

## ASSIGNMENT 4

<b>TEAM ID</b>	PNT2022TMID36685
<b>PROJECT NAME</b>	Gas Leakage monitoring & Alerting system for Industries
<b>TEAM LEAD NAME</b>	Divya S

### Question:

Write a code and connections in wokwi for the ultrasonic sensor. Whenever the distance is less than 100cms send an “Alert” to IBM cloud and display in the device recent events.

### Solution :

#### Code:

```
// defines pins numbers
const int trigPin = 2;
const int echoPin = 5;
// defines 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.print(distance);
Serial.println(" cm");
if(distance <= 100){
Serial.println("Alert Distance is less than 100 cm");

}
}
DIAGRAM.JSON
{
"version": 1,
"author": "Uri Shaked",
"editor": "wokwi",
"parts": [
{ "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -21.91,
"left": -66.98, "attrs": {} },
{ "type": "wokwi-hc-sr04", "id": "ultrasonic1", "top": -56.74,
"left": 85.25, "attrs": {} }
],
"connections": [
[ "esp:TX0", "$serialMonitor:RX", "", [] ],
[ "esp:RX0", "$serialMonitor:TX", "", [] ],
[ "ultrasonic1:ECHO", "esp:D5", "green", [ "v0" ] ],
[ "ultrasonic1:VCC", "esp:3V3", "red", [ "v99.32", "h-11.05" ] ],
[ "esp:GND.1", "ultrasonic1:GND", "black", [ "h0" ] ],
[ "esp:D2", "ultrasonic1:TRIG", "green", [ "h0" ] ]
]
}

```

## OUTPUT:

## IBM CLOUD OUTPUT:

The screenshot displays the IBM Watson IoT Platform interface. At the top, the header shows 'IBM Watson IoT Platform' and user information '112819106004@smartinternz.com ID: p9o7pu'. The main navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains icons for various functions. The main content area shows a device named '1107' with status 'Disconnected' and user 'divya.s'. Below this, a tabbed interface shows 'Recent Events' selected. A message states: 'The recent events listed show the live stream of data that is coming and going from this device.' A table lists five recent events, each with a unique ID, a JSON payload containing distance and alert information, the format 'json', and a timestamp 'a few seconds ago'. At the bottom right, a status box indicates '2 Simulations running'.

Event	Value	Format	Last Received
event_1	{"Distance":75,"Alert":"Distance less than 100"}	json	a few seconds ago
event_1	{"Distance":41,"Alert":"Distance less than 100"}	json	a few seconds ago
event_1	{"Distance":99,"Alert":"Distance less than 100"}	json	a few seconds ago
event_1	{"Distance":33,"Alert":"Distance less than 100"}	json	a few seconds ago
event_1	{"Distance":41,"Alert":"Distance less than 100"}	json	a few seconds ago

WOKWI
SAVE
SHARE
esp32-dht22.ino by urish
Docs SIGN IN

esp32-dht22.ino
diagram.json
libraries.txt
Library Manager

```

1  const int trigPin = 2;
2  const int echoPin = 5;
3  // defines variables
4
5  long duration;
6  int distance;
7  void setup() {
8    pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
9    pinMode(echoPin, INPUT); // Sets the echoPin as an Input
10   Serial.begin(9600); // Starts the serial communication
11 }
12 void loop() {
13   // Clears the trigPin
14   digitalWrite(trigPin, LOW);
15   delayMicroseconds(2);
16   // Sets the trigPin on HIGH state for 10 micro seconds
17   digitalWrite(trigPin, HIGH);
18   delayMicroseconds(10);
19   digitalWrite(trigPin, LOW);
20   // Reads the echoPin, returns the sound wave travel time in microseconds
21   duration = pulseIn(echoPin, HIGH);
22   // Calculating the distance
23   distance= duration*0.034/2;
24   // Prints the distance on the Serial Monitor
25   Serial.print("Distance: ");
26   Serial.print(distance);
27   Serial.println(" cm");
28   if(distance <= 100){
29     Serial.println("Alert Distance is less than 100 cm");
30   }
31 }

```

Simulation
00:10.531 46%

Editing Ultrasonic Distance Sensor
Distance: 23cm

Distance: 23 cm  
Alert Distance is less than 100 cm  
Distance: 22 cm  
Alert Distance is less than 100 cm  
Distance: 23 cm  
Alert Distance is less than 100 cm  
Distance: 22 cm

WOKWI
SAVE
SHARE
esp32-dht22.ino by urish
Docs SIGN IN

esp32-dht22.ino
diagram.json
libraries.txt
Library Manager

```

1  {
2    "version": 1,
3    "author": "Uri Shaked",
4    "editor": "wokwi",
5    "parts": [
6      { "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -53.33, "left": -99.5, "width": 100, "height": 100 },
7      { "type": "wokwi-hc-sr04", "id": "ultrasonic1", "top": -66.84, "left": 77.5, "width": 50, "height": 50 },
8    ],
9    "connections": [
10     [ "esp:TX0", "$serialMonitor:RX", "", [ ] ],
11     [ "esp:RX0", "$serialMonitor:TX", "", [ ] ],
12     [ "ultrasonic1:VCC", "esp:3V3", "red", [ "v0" ] ],
13     [ "ultrasonic1:TRIG", "esp:D2", "green", [ "v0" ] ],
14     [ "ultrasonic1:ECHO", "esp:D5", "green", [ "v0" ] ],
15     [ "ultrasonic1:GND", "esp:GND.1", "black", [ "v0" ] ]
16   ]
17 }

```

Simulation
00:12.831 42%

Editing Ultrasonic Distance Sensor
Distance: 59cm

Alert Distance is less than 100 cm  
Distance: 58 cm  
Alert Distance is less than 100 cm  
Distance: 58 cm  
Alert Distance is less than 100 cm  
Distance: 59 cm  
Alert Distance is less than 100 cm