

Embedded System Programming

(4CS016)

Ultrasonic radar

Report on Mini Project

University Id : 2332917

Module leader : Jeffrey Ting

Lecture : Subash Bista

Student Name : Niraj Chaudhary

Word Count : 2696

Group : L4CG16

Submission date : 21thMay,2023

# Abstract

In the report mini project, I have an implementation of an Ultrasonic Radar using an Arduino Uno, a microcontroller, an ultrasonic sensor, a micro servo, a breadboard, and jumper cables. The project attempts to move the servo motor to face the detected object and display the distance of an object from the sensor on an LCD screen.

# Acknowledgment-

Gratitude and appreciation are lovely expressions of a positive attitude toward learning and growth. We thanked the module leader and the embedded systems module staff for giving the student the opportunity to work on such interesting projects. We also recognized that this coursework provided an excellent platform for learning and gaining knowledge on a variety of embedded system issues. We were able to put theory into reality and gain hands-on experience dealing with various components such as the Arduino Uno, ultrasonic sensor, micro servo, breadboard, and jumper cables through this project. This experience has most likely allowed to gain skills and information that will be useful for future work in the field of embedded systems.

Table of Contents

[1. Introduction 1](#_Toc135507034)

[2. Materials or Methods used in the project: 1](#_Toc135507035)

[2.1 Arduino UNO- 1](#_Toc135507036)

[2.2 Ultrasonic Distance Sensor - HC-SR04**-** 3](#_Toc135507037)

[2.3. Micro servo 9g – 4](#_Toc135507038)

[2.4. Breadboard- 4](#_Toc135507039)

[2.5. Jumper cables- 5](#_Toc135507040)

[3. Creating an Ultrasonic Radar: 6](#_Toc135507041)

[4. TinkerCard 7](#_Toc135507042)

[5. Fritzing 7](#_Toc135507043)

[6. Working Principle 8](#_Toc135507044)

[7. Code 10](#_Toc135507045)

[7.1 Arduino Source Code 10](#_Toc135507046)

[7.2 Processing Code 11](#_Toc135507047)

[8. Testing 14](#_Toc135507048)

[9. Conclusion: 18](#_Toc135507049)

[10. References 18](#_Toc135507050)

# 1. Introduction

ULTRASONIC RADAR is a detection method that measures the attributes of objects using radio waves. It is widely utilized in a variety of applications, including air traffic control, surveillance systems, and missile guidance. The United States Navy invented the acronym "RADAR" in 1940 to stand for Radio Detection and Ranging. Radar technology has now become widely used, influencing the development of self-driving cars and self-parking systems. The project we designed can be tailored to other applications such as vehicles, bicycles, or any other system. The Arduino integration allows for greater freedom in tailoring the module to specific needs (Nevonprojects, 2012).

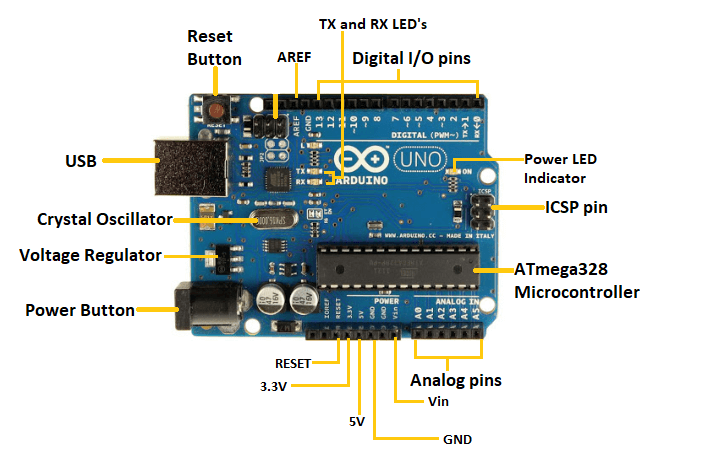
# 2. Materials or Methods used in the project:

## 2.1 Arduino UNO-

The Arduino UNO is a popular Arduino board. The title "UNO" means "one" in Italian, and it was so named since it was the first Arduino software release. The Arduino UNO board was similarly the first to include a USB interface. It is a versatile and durable boarding that is utilized in many different projects. Arduino.cc shaped the Arduino UNO board.

The ATmega328P microcontroller powers the Arduino UNO. It is well-known for being user-friendly, especially when associated to comparable boarding such as the Arduino Mega. The board has numerical and analog input/output pins, as well as protections and additional hardware. Six analog input pins, fourteen digital pins, a USB connector, a control card, and an ICSP header for in-circuit serial program writing are all comprised in the Arduino UNO. For programming, it employs an Combined Growth Situation that is well-matched with together online and disconnected stages (javatpoint, 2011).

The following are the components of the Arduino UNO board:



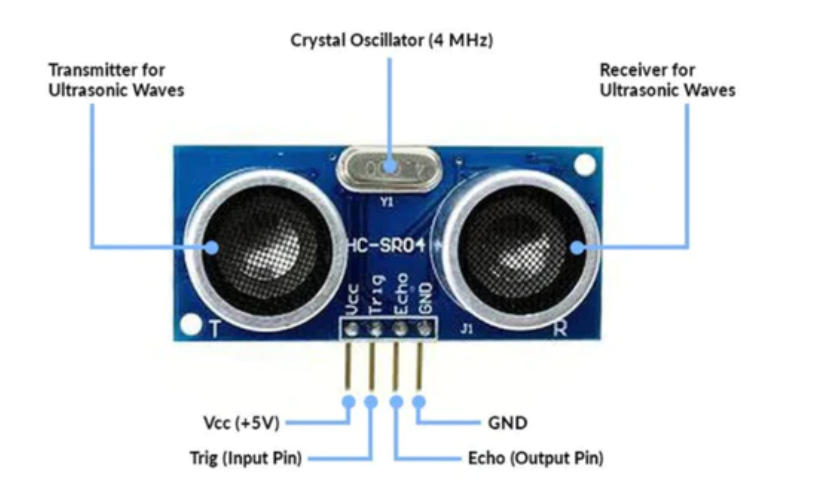
Let's analyze each element in details.

