kamalakkannan -Arduino IDE code implementation

Pravin prabhu- model design

REFERENCES:

[1]https://www.allaboutcircuits.com/projects/servo-motor-control-with-an-arduino/

[2]https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/

[3]https://maker.pro/arduino/tutorial/how-to-make-arduino-and-processing-ide-communicate

[4]https://howtomechatronics.com/tutorials/arduino/processing/

GITHUB LINK:

https://github.com/ashoksanjai/Embedded-System-Laboratory