Experiment 4

Student Name: Harsh Kumar **UID:22BCS15754**
Branch:BE-CSE **Section/Group: FL_IOT_603(B)**
Semester: 5th **Date of Performance:07/08/24**
Subject Name:IOT Architecture and its Protocol Lab
Subject Code:22CSP-329

1. **Aim:** To Formulate distance of an object using an ultrasonic sensor.

2. **Objective:**

   1. Learn about IoT based simulations.
   2. Testing and model in IoT based simulation platform

3. **Requirements(Hardware/Software):**1x Arduino Uno R3 board, 1x Ultrasonic sensor (HC-SR04), 4x Jumper Wires

4. **Working Principle of Ultrasonic Sensor:**

Ultrasonic sensors measure distance by sending and receiving the ultrasonic wave.The ultrasonic sensor has a sender to emit the ultrasonic waves and a receiver to receive the ultrasonic waves. The transmitted ultrasonic wave travels through the air and is reflected by hitting the Object. Arduino calculates the time taken by the ultrasonic pulse wave to reach the receiver from the sender. We know that the speed of sound in air is nearly 344 m/s, So, the known parameters are time and speed (constant). Using these parameters, we can calculate the distance traveled by the sound wave.
Formula:
Distance = Speed * Time
Distance = Speed of Sound in Air * (Time Taken / 2).

5. Procedure:

1. Connect the Echo pin of the sensor to the D8 pin of the Arduino.
2. Connect the Trig pin of the sensor to the D4 pin of the Arduino.
3. Navigate to Tools and select board and port.
4. Verify and compile the code, then upload the code to the Arduino Uno R3board.
5. Monitor the output in the Serial monitor (Set the baud rate as 9600).
   To open Serial monitor Tools>Serial Monitor or (Ctrl+Shift+M)

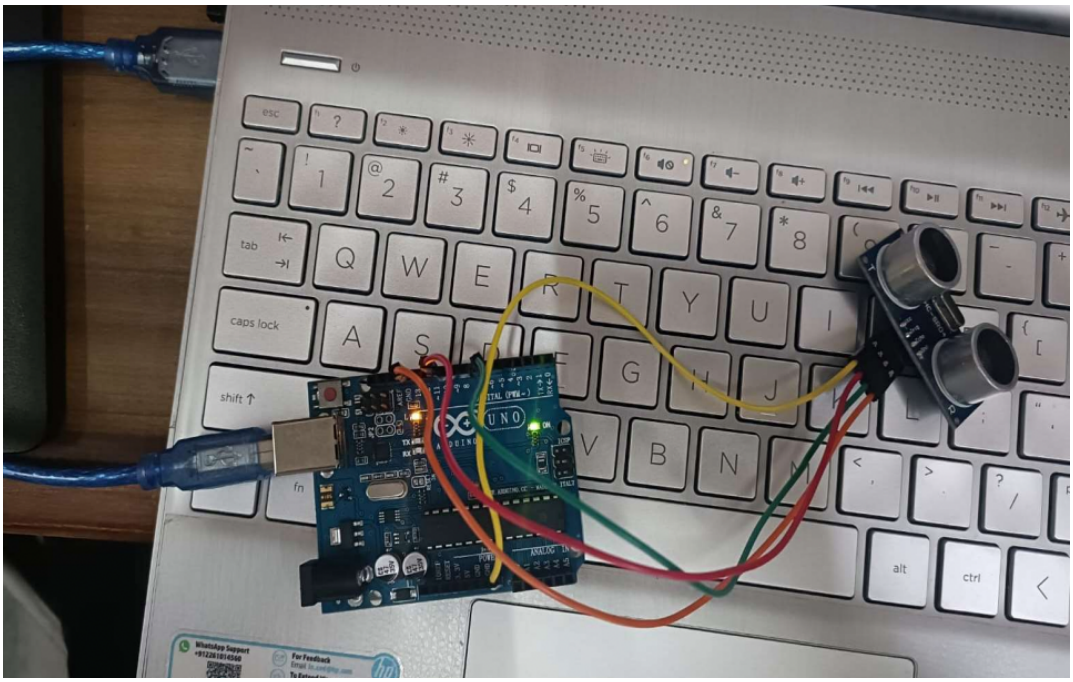
Code:

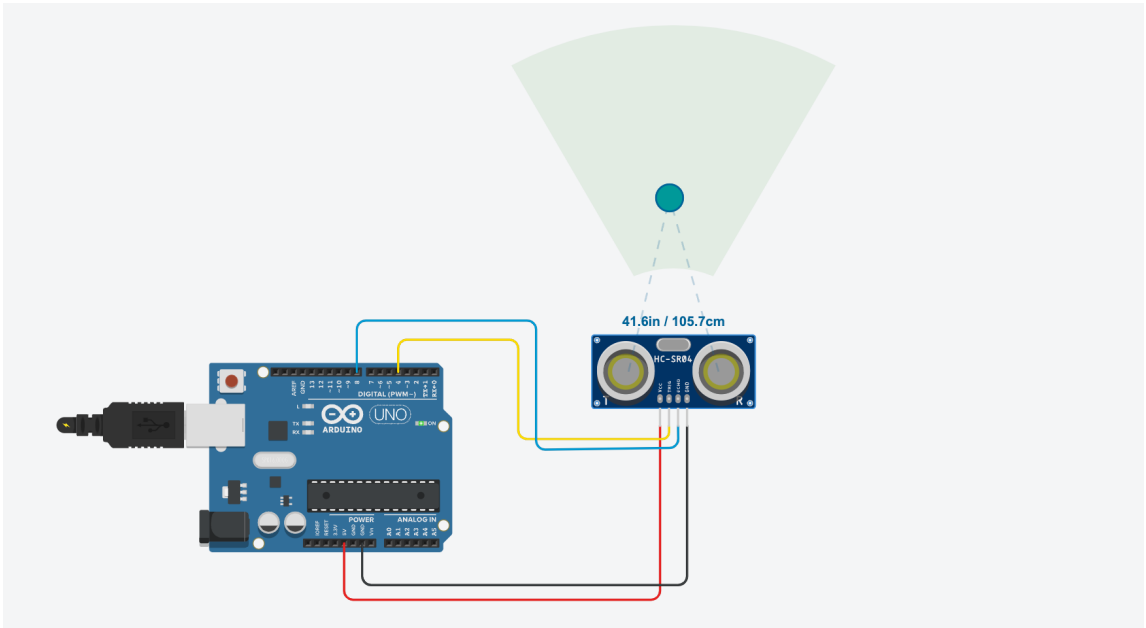
```
#define echoPin 8 // attach pin D2 Arduino to pin Echo of HC-SR04
#define trigPin 4 // attach pin D3 Arduino to pin Trig of HC-SR04
long duration; // Variable to store time taken to the pulse
int distance; // Variable to store distance calculated using
void setup()
{
pinMode(trigPin,OUTPUT); // Sets the trigPin as an OUTPUT
pinMode(echoPin, INPUT); // Sets the echoPin as an INPUT
Serial.begin(9600);
Serial.println("Distance measurement using Arduino Uno.");
delay(5000);
}
void loop()
{
digitalWrite(trigPin, LOW);
delayMicroseconds(2); // wait for 2 ms to avoid collision in serial monitor
digitalWrite(trigPin,HIGH); // turn on the Trigger to generate pulse
delayMicroseconds(10); // keep the trigger "ON" for 10 ms to generate pulse for 10 ms.
digitalWrite(trigPin,LOW); // Turn off the pulse trigger to stop
duration = pulseIn(echoPin, HIGH);
distance= duration * 0.0344 / 2; // Expression to calculate distance using time
Serial.print("Distance: ");
Serial.print(distance); // Print the output in serial monitor
Serial.println(" cm");
delay(10000);
}
```

6. Result:

After completing the setup and uploading the code, the Arduino successfully measured the distance to an object using the ultrasonic sensor. The calculated distance was displayed in real-time on the Serial

Monitor in centimetres, demonstrating the sensor's capability to accurately measure distance in an IoT-based simulation.





7. Conclusion:

The project successfully demonstrates distance measurement using an ultrasonic sensor and an Arduino with an LCD display. By connecting the HC-SR04 sensor to the Arduino and the system measures the distance to an object. The ultrasonic sensor emits sound waves, which reflect off the object and return, allowing the Arduino to calculate the distance based on the time of flight. The calculated distance is then displayed in centimeters. This setup provides a simple and effective means to measure and display distances in real-time.

8. Learning outcomes:

1. Learnt about ultrasonic sensors.
2. Learnt about measuring distance using ultrasonic sensor.
3. Learnt about different connections.