* ATmega328 Microcontroller - It is a microcontroller on a single chip with an 8-bit processor. Analog to Digital Converter, serial ports, I/O lines, timers, interrupts, and an oscillator are all included.
* ICSP pin - It is used to program the Arduino board's firmware via In-Circuit Serial Programming.
* Power LED indicator - It indicates whether or not the electricity is turned on. Once the control is turned on, the LED illuminates.
* Digital I/O pins - These pins can be configured to be HIGH or LOW. They are classified D0 through D13.
* TX – These LEDs show the data movement during communication.
* AREF – It used to source a orientation current to the Arduino board from an exterior source.
* Reset button – It enables the Arduino board to be reset.
* USB – It agrees connection between the board and the processer, which is required for Arduino program writing.
* Crystal Oscillator – A 16MHz crystal oscillator for the Arduino UNO provides a reliable clock frequency.
* Voltage Regulator – It brings the input voltage down to a consistent 5 volts.
* GND – The ground pin acts as the voltage reference.
* Vin – It is a reference to the input voltage.
* Analog Pins - These A0 to A5 pins are used to read analog sensors and can also be utilized as to connect in input/output (GPIO) pins.

## 2.2 Ultrasonic Distance Sensor - HC-SR04­**-**

The HC-SR04 ultrasonic distance sensor is a low-cost module commonly used in a change of projects to detect distances with sound waves. It is made up of an ultrasonic source, a telephone, and a switch journey all moved into one module. The HC-SR04 sensor has a strictness of up to 3mm and can reliably detect distances ranging from 2cm to 400cm, making it perfect for applications needing accurate distance measurements.

The HC-SR04 sensor contains four different kinds of pins. The VCC (electricity) pin supplies electricity to the module. The distance measurement process is started by the Trig (Trigger) pin. When this pin receives a strong signal, the module releases a 40kHz sound wave. The reflected sound waves are received by the Echo (Receive) pin. The Echo Pin gets a signal when the transmitted sound wave collides with an item and bounces back. The length of the signal received on the Echo pin is proportional to the object's distance from the sensor. The GND (Ground) pin links the module to the ground of the power source (robocraze, 2022).



## 

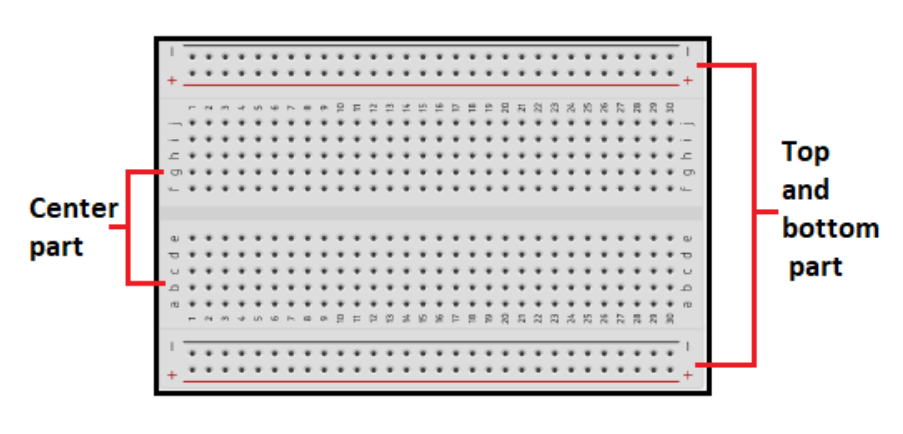
## 2.3. Micro servo 9g –

The 9g micro servo is a small electromechanical device used in robotics, remote control vehicles, and other small-scale tasks. A servo is made up of a direct current motor, a reduction gear system, and a control circuit. It can rotate up to 180 degrees. We can scan the surroundings in multiple directions and detect items at varying angles by rotating the servo. A tiny potentiometer on the servo also offers feedback to the external device regarding its present position.



## 2.4. Breadboard-

The breadboard is a versatile and commonly used equipment in electronics prototyping that enables quick and easy circuit assembly without the use of soldering. It is made out of a plastic board with a grid of small holes for inserting and connecting electronic components. A center gap divides the board into two halves, with two columns of power rails running along the sides of each half. Electricity rails are used to supply electricity to the circuit and are often labeled "+" and "-" for positive and negative voltages. The breadboard's main area is a grid of holes used to insert and connect electronic components. The holes are often placed in rows that are internally connected. This means that if you insert a component into one of the holes in a row, it will be connected to all of the other holes in that row. To make it easier to detect and connect components, the rows of holes are frequently labeled with numbers and letters. For example, the first row of holes could be designated "A", the second row "B", and so on. Each row's holes are normally numbered from left to right. Jumper wires are used to connect components on a breadboard. They come in a variety of lengths and colors to assist keep the circuit tidy and simple to grasp.

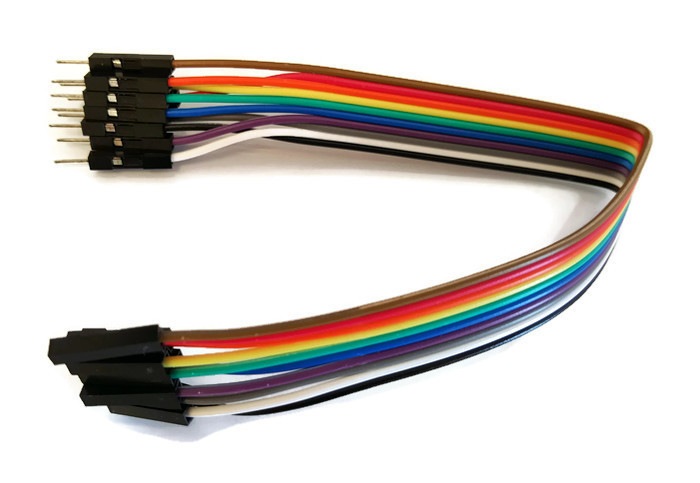


## 2.5. Jumper cables-

Jumper cables, sometimes known as jumper wires or DuPont wires, are necessary components in many electronic projects. They are used to link different electronic components on a breadboard or to connect an Arduino or other microcontroller to different sensors or actuators. Jumper cables are made up of a flexible wire with connectors on both ends, which are usually male or female pins. The pins can be placed into breadboard holes or the pins of an Arduino or other microcontroller.

Jumper cables are available in a variety of lengths and colors, which aids in keeping wiring orderly and easy to follow. The following are the most popular types of jumper cables:

1. Male-to-male- These cables are used to link two female connectors and have pins on both ends.
2. Male-to-Female- These cables are used to link a male connection to female connector and have a pin on one end and a socket on the other.
3. Female-to-female- These cables are used to link two male connectors and have sockets on both ends.

