**Ultrasonic Sensor Based Tracking System**

Project Submitted in Partial Fulfillment of the Requirements for the Degree of Bachelor of Technology in the field of Computer Science and Engineering

##### BY

Sayan Ghosh(123180703078)

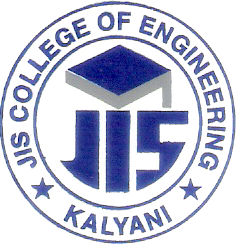
Sourav Sahoo (123180703089)

Sankha Sarkar (380117011033)

Under the supervision

of

Amit Majumder

**

##### Department of Computer Science and Engineering

JIS College of Engineering

Block-A, Phase-III, Kalyani, Nadia, Pin-741235

West Bengal, India

May, 2020



**CERTIFICATE**

This is to certify that **Sayan Ghosh(123180703078), Sourav Sahoo(123180703089), Sankha Sarkar(380117011033)** has completed his/her project entitled **Ultrasonic Sensor based tracking System,** under the guidance of  **Amit Majumder** in partial fulfillment of the requirements for the award of the **Bachelor of Technology in Computer Science and Engineering** from JIS college of Engineering (An Autonomous Institute)is an authentic record of their own work carried out during the academic year 2020-21 and to the best of our knowledge, this work has not been submitted elsewhere as part of the process of obtaining a degree, diploma, fellowship or any other similar title.

**--------------------------------- ------------------------------- ------------------------------**

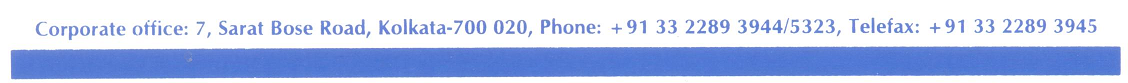
**Signature of theSupervisor Signature of the HOD Signature of the Principal**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Signature of the External Expert**

**Place:**

**Date:**



ACKNOWLEDGEMENT

The analysis of the project work wishes to express our gratitude to Guide Name for allowing the degree attitude and providing effective guidance in development of this project work. His conscription of the topic and all the helpful hints, he provided, contributed greatly to successful development of this work, without being pedagogic and overbearing influence.

We also express our sincere gratitude to Dr. Dharmpal Singh, Head of the Department of Computer Science and Engineering of JIS College of Engineering and all the respected faculty members of Department of CSE for giving the scope of successfully carrying out the project work.

Finally, we take this opportunity to thankProf.**(Dr.) Partha Sarkar**, Principal of JIS College of Engineering for giving us the scope of carrying out the project work.

Date:

**…………………………………….………………………………….**

Sayan Ghosh

B.TECH in Computer Science and Engineering

………6th….YEAR/…3rd…….. SEMESTER

Univ Roll--123180703078

**…………………………………….………………………………….**

Sourav Sahoo

B.TECH in Computer Science and Engineering

……6th……YEAR/……3rd….SEMESTER.

Univ Roll--123180703089

**…………………………………….………………………………….**

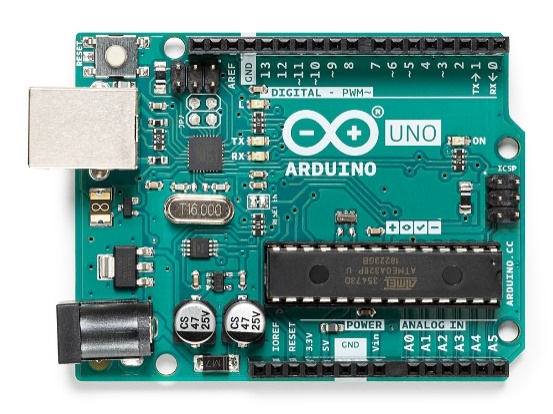
Sankha Sarkar

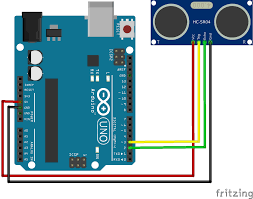
B.TECH in Computer Science and Engineering

