



---

## **Lab no 04: Radar System Prototype Servo motor and Ultrasonic sensor Interfaces**

The purpose of this Lab is to learn interfaces with servo motor and ultrasonic sensor. To do that, we are going to design an Arduino radar project using an Ultrasonic sensor for detection.

This Arduino radar project aims to achieve a radar system prototype based on an Arduino board that detects stationary and moving objects. When detecting an object, there will be an alarm in form of a flasher and buzzer. If the object is moving closer to the radar the alarm should indicate that by increasing the flash rate and the sound of the buzzer.

### **Parts: -**

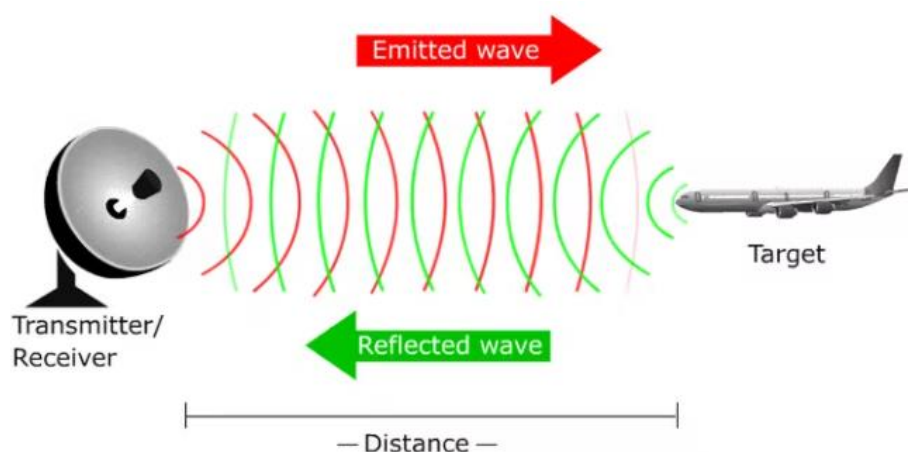
1. How does Radar Work?
2. What is Ultrasonic Sensor?
3. What is Servo Motor?
  - a) Controlling Servo Motor
4. Hardware Connection & Preparing the Fixture.
5. Radar System Arduino code.



## Part 1. How does Radar Work?

The word RADAR means Radio Detection and Ranging. Radar is an object detection system that uses microwaves to determine the range, altitude, direction, and speed of objects within about a 100-mile radius of their location.

The radar antenna transmits radio waves or microwaves that bounce off any object in its path. Due to this, we can easily determine the object in the radar range.



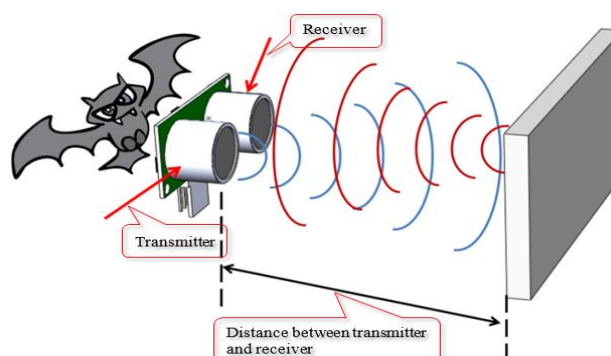
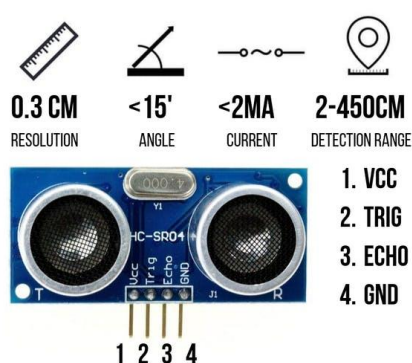
Radar is an electromagnetic sensor that is used to detect and locate an object. Radio waves or microwaves are radiated out from the radar into free space. Some of these waves are intercepted by reflecting objects. These intercepted radio waves hit the target and are reflected in many different directions. Some of these waves can be directed back toward the radar, where they are received and amplified. If these waves are received again at their origin, then it means an object is in the propagation direction.

The modern radar system is very advanced and used in highly diverse applications such as Air traffic control, Air-defense system, radar Astronomy, Antimissile system, Outer space Surveillance system, and many more.