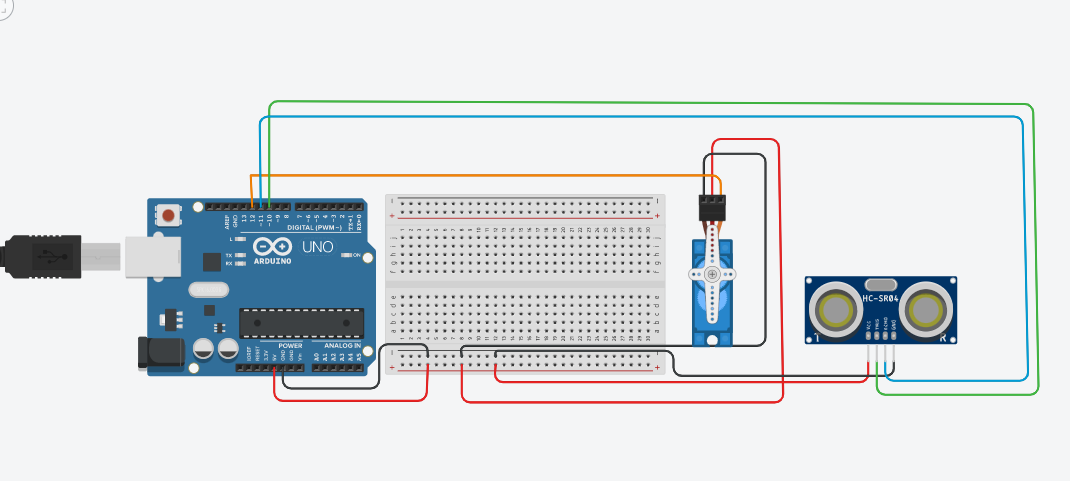


# 3. Creating an Ultrasonic Radar:

To construct the Ultrasonic Radar, we will use a breadboard and jumper cables to link the ultrasonic sensor and micro servo 9g to the Arduino Uno. Here are some steps to take:

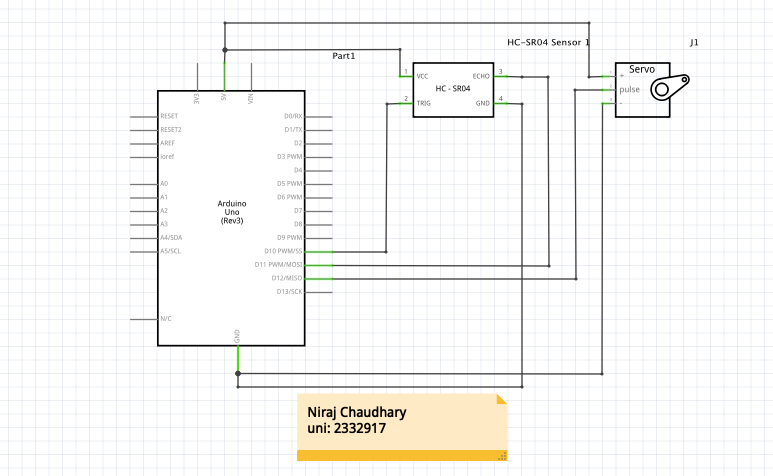
1. VCC pin of ultrasonic senor is joined to Arduino Uno of 5v in positive pin.
2. GND pin of ultrasonic sensor is joined to Arduino Uno of negative pin of GND.
3. Trin pin of ultrasonic senor is joined to Arduino Uno pin 10.
4. Echo pin of ultrasonic sensor is joined to Arduino Uno pin 11
5. Micro servo 9g of red color of jumper wire is joined to 5V.
6. Micro servo 9g of black color of jumper wire is joined to the GND.
7. Micro servo 9g of yellow color of jumper wire is joined to pin 12.

# 4. TinkerCard



We connected the components in Tinkercard. Create a new circuit with Tinkercad. Search for and include the previous materials in the circuit Arduino Uno, Breadboard, Ultrasonic sensor (HC-SR04), and Micro Servo 9g. Tinkercad has a strong simulation capability that allows us to virtually test our circuits. We can run our code and examine how the components interact without causing physical damage. This saves time and money since we can find and solve problems in our circuit before putting it into action.

# 5. Fritzing

****

Fritzing is an additional standard tool used for designing and documenting electronic circuits. It offers a graphical user interface that makes it simple to drag and drop components onto an electronic breadboard, making it simpler to grasp how the circuit is put together. Additionally, Fritzing enables us to design bespoke components, see our circuit from many perspectives (breadboard, schematic, PCB), and produce circuit documentation that appears professional. It's an excellent resource for teaching and learning circuit design effectively for both learners and instructors.

# 6. Working Principle

The working principle of the Ultrasonic Radar project includes the resulting steps:

1. Distance Measurement:

* High-frequency sound waves (ultrasonic waves) are transmitted by the ultrasonic sensor in the direction of the objective item.
* These sound waves impact the object afterward traveling through the atmosphere.
* The sensor then receives the ultrasonic waves after they have returned from the item.
* The distance is calculated using the instrument's measurement of the period it incomes for waves to reach and leave the item.

1. Scanning :

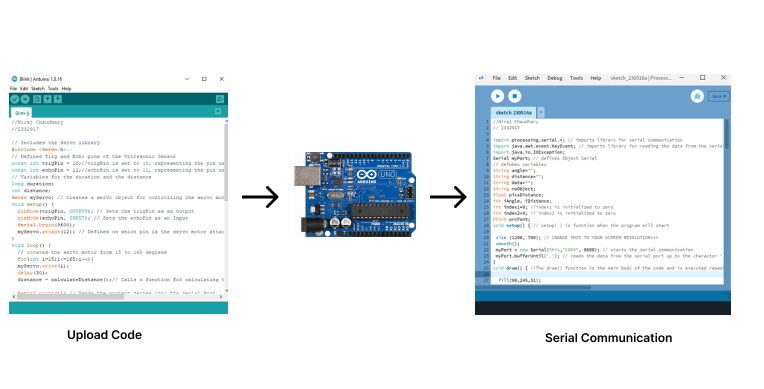
* The ultrasonic sensor is interchanged in a scanning motion by the micro servo.
* The ultrasonic sensor can scan the environment in various directions by regulating the servo's angle.
* This scanning enables a larger coverage area while helping in item recognition from various angles.

1. Data Processing

* The ultrasonic sensor directs the distance readings to the Arduino Uno microcontroller.
* It evaluates this information to calculate the position and separation of the identified items.
* Dependent on the location or distance of the detected object, the Arduino can be programmed to take various actions.

