CHAPTER-1

INTRODUCTION

Radar is an object detection system which uses radio waves to determine the range, altitude, direction or speed of objects. Radar systems come in a variety of sizes and have different performance specifications. Some radar systems are used for air-traffic control at airports and others are used for long range surveillance and early-warning systems. A radar system is the heart of a missile guidance system. Small portable radar systems that can be maintained and operated by one person are available as well as systems that occupy several large rooms

Radar was secretly developed by several nations before and during the World War II. The term RADAR itself, not the actual development, was coined in 1940 by United States Navy as an acronym for Radio Detection and ranging. The modern uses of radar are highly diverse, including air traffic control, radar, astronomy, air Defence systems, antimissile systems, antimissile systems; marine radars to locate landmarks and other ships; aircraft anti-collision systems; ocean surveillance systems, outer space surveillance and rendezvous systems; meteorological precipitation monitoring; altimetry and flight control precipitation monitoring; altimetry and flight control systems; guided missile target locating systems; and ground-penetrating radar for geological observations.

High tech radar systems are associated with digital signal processing RADAR stands for Radio Detection and Ranging System. It is basically an electromagnetic system used to detect the location and distance of an object from the point where the RADAR is placed. It works by radiating energy into space and monitoring the echo or reflected signal from the objects. It operates in the UHF and microwave range.

The RADAR system generally consists of a transmitter which produces an electromagnetic signal which is radiated into space by an antenna. When this signal strikes any object, it gets reflected or reradiated in many directions. This reflected or echo signal is received by the radar antenna which delivers it to the receiver, where it is processed to determine the geographical statistics of the object.

The range is determined by the calculating the time taken by the signal to travel from the RADAR to the target and back. The target’s location is measured in angle, from the direction of

maximum amplitude echo signal, the antenna points to. To measure range and location of moving objects, Doppler Effect is used.

RADAR system is 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 radar is 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 systems 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 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".

The project works on the principle of radar echo effect of the transmitting signal. Arduino control the servo motor for the direction of ultrasonic sensor and it moves from 0 degree to 180 degree. Ultrasonic sensor transmits the signal in all direction and if any obstacle that is target is detected then echo pulse sense. With the help of this echo pulse Arduino program find out the distance and direction angle of the target. Arduino controller and ultrasonic sensor is the base of this project. With the help of Arduino whole radar system forward, backward, left and right. Therefore, user can shift the radar at the place of required.

1.1 Arduino UNO

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the

microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing. Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike. Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IOT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board has 14 digital I/O pins (six capable of PWM output), 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable. It can be powered by the USB cable or by an external 9-volt battery, though it accepts voltages between 7 and 20 volts. It is also similar to the Arduino Nano and Leonardo. The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available the word "uno" means "one" in Italian and was chosen to mark the initial release of Arduino Software. The Uno board is the first in a series of USB-based Arduino boards; it and version 1.0 of the Arduino IDE were the reference versions of Arduino, which have now evolved to newer releases. The ATmega328 on the board comes preprogrammed with a boot loader that allows uploading new code to it without the use of an external hardware programmer. While the Uno communicates using the original STK500 protocol, it differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it uses the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter.

Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. 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.

1.2 HC-SRo4 Ultrasonic Sensor

HC-SR04 Ultrasonic (US) sensor is a 4-pin module, whose pin names are Vcc, Trigger, Echo and Ground respectively. This sensor is a very popular sensor used in many applications where measuring distance or sensing objects are required. The module has two eyes like projects in the front which forms the Ultrasonic transmitter and Receiver. The sensor works with the simple high school formula that

Distance = Speed × Time

The Ultrasonic transmitter transmits an ultrasonic wave, this wave travels in air and when it gets objected by any material it gets reflected back toward the sensor this reflected wave is observed by the Ultrasonic receiver module. HC-SR04 ultrasonic sensor. "The HC-SR04 ultrasonic sensor uses sonar to determine distance to an object like bats or dolphins do. It offers excellent non-contact range detection with high accuracy and stable readings in an easy-to-use package. From 2cm to 400 cm or 1" to 13 feet.

1.3 Servo motor

A servo motor is an electrical device which can push or rotate an object with great precision. If you want to rotate and object at some specific angles or distance, then you use servo motor. It is just made up of simple motor which run through servo mechanism. A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors. Servomotors are not a specific class of motor, although the

term servomotor is often used to refer to a motor suitable for use in a closed-loop control system. Servomotors are used in applications such as robotics, CNC machinery or automated manufacturing. A servomotor is a closed-loop servomechanism that uses position feedback to control its motion and final position. The input to its control is a signal (either analogue or digital) representing the position commanded for the output shaft.

The motor is paired with some type of position encoder to provide position and speed feedback. In the simplest case, only the position is measured. The measured position of the output is compared to the command position, the external input to the controller. If the output position differs from that required, an error signal is generated which then causes the motor to rotate in either direction, as needed to bring the output shaft to the appropriate position. As the positions approach, the error signal reduces to zero and the motor stops.

The very simplest servomotors use position-only sensing via a potentiometer and bang-bang control of their motor; the motor always rotates at full speed (or is stopped). This type of servomotor is not widely used in industrial motion control, but it forms the basis of the simple and cheap servos used for radio-controlled models.

More sophisticated servomotors use optical rotary encoders to measure the speed of the output shaft and a variable-speed drive to control the motor speed. Both of these enhancements, usually in combination with a PID control algorithm, allow the servomotor to be brought to its commanded position more quickly and more precisely, with less overshooting.

1.4 Jumper wires

Jumper wires are simply wiring 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 and other prototyping tools in order to make it easy to change a circuit as needed. Jump Wire also known as jumper wire, or jumper is an electrical wire, or group of them in a cable, with a connector or pin at each end or sometimes without them simply tinned, which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. Individual jump wires are fitted by inserting their end connectors into the slots provided in a breadboard, the header

connector of a circuit board, or a piece of test equipment. Jump wires are simply wiring 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 and other prototyping tools in order to make it easy to change a circuit as needed. 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. A jump wire (also known as jumper wire, or jumper) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. Individual jump wires are fitted by inserting their "end connectors" into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment.

1.5 Mini bread board

Breadboards are designed to work with through-hole electronic components. These components have long metal leads that are designed to be inserted through holes in a printed circuit board (PCB) that are plated with a thin copper coating, which allows the components leads to be soldered to the board. A breadboard is a construction base for prototyping of electronics. Originally the word referred to a literal bread board, a polished piece of wood used for slicing bread. In the 1970s the solder less breadboard (a.k.a. plug board, a terminal array board) became available and nowadays the term breadboard is commonly used to refer to these. Because the solder less breadboard does not require soldering, it is reusable. This makes it easy to use for creating temporary prototypes and experimenting with circuit design. For this reason, solder less breadboards are also popular with students and in technological education. Older breadboard types did not have this property. A strip board (Veroboard) and similar prototyping printed circuit boards, which are used to build semi-permanent soldered prototypes or one-offs, cannot easily be reused. A variety of electronic systems may be prototyped by using breadboards, from small analog and digital circuits to complete central processing units (CPUs) n the early days of radio, amateurs nailed bare copper wires or terminal strips to a wooden board (often literally a

board to slice bread on) and soldered electronic components to them. Sometimes a paper schematic diagram was first glued to the board as a guide to placing terminals, then components and wires were installed over their symbols on the schematic. Using thumbtacks or small nails as mounting posts was also common. Breadboards have evolved over time, with the term now being used for all kinds of prototype electronic devices. For example, US Patent 3,145 ,483, was filed in 1961 and describes a wooden plate breadboard with mounted springs and other facilities. US Patent 3,496,419, was filed in 1967 and refers to a particular printed circuit board layout as a Printed Circuit Breadboard. Both examples refer to and describe other types of breadboards as prior art. The breadboard most commonly used today is usually made of white plastic and is a pluggable (solder less) breadboard. It was designed by Ronald J. Portugal in 1971.

1.6 Arduino USB cable

A Universal Serial Bus (USB) is a common interface that enables communication between devices and a host controller such as a personal computer (PC). It connects peripheral devices such as digital cameras, mice, keyboards, printers, scanners, media devices, external hard drives and flash drives. Because of its wide variety of uses, including support for electrical power, the USB has replaced a wide range of interfaces like the parallel and serial port. Micro USB cable is a common interface that enables communication between devices and a host controller such as a personal computer (PC). It connects peripheral devices such as digital cameras, mice, keyboards, printers, scanners, media devices, external hard drives and flash drives. Because of its wide variety of uses, including support for electrical power, the USB has replaced a wide range of interfaces like the parallel and serial port. Universal Serial Bus (USB) is an industry standard that establishes specifications for cables and connectors and protocols for connection, communication and power supply between computers, peripheral devices and other computers. Released in 1996, the USB standard is currently maintained by the USB Implementers Forum (USB-IF). There have been four generations of USB specifications: USB 1.*x*, USB 2.0, USB 3.*x* and USB4. USB was designed to standardize the connection of peripherals to personal computers, both to communicate with and to supply electric power. It has largely replaced interfaces such as serial

ports and parallel ports, and has become commonplace on a wide range of devices' connectors have been increasingly replacing other types for battery chargers of portable devices. Examples of peripherals that are connected via USB include keyboards, pointing devices, digital still and video cameras, printers, portable media players, disk drives and adapters. From the computer user's perspective, the USB interface improved ease of use in several ways. The USB interface is self-configuring, so the user need not adjust settings on the device and interface for speed or data format, or configure interrupts, input/output addresses, or direct memory access channels. USB connectors are standardized at the host, so any peripheral can use any available receptacle. USB takes full advantage of the additional processing power that can be economically put into peripheral devices so that they can manage themselves; USB devices often do not have user-adjustable interface settings.

