**EMBEDDED SYSTEMS**

**(EL-419)**

# LABORATORY MANUAL

# Spring 2019



**Engr. Aneela Sabir**

## Interfacing Ultrasonic Sensors with Arduino Uno

**(LAB # 03)**

Student Name: \_\_\_\_\_\_Kamran,Moiz,Aqib\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Roll No: \_\_\_i140420,i15-402,443\_\_\_\_\_\_\_\_\_\_\_\_\_ Section: \_\_\_A\_

Date performed: \_\_\_\_\_13/2\_\_\_\_\_\_\_\_, 2019

**MARKS AWARDED: \_\_\_\_\_\_\_\_ / 10**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ **NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES, ISLAMABAD**

|  |  |  |  |
| --- | --- | --- | --- |
| Prepared by: | Engr. Ahsan Khan |  | Version: 1.0.2 |
| Last Edited by: | Engr. Aneela Sabir |  | February 5, 2019 |
| Verified by: | Dr. Durdana Habib |  | Updated: Spring 2019 |

**Lab # 04: Interfacing Ultrasonic Sensors with Arduino Uno**

#### Lab Objective:

The purpose of this lab is to interface different sensors with microcontrollers.

**Equipment Required:**

* Arduino Uno/Mbed LPC1768
* Ultrasonic sensors – HC-SR04
* Jumper Wires
* Breadboard
* DC Motors
* L298 - Motor driver IC

**Introduction:**

An Ultrasonic Sensor is a device that measures distance to an object using **Sound Waves**. It works by sending out a sound wave at ultrasonic frequency and waits for it to bounce back from the object. Then, the time delay between transmission of sound and receiving of the sound is used to calculate the distance.

It is done using the formula **Distance = (Speed of sound \* Time delay) / 2**

We divide the distance formula by 2 because the sound waves travel a round trip i.e. from the sensor and back to the sensor which doubles the actual distance. The HC-SR04 is a typical ultrasonic sensor which is used in many projects such as obstacle detector and electronic distance measurement tapes. In this lab, you have to interface the HC-SC04 with an Arduino Uno.

**HC-SR04 Sonar Sensor:**

The HC-SR04 is an ultrasonic ranging module. This economical sensor provides 2cm to 400cm of non-contact measurement functionality with a ranging accuracy that can reach up to 3mm. Each HC-SR04 module includes an ultrasonic transmitter, a receiver and a control circuit.

There are **Four Pins**on the HC-SR04. They are :

* Vcc (5V supply)
* Gnd (Ground)
* Trig (Trigger)
* Echo (Receive)

The **key features** to be noted are:

* Operating Voltage: 5V DC
* Operating Current: 15mA
* Measure Angle: 15°
* Ranging Distance: 2cm - 4m

**Ultrasonic sensor connections with Arduino Uno:**

The connections are as follows:

Vcc to 5V Pin of the Arduino.