…6th……….YEAR/……3rd….. SEMESTER

Univ Roll--380117011033

**List of Figures**

****

****

**List of Tables**

|  |  |  |
| --- | --- | --- |
| **No.** | **Pin Name** | **Pin Description** |
| 1 | VCC | The power supply pin of the sensor that mainly operates at 5V DC. |
| 2 | Trig Pin | It plays a vital role to initialize measurement for sending ultrasonic waves. It should be kept high for 10us for triggering the measurement. |
| 3 | Echo Pin | This pin remains high for short period based on the time taken by the ultrasonic waves to bounce back to the receiving end. |
| 4 | Ground | This pin is connected to ground. |

**CONTENTS**

**Title page 1-2**

**Certificate 3**

**Acknowledgement 4**

**List of Figures & Tables 5**

**Abstract 11**

**1. Introduction**

**2. Literature Survey 13**

**3. Methodology 4**

**4. Proposed Method 19**

**5. Result and Discussion 20-25**

**6. Conclusion 26**

**Reference**

ABSTRACT

Ultrasonic motion sensors consist of transmitters that emit ultrasonic sound pulses into the monitored area, which would return immediately when it encounters obstacles on the way and a receiver acknowledges the return of the signal. Transmitted sound uses air as medium. Distance measurement is done with respect to the time delay between transmission and reception. This system is based on many such detectors that are arranged in an efficient manner, the outputs of these various sensors are used to determine and decide the movement of the object. The aim of this paper is to realize the design of a sensor array which can detect motion in any environment or medium and also be economically viable and portable. The system is aimed at short range but accurate detection. The system can also be used to trigger external circuitry that can perform certain tasks without human intervention which makes it ideal for use in remote security systems. The sound used are pulses of 50 or 40khz frequency and hence cannot be detected by the human ear, making the system undetectable. The system is useful in environments with moving objects, without which the system continues to scan for moving objects indefinitely till said object is found. This system will be achieved with very little computational power which makes this ideal for many situations. The system after design and construction was found to work as per the conceptual idea.

INTRODUCTION

RADAR system is an object detection or tracking system which uses radio waves to decide or get the range, height, heading, or speed of items or objects. Radar frameworks or system arrive in an assortment of sizes and have distinctive performance particulars. Some radars are utilized for aviation authority at air terminals and others are utilized for long range observation and early cautioning frameworks . There are some ways to show radar working data. There are also some modified radar systems which have advance technology of handling the systems. These modified system are used at higher levels to get or extract the helpful or important data . Our proposed system’s working principle is linked by the following components which are is ultra-sonic sensor connected to the microcontroller (we have chosen Arduino) digital input and output pins. Then we have servo motor which is also connected to digital output and input pins. Our both main components ultra-sonic sensor and servo motor are connected simultaneously, so that when our servo motor rotates from 0 degree to 180 degree from extreme right to extreme left the motor will rotate nearby its axis [. We utilize Computer screen to demonstrate the data (distance and angle) through software called “Processing development Environment”

LITERATURE SURVEY

ARDUINO:

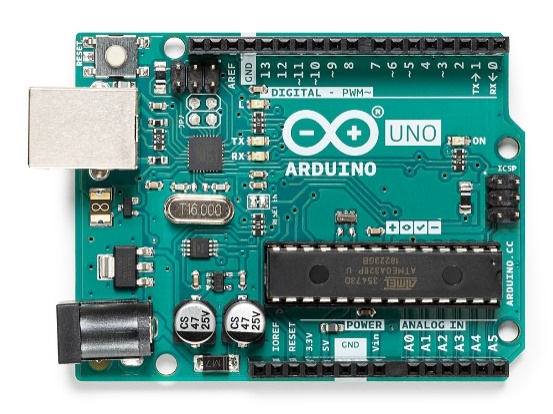
The Arduino Uno is a microcontroller board based on the ATmega328. It has 20 digital input/output pins (of which 6 can be used as PWM outputs and 6 can be used as analog inputs), a 16 MHz resonator, a USB connection, a power jack, an in-circuit system programming (ICSP) header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a [AC-to-DC adapter](https://www.pololu.com/product/1463) or battery to get started.

The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features an ATmega16U2 programmed as a USB-to-serial converter. This auxiliary microcontroller has its own USB bootloader, which allows advanced users to reprogram it.

