IoT Project using Ultrasonic Sensor HC-SR04 and Arduino to distance calculation using Processing App

Let's build an IoT project using Ultra Sonic HC-SR04 and Arduino (Arduino UNO) to calculate distance between Ultra Sonic HC-SR04 device and an object. In this project, we will use a Processing app to display the distance between Ultra Sonic device and object on the Laptop's (Monitor) screen.

The working principle of Arduino-Bluetooth Module

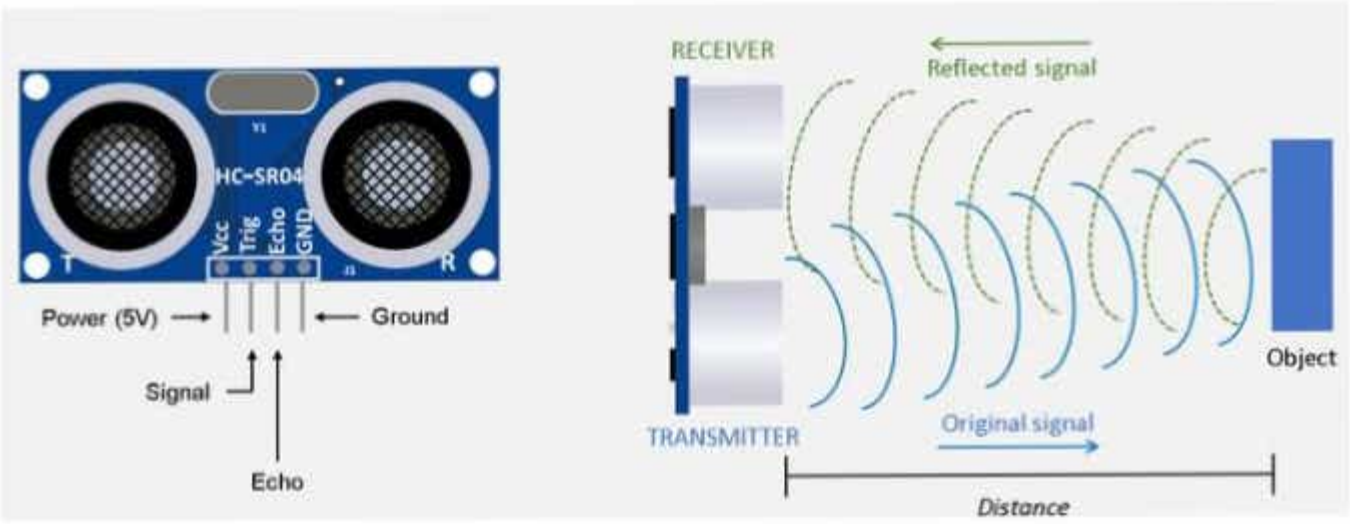
The Ultra Sonic HC-SR04 emits ultrasound at 40,000Hz that travels in the air. If there is an object or obstacle in its path, then it collides and bounces back to the Ultra Sonic module.

The formula **distance = speed*time** is used to calculate the distance.

Suppose, an object is placed at a distance of 10 cm away from the sensor, the speed of sound in air is 340 m/s or 0.034 cm/µs. It means the sound wave needs to travel in 294 µs. But the Echo pin double the distance (forward and bounce backward distance). So, to get the distance in cm multiply the received travel time value with echo pin by 0.034 and divide it by 2.

The distance between Ultra Sonic HC-SR04 and an object is:

$$\text{distance} = \frac{\text{speed} * \text{time}}{2}$$



Speed of sound:

$$speed = 340\ m/s = 0.034\ cm/\mu s$$

$$time = distance/speed$$

$$time = \frac{10}{0.034}\mu s = 294\ \mu s$$

$$distance = \frac{speed * time}{2}$$

$$distance = \frac{0.034 * 294}{2}$$

Open Arduino IDE and paste the following code.

```
#include <Mouse.h>


const int trigpin= 8;
const int echopin= 7;
long duration;
int distance;
void setup(){
  pinMode(trigpin,OUTPUT);
  pinMode(echopin,INPUT);
  Serial.begin(9600);
}


void loop(){
  digitalWrite(trigpin,HIGH);
  delayMicroseconds(10);
  digitalWrite(trigpin,LOW);
```

Digital circuit diagram

Ultrasonic Sensor HC-SR04 Arduino UNO

VCC ----------> 5v

Trig ----------> Pin 8

Echo ----------> Pin 7

GND ----------> GND

```
import processing.serial.*;

Serial myPort;

String data="" ;

PFont  myFont;


void setup(){

 size(1366,900); // size of processing window

 background(0);// setting background color to black

 myPort = new Serial(this, "COM3", 9600);

 myPort.bufferUntil('\n');

}


void draw(){

 background(0);

 textAlign(CENTER);

 fill(255);

 text(data,820,400);

 textSize(100);
```

```
  text("          Distance :      cm",450,400);
  noFill();
  stroke(#4B5DCE);
}

void serialEvent(Serial myPort){
   data=myPort.readStringUntil('\n');
}
```

Initially, it displays the 0 as no activity in the process.