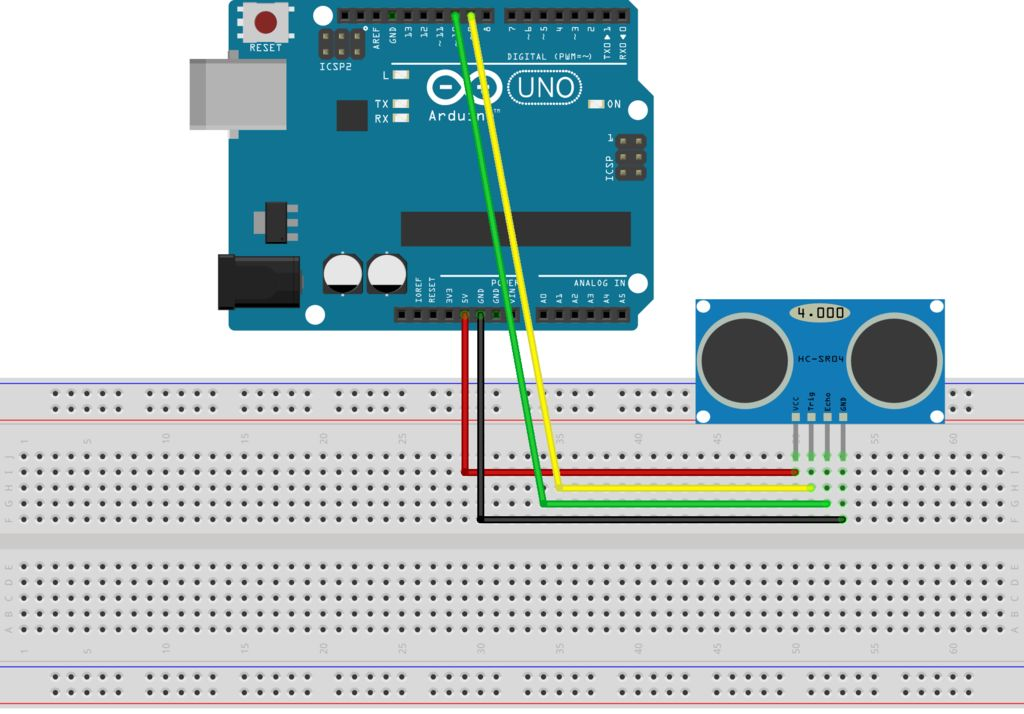
Gnd to Gnd Pin of the Arduino.

Trig to Digital Pin 9.

Echo to Digital Pin 10.

Refer the schematics for more clarity on the connections. Few things to remember while building the circuit

1. Avoid placing the sensor on metal surfaces to avoid short circuits which might burn the sensor.
2. It is recommended to put electrical tape on the back side of the sensor.
3. You can also directly connect the Ultrasonic sensor to the Arduino with jumper wires directly.

****

**Arduino Code:**

/\*

\* Ultrasonic Sensor HC-SR04 interfacing with Arduino.

\*/

// defining the pins

const int trigPin = 9;

const int echoPin = 10;

// defining variables

long duration;

int distance;

void setup() {

pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin, INPUT); // Sets the echoPin as an Input

Serial.begin(9600); // Starts the serial communication

}

void loop() {

// Clears the trigPin

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds

duration = pulseIn(echoPin, HIGH);

// Calculating the distance

distance= duration\*0.034/2;

// Prints the distance on the Serial Monitor

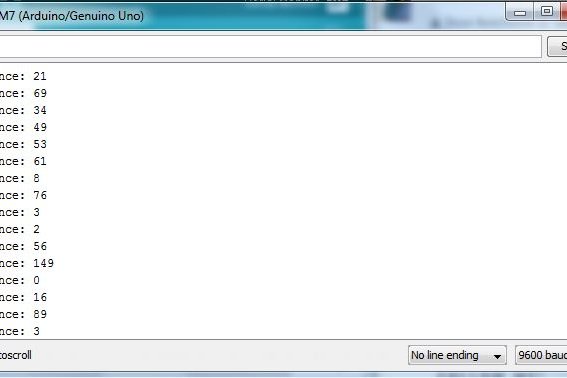
Serial.print("Distance: ");

Serial.println(distance);

