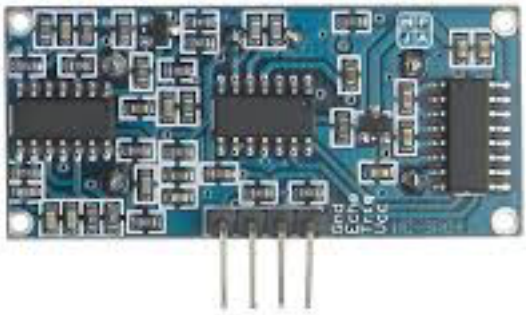
Sensor: Ultrasonic Ranging

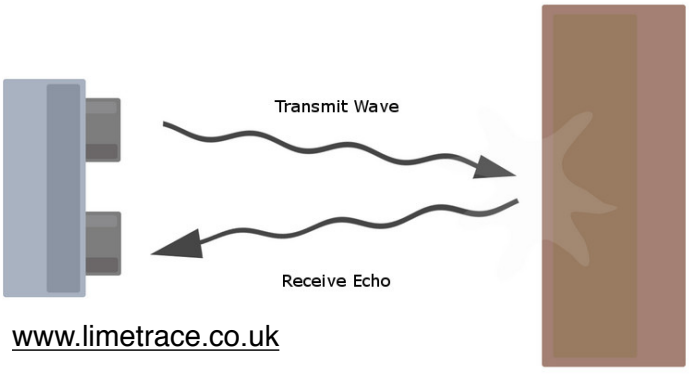


mpja.com

This sensor also uses the BIS0001 microcontroller internally, just as the motion detector did. It is also fairly simple to use, but quite complex in its actual design.

Essentially the two cans on the front of the unit act as a speaker and a microphone. One sends a sound and the other then listens for that sound to reflect back. The sounds are ultrasonic, meaning that they are above the range of human hearing (i.e. well over 20kHZ). Higher frequency sounds are more reflective than those of a lower frequency, and they occur less often naturally, they are also more directional in nature. Hence why a subwoofer, which makes low frequency sounds, can be placed anywhere in the room, but the placement of left, right, center, etc. speakers requires more thought and adjustment.

Some distance sensors use InfraRed light, though the many other IR light sources may cause interference. Lasers and RF are used in more high-precision units.



www.limetrace.co.uk

By carefully timing how long it took to receive the echo, the sensor can determine the distance from it to the nearest object, with a resolution of around 3mm. This is done for a range of around 1" up to 12' or so. Certain materials with high sound absorption, or unusual reflectivity may cause abnormal readings.

To use the unit requires 4 wires. The usual power and ground, but both an In and Out — labeled Trig and Echo. When the Trig line is brought HIGH by our Nano, the sensor will send a ping and calculate the distance to the nearest object. In order to communicate that distance back to the Nano, it brings the Echo pin HIGH when it sends the ping, and back to LOW when it receives it.

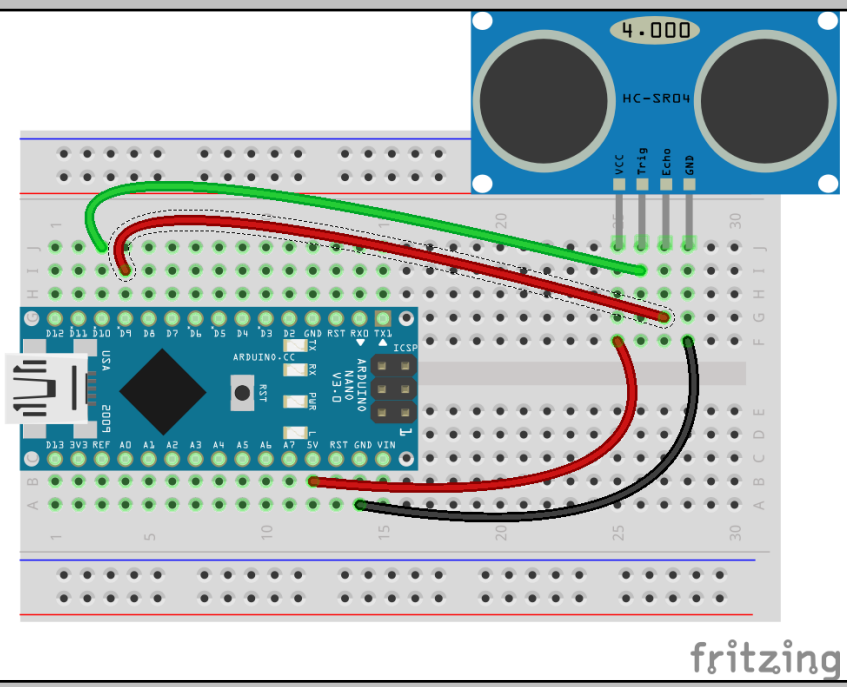
If you measure the time Echo was HIGH and multiply that by the speed of sound, you have the total distance the sound traveled. Since it went out and back, if you divide that by 2, you have the distance of the sensor from the object. You could use a temperature and pressure compensated value for the speed of sound for increased accuracy, though the device's internal timing limits its overall accuracy. Alternatively you can just take the HIGH time and divide by 148 for the distance in inches, or by 58 yields for centimeters.

Since sound can go pretty far in just a microsecond, after setting Trig HIGH, we need to start reading Echo immediately, and we need to read as often as possible. Once it goes HIGH we need to record the time in microseconds and continue reading as quickly as possible. Then, once it finally goes LOW, record that time in microseconds. Subtract the last recorded time from the first, and we have the delay, and thus the distance.

While this wouldn't be too terribly difficult, as with the other sensors we have used, there exists some software to simplify the usage even further.

We will be using the NewPing library, which handles all of that stuff and more. This exercise involves only a trivial amount of actual programming, but you will need to consult the documentation for the NewPing library in order to find out how to complete the first simple task.

Experiment: Who goes there?

Components	Wiring Diagram
✓Battery Pack ✓Breadboard ✓HC-SR04 ultrasonic ranging sensor	

Connection Instructions

You may choose to plug the sensor directly into the breadboard and use M to M jumpers, or you may use M to F jumpers to connect the sensor to be able to move it. The sensor will need to remain stable while in use, and you will need to be able to move objects in front of it at various distances to test your code.

Use pin 10 for the Trig connection and pin 9 for the Echo connection. The standard 5V and GND are also required.

Sketch(es)	basicDistance.ino

Analysis Questions

Could these sensors be used to determine the relative shape of an object in front of them? If so how?

Can you think of any possible uses for such a sensor?

What challenges would using a laser present vs ultrasonics?

Programming Tasks

Compile the provided program and, using the Serial monitor, verify that it works and senses distance. You will probably notice that even when the object in front of the sensor is not moving, that the distance may vary slightly.

Locate and read the documentation for the NewPing library to find an alternative function that may provide more stable results, possibly by taking several readings over a short period of time. Modify the code to use this new solution and see if the results are more stable.

Time permitting, change the original and/or your updated code to display fractional CM readings.