

Assignment-4

Team ID: PNT2022TMID28270

Name: Mary Merlin V

1. Write Code and connections in wokwi for ultrasonic sensor. whatever distance is less than 100 cm 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
  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 (- 340m/s)
cm=pulse_Width / 58 ;
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!!");
        //}
    }

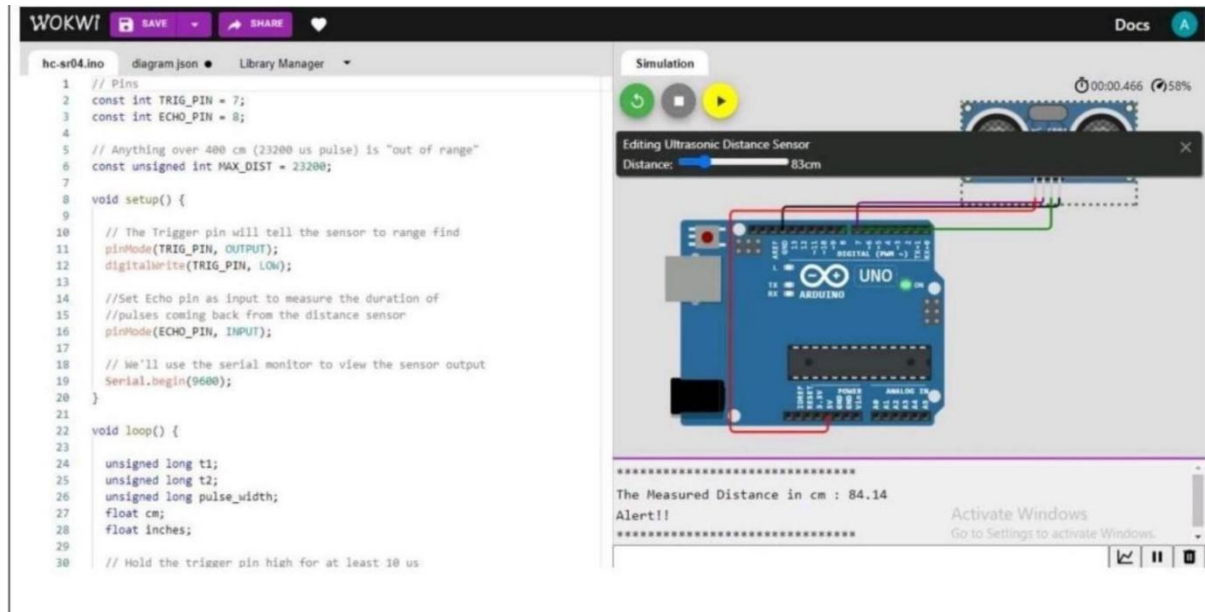
Serial.print("*****");
}

//wait at least 1000ms before next measurement
Delay(1000);
}

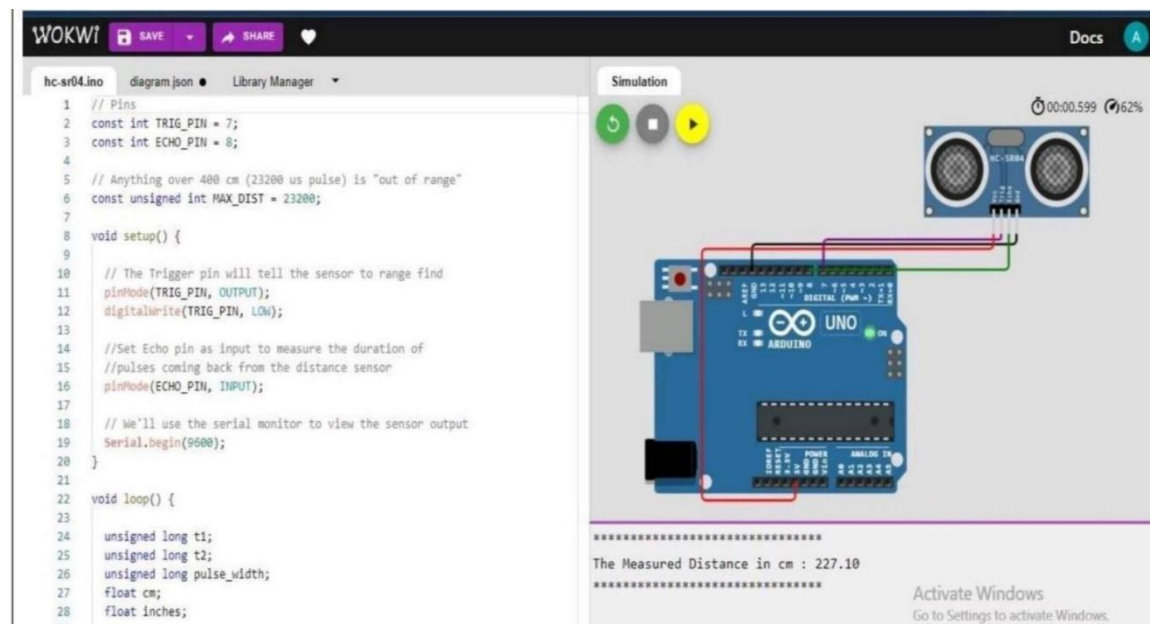
```

Output:

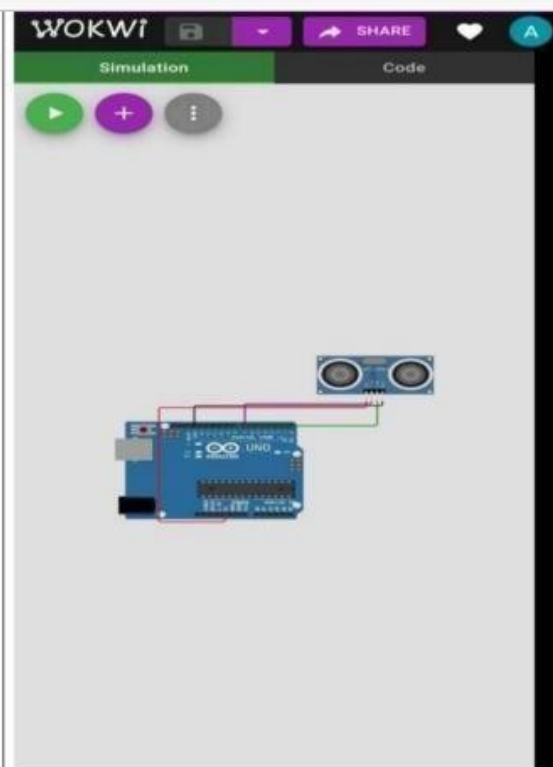
1.If the distance is less than 100 cm, it alerts.



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



3.Simulation and code execution



```
1 // Pin
2 const int TRIG_PIN = 5;
3 const int ECHO_PIN = 6;
4
5 // Anything over 400 cm (1300 us pulse) is "out of range"
6 const unsigned int MAX_DIST = 2200;
7
8 void setup() {
9   // The trigger pin will tell the sensor to range find
10  pinMode(TRIG_PIN, OUTPUT);
11  digitalWrite(TRIG_PIN, LOW);
12
13  // Set the pin as input to measure the duration of
14  // pulses, coming back from the distance sensor
15  pinMode(ECHO_PIN, INPUT);
16
17  // We'll use the serial monitor to view the sensor output
18  Serial.begin(9600);
19 }
20
21 void loop() {
