ASSIGNMENT 4

wokwi link:

Program code:

```
/*

  HC-SR04 Ultrasonic Sensor Example.

  Turn the LED on when an object is within 100cm range.

  Copyright (C) 2021, Uri Shaked

*/

#define ECHO_PIN 2

#define TRIG_PIN 3

#define organization = "hfblrc"

#define device type ="type1"

#define deviceId="01"

#define authmethod="token"

#define authToken="12345678"

void setup() {

Serial.begin(115200);

  pinMode(LED_BUILTIN, OUTPUT);

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("Measured distance:");

        Serial.println(readDistanceCM());

    }

delay(1000);

}
```

IBM CLOUD:

IBM Watson IoT Platform

board1

Line chart

150
100
50
0
22:04   22:06   22:08

5 minutes ▾   now
● distance

Device Type: type1

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

WOKWI   💾 SAVE ▾   ➡ SHARE   ♡   hc-sr04.ino
by urish   Docs

hc-sr04.ino ● | diagram.json | Library Manager ▾   Simulation

```
5   */
6   #define ECHO_PIN 2
7   #define TRIG_PIN 3
8   #define organization ="hfblrc"
9   #define device type ="type1"
10  #define deviceId="01"
11  #define authmethod="token"
12  #define authToken="12345678"
13  void setup() {
14    Serial.begin(115200);
15    pinMode(LED_BUILTIN, OUTPUT);
16    pinMode(TRIG_PIN, OUTPUT);
17    pinMode(ECHO_PIN, INPUT);
18  }
19  float readDistanceCM() {
20    digitalWrite(TRIG_PIN, LOW);
21    delayMicroseconds(2);
22    digitalWrite(TRIG_PIN, HIGH);
23    delayMicroseconds(10);
24    digitalWrite(TRIG_PIN, LOW);
25    int duration = pulseIn(ECHO_PIN, HIGH);
26    return duration * 0.034 / 2;
27  }
28  void loop(){
29    float distance=readDistanceCM();
30    if(distance<=100)
31    {
32      Serial.println("person_detected");
```

00:05.765 103%

Measured distance:177.33
Measured distance:177.34
Measured distance:177.24
Measured distance:177.40
Measured distance:177.24
Measured distance:177.33