The USB interface is hot pluggable, meaning devices can be exchanged without rebooting the host computer. Small devices can be powered directly from the USB interface, displacing extra power supply cables. Because use of the USB logos is only permitted after compliance testing, the user can have confidence that a USB device will work as expected without extensive interaction with settings and configuration; the USB interface defines protocols for recovery from common errors, improving reliability over previous interfaces. Installation of a device relying on the USB standard requires minimal operator action. When a device is plugged into a port on a running personal computer system, it is either entirely automatically configured using existing device drivers, or the system prompts the user to locate a driver which is then installed and configured automatically.

For hardware manufacturers and software developers, the USB standard eliminates the requirement to develop proprietary interfaces to new peripherals. The wide range of transfer speeds available from a USB interface suits devices ranging from keyboards and mice up to streaming video interfaces. A USB interface can be designed to provide the best available latency for time-critical functions, or can be set up to do background transfers of bulk data with little impact on system resources. The USB interface is generalized with no signal lines dedicated to only one function of one device.

1.7 Arduino IDE

The Arduino integrated development environment is a cross-platform application that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development board. Arduino is a both an open source software library and an open-source breakout board for the popular AVR micro-controllers. The Arduino IDE (Integrated Development Environment) is the program used to write code, and comes in the form of a downloadable file on the Arduino website. The Arduino board is the physical board that stores and performs the code uploaded to it. Both the software package and the board are referred to as "Arduino." A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more. Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code. The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex Filen which is then transferred and uploaded in the controller on the board. The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module. This environment supports both C and C++ languages. The Arduino (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development boards.

The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program avrdude to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware. By default, avrdude is used as the uploading tool to flash the user code onto official Arduino boards.

1.8 Processing application

The application processing means the use of transaction data for bringing out a particular status. The application could be designed to change the number of different files holding a variety of information. Processing is an open-source graphical library and integrated development environment (IDE) built for the electronic arts, new media art, and visual design communities with the purpose of teaching non-programmers the fundamentals of computer programming in a visual context. Processing uses the Java language, with additional simplifications such as additional classes and aliased mathematical functions and operations. It also provides a graphical user interface for simplifying the compilation and execution stage.

The Processing language and IDE were the precursor to other projects including Arduino, Wiring and P5js. Processing includes a sketchbook, a minimal alternative to an integrated development environment (IDE) for organizing projects. Processing is a flexible software sketchbook and a language for learning how to code within the context of the visual arts. Since 2001, Processing has promoted software literacy within the visual arts and visual literacy within technology. There are tens of thousands of students, artists, designers, researchers, and hobbyists who use Processing for learning and prototyping

Every Processing sketch is actually a subclass of the PApplet Java class (formerly a subclass of Java's built-in Applet) which implements most of the Processing language's features.

When programming in Processing, all additional classes defined will be treated as inner classes when the code is translated into pure Java before compiling. This means that the use of static variables and methods in classes is prohibited unless Processing is explicitly told to code in pure Java mode.

Processing also allows for users to create their own classes within the P Applet sketch. This allows for complex data types that can include any number of arguments and avoids the limitations of solely using standard data types such as: int (integer), char (character), float (real number), and color (RGB, RGBA, hex).

CHAPTER-2

LITERATURE REVIEW

2.1 'The Idea'

Army, Navy and the Air Force make use of this technology. The use of such technology has been seen recently in the self-parking car systems launched by AUDI, FORD etc. And even the upcoming driverless cars by Google like Prius and Lexus.

The project made by us can be used in any systems the customer may want to use like in a car, a bicycle or anything else. The use of Arduino in the project provides even more flexibility of usage of the above-said module according to the requirements.

The idea of making an Ultrasonic RADAR came as a part of a study carried out on the working and mechanism of "Automobiles of Future". Also, being students of ECE, we have always been curious about the latest ongoing technology in the world like Arduino, Raspberry Pi, Beagle-Bone boards etc. A hence this time we were able to get a hold of one of the Arduino boards, Arduino UNO R3. So, knowing about the power and vast processing capabilities of the Arduino, we thought of making it big and a day to day application specific module that can be used and configured easily at any place and by anyone.

Moreover, in this fast-moving world there is an immense need for the tools that can be used for the betterment of the mankind rather than devastating their lives. Hence, we decided to make some of the changes and taking the advantage of the processing capabilities of Arduino, we decided to make up the module more application specific.

Hence, from the idea of the self-driving cars came the idea of self-parking cars. The main problem of the people in India and even most of the countries is safety while driving. So, we came up with a solution to that by making use of this project to continuously scan the area for traffic, population etc. and as well as protection of the vehicles at the same time to prevent accidents or minor scratches to the vehicles.

- Military use to protect the ground borders and the airspace as well as for military aviation control.

- Police use to catch speed law offenders as well as border control to prevent smugglers and illegal aliens from crossing the border.

- Civil Aviation use for air traffic control to control the airspace and prevent airplane collisions, as well as Ground Control Approach in critical visibility situations in lesser equipped airports.

- Meteorological use ground and airborne to detect weather conditions especially heavy cloud accumulation (specifically the dangerous Cumulus nimbus type).

Subsequent to experiencing a portion of the papers with respect to usage utilizing ultrasonic sensors and ARDUINO, it was found that this idea is searched a lot and is a mainstream idea which is still in advance. The advances utilized were not just productive and solid yet in addition financially achievable Arduino based radar system. Not only this, here other very useful applications of ultrasonic sensors were observed too.

This paper discusses about a monitoring system which is designed measure to speed of waves and height of river through ultra-sonic sensor using microcontroller (Arduino). On the off chance that the waterway can't oblige the volume of water, then all the water will submerge with land and this phenomenon is called as flood or surge. We can overcome this flood problem by earlier identification in height of water and observing speed. If we identify problem earlier, we can overcome this problem before it become crisis. By testing the system i.e. simple water level, it was observed that ultra-sonic have accuracy of 96.6%. But when it is implemented in the rivers there are many errors because of different type of water levels due to heavy waves and speed of water and also due to floating of heavy objects. Unlike Previous testing results, author directed this analysis on tracking of speed of water improvement or modification and level of water in flooding. The test was completed when the Arduino used as controller of application. For more research, information of depth level and speed of water of this system will be sent to database server website to be checked regularly.

An intelligent driver monitoring and vehicle control system is introduced in this research. This technology is creating to avoid accidents by monitoring the driver’s activities. The writer states some of the main reasons of accidents today. These are alcohol consumption by the driver,

carelessness, drowsiness or medical illness. The various units in the framework, including motors, relays, power unit and ESP8299 module are tried and are observed to be in working condition. Ultrasonic sensor is utilized to alarm the driver if any vehicle draws close to his vehicle. The status of the driver can be observed by the assistance of sensors executed in the vehicle and the subtle elements are refreshed to the proprietor. This system overcomes all the different aspects due to which other technologies designed for this purpose have failed, making the system more useful, efficient and less costly and less time consuming. In this research paper authors have given information about the detection of radio waves and tracking or ranging through radar set which is built from components like an ultra-sonic sensor, a servo motor and an Arduino. The author discuss about the linear measurement problem because of which distance measurement was not possible between some objects, was resolved with the introduction of Ultrasonic distance measurer. It allows to take noncontact measurements. This radar system can drastically reduce power consumption. The author says, that this system is an extremely handy radar system, it can read or track the distance and angle of an obstacle and shown it up on the monitor screen.

