# **Technical Training Manual for Ultrasonic Sensor with Arduino Code**

Introduction to Ultrasonic Sensor:

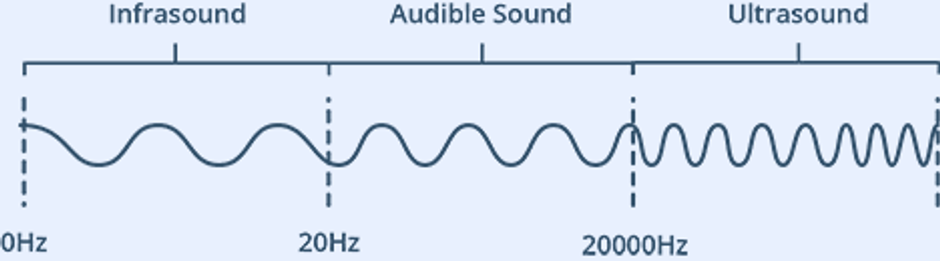
An ultrasonic sensor is a type of distance measuring device that uses sound waves to detect the distance of an object. It emits high-frequency sound waves and measures the time it takes for the sound waves to bounce back after hitting an object. By knowing the speed of sound in the medium, the sensor can calculate the distance to the object.

In this training manual, we will explore how to use an ultrasonic sensor with Arduino to measure distances and integrate it into your projects.

💡

What is Ultrasound?

Ultrasound is a high-pitched sound wave whose frequency exceeds the audible range of human hearing.

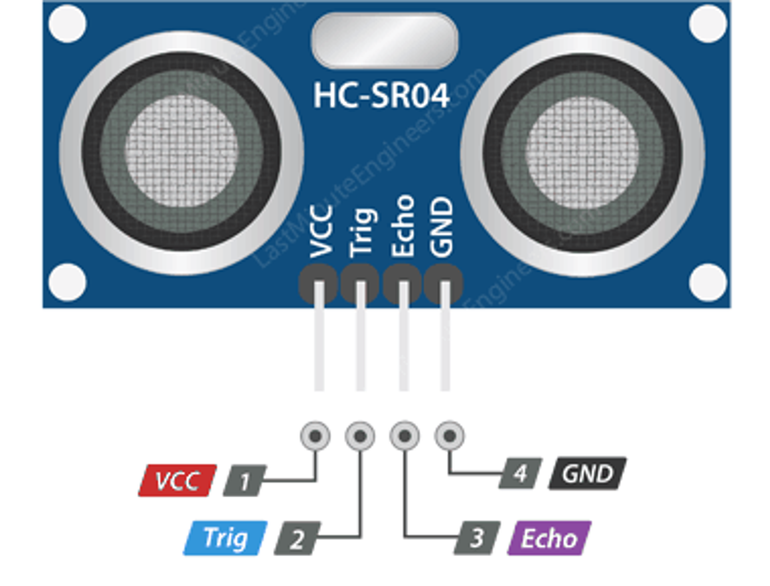


Humans can hear sound waves that vibrate in the range of about 20 times a second (a deep rumbling noise) to 20,000 times a second (a high-pitched whistle). However, ultrasound has a frequency of more than 20,000 Hz and is therefore inaudible to humans.

#### **Technical Specifications**

Here are the specifications:

| Operating Voltage | DC 5V |
| --- | --- |
| Operating Current | 15mA |
| Operating Frequency | 40KHz |
| Max Range | 4m |
| Min Range | 2cm |
| Ranging Accuracy | 3mm |
| Measuring Angle | 15 degree |
| Trigger Input Signal | 10µS TTL pulse |
| Dimension | 45 x 20 x 15mm |

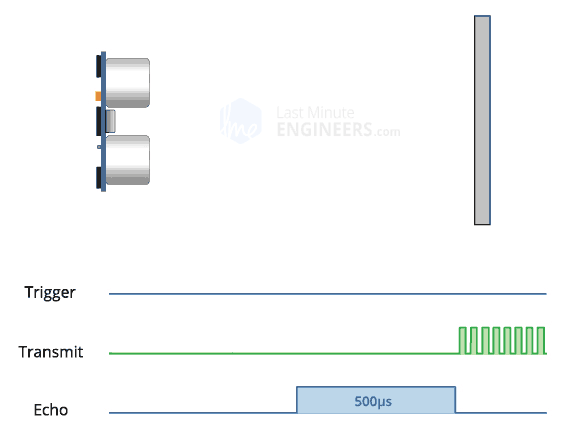


VCC supplies power to the HC-SR04 ultrasonic sensor. You can connect it to the 5V output from your Arduino.

Trig (Trigger) pin is used to trigger ultrasonic sound pulses. By setting this pin to HIGH for 10µs, the sensor initiates an ultrasonic burst.

Echo pin goes high when the ultrasonic burst is transmitted and remains high until the sensor receives an echo, after which it goes low. By measuring the time the Echo pin stays high, the distance can be calculated.

GND is the ground pin. Connect it to the ground of the Arduino.



#### **Calculating the Distance**

The width of the received pulse is used to calculate the distance from the reflected object. This can be worked out using the simple distance-speed-time equation we learned in high school.