}

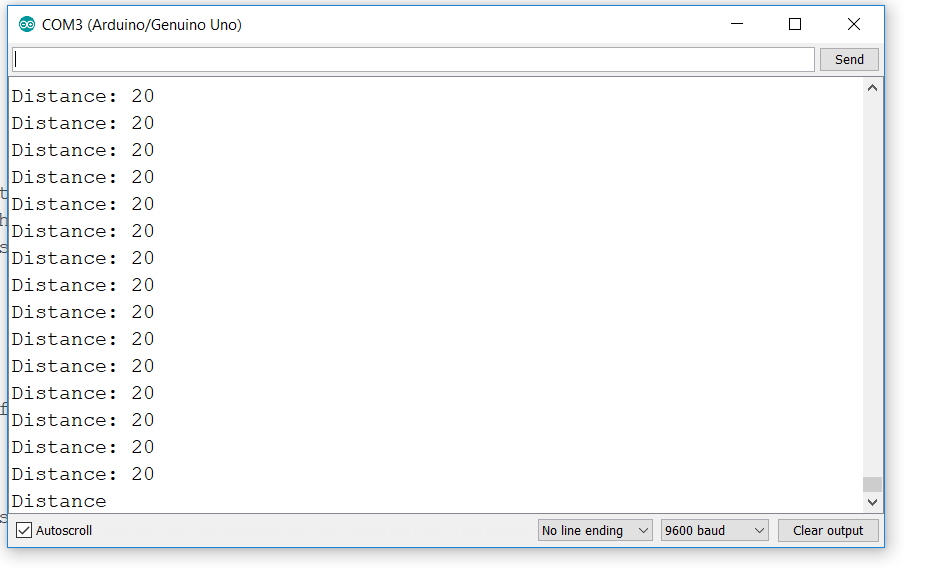
**Uploading and testing code:**

Connect the Arduino to your computer and upload the code. Once you've uploaded the code, the board will begin to transmit data to the computer. You will know this when you see the TX LED on the Arduino blinking each time it transmits a data. Now if you open the Serial Monitor, you'll see the distance being displayed.



**Task 4.1**

Interface HCSR-04 with Arduino Uno. Read the values of sonar sensor and display it on Serial Monitor in Arduino IDE.



**Task 4.2**

Design automatic door opening using Ultrasonic Sensor HC-SR04. You can use LED for the indication of door opening.

\*

\* Ultrasonic Sensor HC-SR04 interfacing with Arduino.

\*/

// defining the pins

const int trigPin1 = 9;

const int echoPin1 = 10;

const int trigPin2 = 6 ;

const int echoPin2 = 5;

const int led=4;

// defining variables

long duration1;

int distance1;

long duration2;

int distance2;

void setup() {

pinMode(trigPin1, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin1, INPUT); // Sets the echoPin as an Input

pinMode(trigPin2, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin2, INPUT); // Sets the echoPin as an Input

Serial.begin(9600); // Starts the serial communication

}

void loop() {

// Clears the trigPin

digitalWrite(trigPin1, LOW);

digitalWrite(trigPin2, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin1, HIGH);

digitalWrite(trigPin2, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin1, LOW);

duration1 = pulseIn(echoPin1, HIGH);

digitalWrite(trigPin2, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds

duration2 = pulseIn(echoPin2, HIGH);

// Calculating the distance

distance1= duration1\*0.034/2;

distance2= duration2\*0.034/2;

// Prints the distance on the Serial Monitor

Serial.print("Distance: ");

Serial.print(distance1);

Serial.print(" ");

Serial.println(distance2);

if (distance1< 10 || distance2< 10 )