The ultra-sonic was attached on top of the servomotor to detect obstacles at 0 degree to 180 degree from right to left. Both the ultra-sonic sensor and the servo were fuelled and controlled by the Arduino controller. The GUI was built using the JAVA programming language to show the result on the monitor. This paper represents a system for obstacle detection in a known environment. This system works through an android based mobile camera. People who are Arduino based radar system DOI: http://dx.doi.org/10.17993/3ctecno.2019.specialissue.14 160 visually impaired, face difficulties in detecting obstacles and navigation while they walk. They use sticks for this problem nonetheless this manner or technique is not right way of doing it. Object indicator or detector can overcome accidents or collision problems of people or the other way is they can to accurate map reading. The algorithm which is proposed in this paper is made for indoor mapping. In indoor surrounding all distinctive floors are taken in consideration and single image is kept or stored for distinctive floors. These images of floor are taken as reference image. The author mentions that this algorithm is 96% accurate and works in real time. There are different techniques discussed in this paper for obstacle detection. For these types of problems, we can use the approach of SONAR sensor and also laser camera. In this paper introduced a

calculation for identifying hindrance in known condition with an android based versatile camera which scans chosen territory before the camera for impediment location.

This research is about a blind walking stick made for blind people through whom they can avoid obstacles while they walk and recognize currency. With the thought of visually impaired individuals, it is to some degree troublesome job to distinguish the cash or any unexpected obstacle. Despite the fact that currency dependent on size could possibly be recognized however it is relatively hard to distinguish that whether the note is unique or phony. So, to overcome this issue the authors have designed the Currency Recognition Blind Walking Stick. A lot of work is done on currency recognition and obstacle detection using advanced technologies like optical character recognition, SURF and pattern extraction through colours. But none of these systems had the feature for obstacle detection for blind people. Therefore, this framework is efficient as the other ones having an extra feature for helping the visually impaired.

The Arduino Uno is a microcontroller board based on the ATmega328. It has14 digital Input /Output pins (of which 6 can be used as PWM outputs), 6 analoginputs, a 16MHz ceramic resonator, USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to computer with a USB cable or power it with a AC-to-DC adapter 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 the Atmega16U2 programmed as a USB-to-serial converter. Changes in Uno R3.

Pin out: added SDA and SCL pins that are near to the AREF pin and two other new pins placed near to the reset pin, the IOREF that allow the shields to adapt to the voltage provided from the board.with both the board that uses the AVR, which operates with 5v and with the Arduino due that operates with 3.3v.2. Stronger RESET circuit.3.

ATmega16U2 replace the 8U2."Uno" means one in Italian and is named to mark the upcoming release of Arduino1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, seethe index of Arduino Board.

Literature Survey

Sr no.	Author	Work of Result	Year
1	Christan Hulsmeyer	Use of radio echo to detect ship	1904
2	Robart Alexander Watson-Watt	Robart Alexander Watson-Watt	1939

Table 1 (Source: www.google.com)

CHAPTER-3

METHODOLOGY

Arduino program controls the all interfaced devices. For moving the radar, we have robotics setup. WI-FI is connected to Arduino serially. Arduino controls the motors clockwise and anti-clockwise by giving signal to the motor driver IC (L293D IC). User sends the command by mobile to the WI-FI and gives this command to the Arduino. Arduino program compare this code with the predefined code. If it is match then program gives digital signal to motor driver IC and perform the required operation like left, right, forward and backward. Arduino control the servo motor in angle between 0 to 180 degree. On this servo motor we are putting ultrasonic sensor which is connected to the Arduino board. Ultrasonic sensor works on the trigger and echo pulse. As per object distance it will generate echo pulse. If object is having less distance than threshold distance saved in program then program will find out the angle of servo motor. In next operation \distance data and angle data is send to the predefined mobile number in SMS using SIM900 module which is interfaced to the Arduino. The beauty of the project is camera application. If object is detected then camera move in the object direction with the help of another servo motor

3.1 OBJECTIVE

- The basic objective of our design is to ascertain the distance position and speed of the obstacle set at some distance from the sensor.
- This Arduino based radar system can provide cheapest radar to the customer. It reduces the cost.
- It can help the blind person to walk and move easily. Without any fear of collision, we can embed a buzzer which buzz when any object is in the range of the waves.
- We during this learn how to programme the microcontroller with the help of software and make it for use.
- We use Arduino UNO microcontroller and Arduino IDE for programme it and processing application to display the objects detected in the radio wave
- The sensor produces radio wave and the uses pulses system to detect the objects.

3.2 SCOPE

In order to testify the working of this system, after its designing, construction and programming we placed few objects in front of the ultrasonic sensor. As the motor started to rotate, our monitor started to display the output through processing IDE. Hence, when the sensor crossed over the object it showed a red segment with the distance and angle where the object is paced. The first object was placed at the distance of 30.5cm measured through a ruler and the system measured the distance at 32cm.While the second object was placed at a distance of 20 cm and the system measured it as 21cm. Hence the calculated efficiency turned out to be 95% system.

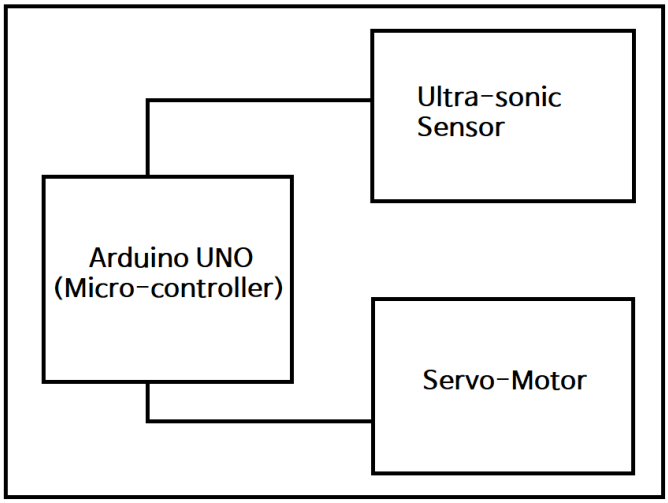

Fig 3.2 (Source: Created in paint)

3.3 PROCESS

The basic objective of our radar is to obtained the distance position and angle of the obstacle set at some distance from the sensor. Ultrasonic sensor sends the ultrasonic wave in various ways by rotating with help of servo motors. This wave goes in air and gets reflected back subsequent to striking some object. This wave is again detected by the sensor and its qualities is analysed and output is shown in screen indicating parameters, for example, distance and position angle of object. Arduino IDE is utilized to compose code and transfer coding in Arduino and causes us to detect position or angle of servo motor and it is communicated through the serial port alongside the covered distance of the nearest object in its way. Output of all of this working is shown in the

software processing 3, it will display the input/output and the range of the object Implementations of the sensors are done in such a way that ultra-sonic sensor is attached on top of the servo motor because it have to detect the object and its distance. Arduino (micro-controller) will control the ultra-sonic sensor and servo motor and also powered will be given to both of them through micro-controller Flow chart of Radar System.

Flow Chart

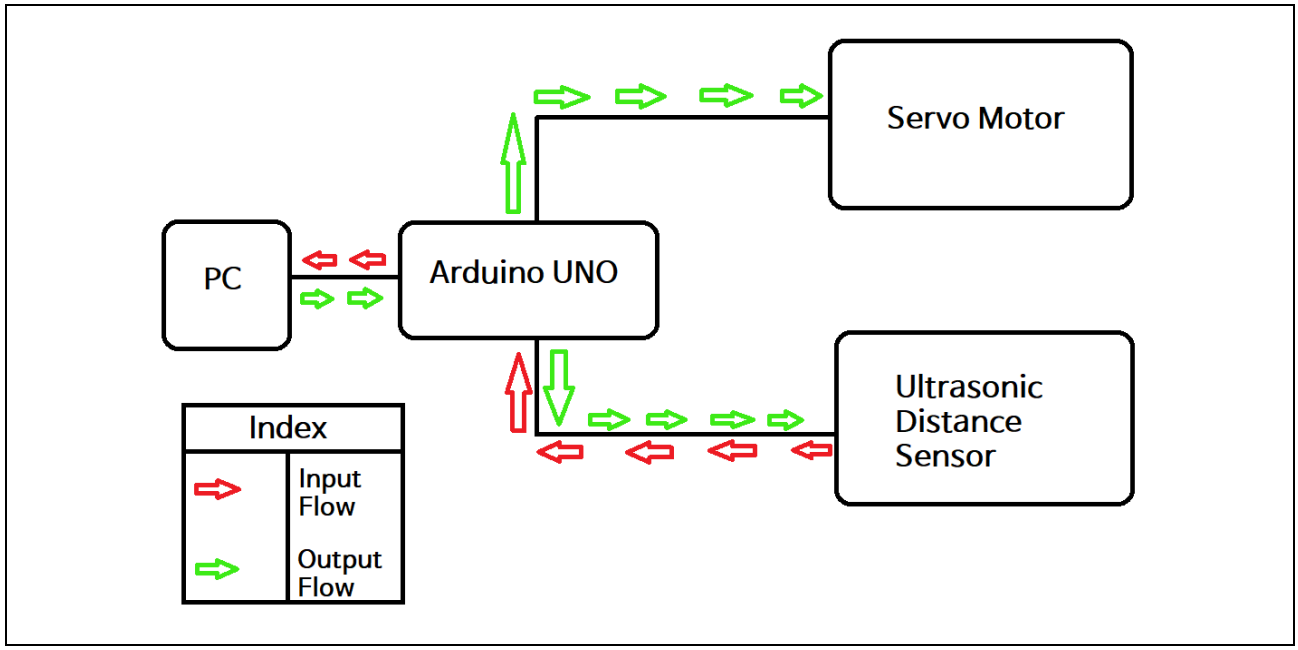

Fig 3.3 (Source: Created in paint)

The above flow chart explains the working and the decision flow of this framework. As it can be seen the system starts with an input i.e. when the ultrasonic sensor detects an object, or does not detects any object, at any condition the encoder feeds the information in the controller while the servo keeps constantly rotating.