💡

Distance = Speed x Time

we know the speed. Of course it’s the speed of sound! It is 340 m/s

Distance = 0.034 cm/µs x Time

📎

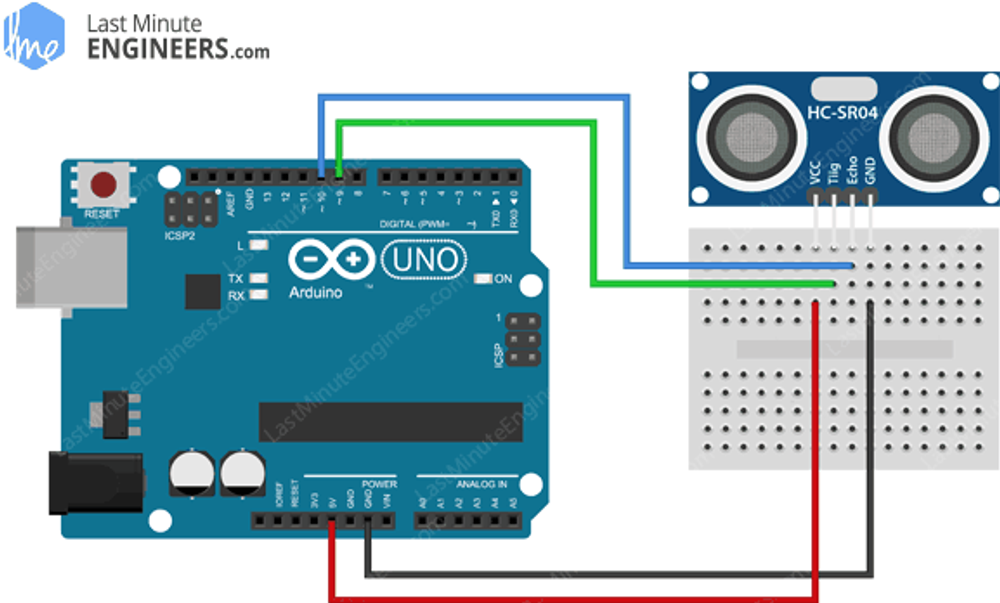
But we’re not done yet! Remember that the echo pulse indicates the time it takes for the signal to be sent and reflected back. So to get the distance, you have to divide your result by two.Time = Duration / 2

⚙

Time = Duration / 2

| HC-SR04 Sensor |  | Arduino |
| --- | --- | --- |
| VCC |  | 5V |
| Trig |  | 9 |
| Echo |  | 10 |
| GND |  | GND |

When you are done you should have something that looks similar to the image shown below.



Components Required:

Arduino board (e.g., Arduino UNO)

Ultrasonic sensor (HC-SR04 or similar)

Breadboard and jumper wires

USB cable for Arduino

Computer with Arduino IDE installed (Download from:<https://www.arduino.cc/en/software>)

Circuit Connection:

Connect VCC of the ultrasonic sensor to the 5V pin of the Arduino.

Connect GND of the ultrasonic sensor to the GND pin of the Arduino.

Connect Trig (trigger) of the ultrasonic sensor to a digital pin (e.g., D7) of the Arduino.

Connect Echo of the ultrasonic sensor to another digital pin (e.g., D6) of the Arduino.

Arduino Code:

Now, let's write the Arduino code to read data from the ultrasonic sensor and calculate the distance.

// Define pins, where arduino is connected with the sensor

const int trigPin = 7;

const int echoPin = 6; // Variables for duration and distance

long duration;

int distance;

void setup()

{

// Set pin modes

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

// Initialize Serial Monitor

Serial.begin(9600);

}

void loop()

{

// Clear the trigger pin

digitalWrite(trigPin, LOW);

delayMicroseconds(2); // Activate the trigger pulse for 10 microseconds digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// Measure the duration of the echo pulse

duration = pulseIn(echoPin, HIGH);

// Calculate the distance in centimeters

distance = duration \* 0.034 / 2;

// Print the distance to Serial Monitor

Serial.print("Distance: ");

Serial.print(distance);

Serial.println(" cm");

// Add a small delay before the next reading

delay(500);

}

### Uploading the Code:

Open the Arduino IDE on your computer.

Connect the Arduino board to your computer using the USB cable.

Select the correct board and port from the "Tools" menu.

Copy and paste the above code into the Arduino IDE.

Click the "Upload" button to upload the code to your Arduino board.

### Testing:

Once the code is uploaded successfully, open the Serial Monitor from the Arduino IDE (Ctrl + Shift + M). You should see the distance readings in centimeters printed continuously.

Calibration and Adjustment:

The distance measured by the ultrasonic sensor might not be accurate in all cases. Factors like temperature, humidity, and surface properties can affect the results. Calibration and adjustments might be necessary depending on your specific application.

### Conclusion:

In this training manual, we learned how to use an ultrasonic sensor with an Arduino board to measure distances. You can now integrate this sensor into various projects like obstacle avoidance robots, distance measurement devices, or smart sensing applications. Experiment with different projects to get a better understanding of ultrasonic sensors and their capabilities. Happy tinkering!