1. Component Integration

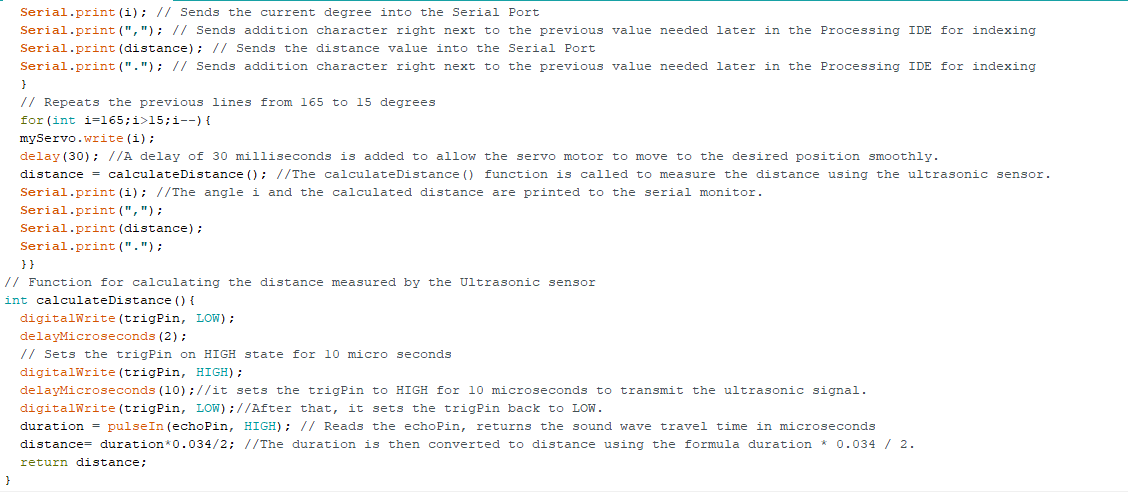
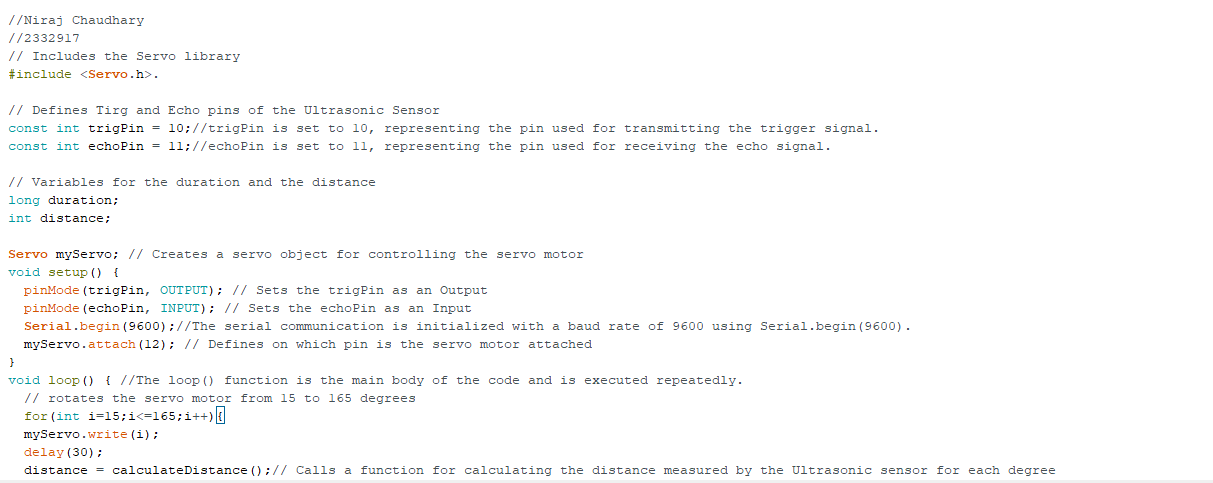
* As the project's primary control, the Arduino Uno is involved to the breadboard.
* Jumper cables are recycled to connection the Arduino Uno and the ultrasonic sensor.
* Moreover, using jumper cables, an immaterial servo is associated to the Arduino Uno.
* The breadboard deals an easy-to-use platform for circuit prototyping by helping connections between the several components.



