

## ASSIGNMENT-4

### 1. CODE:

```
#define ECHO_PIN 2

#define TRIG_PIN 3

#define LED_BUILTIN 13

#define organization = "57kmpx"

#define deviceType = "abcd"

#define deviceId = "13"

#define authMethod = "token"

#define authToken = "123456789"


void setup(){

    Serial.begin(9600);

    pinMode(TRIG_PIN,OUTPUT);

    pinMode(ECHO_PIN,INPUT);

    pinMode(LED_BUILTIN, OUTPUT);

}


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();
```

```
{  
  digitalWrite(LED_BUILTIN, HIGH);  
  delay(1000);  
  digitalWrite(LED_BUILTIN, LOW);  
  delay(1000);  
}  
  
if(distance<=100)  
{  
  Serial.println("person detected");  
  Serial.print("Measured distance:");  
  Serial.println(readDistanceCM());  
  Serial.println("HIGH");  
}  
else{  
  Serial.print("Measured distance:");  
  Serial.println(readDistanceCM());  
  Serial.println("LOW");  
}  
  delay(1000);  
}
```

## 2. LINK:

<https://wokwi.com/projects/346845514621256275>

### 3. IBM CLOUD:

The screenshot shows the IBM Watson IoT Platform interface. The main panel displays the 'Recent Events' for device 'abcd'. A modal window is open for creating a new event type.

**Recent Events Table:**

Event	Value	Format	Last Received
Ultrasonic	{"randomNumber":55,"distance":137}	json	a few seconds ago
Ultrasonic	{"randomNumber":55,"distance":113}	json	a few seconds ago
Ultrasonic	{"randomNumber":13,"distance":87}	json	a few seconds ago
Ultrasonic	{"randomNumber":85,"distance":79}	json	a few seconds ago
Ultrasonic	{"randomNumber":60,"distance":109}	json	a few seconds ago

**Modal Window: Device Type: abcd**

- Event type name:** Ultrasonic
- Schedule:** 20 Every Minute
- Payload:**

```
{
  "randomNumber": random(0, 100),
  "distance": random(50, 150)
}
```

The screenshot shows the IBM Watson IoT Platform interface for the 'Ultrasonic Sensor'. The dashboard displays two charts: a line chart and a donut chart.

**Line Chart:** Shows the values of 'randomNumber' and 'distance' over time. The x-axis represents time (11:44, 11:46, 11:48). The y-axis represents values from 0 to 140. The legend indicates that the purple line represents 'randomNumber' and the blue line represents 'distance'.

**Donut Chart:** Shows the total distance traveled. The total is 149 m. The legend indicates that the purple circle represents 'distance' (133.0 km/h) and the blue circle represents 'randomNumber' (16.0 m).

**Modal Window: Device Type: abcd**

- Event type name:** Ultrasonic
- Schedule:** 20 Every Minute
- Payload:**

```
{
  "randomNumber": random(0, 100),
  "distance": random(50, 150)
}
```

#### 4. WOKKI:

W sketch.ino - Wokwi Arduino and x IBM IBM Watson IoT Platform x +

wokwi.com/projects/346845514621256275

WOKWI SAVE SHARE

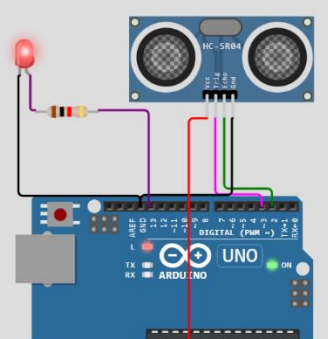
Docs 6

sketch.ino diagram.json Library Manager

```
15 }
16
17 float readDistanceCM(){
18   digitalWrite(TRIG_PIN, LOW);
19   delayMicroseconds(2);
20   digitalWrite(TRIG_PIN, HIGH);
21   delayMicroseconds(10);
22   digitalWrite(TRIG_PIN, LOW);
23   int duration=pulseIn(ECHO_PIN,HIGH);
24   return duration*0.034/2;
25 }
26
27 void loop(){
28   float distance=readDistanceCM();
29   {
30     digitalWrite(LED_BUILTIN, HIGH);
31     delay(1000);
32     digitalWrite(LED_BUILTIN, LOW);
33     delay(1000);
34   }
35   if(distance>=100)
36   {
37     Serial.println("person detected");
38     Serial.print("Measured distance:");
39     Serial.println(readDistanceCM());
40     Serial.println("HIGH");
41   }
42   else{
43     Serial.print("Measured distance:");
44     Serial.println(readDistanceCM());
45     Serial.println("LOW");
46   }
47   delay(1000);
48 }
```

Simulation

00:49.117 100%



HIGH  
person detected  
Measured distance:394.60  
HIGH  
person detected  
Measured distance:394.60  
HIGH

11:12 31-10-2022