

## ASSIGNMENT – 4

|              |   |
|--------------|---|
| Date         | 29 October 2022   |
| Name         | M Sakthi Maheshwari   |
| Roll No      | 420419205012  |
| Team ID      | PNT2022TMID38668  |
| Project Name | Gas Leakage Monitoring And Alerting System For Industries . |

### QUESTION :

Write code and connections in wokwi for the ultrasonic sensor. Whenever the distance is less than 100 cms send an "alert" to the IBM cloud and display in the device recent events. Upload document with wokwi share link and images of IBM cloud

### CODE :

```
long int echoPin=7;
long int trigPin=5;
void setup()
{
  Serial.begin(9600);
  pinMode(7,OUTPUT);
  pinMode(5,INPUT);
}

void loop()
{
  int duration, distance;
  digitalWrite(3,LOW);
  delayMicroseconds(2);
  digitalWrite(3,HIGH);
  delayMicroseconds(10);
  digitalWrite(3,LOW);

  duration=pulseIn(7,HIGH);
  distance=(duration*0.034/2);
  delay(1000);
  if (distance<=100)
  {
    Serial.println("ALERT!!");
  }
}
```

## OUTPUT :

The screenshot shows the Wokwi web IDE interface. On the left, the 'sketch.ino' file is open, displaying an Arduino sketch. The sketch defines two pins: echoPin (7) and trigPin (5). In the setup function, it initializes a serial connection at 9600 baud and configures the pins as OUTPUT and INPUT. The loop function uses the ultrasonic library to measure distance. It sends a pulse to the trigPin, reads the duration, calculates the distance (using the formula  $\text{distance} = \text{duration} \times 0.034 / 2$ ), and prints 'ALERT!!' if the distance is less than or equal to 100 cm.

```
1 long int echoPin=7;
2 long int trigPin=5;
3 void setup()
4 {
5   Serial.begin(9600);
6   pinMode(7,OUTPUT);
7   pinMode(5,INPUT);
8 }
9
10 void loop()
11 {
12   int duration, distance;
13   digitalWrite(3,LOW);
14   delayMicroseconds(2);
15   digitalWrite(3,HIGH);
16   delayMicroseconds(10);
17   digitalWrite(3,LOW);
18
19   duration=pulseIn(7,HIGH);
20   distance=duration*0.034/2;
21   delay(1000);
22   if (distance<=100)
23   {
24     Serial.println("ALERT!!");
25   }
26 }
27
28
```

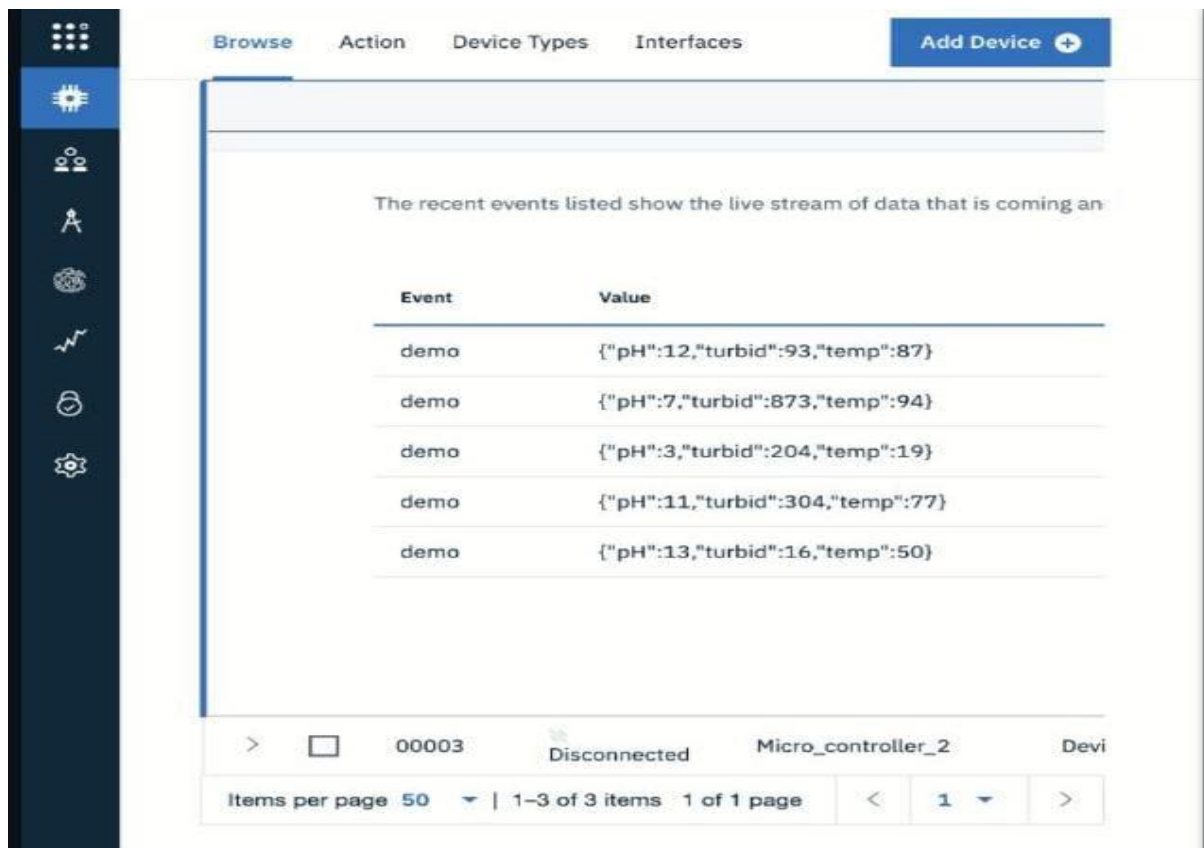
On the right, the 'Simulation' tab shows a virtual Arduino Uno board connected to an ultrasonic sensor. The simulation is running, and the console output shows 'ALERT!!' being printed repeatedly.

The screenshot shows the IBM Watson IoT Platform interface. On the left, a Python script named 'Test\_python\_3.7.4.py' is displayed. The script generates random values for pH, turbidity, and temperature, and publishes them as events. The console output shows a stream of published events.

```
Test_python_3.7.4.py
1 pH = random.random()
2 turbidity = random.randint(1,10)
3 temperature = random.randint(0,100)
4 data = {'pH': pH, 'turbid': turbidity, 'temp': temperature}
5 # print(data)
6 def myOnPublishCallback():
7     while True:
8         # Publish data
9         publish(data)
10
```

On the right, the 'IBM Watson IoT Platform' interface shows a table of recent events. The table has two columns: 'Event' and 'Value'. The events are listed in descending order of time.

| Event | Value                            |
|-------|----------------------------------|
| demo  | ["pH":12,"turbid":93,"temp":87]  |
| demo  | ["pH":7,"turbid":873,"temp":94]  |
| demo  | ["pH":3,"turbid":204,"temp":19]  |
| demo  | ["pH":11,"turbid":304,"temp":77] |
| demo  | ["pH":13,"turbid":16,"temp":50]  |



**LINK :**

<https://wokwi.com/projects/346826481011262036>