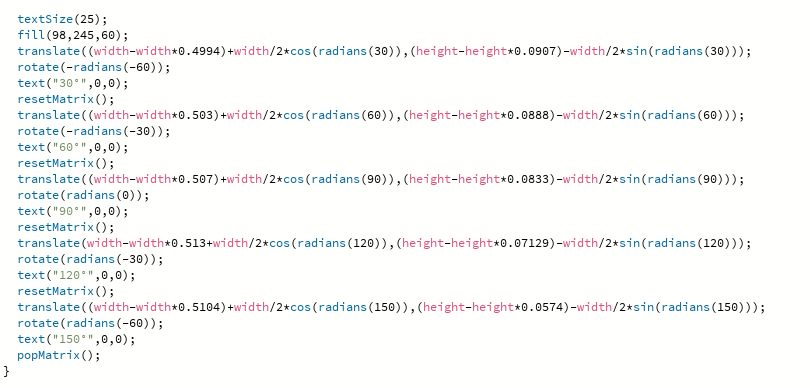
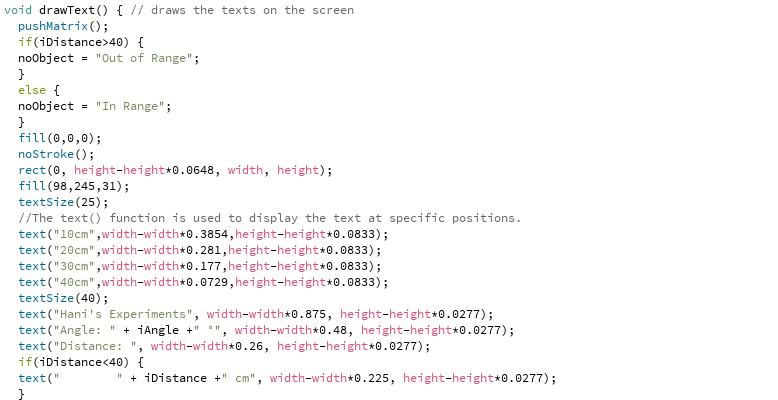
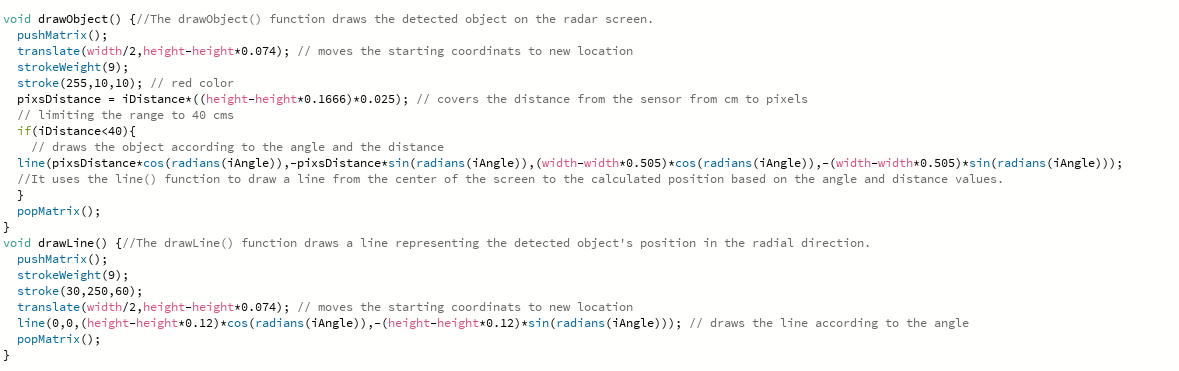
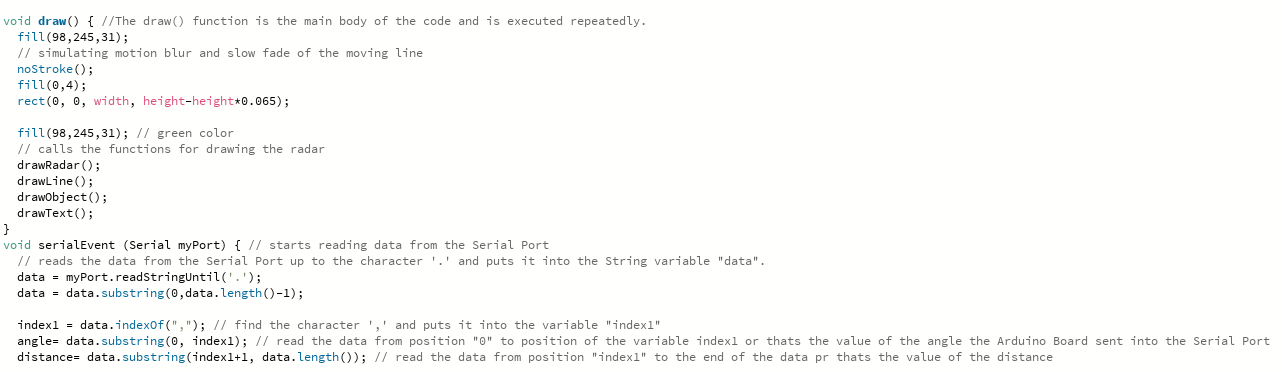
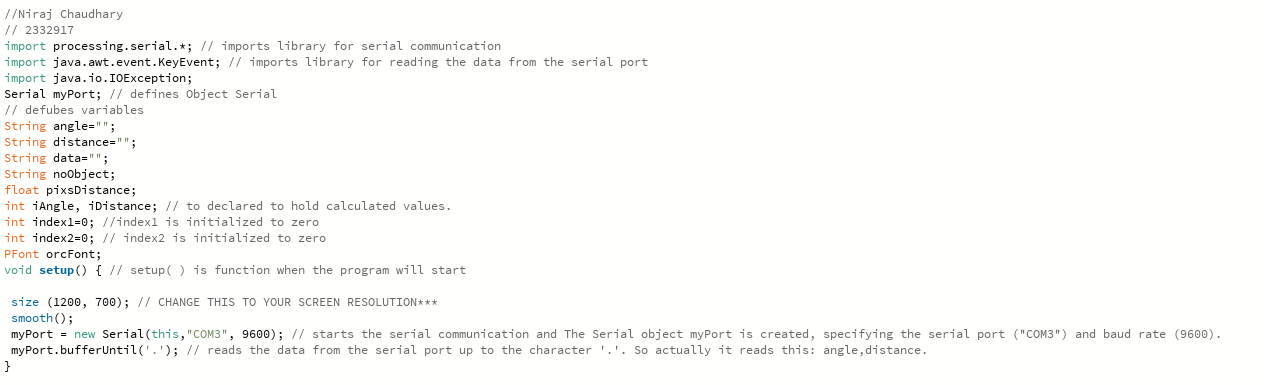
# 

# 7. Code

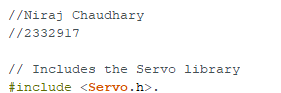
## 7.1 Arduino Source Code



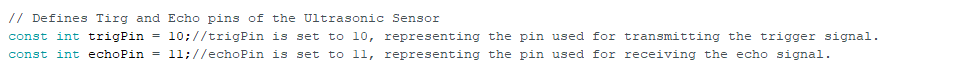
## 7.2 Processing Code



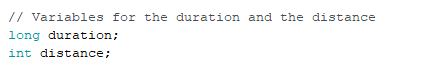
The Arduino Source Code, which includes a description of each line of code:



* The Servo library, which is required to control the servo motor, is involved in this line.



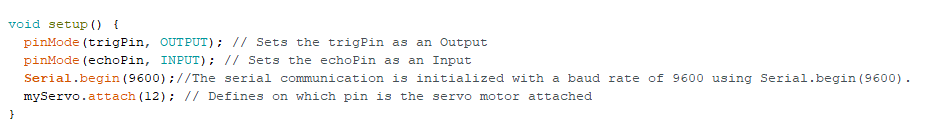
* These lines define the pin numbers for the trigPin (trigger pin connected to the ultrasonic sensor) and echoPin ( echo pin connected to the ultrasonic sensor).



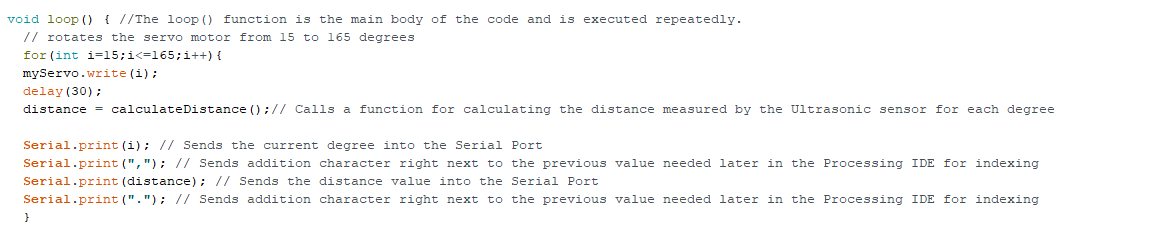
* These variables will be used to hold the ultrasonic pulse length and determined distance.