As soon as any obstacle/object is detected by the ultrasonic sensor the data is immediately processed by the controller and is fed to the IDE which shows it on the display screen. Here the

process ends with an estimated distance of the object from the system with the angle at which it is placed.

3.4 VARIABLES AND DATA

3.4.1 Power supply

The System (PC) is used as power house for over project. We use system USB port as power supply for our Arduino based radar system. The Arduino USB cable is connected one end to System with USB port and the other end to the Arduino board. Genuinely the Arduino board is powered by the ac adapter which is provided along with it. But we can also use the USB cable as power cable for our Arduino board system provides its sufficient power to Arduino board which further work as power house for ultrasonic sonar and servo motor.

3.4.2 Arduino UNO board at Arduino application

We use the Arduino ide for over Radar system based on the Arduino UNO. We use the Arduino platform for installing the Arduino UNO board for coding in suitable programming language. Here are some easy points we take during the coding phase of over Radar System as mention below:

1. In the Arduino IDE open the Preferences window and enter the URL: http://arduino.esp8266.com/stable/package_esp8266com_index.json into the Additional Boards Manager URLs field, and click OK.

2. In the MENU select: Tools → Board → Boards Manager and scroll down and to choose esp8266 by ESP8266 Community and click INSTALL. Installing takes about 3 minutes.

3. Install the USB Driver (in my case CH340).

4. Restart the Arduino IDE and select our board from the menu option: Tools → Board → NodeMCU 1.0 (ESP-12E Module).

5. Then, we specify the correct CPU Frequency (Tools → CPU Frequency→ 80MHz) and Upload Speed (Tools → Upload Speed → 115200).

6. Select the correct option for the Port (Tools → Port → COM5).

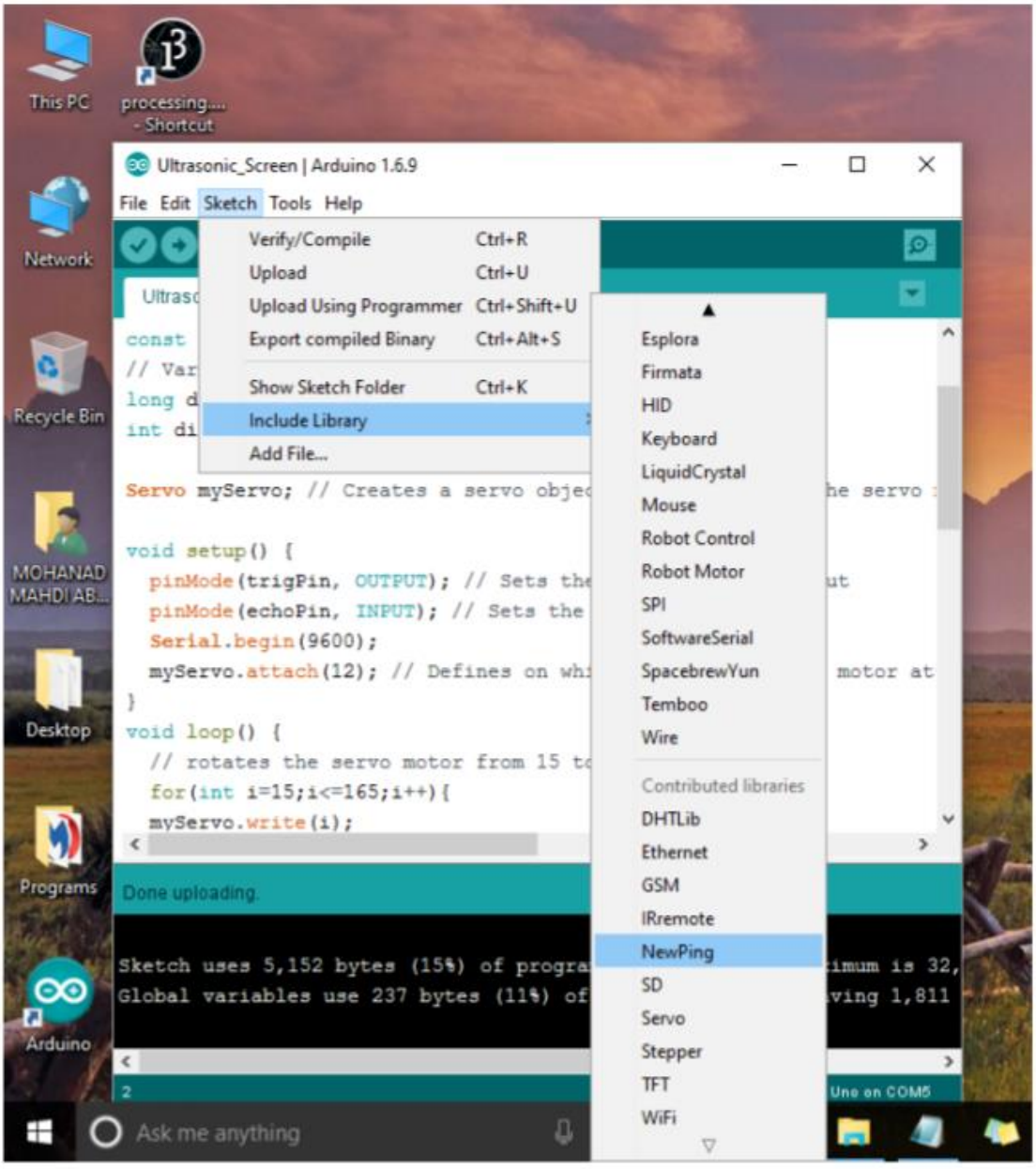

Fig 3.4.2 (Source: www.circuitio.com)

3.4.3 Processing 3 IDE

1. The values for the angle and the distance measured by the sensor will be read from the Arduino board by the Processing IDE using the SerialEvent() function which reads the data from the Serial Port. These values will be used for drawing the lines, the detected objects and some texts.
2. For drawing the radar display we make this function drawRadar() which consist of arc() and line() functions.

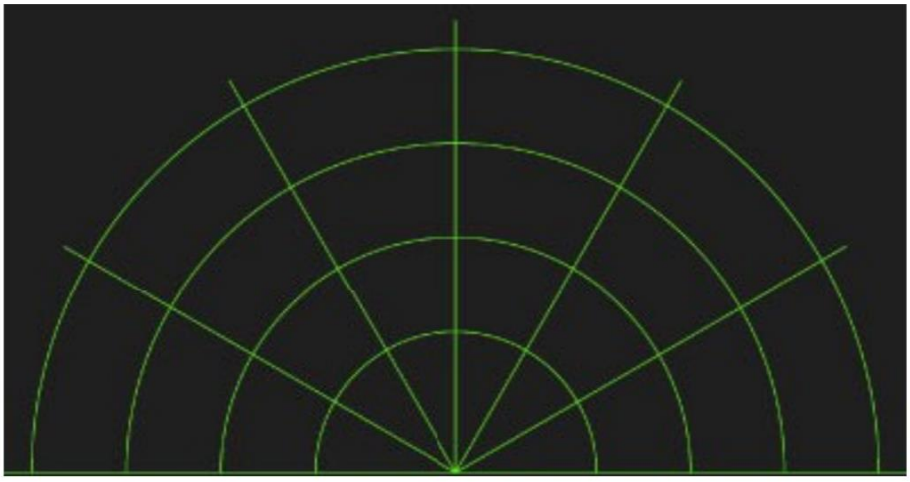

Fig 3.4.3 (a) (Source: www.circuitio.com)

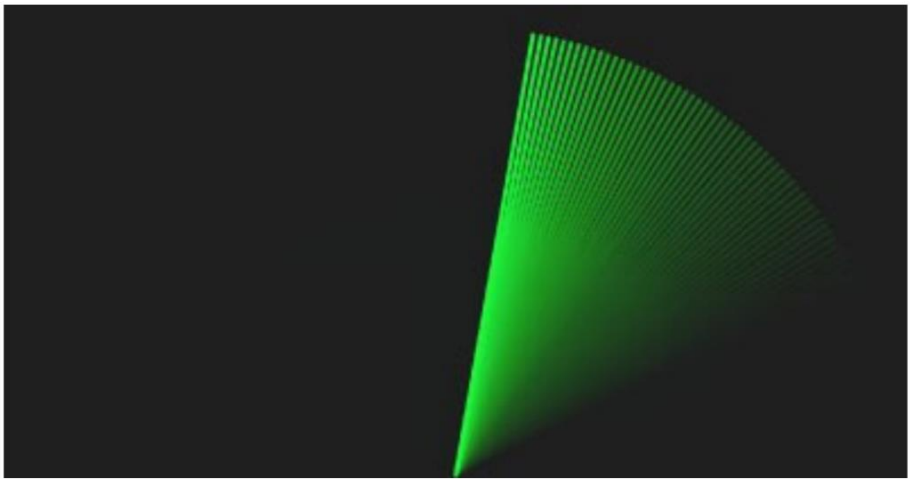

Fig 3.4.3 (b) (Source: www.circuitio.com)

3. For drawing the moving lines we make this function drawLine(). Its center of rotation is set with the translate() function and using the line() function in which the iAngle variable is used to redraw the line for each degree.

