

## Assignment-4

### Question-1:

Write code and connections in wokwi for ultrasonic. Whenever distance is less than 100 cm send "alert" to IBM Cloud and display in device recent events.

Solution:

```
#define ECHO_PIN 2

#define TRIG_PIN 3

#define organization = "fkxdqs"

#define deviceType = "Arduino"

#define deviceId = "1200"

#define authMethod = "use-token-auth" #define

authToken = "00000000" void setup() {

Serial.begin(9600); pinMode(TRIG_PIN,

OUTPUT); pinMode(ECHO_PIN, INPUT);

}

float readDistanceCM() {

digitalWrite(TRIG_PIN, LOW);

delayMicroseconds(2);

digitalWrite(TRIG_PIN, HIGH);

delayMicroseconds(10);

digitalWrite(TRIG_PIN, LOW); int duration

=pulseIn(ECHO_PIN, HIGH); return

duration * 0.034 / 2;
```

```

}

void loop() { float distance = readDistanceCM();
if(distance <=

100)

{

Serial.println("person detected ");

}

else{

Serial.print("Measureddistance: ");

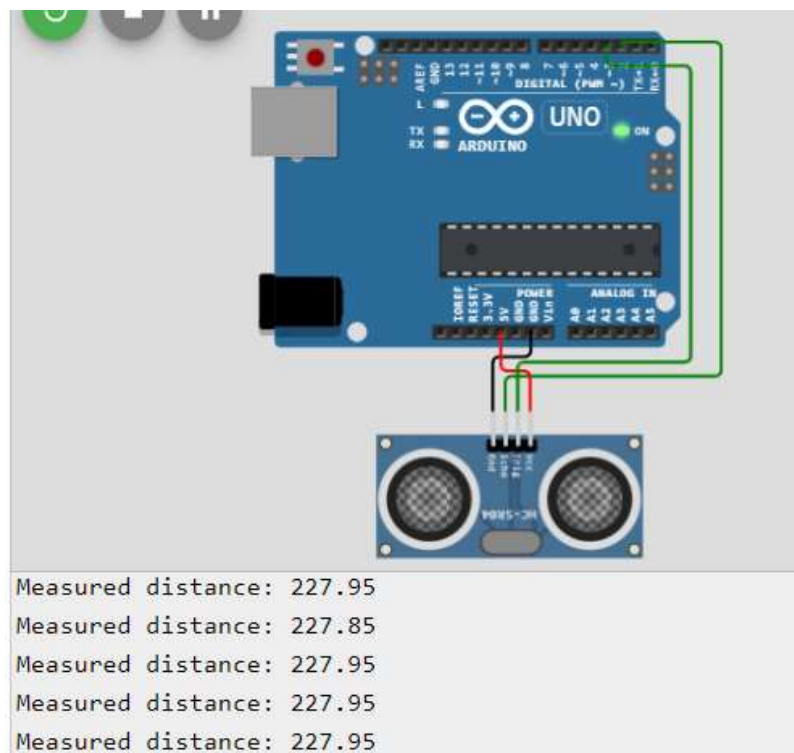
Serial.println(readDistanceCM());

}

delay(1000);

}

```



**Wokwi Link:** <https://wokwi.com/projects/347653583594848852>

## **IBM Cloud**

### **Device Recent Events**

Identity

Device Information

Recent Events

State

Logs

X

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
event_1	"PersonDetected"	json	a few seconds ago
event_1	"PersonDetected"	json	a few seconds ago
event_1	"PersonDetected"	json	a few seconds ago
event_1	"PersonDetected"	json	a few seconds ago
event_1	"PersonDetected"	json	a few seconds ago

1 Simulation running

WOWKI EXECUTIONPART WITH ANOTHER CODE:

sketch.ino

diagram.json

Library Manager

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

Simulation

▶

+

⋮