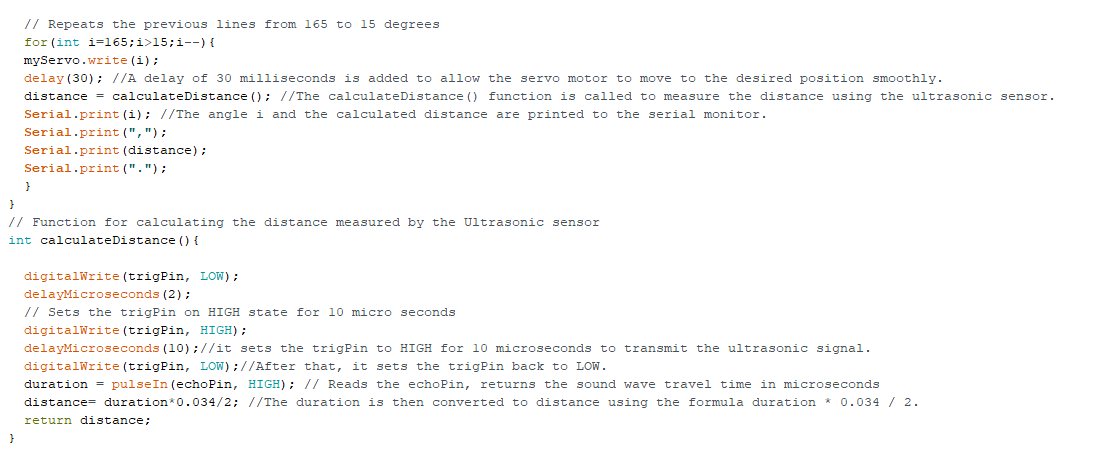
* A Servo class instance is produced, which will control the servo motor.



* The trigPin is set as an output and the echoPin is set as an input in the setup() method. For debugging purposes, serial message is happening at a degree of 9600. Using the myServo.attach() function, the servo motor is joined to pin 12.

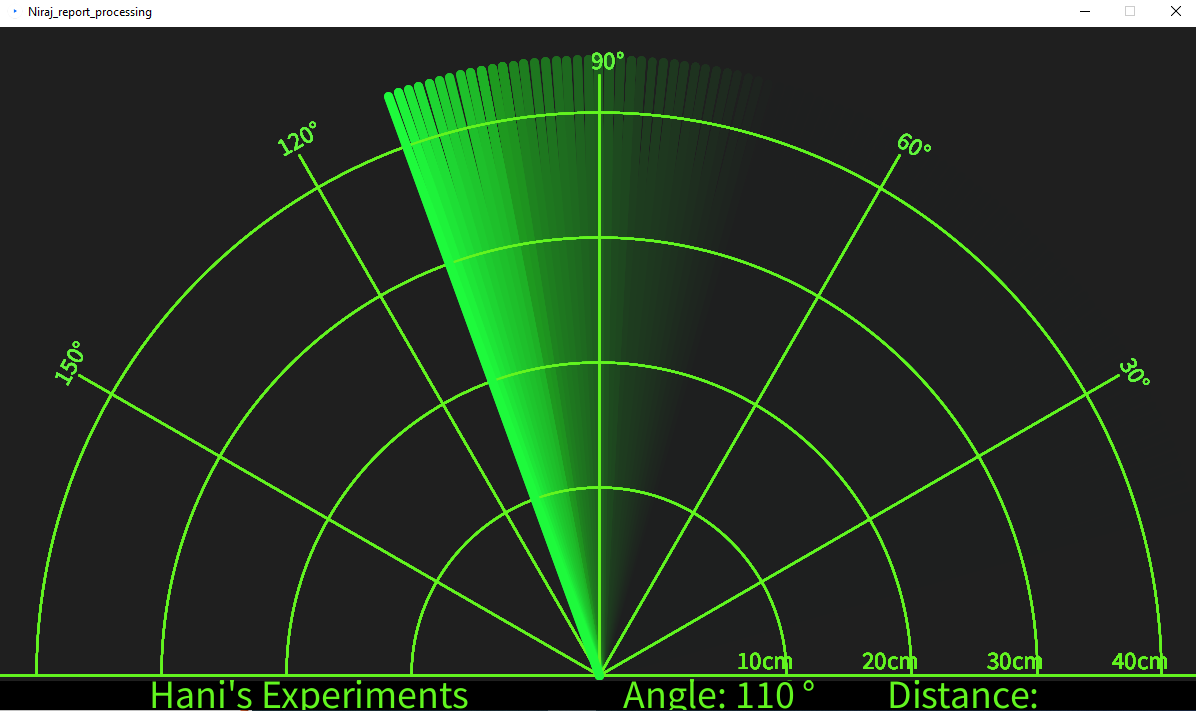


* There are two for loops in the loop() function. The first loop improvements the servo motor in 1 degree increments from 15 to 165 degrees. MyServo is used to set the servo position within the loop.After appealing write(), a 30ms delay is introduced before measuring the distance with the calculateDistance() function. For debugging resolutions, the location and distance beliefs are reported to the serial monitor.
* The second loop, which practices the similar logic as the first, switches the servo motor from 165 to 15 degrees in 1 degree increments.



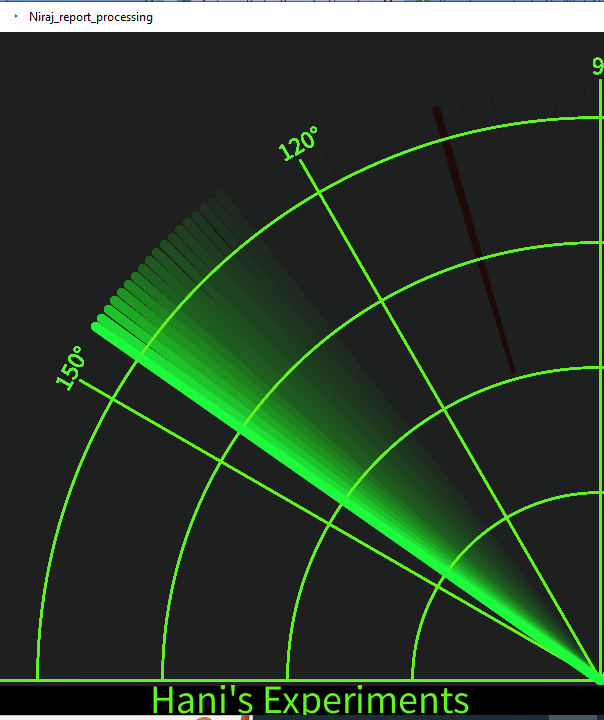
* The calculateDistance() function is in charge of measuring the distance using the ultrasonic sensor. It begins by situation the trigPin to LOW and adding a 2 microsecond delay. The ultrasonic sensor is then triggered by situation the trigPin to HIGH for 10 microseconds and quickly returning to LOW. The pulseIn() function is used to determine the length of the pound received on the echoPin. The reserve is calculated by increasing the duration by the rapidity of sound and in-between by two. The calculated distance is then returned.
* Overall, this code moves the servo motor back and forth while detecting distance with the ultrasonic sensor. The position and distance values are reported to the serial monitor for examination and restoring.