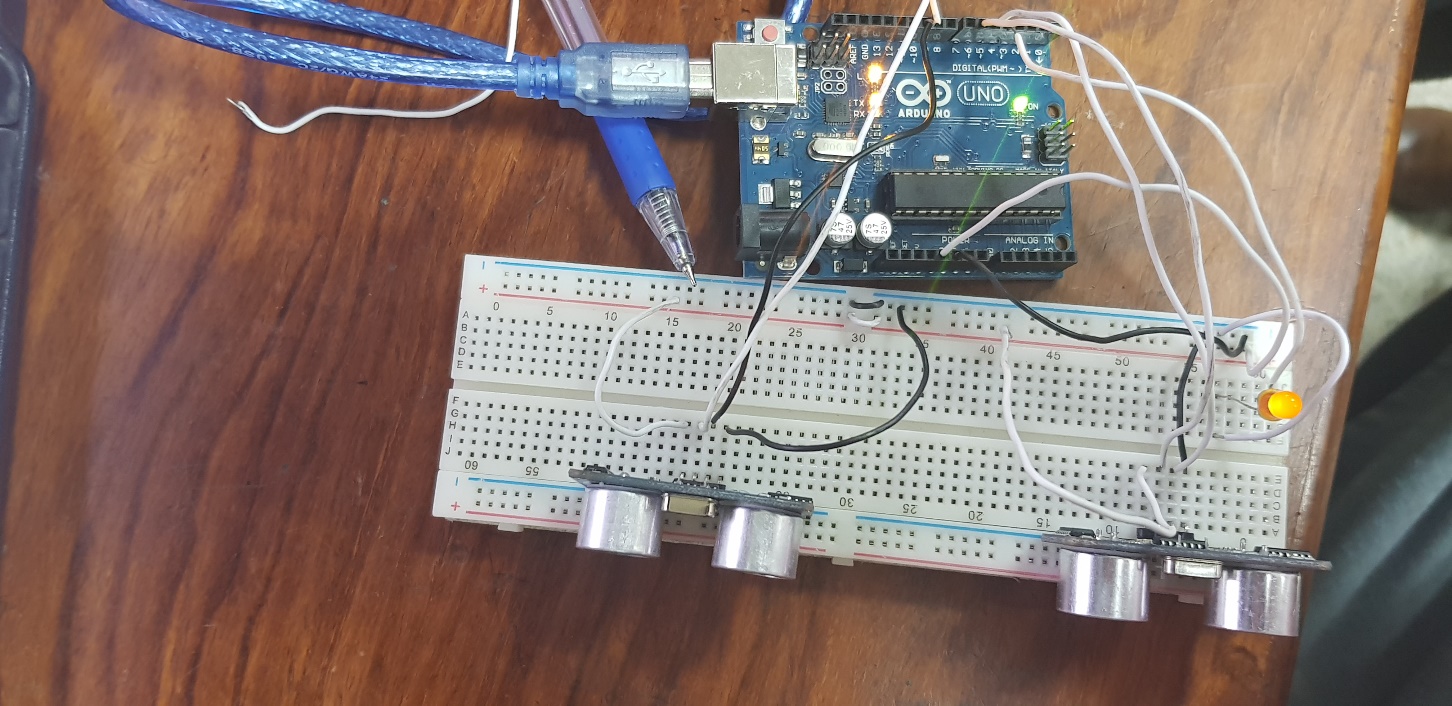
digitalWrite(led, HIGH);