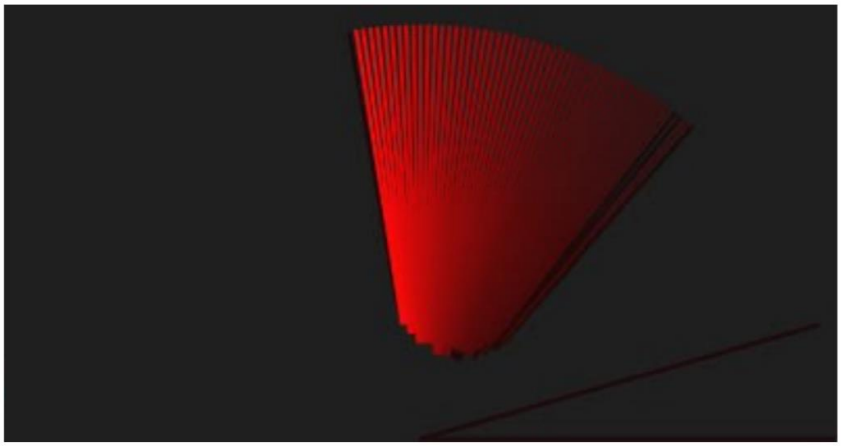

Fig 3.4.3 (c) (Source: www.circuitio.com)

4. For drawing the detected objects we made the drawObject() function. It receives the distance from the ultrasonic sensor, transforms it into pixels. Then, using the angle detected by the sensor it draws the object on the radar screen
5. To illustrate the text on the screen, we make the drawText() function that draws texts on some particular locations. All of these functions are called in the main draw() function which is repeated in each iteration to draw the screen details

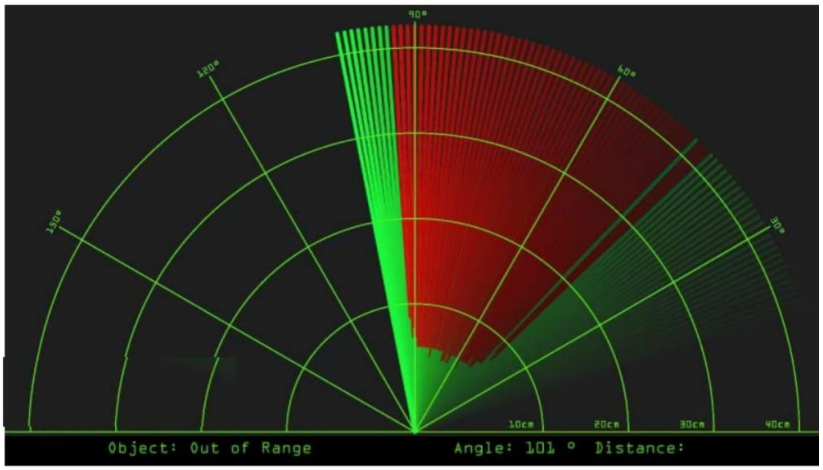

Fig 3.4.3 (d) (Source: www.circuitio.com)

3.4.4 Circuit Diagram

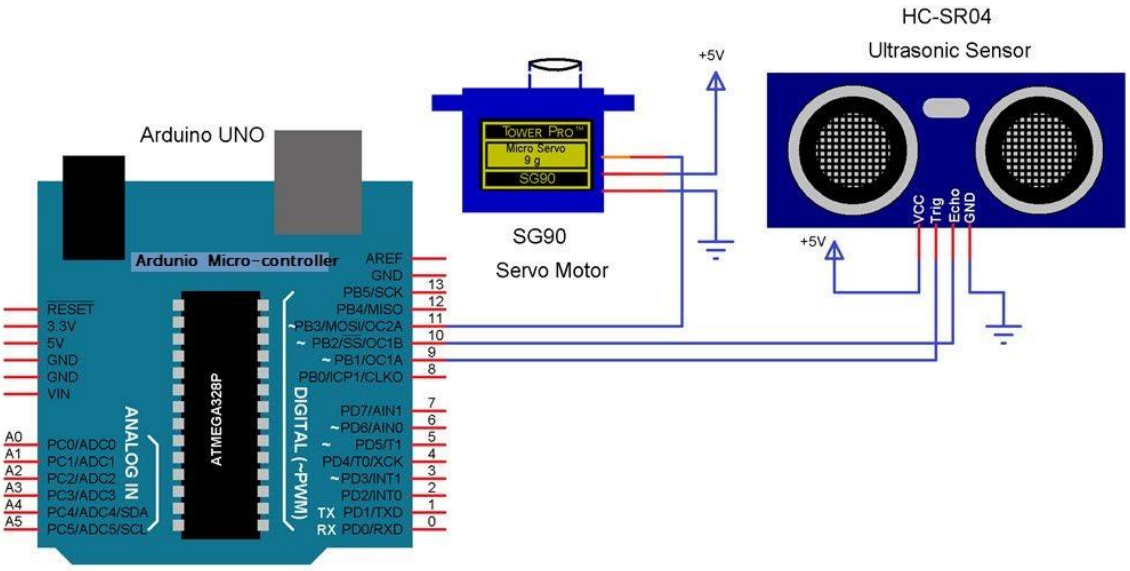

Fig 3.4.4 (Source: www.circuitio.com)

3.5 TOOLS AND TECHNOLOGY

3.5.1 Hardware Requirements

- Arduino UNO
- HC-SRo4 Ultrasonic Sensor
- Servo Motor
- Jumper Cables
- Mini bread board
- USB Cable for Arduino

3.5.2 Software Requirements

- Arduino IDE
- Processing Application

3.5.1.1 Arduino UNO

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. Arduino, an easy-to-use programmable device for interactive art design projects, at the Interaction Design Institute Ivrea in Ivrea, Italy. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing. Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments.

Different Arduino boards		
Arduino Board	**Processor**	**Digital I/O**
Arduino Uno	16Mhz ATmega328	14
Arduino Due	84MHz AT91SAM3X8E	54
Arduino Mega	16MHz ATmega2560	54
Arduino Leonardo	16MHz ATmega32u4	20

Table 2 (Source: www.google.com)

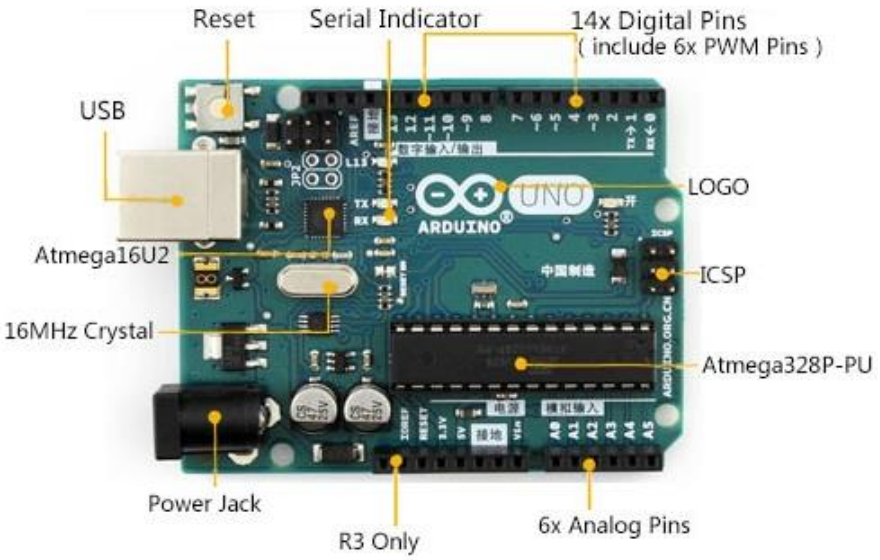

Fig 3.5.1.1 (Source: www.google.com)

3.5.1.2 HC-SRo4 Ultrasonic Sensor

HC-SR04 Ultrasonic (US) sensor is a 4-pin module, whose pin names are Vcc, Trigger, Echo and Ground respectively. This sensor is a very popular sensor used in many applications where measuring distance or sensing objects are required. The module has two eyes like projects in the front which forms the Ultrasonic transmitter and Receiver. The Ultrasonic transmitter transmits an ultrasonic wave, this wave travels in air and when it gets objected by any material it gets reflected back toward the sensor this reflected wave is observed by the Ultrasonic receiver module. HC-SR04 ultrasonic sensor. The amount of time it takes to send and receive waves will determine how far the object is placed from the sensor. It mainly depends on the sound waves working on "non-contact" technology. The required distance of the target object is measured without any damage, giving you accurate and precise details. This sensor comes with a range between 2cm to 400cm and is used in a wide range of applications including speed and direction measurement, wireless charging, humidifiers, medical ultrasonography, sonar, burglar alarms, and non-destructive testing.