# 8. Testing

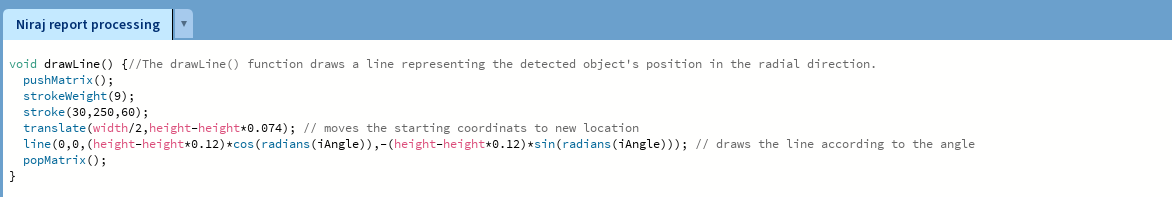


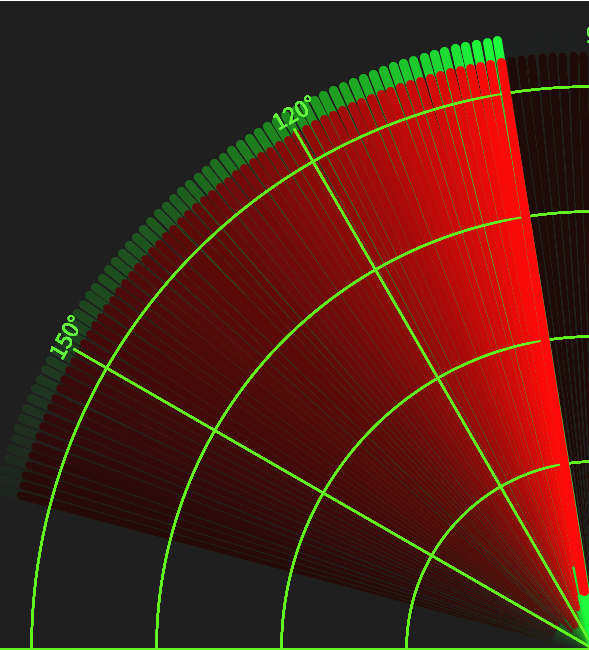
* Now, we'll read the angle besides distance values unrushed by the sensor from the Arduino Board interested in the Processing IDE using the SerialEvent() method, which gets data from the Serial Port, and save them in the variables iAngle and iDistance. The radar, lines, identified matters, and some typescript will be drawn using these variables.
* I twisted the drawRadar() function, which contains of the arc() and line() functions, to draw the radar.



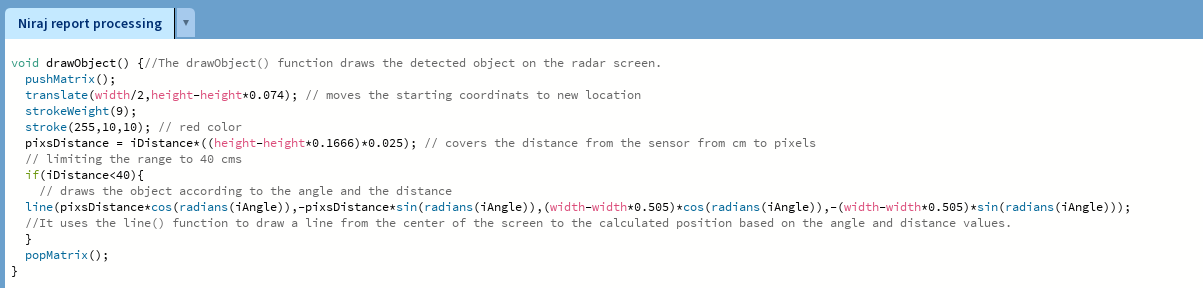
****

* To attraction the moving line on the radar, I inscribed the drawLine() function. The translate() function is used to resolve its center of rotation, and the line is redrawn for individually degree by means of the line() function and the iAngle variable.

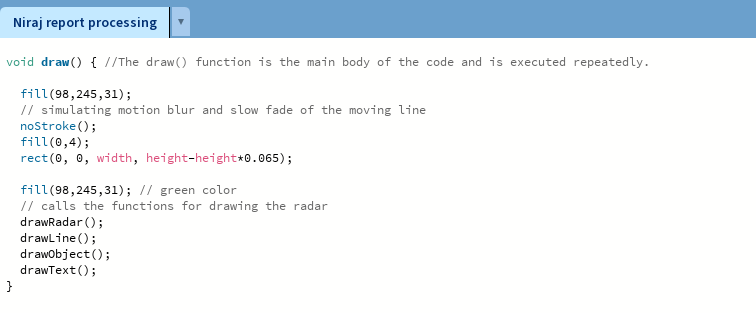
****

****

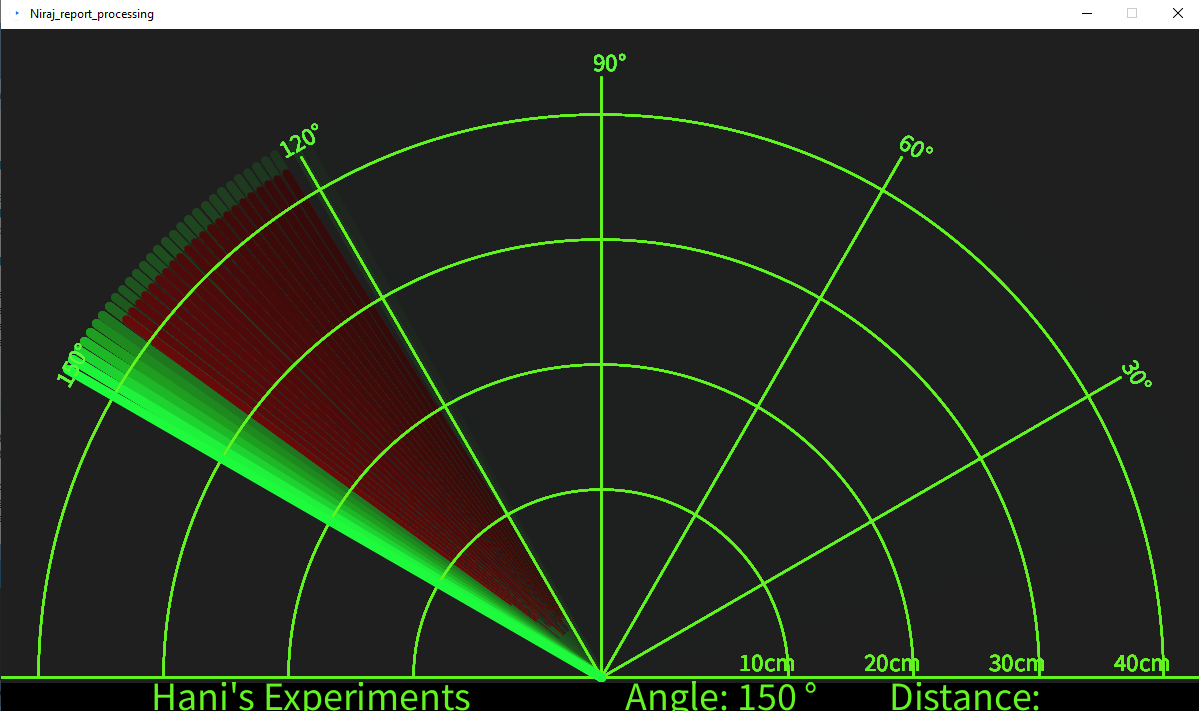
* To draw the discovered items, this drawObject() purpose was created. It retrieves the reserve from the ultrasonic sensor, transforms it to pixels, and then attractions the item on the radar in conjunction with the sensor's angle.

