

ASSIGNMENT 4

Ultrasonic sensor simulation in Wokwi

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.

Wokwi simulation link:

<https://wokwi.com/projects/347020199811088980>

Code:

```
1  #define ECHO_PIN 2
2  #define TRIG_PIN 3
3  #define organization = "cqj124"
4  #define deviceType = "IBM"
5  #define deviceId = "69"
6  #define authMethod = "token"
7  #define authToken = "12345678"
8
9  void setup(){
10     Serial.begin(9600);
11     pinMode(TRIG_PIN,OUTPUT);
12     pinMode(ECHO_PIN,INPUT);
13 }
14
15 float readDistanceCM(){
16     digitalWrite(TRIG_PIN,LOW);
17     delayMicroseconds(2);
18     digitalWrite(TRIG_PIN,HIGH);
19     delayMicroseconds(10);
20     digitalWrite(TRIG_PIN,LOW);
21     int duration=pulseIn(ECHO_PIN,HIGH);
22     return duration*0.034/2;
23 }
24 void loop(){
```

```

25     float distance=readDistanceCM();
26     if(distance<=100)
27     {
28         Serial.println("person detected");
29     }
30     else{
31         Serial.print("Measured distance:");
32         Serial.println(readDistanceCM());
33     }
34     delay(1000);
35 }

```

IBM Cloud:

The screenshot displays the IBM Watson IoT Platform interface. The main dashboard shows a list of devices, with one device selected (Device ID: 69, Status: Disconnected, Device Type: IBM). The 'Recent Events' tab is active, showing a table of events with columns for Event, Value, Format, and Last Received. The event data is as follows:

Event	Value	Format	Last Received
event_1	{"randomNumber":76,"distance":78}	json	a few seconds ago
event_1	{"randomNumber":10,"distance":119}	json	a few seconds ago
event_1	{"randomNumber":19,"distance":110}	json	a few seconds ago
event_1	{"randomNumber":47,"distance":141}	json	a few seconds ago

An 'Add Device' modal is open, showing the 'Device Type: IBM' configuration. The 'Events' tab is selected, showing a list of events. The 'Event type name' is 'event_1'. The 'Schedule' is set to '20' and 'Every Minute'. The 'Payload' is defined as:

```

{
  0: {
    1: "randomNumber": random(0, 100)
    2: "distance": random(0, 150)
    3: }
  4: }

```

The 'Send' button is visible, and the 'Upload a CSV file' button is also present.

Service Details - IBM Cloud x IBM Watson IoT Platform x +

← → ↻ cql124.internetofthings.ibmcloud.com/dashboard/devices/browse

IBM Watson IoT Platform 613519106034@smartinternz.com ID: cql124

Browse Action Device Types Interfaces Add Device +

criteria. To get started, you can add devices by using the Add Device button, or by using API.

Device Type: IBM

Events 1 New event type + 1 item selected Cancel

Event type name event_1 Send

Schedule 20 Every Minute

Payload Specify the event payload in the editor window or by uploading a CSV file.

```
0 {
1   "randomNumber": random(0, 100),
2   "distance": random(0, 150)
3 }
4
```

Upload a CSV file

Cancel Save

Delete

Device ID	Status	Device Type	Class ID	Date
69	Disconnected	IBM	Device	Oct

Identity Device Information Recent Events State Logs

Device ID 69

Device Type IBM

Date Added Oct 31, 2022 1:07 PM

Added By 613519106034@smartinternz.com

Connection Status Disconnected

Items per page 50 | 1-1 of 1 item

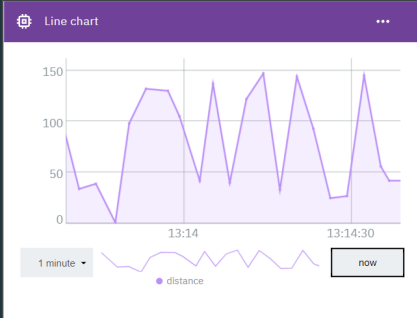
Service Details - IBM Cloud x IBM Watson IoT Platform x Wokwi - Online Arduino and ESP x WhatsApp x +

← → ↻ cql124.internetofthings.ibmcloud.com/dashboard/boards/58d0134a-d53e-4ca4-aaca-5ab7c568a212

IBM Watson IoT Platform 613519106034@smartinternz.com ID: cql124

ultrasonic + Add New Card Settings

Line chart



1 minute now

Device Type: IBM

Events 1 New event type +

Event type name event_1 Send

Schedule 20 Every Minute

Payload Specify the event payload in the editor window or by uploading a CSV file.

```
0 {
1   "randomNumber": random(0, 100),
2   "distance": random(0, 150)
3 }
4
```

Upload a CSV file

Cancel Save

Wokwi Simulation:

Wokwi Simulation interface showing a sketch and a simulation of an Arduino Uno connected to an HC-SR04 ultrasonic sensor.

Sketch (sketch.ino):

```
1 #define ECHO_PIN 2
2 #define TRIG_PIN 3
3 #define organization = "cqj124"
4 #define deviceType = "IBM"
5 #define deviceId = "69"
6 #define authMethod = "token"
7 #define authToken = "12345678"
8
9 void setup(){
10   Serial.begin(9600);
11   pinMode(TRIG_PIN,OUTPUT);
12   pinMode(ECHO_PIN,INPUT);
13 }
14
15 float readDistanceCM(){
16   digitalWrite(TRIG_PIN,LOW);
17   delayMicroseconds(2);
18   digitalWrite(TRIG_PIN,HIGH);
19   delayMicroseconds(10);
20   digitalWrite(TRIG_PIN,LOW);
21   int duration=pulseIn(ECHO_PIN,HIGH);
22   return duration*0.034/2;
23 }
24 void loop(){
25   float distance=readDistanceCM();
26
27   if(distance<=100)
28   {
29     Serial.println("person detected");
30   }
31   else{
32     Serial.print("Measured distance:");
33     Serial.println(readDistanceCM());
34   }
35   delay(1000);
36 }
37 }
```

Simulation:

The simulation shows an Arduino Uno connected to an HC-SR04 ultrasonic sensor. The sensor is connected to the Arduino's digital pins. The simulation output shows the measured distance in centimeters and the detection of a person.

Editing Ultrasonic Distance Sensor
Distance: 48cm

Measured distance:108.60
Measured distance:108.60
Measured distance:258.49
Measured distance:275.26
person detected
person detected
person detected