## Part 2. What is Ultrasonic Sensor?

An ultrasonic sensor is a proximity sensor that is used to measure the distance of a target or object. It detects the object by transmitting ultrasonic waves and converts the reflected waves into an electrical signal. These sound waves travel faster than the speed of the sound that humans can hear.



It has two main components: the transmitter & receiver. The transmitter emits the sound using a piezoelectric crystal, and the receiver encounters the sound after it has traveled to and from the target.

For the calculation of the object distance, the sensor measures the time taken by the signal to travel between the transmission of the sound by the transmitter to the reflecting back towards the receiver. The formula for this calculation is,

$$D = \frac{1}{2} T \times C$$

Where, **D = distance**, **T = time**, **C = speed of sound which is 343 meters/second.**

These sensors are mostly found in automobile self-parking technology and anti-collision safety systems. Also, used in robotic obstacle detection systems, manufacturing technology, and many more. To know more about the ultrasonic sensor.



### Part 3. What is Servo Motor?

The servo motor is a simple DC motor that can be controlled for specific angular rotation with the help of additional servomechanism. This motor will only rotate as much we want and then stop. The servo motor is a closed-loop mechanism that uses positional feedback to control the speed and position.

This closed-loop system includes a control circuit, servo motor, shaft, potentiometer, drive gears, amplifier, and either an encoder or resolver.



The servo motor is unlike a standard electric motor which starts and stops according to the power input. According to the signal, the servo motor will work.

Nowadays, servo motors are widely used in industrial and robotics applications. They are also commonly seen in remote-controlled toy cars, RC planes, and in the CD or DVD player. Besides these, we see hundreds of applications in our daily life that use a servo motor. To know more about the servo motor.

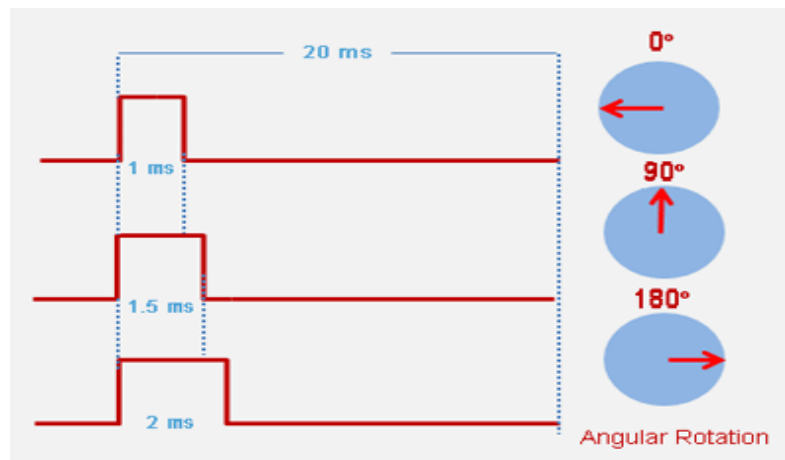


### Part 3. a) Controlling Servo Motor

All motors have three wires coming out of them. Out of which two will be used for Supply (positive and negative) and one will be used for the signal that is to be sent from the MCU.

Servo motor is controlled by PWM (Pulse with Modulation) which is provided by the control wires. There is a minimum pulse, a maximum pulse, and a repetition rate. Servo motor can turn 90 degrees from either direction from its neutral position. The servo motor expects to see a pulse every 20 milliseconds (ms) and the length of the pulse will determine how far the motor turns. For example, a 1.5ms pulse will make the motor turn to the 90° position, such as if the pulse is shorter than 1.5ms shaft moves to 0°, and if it is longer than 1.5ms then it will turn the servo to 180°.

Servo motor works on the **PWM(Pulse width modulation)** principle, which means its angle of rotation is controlled by the duration of applied pulse to its Control PIN. Basically, **the servo motor is made up of a DC motor which is controlled by a variable resistor (potentiometer) and some gears.** A high-speed force of DC motor is converted into torque by Gears. We know that **WORK= FORCE X DISTANCE**, in DC motor Force is less and distance (speed) is high and in Servo, force is High, and distance is less. The potentiometer is connected to the output shaft of the Servo, to calculate the angle and stop the DC motor at the required angle.