There are four major different types of ultrasonic sensors are currently in used,

1. Ultrasonic Proximity Sensors
2. Ultrasonic 2 Point Proximity Switches
3. Ultrasonic Retro-reflective Sensors
4. Ultrasonic Through Beam Sensors

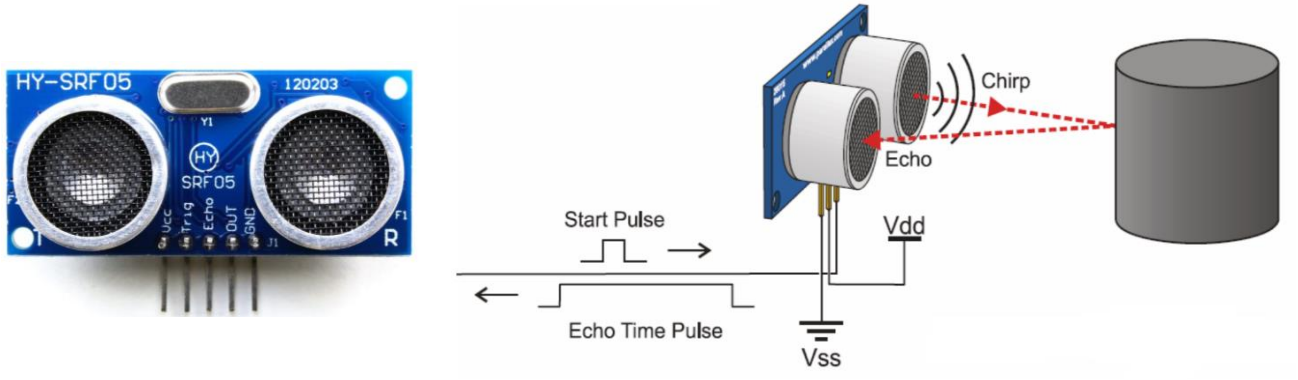

Fig 3.5.1.2 (Source: www.google.com)

3.5.1.3 Servo motor

A servo motor is an electrical device which can push or rotate an object with great precision. If you want to rotate and object at some specific angles or distance, then you use servo motor. It is just made up of simple motor which run through servo mechanism. A servomotor is a rotary actuator or linear actuator that allows for precise control of angular or linear position, velocity and acceleration. It consists of a suitable motor coupled to a sensor for position feedback. It also requires a relatively sophisticated controller, often a dedicated module designed specifically for use with servomotors. Servomotors are not a specific class of motor, although the term servomotor is often used to refer to a motor suitable for use in a closed-loop control system.

Fig 3.5.1.3 (Source: www.google.com)

3.5.1.4 Jumper wires

Jumper wires are simply wiring 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 and other prototyping tools in order to make it easy to change a circuit as needed. Jump Wire also known as jumper wire, or jumper is an electrical wire, or group of them in a cable, with a connector or pin at each end or sometimes without them simply tinned, which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. Individual jump wires are

fitted by inserting their end connectors into the slots provided in a breadboard, the header connector of a circuit board, or a piece of test equipment. Jump wires are simply wiring 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 and other prototyping tools in order to make it easy to change a circuit as needed.

Fig 3.5.1.4 (Source: www.google.com)

3.5.1.5 Mini bread board

Breadboards are designed to work with through-hole electronic components. These components have long metal leads that are designed to be inserted through holes in a printed circuit board (PCB) that are plated with a thin copper coating, which allows the components leads to be soldered to the board.

A breadboard is a construction base for prototyping of electronics. Originally the word referred to a literal bread board, a polished piece of wood used for slicing bread. In the 1970s the solder less breadboard (a.k.a. plug board, a terminal array board) became available and nowadays the term breadboard is commonly used to refer to these. Because the solder less breadboard does not require soldering, it is reusable.

This makes it easy to use for creating temporary prototypes and experimenting with circuit design. For this reason, solder less breadboards are also popular with students and in

technological education. Older breadboard types did not have this property. A strip board (Veroboard) and similar prototyping printed circuit boards, which are used to build semi-permanent soldered prototypes or one-offs, cannot easily be reused.

Fig 3.5.1.5 (Source: www.google.com)

3.5.1.6 Arduino USB cable

A Universal Serial Bus (USB) is a common interface that enables communication between devices and a host controller such as a personal computer (PC). It connects peripheral devices such as digital cameras, mice, keyboards, printers, scanners, media devices, external hard drives and flash drives. Because of its wide variety of uses, including support for electrical power, the USB has replaced a wide range of interfaces like the parallel and serial port. Micro USB cable is a common interface that enables communication between devices and a host controller such as a personal computer (PC).

It connects peripheral devices such as digital cameras, mice, keyboards, printers, scanners, media devices, external hard drives and flash drives. Because of its wide variety of uses, including support for electrical power, the USB has replaced a wide range of interfaces like the parallel and serial port.

Universal Serial Bus (USB) is an industry standard that establishes specifications for cables and connectors and protocols for connection, communication and power supply between computers, peripheral devices and other computers. Released in 1996, the USB standard is currently maintained by the USB Implementers Forum (USB-IF).

Fig 3.5.1.6 (Source: www.google.com)

3.5.2.1 Arduino IDE

The Arduino integrated development environment is a cross-platform application that is written in the programming language Java. It is used to write and upload programs to Arduino compatible boards, but also, with the help of 3rd party cores, other vendor development board. Arduino is a both an open source software library and an open-source breakout board for the popular AVR micro-controllers. The Arduino IDE (Integrated Development Environment) is the program used to write code, and comes in the form of a downloadable file on the Arduino website. The Arduino board is the physical board that stores and performs the code uploaded to it. Both the software package and the board are referred to as "Arduino."

A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more. Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code. The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex Filen which is then transferred and uploaded in the controller on the board. The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module. This environment supports both C and C++ languages.

Fig 3.5.2.1(Source: www.google.com)

3.5.2.2 Processing application

The application processing means the use of transaction data for bringing out a particular status. The application could be designed to change the number of different files holding a variety of information. Processing is an open-source graphical library and integrated development environment (IDE) built for the electronic arts, new media art, and visual design communities with the purpose of teaching non-programmers the fundamentals of computer programming in a visual context.

Processing uses the Java language, with additional simplifications such as additional classes and aliased mathematical functions and operations. It also provides a graphical user interface for simplifying the compilation and execution stage.

Fig 3.5.2.2(Source: www.google.com)

3.6 LANGUAGES USED

In over Arduino Based Radar System Major project to code the microcontroller Arduino UNO. With the help of Arduino IDE software, we install the Arduino software because this is the genuine software to code in the Arduino UNO. For the coding section we have two parts. One for the microcontroller codding which commands the other devices such as servo motor and ultrasonic sonar sensor connected to the microcontroller. And the second is for the output graph. We use the processing 3 software for the creation of graph in which the preloaded layout is coded to make the output graph which is further seen on the screen as shown in the given below figure.

