**Distance measure display**

Introduction

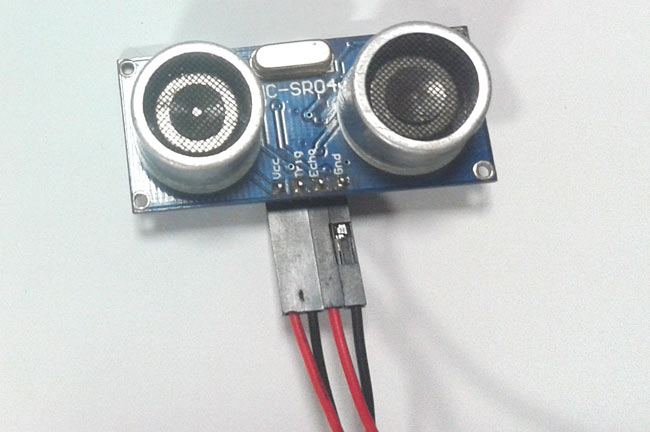
In this project, we have used the **HC-SR04** **Ultrasonic Sensor with Arduino** to determine the distance of an obstacle from the sensor. The basic principle of ultrasonic distance measurement is based on ECHO. When sound waves are transmitted in the environment then waves return back to the origin as ECHO after striking on the obstacle. So we only need to calculate the traveling time of both sounds means outgoing time and returning time to origin after striking on the obstacle. As the speed of the sound is known to us, after some calculation we can calculate the distance. We are going to use this same technique for this **Arduino distance measurement**project.

**VCC** is the power supply for the HC-SR04 Ultrasonic distance sensor which we connect the 5V pin on the Arduino.

**Trig** (Trigger) pin is used to trigger the ultrasonic sound pulses.

**Echo** pin produces a pulse when the reflected signal is received. The length of the pulse is proportional to the time it took for the transmitted signal to be detected.

**GND** should be connected to the ground of Arduino.



Components

1. Arduino Uno
2. Ultrasonic sensor Module
3. 16x2 LCD
4. Scale
5. Bread board
6. 9 volt battery
7. Connecting wires

Application

 measuring liquid level, checking proximity and even more popularly in automobiles to assist in self-parking or anti-collision systems. Previously we have also build many [Ultrasonic Sensor projects](https://circuitdigest.com/tags/ultrasonic-sensor) like [water level detecting](https://circuitdigest.com/microcontroller-projects/water-level-indicator-project-using-arduino), [Ultrasonic Radar](https://circuitdigest.com/microcontroller-projects/arduino-radar-using-android-and-ultrasonic-sensor)

Objective

During this activity ,you will help students to achieve following objectives

1. Understanding the principle and operation of ultrasonic distance sensor

2. Design algorithm and flowchart to detect obstacle and get alerted

3. Programming ultrasonic distance sensor using Arduino uno

4. Interfacing ultrasonic distance sensor withArduino uno

Algorithm

1. initialise trigger signal

2. initialise echo signal

3. initialise LCD display as output

4. send trigger signal continusly

5.check for echo signal if echo signal not receive then continue

6. if echo signal receives means there is obstacle ,read the time and distance

7. display time and distance to the LCD

8.end

Flowchart

Programming

#include <LiquidCrystal.h>  
   
#define trigger 18  
#define echo 19  
   
LiquidCrystal lcd(2,3,4,5,6,7);  
   
float time=0,distance=0;  
   
void setup()  
{  
 lcd.begin(16,2);  
 pinMode(trigger,OUTPUT);  
 pinMode(echo,INPUT);  
 lcd.print(" Ultra sonic");  
 lcd.setCursor(0,1);  
 lcd.print("Distance Meter");  
 delay(2000);  
 lcd.clear();  
 lcd.print(" Circuit Digest");  
 delay(2000);  
}  
   
void loop()  
{  
 lcd.clear();  
 digitalWrite(trigger,LOW);  
 delayMicroseconds(2);  
 digitalWrite(trigger,HIGH);  
 delayMicroseconds(10);  
 digitalWrite(trigger,LOW);  
 delayMicroseconds(2);  
 time=pulseIn(echo,HIGH);  
 distance=time\*340/20000;  
 lcd.clear();  
 lcd.print("Distance:");  
 lcd.print(distance);  
 lcd.print("cm");  
 lcd.setCursor(0,1);  
 lcd.print("Distance:");  
 lcd.print(distance/100);  
 lcd.print("m");  
 delay(1000);  
}

Hardware

Instruction

1. Connect VCC and GND OF sensor module to 5 v supply pin and ground pin of arduino
2. Connect trigger pin to analog input pin 18 and echo signal to pin 19
3. Connect potentiometer vcc and ground connection to arduino and pot variable resistor pin to pin 3 of LCD
4. Connect register select pin to digital pin 2 and enble pin to ground
5. Connect data pins of LCD (4,5,6,7) to the digital input pin (D7,D6,D5,D4)
6. Connect vcc and GND pin of LCD to arduino board

