

## Assignment – 4

Assignment Date	28 October 2022
Student Name	Swetha S
Student Roll Number	611819106055
Maximum Marks	2 Marks

### 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:

```
// Pins
```

```
const int TRIG_PIN = 7;
```

```
const int ECHO_PIN = 8;
```

```
// Anything over 400 cm (23200 us pulse) is "out of range"
```

```
const unsigned int MAX_DIST = 23200;
```

```
void setup() {
```

```
    // The Trigger pin will tell the sensor to range find
```

```
    pinMode(TRIG_PIN, OUTPUT);
```

```
    digitalWrite(TRIG_PIN, LOW);
```

```
    //Set Echo pin as input to measure the duration of
```

```
    //pulses coming back from the distance sensor
```

```
    pinMode(ECHO_PIN, INPUT);
```

```
// We'll use the serial monitor to view the sensor output
```

```
Serial.begin(9600);  
}
```

```
void loop() {
```

```
    unsigned long t1;  
    unsigned long t2;  
    unsigned long pulse_width;  
    float cm;  
    float inches;
```

```
    // Hold the trigger pin high for at least 10 us  
    digitalWrite(TRIG_PIN, HIGH);  
    delayMicroseconds(10);  
    digitalWrite(TRIG_PIN, LOW);
```

```
    // Wait for pulse on echo pin  
    while ( digitalRead(ECHO_PIN) == 0 );
```

```
    // Measure how long the echo pin was held high (pulse width)  
    // Note: the micros() counter will overflow after ~70 min  
    t1 = micros();  
    while ( digitalRead(ECHO_PIN) == 1);  
    t2 = micros();  
    pulse_width = t2 - t1;  
    // Calculate distance in centimeters and inches. The constants
```

```

// are found in the datasheet, and calculated from the assumed speed
//of sound in air at sea level (~340 m/s).
cm = pulse_width / 58.0;
inches = pulse_width / 148.0;
// Print out results
if ( pulse_width > MAX_DIST ) {
Serial.println("Out of range");
} else {
Serial.println("*****");
Serial.print("The Measured Distance in cm : ");
Serial.println(cm);
if(cm<100){
  // while(true){
    Serial.println("Alert!!");
  // }
}
Serial.print("*****");
}
// Wait at least 1000ms before next measurement
delay(1000);
}

```

## Output:

1) If the distance is less than 100 cms , it alerts.

WOKWI

```
1 // Pins
2 const int TRIG_PIN = 7;
3 const int ECHO_PIN = 8;
4
5 // Anything over 400 cm (21000 us pulse) is "out of range"
6 const unsigned int MAX_DIST = 21000;
7
8 void setup() {
9
10 // The Trigger pin will tell the sensor to range find
11 pinMode(TRIG_PIN, OUTPUT);
12 digitalWrite(TRIG_PIN, LOW);
13
14 //Set Echo pin as input to measure the duration of
15 //pulses coming back from the distance sensor
16 pinMode(ECHO_PIN, INPUT);
17
18 // We'll use the serial monitor to view the sensor output
19 Serial.begin(9600);
20 }
21
22 void loop() {
23
24   unsigned long t1;
25   unsigned long t2;
26   unsigned long pulse_width;
27   float cm;
28   float inches;
29
30   // Hold the trigger pin high for at least 10 us
```

Simulation

00:00:44 50%

Editing Ultrasonic Distance Sensor

Distance: 84cm

The Measured Distance in cm : 84.14

Alert!!

Activate Windows  
Go to Settings to activate Windows.

2) If the distance is more than 100 cms, it won't alert.

WOKWI

```
1 // Pins
2 const int TRIG_PIN = 7;
3 const int ECHO_PIN = 8;
4
5 // Anything over 400 cm (21000 us pulse) is "out of range"
6 const unsigned int MAX_DIST = 21000;
7
8 void setup() {
9
10 // The Trigger pin will tell the sensor to range find
11 pinMode(TRIG_PIN, OUTPUT);
12 digitalWrite(TRIG_PIN, LOW);
13
14 //Set Echo pin as input to measure the duration of
15 //pulses coming back from the distance sensor
16 pinMode(ECHO_PIN, INPUT);
17
18 // We'll use the serial monitor to view the sensor output
19 Serial.begin(9600);
20 }
21
22 void loop() {
23
24   unsigned long t1;
25   unsigned long t2;
26   unsigned long pulse_width;
27   float cm;
28   float inches;
29
30   // Hold the trigger pin high for at least 10 us
```

Simulation

00:00:59 62%

The Measured Distance in cm : 227.10

Activate Windows  
Go to Settings to activate Windows.

### 3) Simulation and code execution