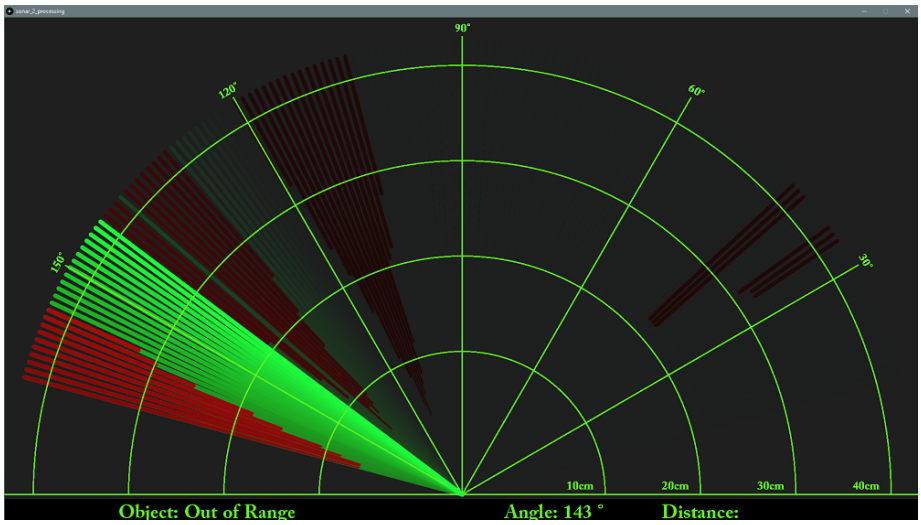

Fig 3.6 (Source: Output screenshot)

Software Description

Sr no.	Software name	Usage	Description
1.	**Arduino IDE**	The Arduino IDE is used to programme the Arduino board. We use this software to embed the code in the flash memory of the Arduino board.	The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in functions from C and C++. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures.
2.	**Processing 3**	Processing 3 software is used to display the radar output with the help of the graph. Processing 3 is also an IDE and in this we write create a screen graph to display the output of the radar.	Processing 3 is a flexible software sketchbook and a language for learning how to code and create visual arts for the projects. Processing has promoted software literacy within visual arts and visual technology. Its free to download and open source.

Table 3 (Source: Project)

Hardware Description

Sr no.	Hardware name	Usage	Description
1.	**Arduino UNO (Board)**	This is the main brain of our project in which code is embed by the Arduino ide software. It gives the command to other hardware component connect within the project circuit.	Arduino UNO is an open source electronics platform based on easy to use hardware and software. Arduino board are able to read input- light on a sensor, a finger on a button and turn it into an output- activating monitor, turn on LED, publishing something online
2.	**HC-SRo4 Ultrasonic Sensor**	The senor transmits radio wave and then when the object is come in its wave range. Then the wave strike and reflect backwards to	Ultrasonic transducers or ultrasonic sensors are a type of acoustic sensor divided into three broad categories: transmitters, receivers and transceivers

		wards the sensor and then the sensor receive it as input which is further processed.	
3.	**Servo Motor (Mini)**	It uses to rotate the sonar sensor in semicircle	A servo motor is an electrical device which can push or rotate an object with great precision. If you want to rotate and object at some specific angles or distance, then you use servo motor. It is just made up of simple motor which run through servo mechanism.
4.	**Bread Board (Mini)**	To connect Arduino board, sonar sensor and servo motor. We use the breadboard	Breadboard is a platform you can use to build and test electronic circuits, usually without having to do any soldering.
5.	**Jumper Wires**	To make the connection between our hardware and let then to communicate each other we use jumper wires	Jumper wires are simple wires that have connector pins at each end, allowing them to be used to connect two points to each other without soldering.
6.	**USB (A to B) Cable**	To connect the Arduino board with our system we use USB A to B cable and also it helps us to power all the hardware components without any other power supply	The USB A to B Cable comes with the Arduino UNO to connect it**.**

Table 4 (Source: Project)

CHAPTER 4

RESULT, ANALYSIS AND DISCUSSION

(I) Codding phase for the Arduino UNO: -

```
// Includes the Servo library
#include <Servo.h>.
// Defines Tirg and Echo pins of the Ultrasonic Sensor
const int trigPin = 10;
const int echoPin = 11;
// Variables for the duration and the distance
long duration;
int distance;
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(12); // Defines on which pin is the servo motor attached
}
void loop() {
  // rotates the servo motor from 15 to 165 degrees
  for(int i=15;i<=165;i++){
  myServo.write(i);
  delay(30);
  distance = calculateDistance();

  Serial.print(i); // Sends the current degree into the Serial Port
  Serial.print(",");
  Serial.print(distance);
```

```
  Serial.print(".");
  }
  // Repeats the previous lines from 165 to 15 degrees
  for(int i=165;i>15;i--){
  myServo.write(i);
  delay(30);
  distance = calculateDistance();
  Serial.print(i);
  Serial.print(",");
  Serial.print(distance);
  Serial.print(".");
  }
}
// Function for calculating the distance measured by the Ultrasonic sensor
int calculateDistance(){

  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance= duration*0.034/2;

  return distance;
}
```

(II) Codding phase for the output graph: -

```
import processing.serial.*;                 // imports library for serial communication
import java.awt.event.KeyEvent;             // imports library for reading the data from the
serial port
import java.io.IOException;
Serial myPort;                              // defines Object Serial
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);                                  // screen resolution
 smooth();
 myPort = new Serial(this,"COM4", 9600);      // starts the serial communication
 myPort.bufferUntil('.');
// reads the data from the serial port up to the character '.'. So actually it reads this:
angle,distance.
}
void draw() {

  fill(98,245,31);                                 // simulating motion blur and slow fade of the
moving line
  noStroke();
  fill(0,4);
```

```
  rect(0, 0, width, height-height*0.065);

  fill(98,245,31);                          // green color
                                            // calls the functions for drawing the radar
  drawRadar();
  drawLine();
  drawObject();
  drawText();
}
void serialEvent (Serial myPort) {          // starts reading data from the Serial Port

  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);     // moves the starting coordinats to new
location
  noFill();
  strokeWeight(2);
  stroke(98,245,31);
                                              // draws the arc lines
  arc(0,0,(width-width*0.0625),(width-width*0.0625),PI,TWO_PI);
  arc(0,0,(width-width*0.27),(width-width*0.27),PI,TWO_PI);
  arc(0,0,(width-width*0.479),(width-width*0.479),PI,TWO_PI);
  arc(0,0,(width-width*0.687),(width-width*0.687),PI,TWO_PI);
```

```
                                        // draws the angle lines
  line(-width/2,0,width/2,0);
  line(0,0,(-width/2)*cos(radians(30)),(-width/2)*sin(radians(30)));
  line(0,0,(-width/2)*cos(radians(60)),(-width/2)*sin(radians(60)));
  line(0,0,(-width/2)*cos(radians(90)),(-width/2)*sin(radians(90)));
  line(0,0,(-width/2)*cos(radians(120)),(-width/2)*sin(radians(120)));
  line(0,0,(-width/2)*cos(radians(150)),(-width/2)*sin(radians(150)));
  line((-width/2)*cos(radians(30)),0,width/2,0);
  popMatrix();
}
void drawObject() {
  pushMatrix();
  translate(width/2,height-height*0.074);       // moves the starting coordinats to new
location
  strokeWeight(9);
  stroke(255,10,10);                            // red color
  pixsDistance = iDistance*((height-height*0.1666)*0.025);
   // covers the distance from the sensor from cm to pixels
  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);                                    //
moves the starting coordinats to new location
```

```
  line(0,0,(height-height*0.12)*cos(radians(iAngle)),-(height-
height*0.12)*sin(radians(iAngle)));          // draws the line according to the angle
  popMatrix();
}
void drawText() {                            // draws the texts on the screen

  pushMatrix();
  if(iDistance>40) {
  noObject = "Out of Range";
  }
  else {
  noObject = "In Range";
  }
  fill(0,0,0);
  noStroke();
  rect(0, height-height*0.0648, width, height);
  fill(98,245,31);
  textSize(25);

  text("10cm",width-width*0.3854,height-height*0.0833);
  text("20cm",width-width*0.281,height-height*0.0833);
  text("30cm",width-width*0.177,height-height*0.0833);
  text("40cm",width-width*0.0729,height-height*0.0833);
  textSize(40);
  text("Major Project 2020", width-width*0.975, height-height*0.0177);
  text("Angle: " + iAngle +" °", width-width*0.55, height-height*0.0177);
  text("Distance: ", width-width*0.30, height-height*0.0177);
  if(iDistance<40) {
  text("        " + iDistance +" cm", width-width*0.225, height-height*0.0177);
  }
  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();
}
```

CHAPTER 5
SUMMARY

5.1 FINDINGS

In this finding we have mentioned that our system is designed consisting following components such as, a servo- motor, an ultra-sonic sensor and a micro-controller (Arduino). System's objective is to track the distance and angle of the object and to represent this information graphically, means its output should be in graphical form which will be represented through processing software. We can have an idea of an efficiency of this radar by testing objects at different levels and observe how faster or smoothly it detects an object that it finds in a way and gives us an expected range of the obstacle. we have designed a short-range radar therefore our research was specified and limited. This system can only detect objects from 30 to 150 degrees only because the servo motor that we have used can rotate only to this range. We look forward to modify this system and enhance our research work by using a fully 360 degrees rotating servo and a higher ranged ultrasonic sensor. We can further add features to this system i.e. making it mobile, mounting an alarm system to it which turns on when obstacle is detected. Further modifications could be an obstacle avoiding robot with surveillance system. Radar is an electronic device which utilizes electromagnetic waves to determine the altitude, range, direction, or speed of both moving and immovable objects. In contrast, ultrasonic waves are used instead of electromagnetic waves in ultrasonic radar. The low power consumption, low cost and ease of implementation are considered the main features of the ultrasonic radar to be devoted in sever AI applications such as security purposes, object detection and avoidance systems in robotics. This work presents a design and implementation of ultrasonic radar for distance measurements. The design consists of an ultrasonic sensor, an Arduino board as a controller, a servo motor and a java application. The detection range of the proposed system is tested up to 10 m with the angle of rotation from (30 to +150) and (150 to 30) degrees for different types of obstacles or objects (sponge, wood an aluminium). The design is built using open source hardware (Arduino Uno 328) which is coded via Micro C environment as a software entity. The effectiveness of the proposed design is measured using a statistical analysis of the distance error between the radar and the obstacles.

