

## LAMPIRAN

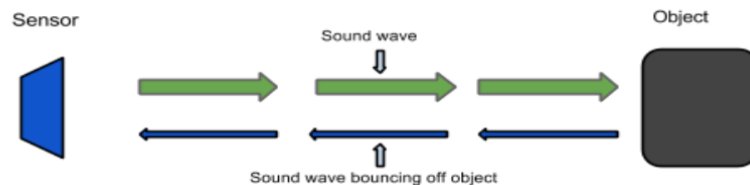
### 1. Data Sheets Sensor Ultrasonik

The purpose of this file is to explain how the HC-SR04 works. It will give a brief explanation of how ultrasonic sensors work in general. It will also explain how to wire the sensor up to a microcontroller and how to take/interpret readings. It will also discuss some sources of errors and bad readings.

1. How Ultrasonic Sensors Work
2. HC-SR04 Specifications
3. Timing chart, Pin explanations and Taking Distance Measurements
4. Wiring HC-SR04 with a microcontroller
5. Errors and Bad Readings

#### 1. How Ultrasonic Sensors Work

Ultrasonic sensors use sound to determine the distance between the sensor and the closest object in its path. How do ultrasonic sensors do this? Ultrasonic sensors are essentially sound sensors, but they operate at a frequency above human hearing.



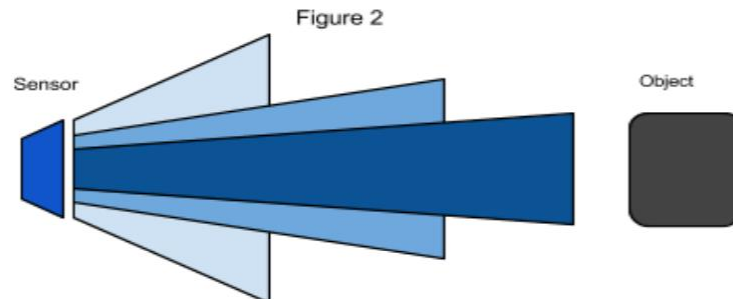
The sensor sends out a sound wave at a specific frequency. It then listens for that specific sound wave to bounce off of an object and come back (Figure 1). The sensor keeps track of the time between sending the sound wave and the sound wave returning. If you know how fast something is going and how long it is traveling you can find the distance traveled with equation 1.

**Equation 1.**  $d = v \times t$

The speed of sound can be calculated based on the a variety of atmospheric conditions, including temperature, humidity and pressure. Actually calculating the distance will be shown later on in this document.

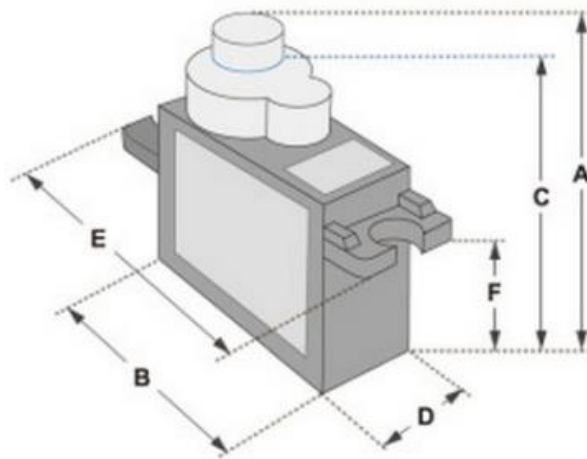
It should be noted that ultrasonic sensors have a cone of detection, the angle of this cone varies with distance, Figure 2 show this relation. The ability of a sensor

to detect an object also depends on the objects orientation to the sensor. If an object doesn't present a flat surface to the sensor then it is possible the sound wave will bounce off the object in a way that it does not return to the sensor.



## 2. Servo Motor SG90

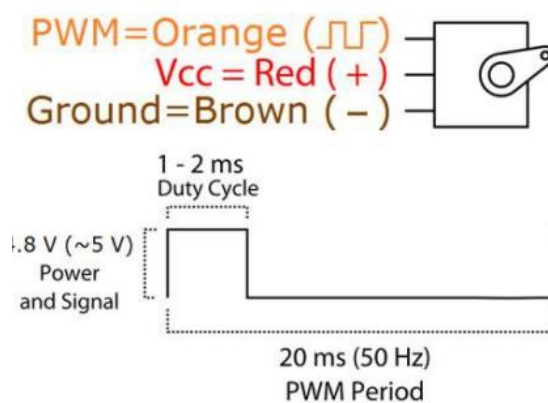
Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.



Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.

| Dimension & Specifications |
|----------------------------|
| A (mm) : 32                |
| B (mm) : 23                |

|                      |
|----------------------|
| C (mm) : 28,5        |
| D (mm) : 12          |
| E (mm) : 32          |
| F (mm) : 19,5        |
| Speed (sec) : 0,1    |
| Torque (kg-cm) : 2,5 |
| Weight (g) : 14,7    |
| Voltage : 4,8 – 6    |



### 3. Arduino Wemos D1 ESP8266

The WeMos D1 is a ESP8266 WiFi based board that uses the Arduino layout with operating voltage of 3.3V. Note that this is NOT an Arduino board, it only uses the Arduino Uno layout for the board design! In summary, the board is controlled by the ESP8266 chip (a 32-Bit processor) and has a larger flash memory compared to an Arduino Uno. It consists of 11 digital I/O pins and 1 analogue (input) pin. The board can be connected using a Micro-B type USB cable. (Aka “Android Cable”).



**Brief Data:**

1. Microcontroller: ESP-8266EX.
2. Operating Voltage: 3.3Vdc.
3. Digital I/O Pins: 11.
4. Analog Input Pins: 1.
5. Clock Speed: 80MHz/160MHz.
6. Flash: 4M bytes.
7. Board Dimension: 68.6mm x 53.4mm.
8. Weight: 25g.