22   unsigned long t1;
23   unsigned long t2;
24   unsigned long pulse_width;
25   float cm;
26   float inches;
27
28   // Hold the trigger pin high for at least 10 us
29   digitalWrite(TRIG_PIN, HIGH);
30   delayMicroseconds(10);
31   digitalWrite(TRIG_PIN, LOW);
32
33   // Wait for pulse on echo pin
34   while (!digitalRead(ECHO_PIN)) {}
35
36   // Measure how long the echo pin was held high (pulse width)
37   // In here, the distance sensor will overflow after ~30 us
38   t1 = micros();
39   while (digitalRead(ECHO_PIN)) {}
40   t2 = micros();
41   pulse_width = t2 - t1;
42
43   // Calculate distance in centimeters and inches. The constants
44   // are found in the datasheet, and calculated from the assumed speed
45   // of sound in air at sea level (340 m/s).
46   cm = pulse_width / 58.8;
47   inches = pulse_width / 148.0;
48
49   // Print out results
50   if (pulse_width > MAX_DIST) {
51     Serial.println("Out of range");
52   } else {
53     Serial.println("The measured distance in cm : ");
54     Serial.println(cm);
55     Serial.println("The measured distance in in : ");
56     Serial.println(inches);
57
58     if (cm < 100) {
59       Serial.println("Alert it !!");
60     }
61   }
62   Serial.println("=====");
63 }
64
65 // Wait at least 100ms before next measurement
66 delay(100);
67 }
```