5.1.1 Applications

This Radar System have various applications for security purposes and it is mainly used for mapping.

o Application in Air Force:

It is used in airplanes or aircraft machines which have implemented radar system in it to detect the objects that comes in a way. It is also used to calculate height readings. In aviation, aircraft are equipped with radar devices that warn of aircraft or other obstacles in or approaching their path, display weather information, and give accurate altitude readings. The first commercial device fitted to aircraft was a 1938 Bell Lab unit on some United Air Lines aircraft. Such aircraft can land in fog at airports equipped with radar-assisted ground-controlled approach systems in which the plane's flight is observed on radar screens while operators radio landing directions to the pilot.

o Application in Marine:

Marine radars are used to measure the bearing and distance of ships to prevent collision with other ships, to navigate, and to fix their position at sea when within range of shore or other fixed references such as islands, buoys, and lightships. In port or in harbour, vessel traffic service radar systems are used to monitor and regulate ship movements in busy waters. This radar system also used in ships or marine. It is implemented on big ships to calculate the distance of other boats or ships, with the help of this sea accidents can also be reduced by not colliding. It can also be implemented on ports to see the distance of other ships and to monitor or control the ship movements.

o Application in Meteorology:

Arduino based radar system Meteorologists also uses radar systems to track or monitor the wind. It has been become an important equipment for climate testing. For example, to detect tornados, storms.

o Application in Army:

Two video cameras automatically detect and track individuals walking anywhere near the system, within the range of a soccer field. Low-level radar beams are aimed at them and then

reflected back to a computer, which analyses the signals in a series of algorithms. It does this by comparing the radar return signal (which emits less than a cell phone) to an extensive library of "normal responses." Those responses are modelled after people of all different shapes and sizes (SET got around to adding females in 2009). It then compares the signal to another set of "anomalous responses" – any anomaly, and horns go off. Literally, when the computer detects a threat, it shows a red symbol and sounds a horn. No threat and the symbol turn green, greeting the operators with a pleasant piano riff.

5.1.2 Advantages

- It is not affected by colour or transparency. Basically, the Ultrasonic Sensors transmit the sound off of the object, hence the colour and transparency have no effect on the radar reading.
- Any dark environments have no effect on this Arduino radar sensor's detection procedure. So, it can also use at night.
- Easy to design and low price. The ultrasonic sensors are available at the market with very cheap price.
- It has high frequency, high sensitivity, therefore, it can easily detect the external or deep objects.
- This radar sensor is not affected by dust, rain, snow, and many more.
- It has a self-cleaning system to continue running and less downtime.
- The Arduino Radar Sensor is easy to use. Also, it is completely safe during the operation to nearby objects, human or equipment.
- The Ultrasonic sensor can easily interface with any types of the microcontroller.

5.1.3 Limitations

- The Arduino Radar Sensor conduct sound to continue the work. So, it is not working in a vacuum as there is no air for the sound to travel through.
- A very soft fabric can absorb more sound waves. Therefore, it is hard to detect objects which are covered with soft fabric.

- If temperature changes of 5 to 10 degree or more then it is the effect on the sensing accuracy. Although this is true that there have many more temperature compensated sensors available.
- Another limitation is the detection range. This depends on which Ultrasonic sensor have used to make the Arduino Radar Sensor.
- While the radar using for inspection purpose, make sure it should be water resistive. Otherwise highly chances of damage.

5.1.3 Design Consideration

ULTRASONIC RADAR has been receiving increasing attention in recent years from researchers, practitioners, and funding agencies. The idea of making an Ultrasonic RADAR came as a part of a study carried out on the working and mechanism of "Automobiles of Future". Also, being an engineering student, we have always been curious about the latest ongoing technology in the world like Arduino, Raspberry Pi, Beaglebone boards. Hence, this time we were able to get a hold of one of the Arduino boards, we thought of making it big and a day to day application specific module that can be used and configured easily at any place and by anyone. Moreover, in this fast-moving world there is an immense need for the tools that can be used for the betterment of the mankind rather than devastating their lives. Hence, we decided to make some of the changes and taking the advantage of the processing capabilities of Arduino, we decided to make up the module more application specific. Hence, from the idea of the self-driving cars came the idea of self-parking cars. The main problem of the people in most of the countries is safety while driving. So, we came up with a solution to that by making use of this project to continuously scan the area for traffic, population, as well as offer protection of the vehicles at the same time to prevent accidents or minor scratches to the vehicle.

5.2 SECURITY

5.2.1 Security Through Radar Technology

The need for security is fundamental for us as people. With the increase in technical possibilities for everyday life, the demand for new solutions in the field of security technology

is increasing. Developments such as increasing crime rates, stricter legal provisions or requirements, growing floods of data or new challenges presented by technological trends require the development of established security measures and the use and promotion of innovative approaches.

Radar sensors successfully established themselves in the security industry years ago. Motion detectors in alarm systems, access controls and area monitoring use electromagnetic waves to secure premises, building complexes or properties. RADAR offers a wide range of solutions for various security requirements. For example, the selection of the optimum radar solution depends on whether the security measure is used to protect personal property, major events, road safety, surveillance of external installations or monitoring of entrance areas.

- **Alarm System**

Although radar-based alarm systems seem to function similarly to the principle of PIR sensors, they raise movement detection to the next level by gathering additional information. As soon as a person or a moving object appears in the detection area or is approaching the installation, an alarm is triggered by the radar motion sensor. Unlike ultrasound, infrared and infrasound technology, radar sensor systems are unaffected by environmental influences, pass through plastic and have a high range.

- **Combination of video and radar technology**

An innovative security concept is the use of synergy effects of two different technology approaches. Through the fusion of CCTV and a radar system, the disadvantages of the optical security system are compensated for by the advantages of high-frequency technology.

The combination of classic video surveillance for the near range with radar application for longer distance form a perfect team for area surveillance.

- **Perimeter surveillance**

Securing an open area, the surroundings of a building as well as fences or walls which surround the object to be secured is summed up under the term perimeter security. Virtually every security concept for critical outside areas includes the use of radar technology. Intruders can be detected and tracked early through the large detection area of the sensors.

- **Tracking radars**

Trackers allow the representation of movement patterns and behaviours of detected targets. These can provide information about the existing security risk. The information on position, speed, movement direction, distance and angle are helpful for predicting the position sequence of a detected object.

- **Access control**

Security measures such as installed access controls keep unauthorised persons or objects from accessing premises or buildings. Radar sensors monitor the opening and closing of gates, doors or barriers and provide information on the number or direction of movement of the recorded objects. This allows the permissible objects to be limited.

Radar allows object classification. For example, during access control with a barrier, a distinction can be made between cars and people.

5.3 FUTURE SCOPE

The first radar has been patented 110 years ago. Meanwhile the applications became numerous and the system concepts have been adopted to the available technologies. Typical applications are speed control, air traffic control, synthetic aperture radar, airborne and spaceborne missions, military applications and remote sensing. Research for medical radar applications is well progressing for breast cancer detection and tumour localization. Automobile radar for save and autonomous driving. Future scope of this design because of its security capacity. It can be used in many applications. This framework can also be developed or modified according to the rising needs and demand. We have represented a project on Ultrasonic RADAR for security system for human or object interference in a short range. The system has been successfully implemented and the aim is achieved without any deviation. There is a lot of future scope of this project because of its security capacity. It can be used in many applications. This project can also be developed or modified according to the rising needs and demand

5.4 CONCLUSION

In this project using Arduino board the radar system was implemented. It succeeds in helping to be widely used to help detect objects in different environments. In the future it can be updated to use for a larger range and advancements. Numerous advanced control methods gave designers to have more command over different advanced applications. In our paper, the recommended mapping method of whole system is assessed on small principles or scale. The field that we have chosen for our design “Radar System” is a very vast field and future scope of this technology is very high. We have tremendous applications in which radar system have been implemented or used. There is a lot of future scope of this design because of its security capacity. It can be used in many applications. This framework can also be developed or modified according to the rising needs and demand. We have represented a project on Ultrasonic RADAR for security system for human or object interference in a short range. The system has been successfully implemented and the aim is achieved without any deviation. There is a lot of future scope of this project because of its security capacity. It can be used in many applications. This project can also be developed or modified according to the rising needs and demand.