

IBM ASSIGNMENT- 4

TEAM ID: PNT2022TMID2663

NAME : BALAMURUGAN.U

ROLL NO: 212919106014

Write Code and connections in wokwi for ultrasonic sensor. whatever distance is less than 100 cms send an “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
```

```
Pin Mode(TRIG_PIN, OUTPUT); digital
```

```
Write(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);

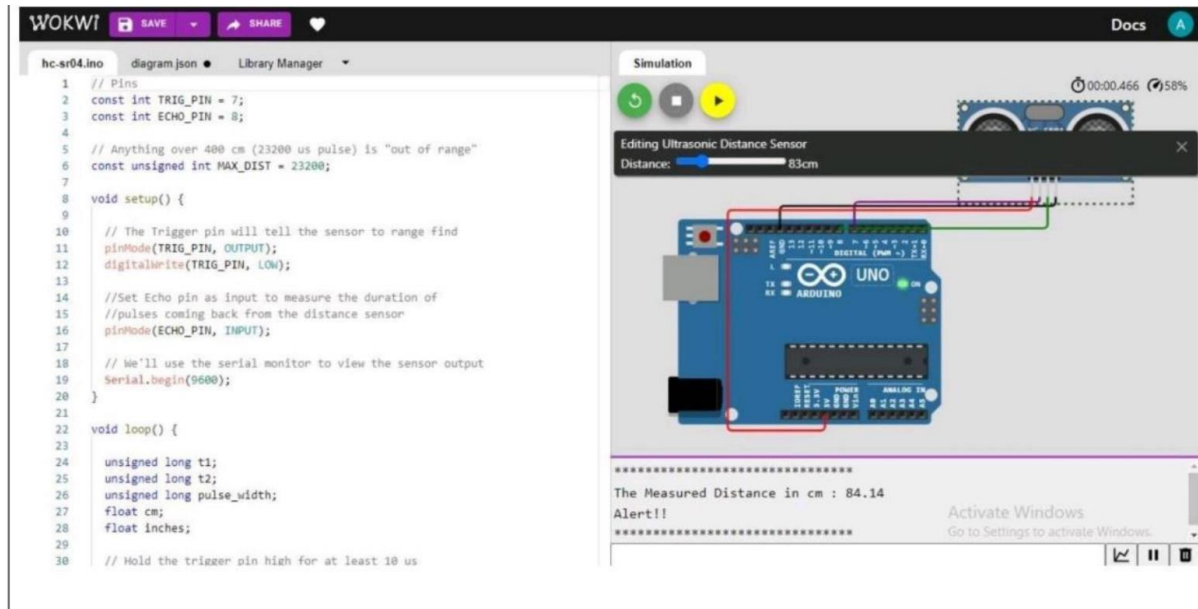
if( cm < 100 ){
    //while(true){
        Serial.println("Alert!!");
        //}
    }
Serial.print("*****");
}