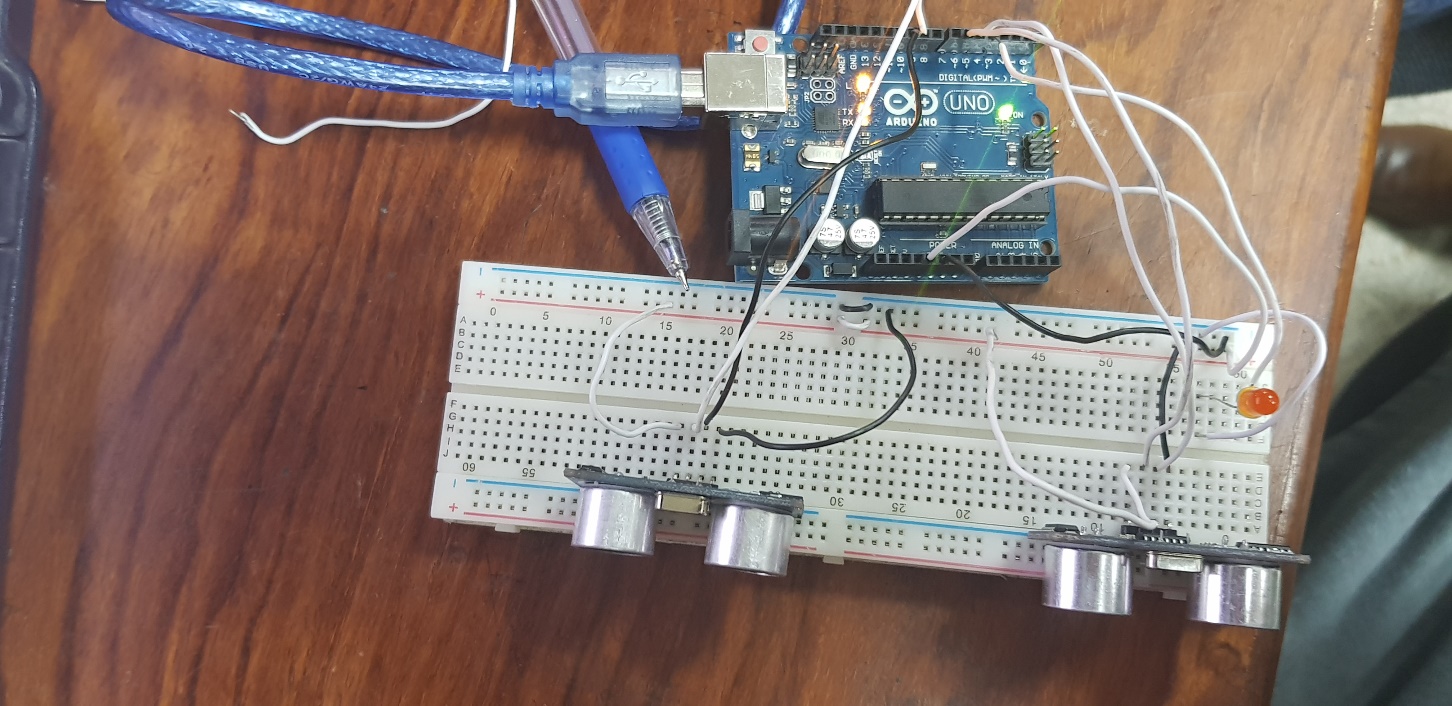
else

digitalWrite(led, LOW);

delayMicroseconds(100);

}



****

**Task 4.3**

Design water level controller system using HC-SR04. If water level increases/decreases from a particular range, the motor should be turned off/on. You can use L298 driver IC to interface DC motor with Arduino Uno. You must need to display different water levels on LED Bar.

/\*

\* Ultrasonic Sensor HC-SR04 interfacing with Arduino.

\*/

// defining the pins

const int trigPin1 = 6;

const int echoPin1 = 5;

const int led1=8;

const int led2=9;

const int led3=10;

const int led4=11;

const int led5=12;

// defining variables

long duration1;

int distance;

void setup() {

pinMode(trigPin1, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin1, INPUT); // Sets the echoPin as an Input

Serial.begin(9600); // Starts the serial communication

}

void loop() {

// Clears the trigPin

digitalWrite(trigPin1, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 micro seconds

digitalWrite(trigPin1, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin1, LOW);

duration1 = pulseIn(echoPin1, HIGH);

// Calculating the distance

distance= duration1\*0.034/2;

// Prints the distance on the Serial Monitor

Serial.print("Distance: ");

Serial.println(distance);

if (distance > 0 && distance < 20)

{digitalWrite(led1, HIGH);

digitalWrite(led2, HIGH);

digitalWrite(led3, HIGH);

digitalWrite(led4, HIGH);

digitalWrite(led5, LOW);

}

else if (distance > 20 && distance < 30)

{ digitalWrite(led1, HIGH);

digitalWrite(led2, HIGH);

digitalWrite(led3, HIGH);

digitalWrite(led4, LOW);

digitalWrite(led5, LOW);

}

else if (distance > 30 && distance < 40)

{ digitalWrite(led1, HIGH);

digitalWrite(led2, HIGH);

digitalWrite(led3, LOW);

digitalWrite(led4, LOW);

digitalWrite(led5, LOW);

}

else if (distance > 40 && distance < 50)

{ digitalWrite(led1, HIGH);

digitalWrite(led2, LOW);

digitalWrite(led3, LOW);

digitalWrite(led4, LOW);

digitalWrite(led5, LOW);