The Arduino has a large [support community](http://arduino.cc/forum/) and an extensive set of support libraries and hardware add-on “[shields](https://www.pololu.com/category/113/shields-for-arduino)” (e.g. you can easily make your Arduino wireless with our [Wixel shield](https://www.pololu.com/product/2513)), making it a great introductory platform for embedded electronics. Note that we also offer a [**SparkFun Inventor’s Kit**](https://www.pololu.com/product/2776), which includes an Arduino Uno along with an assortment of components (e.g. breadboard, sensors, jumper wires, and LEDs) that make it possible to create a number of fun introductory projects.

This is the 3rd revision of the Uno (R3), which has a number of changes:

* The USB controller chip changed from ATmega8U2 (8K flash) to ATmega16U2 (16K flash). This does not increase the flash or RAM available to sketches.
* Three new pins were added, all of which are duplicates of previous pins. The I2C pins (A4, A5) have been also been brought out on the side of the board near AREF. There is a IOREF pin next to the reset pin, which is a duplicate of the 5V pin.
* The reset button is now next to the USB connector, making it more accessible when a shield is used.



# Micro Servo Motor SG90

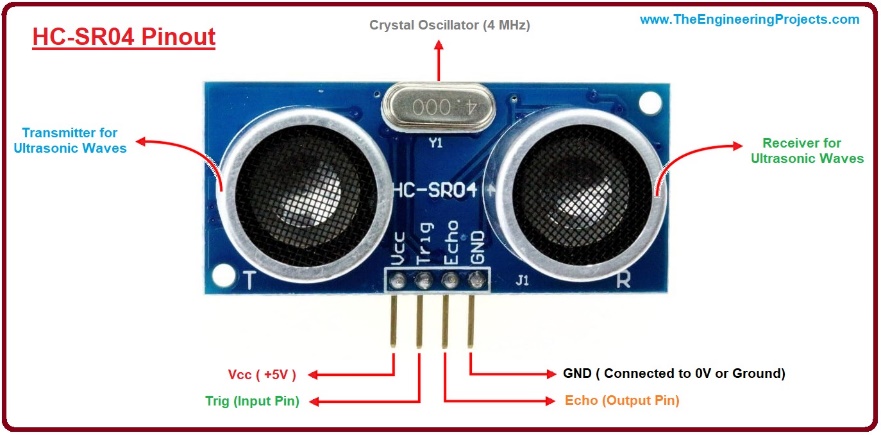
Micro Servo Motor SG90 is a tiny and lightweight server motor with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.

**(ULTRASONIC )HC-SRO4**-As the name indicates, ultrasonic / level sensors measure distance by using ultrasonic waves.  
The sensor head emits an ultrasonic wave and receives the wave reflected back from the target. ultrasonic / level sensors measure the distance to the target by measuring the time between the emission and reception.An optical sensor has a transmitter and receiver, whereas an ultrasonic / level sensor uses a single ultrasonic element for both emission and reception. In a reflective model ultrasonic / level sensor, a single oscillator emits and receives ultrasonic waves alternately. This enables miniaturisation of the sensor head.

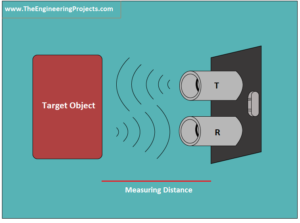
* **HC-SR04**is an ultrasonic sensor mainly used to determine the distance of the target object.
* It measures accurate distance using a non-contact technology - A technology that involves no physical contact between sensor and object.
* Transmitter and receiver are two main parts of the sensor where former converts an electrical signal to ultrasonic waves while later converts that ultrasonic signals back to electrical signals.
* HC-SR04 contain 4 pins in total.

|  |  |  |
| --- | --- | --- |
| **No.** | **Pin Name** | **Pin Description** |
| 1 | VCC | The power supply pin of the sensor that mainly operates at 5V DC. |
| 2 | Trig Pin | It plays a vital role to initialize measurement for sending ultrasonic waves. It should be kept high for 10us for triggering the measurement. |
| 3 | Echo Pin | This pin remains high for short period based on the time taken by the ultrasonic waves to bounce back to the receiving end. |
| 4 | Ground | This pin is connected to ground. |