Servo motor can be rotated from 0 to 180 degrees, but it can go up to 210 degrees, depending on the manufacturing. This degree of rotation can be controlled by applying the Electrical Pulse of proper width, to its Control pin. Servo checks the pulse every 20 milliseconds. The pulse of 1 ms (1 millisecond) width can rotate the servo to 0 degrees, 1.5ms can rotate to 90 degrees (neutral position) and 2 ms pulse can rotate it to 180 degrees.

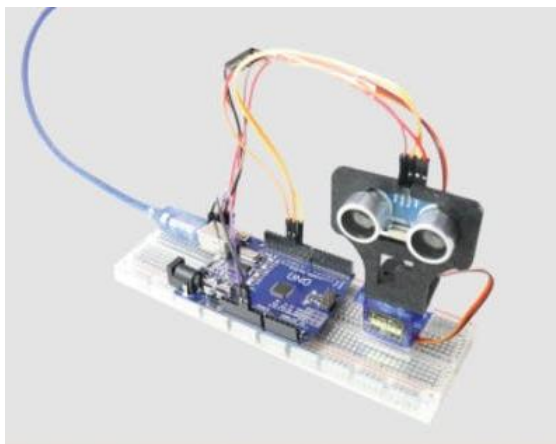
All servo motors work directly with your +5V supply rails but we have to be careful about the amount of current the motor would consume if you are planning to use more than two servo motors a proper servo shield should be designed.



## Part 4. Hardware Connection & Preparing the Fixture.

### a) Hardware Connection

The below image shows the connection of the Arduino radar project. The connections are very simple. Here, we have interfaced the ultrasonic sensor and servo motor with an Arduino Uno. The connection details are,



Arduino Uno	Ultrasonic Sensor	Servo Motor
Vcc	Vcc	Vcc
Gnd	Gnd	Gnd
D10	Trig	-
D11	Echo	-
D12	-	Signal

### b) Preparing the Fixture

After making the connections, to build the fixture of this construction, we can use a mounting bracket as shown in the below image to fix the ultrasonic sensor.

This step is optional. You can also make a simple structure with cardboard to mount the ultrasonic sensor with a Servo motor.





## Part 5. Radar system Arduino code

```
// Includes the Servo library
#include <Servo.h>.
// Defines Trig and Echo pins of the Ultrasonic Sensor
const int trigPin = 10;
const int echoPin = 11;
const int buzzer=7;
const int flash= 6;
// 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
  pinMode(buzzer, OUTPUT);
  pinMode(flash, OUTPUT);
  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(200);
    distance = calculateDistance();// Calls a function for
calculating the distance measured by the Ultrasonic sensor for
each degree
    Serial.print(distance); // Sends the distance value into the
Serial Port
    Serial.println("."); // Sends addition character right next
to the previous value needed later in the Processing IDE for
indexing
    if (distance < 20 &&distance > 10)
    {
      tone(buzzer,100);
      digitalWrite(flash, HIGH);
      delay(70);
      digitalWrite(flash, LOW);
    }
  }
}
```





```
        delay(70);
    }
else if(distance<10)
{
    tone(buzzer,500);
    digitalWrite(flash, HIGH);
    delay(20);
    digitalWrite(flash, LOW);
    delay(20);
}
else
{
    digitalWrite(flash, LOW);
    digitalWrite(buzzer, LOW);
    noTone(buzzer);
}
}
// Repeats the previous lines from 165 to 15 degrees
for(int i=165;i>15;i--){
myServo.write(i);
delay(200);
distance = calculateDistance();
Serial.print(distance);
Serial.println(".");
    if (distance < 20 &&distance > 10)
{
    tone(buzzer,100);
    digitalWrite(flash, HIGH);
    delay(70);
    digitalWrite(flash, LOW);
    delay(70);

}
else if(distance<10)
{
    tone(buzzer,500);
    digitalWrite(flash, HIGH);
    delay(20);
    digitalWrite(flash, LOW);
    delay(20);
}
```



```
else
{
    digitalWrite(flash, LOW);
    digitalWrite(buzzer, LOW);
    noTone(buzzer);
}
}
// Function for calculating the distance measured by the
Ultrasonic sensor
int calculateDistance(){

    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); // Reads the echoPin,
returns the sound wave travel time in microseconds
    distance= duration*0.034/2;
    return distance;
}
```