}

else if (distance > 50 )

{ digitalWrite(led1, LOW);

digitalWrite(led2, LOW);

digitalWrite(led3, LOW);

digitalWrite(led4, LOW);

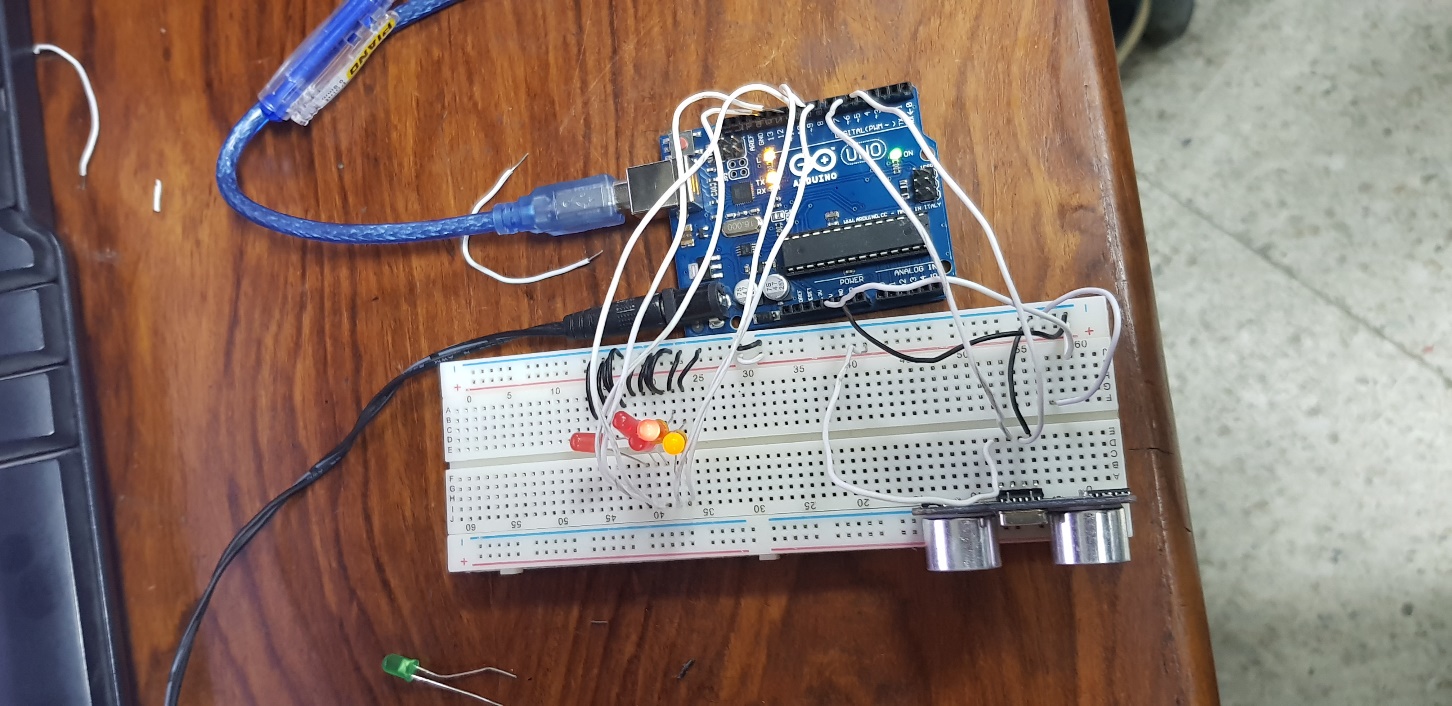
digitalWrite(led5, HIGH);

Serial.print("Motor On! ");

}

delay(10);

}

****

**Assignment 2 | Due Date: 27th February, 2019**

Design Obstacle avoidance robot using Arduino Uno and HC-SR04 sonar sensors.