* I have labelled these HC-SR04 Pinout in below figure for better visualization:

[](https://www.theengineeringprojects.com/wp-content/uploads/2018/10/Introduction-to-HC-SR04.jpg)

### How does it work?

The **HC-SR04 Ultrasonic (US) sensor** is an ultrasonic transducer that comes with 4 pin interface named as Vcc, Trigger, Echo, and Ground. It is very useful for accurate distance measurement of the target object and mainly works on the sound waves.As we connect the module to 5V and initialize the input pin, it starts transmitting the sound waves which then travel through the air and hit the required object. These waves hit and bounce back from the object and then collected by the receiver of the module.[](https://www.theengineeringprojects.com/wp-content/uploads/2018/10/introduction-to-ultrasonic-sensor-hc-sr04-2.png)

Distance is directly proportional to the time these waves require to come back at the receiving end. The more the time taken, more the distance will be.The waves will be generating if the Trig pin is kept High for 10 µs. These waves will travel at the speed of sound, creating 8 cycle sonic burst that will be collected in the Echo pin.The echo pin remains turned on for the time these waves take to travel and bounce back to the receiving end. This sensor is mainly incorporated with Arduino to measure the required distance.Following formula is used to calculate the distance of the object.

**S = (V x t)/2**

Where S is the required distance, V is the speed of sound and t is the time sound waves take to come back after hitting the object. We need to divide the value by 2 because time will be double as the waves travel and bounce back from the initial point. Dividing it by 2 will give the actual distance of the target object.

### Using HC-SR04 with Arduino Module

In order to get the precise distance measurement, HC-SR04 is mostly used in combination with different Arduino Modules like [Arduino Uno](https://www.theengineeringprojects.com/2018/06/introduction-to-arduino-uno.html) and [Arduino Mega](https://www.theengineeringprojects.com/2018/06/introduction-to-arduino-mega-2560.html). You can connect Arduino with this sensor in the following way.

* First, you need to power up the sensor using 5V DC regulated input to the sensor. Connect the ground pin with the ground of the voltage source. You can also power the sensor module using the Arduino 5V pins as the current drawn by the sensor is less than 15mA, won't be affecting the current ratings of the Arduino Module.

After setting up the initial arrangement, connect both Trig and Echo pins to the I/O pins of the Arduino Board. As mentioned earlier, in order to initialize the measurement process, the Trig pin must be kept high for 10us in the start. The sensor module will start generating sound waves with the frequency around 40,000 Hz per second from the transmitter.

* As the waves bounce back, consequently, the Echo pin will turn on until the sounds waves are received by the receiver. This time will be calculated using Arduino Module.

This tutorial that will help you [How to Interface Ultrasonic Sensor with Arduino Module](https://www.theengineeringprojects.com/2017/08/ultrasonic-sensor-arduino-interfacing.html)You can also [Interface Multiple Ultrasonic Sensors with Arduino Module](https://www.theengineeringprojects.com/2015/02/interfacing-multiple-ultrasonic-sensor-arduino.html)

### Applications

HC-SR04 comes with a wide range of applications mainly targeting distance and direction measurements. Following are the major applications it can be used for.

* Speed and direction measurement
* Wireless charging
* Humidifiers
* Medical ultrasonography
* Burglar alarms
* Embedded system
* Depth measurement
* Non-destructive testing
* Jumper wires are simply wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering. Jumper wires are typically used with [breadboards](https://blog.sparkfuneducation.com/what-is-a-breadboard) and other prototyping tools in order to make it easy to change a circuit as needed. Fairly simple. In fact, it doesn’t get much more basic than jumper wires.
* 
* Though jumper wires come in a variety of colors, the colors don’t actually mean anything. This means that a red jumper wire is technically the same as a black one. But the colors can be used to your advantage in order to differentiate between types of connections, such as ground or power.