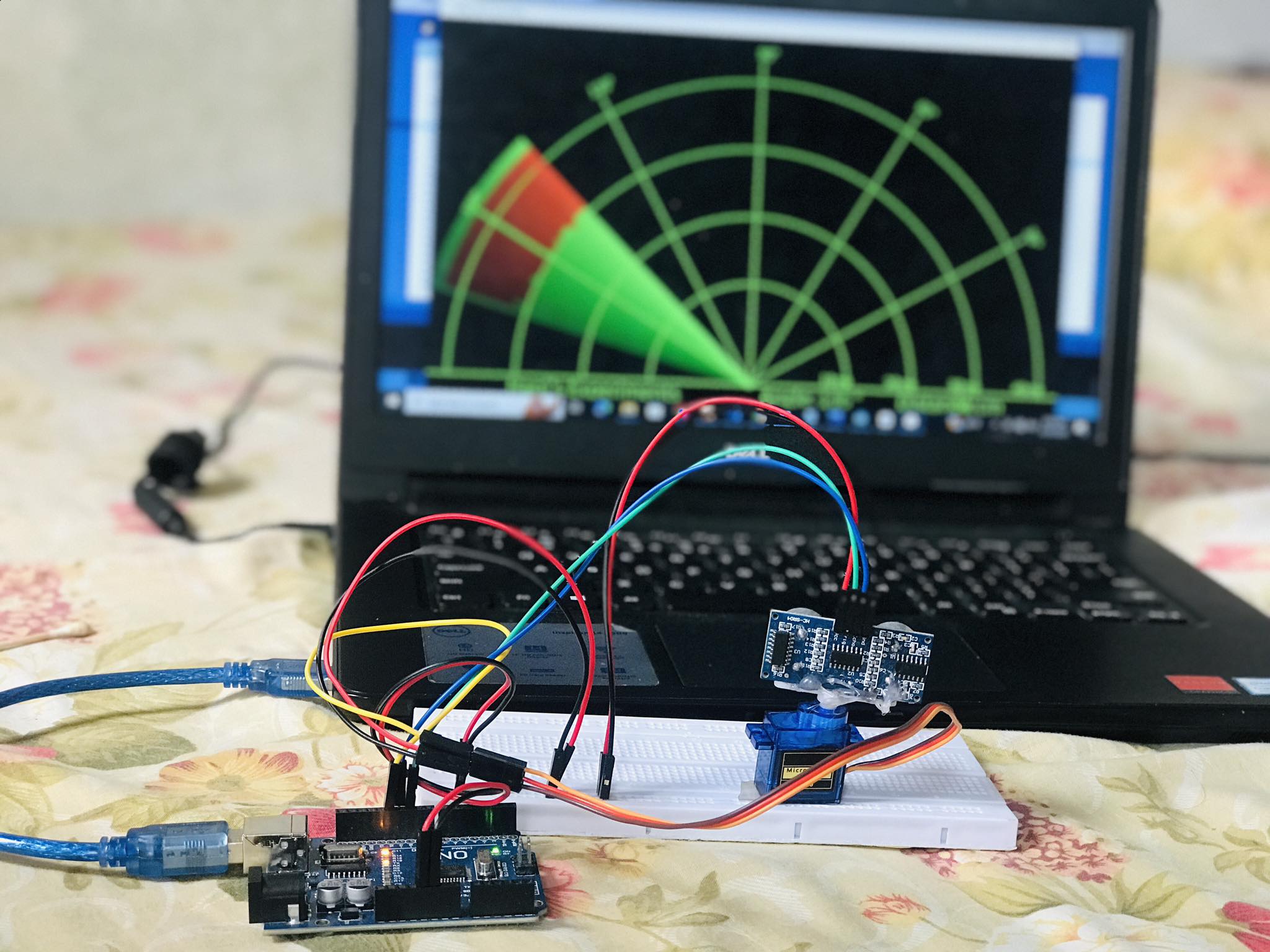


* I generated the drawText() function to draw text on specific areas on the screen.
* The function of main draw() method runs boundlessly and counsellors the shelter, promotions all of these functions. This fill() function is also used with two arguments to imitate motion blur and a consistent fading of a moving line.

****

The radar's final appearance is as follows:

****

****

# 9. Conclusion:

The ultrasonic radar research shows how ultrasonic sensor knowledge can be used to detect objects. A fully working radar system is developed by participating in the Arduino Uno, ultrasonic sensor, micro servo, and other workings. This project will help you better recognize microcontroller programming, sensor integration, and data visualization. It also gives students hands-on practice with prototyping and electronics meetings with the help of a breadboard and jumper cables.

The ultrasonic radar project can be improved and modified additional by including structures such as object tracking, more sensors for greater attention, and incorporation with other systems. This project serves as an outline for extra research into robots, security systems, and automated monitoring.

Overall, the Arduino Uno ultrasonic radar project provides an exciting and informative experience that promotes creativity, problem-solving abilities, and a greater hold of embedded systems and sensor skills.

# 10. References

javatpoint.(2011).Retrievedfromhttps://www.javatpoint.com/arduino-uno?fbclid=IwAR14aciqcTS0ELJHjaYzyJH2HCP\_DaDfhKYbRaYpHVscXz33oh-Rq98fJI

Nevonprojects.(2012).Retrievedfromhttps://nevonprojects.com/ultrasonic-radar-project/

robocraze.(2022,10september).Retrievedfromhttps://robocraze.com/blogs/post/what-isultrasonicsensor?fbclid=IwAR2IiyvmTdF0z2M45\_F1J1\_fafvhXviRPthCZZh8UQejVJ0v4ZGO2Zwy09I