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

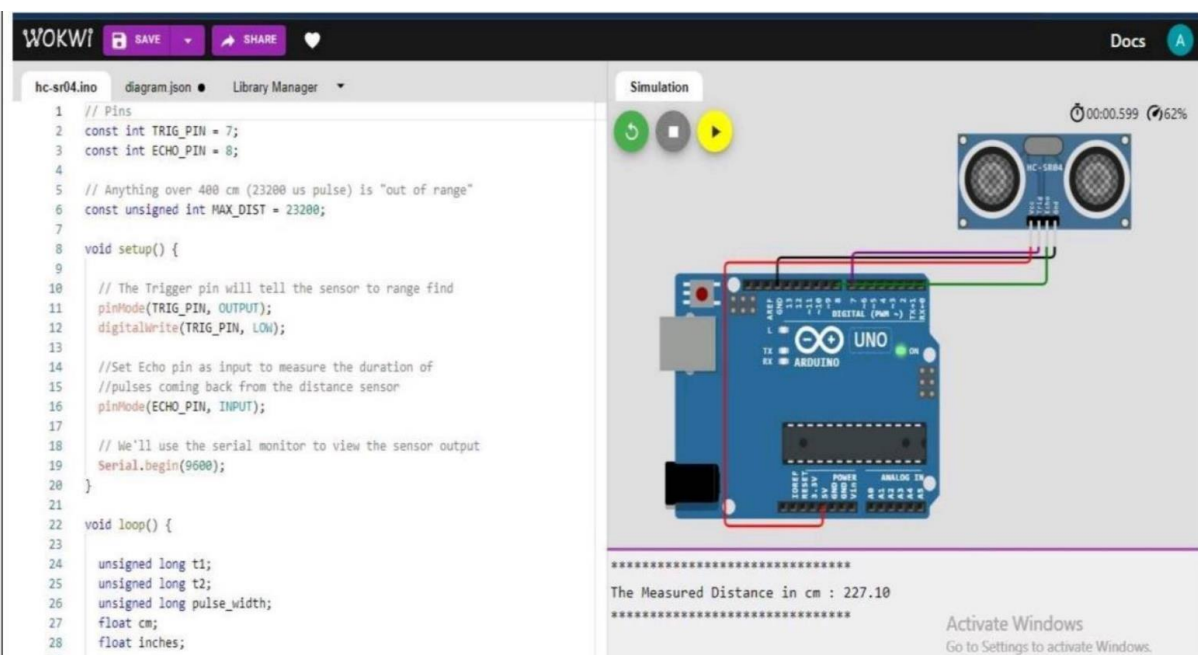
```

Output:

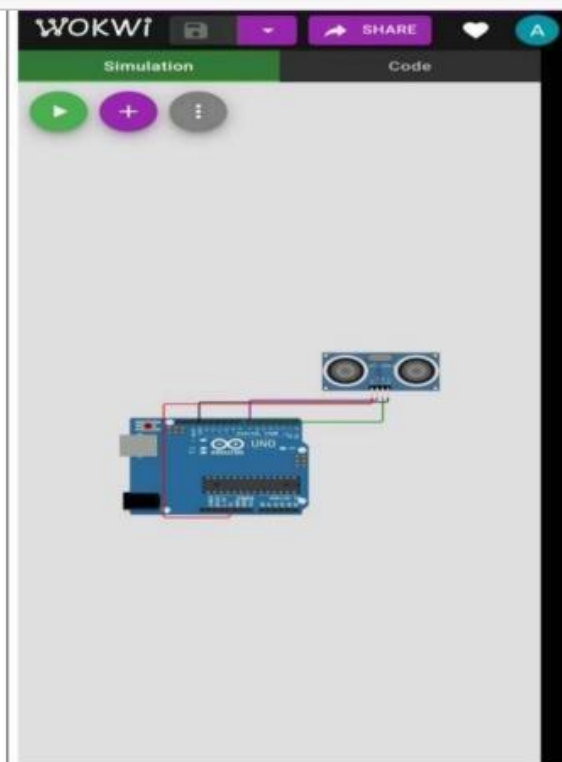
1.If the distance is less than 100 cms ,it alerts.



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



3.Simulation and code execution



```

1 // Pin
2 const int TRIG_PIN = 5;
3 const int ECHO_PIN = 4;
4 // Anything over 400 cm (1200 us pulse) is "out of range"
5 const unsigned int MAX_DIST = 2000;
6
7
8 void setup() {
9   // The Trigger pin will send the sensor to range find
10  pinMode(TRIG_PIN, OUTPUT);
11  digitalWrite(TRIG_PIN, LOW);
12
13  //Set Echo 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
22 void loop() {
23   unsigned long t1;
24   unsigned long t2;
25   unsigned long pulse_width;
26   float cm;
27   float inches;
28
29   // Hold the trigger pin high for at least 10 us
30   digitalWrite(TRIG_PIN, HIGH);
31   digitalWrite(TRIG_PIN, LOW);
32   digitalWrite(TRIG_PIN, LOW);
33
34   // Wait for pulse on echo pin
35   while ( !digitalRead(ECHO_PIN) ) {}
36
37   // Measure how long the echo pin was held high (pulse width)
38   // Note: the microcontroller's clock is 16 MHz, so 1 us = 16 ns
39   t1 = micros();
40   while ( digitalRead(ECHO_PIN) == HIGH ) {}
41   t2 = micros();
42   pulse_width = t2 - t1;
43
44   // Calculate distance in centimeters and inches. The constants
45   // are found in the datasheet, and calculated from the speed of sound
46   // of sound in air at sea level (340 m/s).
47   cm = pulse_width / 58.0;
48   inches = pulse_width / 148.0;
49
50   // Print our results
51   if ( pulse_width < MAX_DIST ) {
52     Serial.println("Out of range");
53   } else {
54     Serial.println("*****");
55     Serial.println("The Measured Distance is cm : ");
56     Serial.println(cm);
57   }
58
59   // Wait 100ms
60   // Serial.println("Waiting 100ms");
61   // delay(100);
62 }
63
64 // Wait at least 100ms before next measurement
65 delay(1000);
66
67
68
69
70

```

