

**S.NITHYA**

## **ASSIGNMENT-4**

**Question-1: Write code and connections in wokwi for ultrasonic sensor. Whenever distance is less than 100 cms send “alert” to ibm cloud and display in device recent events.**

Solution:

```
#define ECHO_PIN 2
#define TRIG_PIN 3
#define organization ="ysx3j0"
#define deviceType=" Arduino"
#define deviceId ="50601"
#define authMethod ="use-token-auth"
#define authToken ="050620010"

void setup() {
  // put your setup code here, to run once:
  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() {
```

```
    // put your main code here, to run repeatedly:
```

```
    float distance = readDistanceCM();
```

```
    if(distance <= 100)
```

```
    {
```

```
        Serial.println("person detected ");
```

```
    }
```

```
    else{
```

```
        Serial.print("Measured distance: ");
```

```
        Serial.println(readDistanceCM());
```

```
    }
```

```
    delay(1000);
```

```
}
```

Service Details - IBM Cloud | IBM Watson IoT Platform | IBM Watson IoT Platform | New Arduino Uno Project - Wokwi

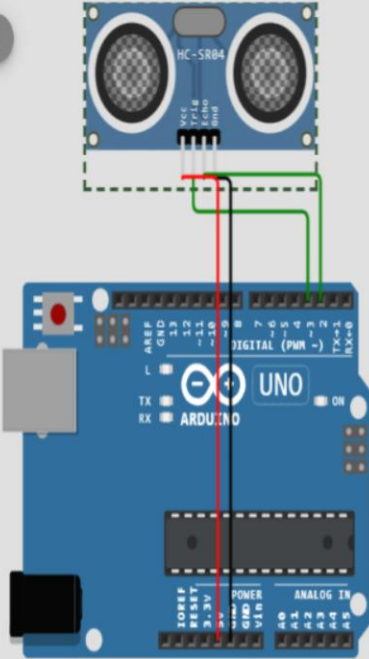
wokwi.com/projects/new/arduino-uno

WOKWI SAVE SHARE Docs

sketch.ino diagram.json Library Manager

```
1 #define ECHO_PIN 2
2 #define TRIG_PIN 3
3 #define organization "ysx3j0"
4 #define deviceType "Arduino"
5 #define deviceId "50601"
6 #define authMethod "use-token-auth"
7 #define authToken "050620010"
8
9 void setup() {
10   // put your setup code here, to run once:
11   Serial.begin(9600);
12   pinMode(TRIG_PIN, OUTPUT);
13   pinMode(ECHO_PIN, INPUT);
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
25 }
26
27 void loop() {
28   // put your main code here, to run repeatedly:
29   float distance = readDistanceCM();
30   if(distance <= 100)
```

Simulation



# Output:

The screenshot displays a Wokwi simulation environment. On the left, the 'sketch.ino' file is open, showing the following code:

```
25 }
26
27 void loop() {
28   // put your main code here, to run repeatedly:
29   float distance = readDistanceCM();
30   if(distance <= 100)
31   {
32     Serial.println("person detected ");
33   }
34   else{
35     Serial.print("Measured distance: ");
36     Serial.println(readDistanceCM());
37   }
38   delay(1000);
39
40
41
42 }
43
```

On the right, the 'Simulation' window shows an Arduino Uno board connected to an HC-SR04 ultrasonic sensor. The sensor's VCC pin is connected to the 5V pin on the Arduino, and its GND pin is connected to a GND pin. The sensor's TRIG pin is connected to digital pin 5, and its ECHO pin is connected to digital pin 4. The simulation is running, and the serial monitor at the bottom shows the following output:

```
Measured distance: 395.25
Measured distance: 395.27
Measured distance: 395.25
Measured distance: 395.35
Measured distance: 395.25
Measured distance: 395.25
Measured distance: 395.25
Measured distance: 395.25
```

**Wokwi Link:** <https://wokwi.com/projects/347041684263010899>

# IBM CLOUD

## Device Recent Events

The screenshot shows the IBM Watson IoT Platform interface. The browser tabs include 'Service Details - IBM Cloud', 'IBM Watson IoT Platform', and 'New Arduino Uno Project - Work'. The URL is `ysx3j0.internetofthings.ibmcloud.com/dashboard/devices/drilldown/arduino:50601?returnTo=/devices/browse`. The user is logged in as `nithyashankar0509@gmail.com` with ID `ysx3j0`.

### Device Drilldown - 50601

[← Back](#)

**Device Credentials**

- Connection Information
- Recent Events
- State
- Device Information
- Metadata
- Diagnostics
- Connection Logs
- Device Actions

**Device Credentials**

You registered your device to the organization. Add these credentials to the device to connect it to the platform. After the device is connected, you can navigate to view connection and event details.

Organization ID	ysx3j0
Device Type	arduino
Device ID	50601
Authentication Method	use-token-auth
Authentication Token	050620010

**Authentication tokens are non-recoverable. authentication token.**

0 Simulations running

[Find out how to add these credentials to your device](#)