CODE(processing IDE)

importprocessing.serial.\*; // imports library for serial communication

importjava.awt.event.KeyEvent; // imports library for reading the data from the serial port

importjava.io.IOException;

Serial myPort; // defines Object Serial

// defubes variables

String angle="";

String distance="";

String data="";

StringnoObject;

floatpixsDistance;

intiAngle, iDistance;

int index1=0;

int index2=0;

PFontorcFont;

void **setup**() {

size (1200, 700); // \*\*\*CHANGE THIS TO YOUR SCREEN RESOLUTION\*\*\*

smooth();

myPort = new Serial(this,"COM3", 9600); // starts the serial communication

myPort.bufferUntil(&apos;D&apos;); // reads the data from the serial port up to the character

&apos;D&apos;. 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();

}

voidserialEvent (Serial myPort) { // starts reading data from the Serial Port

// reads the data from the Serial Port up to the character &apos;D&apos; and puts it into the String variable

"data". data = myPort.readStringUntil(&apos;D&apos;);

data = data.substring(0,data.length()-1);

index1 = data.indexOf("A"); // find the character &apos;A&apos; 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 Port

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

data prthats the value of the distance

// converts the String variables into Integer

iAngle = int(angle);

iDistance = int(distance);

}

voiddrawRadar() {

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();

}

voiddrawObject() {

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

line(pixsDistance\*cos(radians(iAngle)),-pixsDistance\*sin(radians(iAngle)),(width-width\*0.505)\*cos(radia

ns(iAngle)),-(width-width\*0.505)\*sin(radians(iAngle)));

}

popMatrix();

}

voiddrawLine() {

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();

}

voiddrawText() { // 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(" Sourav&apos;s SONAR", 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(15

0)));

rotate(radians(-60));

text("150°",0,0);

popMatrix();

}

SONAR(Arduino)

#include <Servo.h>.

constinttrigPin = 8;

constintechoPin = 12;

// defining time and distance

long duration;

int distance;

Servo myServo; // Object servo

void setup() {

pinMode(trigPin, OUTPUT); // trigPin as an

Output

pinMode(echoPin, INPUT); // echoPin as an

Input

Serial.begin(9600);

myServo.attach(10); // Pin Connected To

Servo

}

void loop() {

// rotating servo i++ depicts increment of

one degree

for(inti=15;i<=165;i++){

myServo.write(i);//rotating servo motor

from 15 to 165 degree by increment of one

degreedelay(30);

distance = calculateDistance();//calling a

function which measures the distance of any

obstacle on per degree rotation

Serial.print(i);//printing rotating angle

of the sensor on serial screen

Serial.print("A");

Serial.print(distance);//printing distance

of an obstacle on serial screen

Serial.print("D");

}

// Repeats the previous lines from 165 to

15 degrees

for(inti=165;i>=15;i--){

myServo.write(i);

delay(30);

distance = calculateDistance();

Serial.print(i);

Serial.print("A");

Serial.print(distance);

Serial.print("D");

}

}

int calculateDistance(){

//clear the trigPin

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10

micro seconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH);

distance= duration\*0.034/2;

return distance;

}

}

RESULTS AND DISCUSSION

Our system can detect and range any object with good resolution inside the range of '40 cm’

There are many kind of tracking system nowadays use in modern world some are radar,so they use radio waves for detection,some are sonar,so they use ultrasonic sounds,some are using infrared for extract any heat or temperature related information of target object.These tracking systems at the same time can do many things like, detection, ranging, extract motion information from the targeted object,the heat signature on any targeted object.

Our system only can range any obstacle less than ‘40cm’

It only can extract information like how much distance the targeted object has from the sensor and in which angle respect to sensor

CONCLUSIONS

The aim objective of this project was to design and implement a simple sonar system. The mentioned system can detect the object and measure the target distance. Short-range radar system a low cost, a simple device for distance measurement. The software results have been verified by using rubber and paper with a drawn angle. Hence, the device calculates the distance with suitable accuracy and resolution. The data converted into visual information. The appliance of the device is numerous. Also Being a miniature design of the actual SONAR system this project has helped us a lot to understand the SONAR being used in submarines or different application. Due to easy distance calculation of the surrounding ,this project can contribute largely and efficiently in many future endeavors.

REFERENCES

**1.YOUTUBE**

**2.GOOGLE**

**3.WIKIPEDIA**