

## **IBM ASSIGNMENT 4**

**TEAM LEADER : V.SANTHIYA (922519104138)**

**TEAM MEMBER 1 : A.SOWMIYA (922519104152)**

**TEAM MEMBER 2 : G.SOWMIYA (922519104153)**

**TEAM MEMBER 3 : K.YUVASHANKARI (922519104184)**

### **TOPIC :**

Write a code and connections in wokwi for ultrasonic sensor.

Whenever distance is less than 100 cms send an “alert” to IBM Cloud and display in device recent events.

### **Source Code:**

```
//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 will 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 atleast 10

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 constant are found in datasheet, and calculated the 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.println("The Measured Distance in cm");
Serial.println(cm);

if(cm<100){
while(true){

```

```

Serial.println("Alert!");
}

Serial.print("*****");
}
//wait atleast 1000ms before next measurement

Delay(1000);
}

```

## OUTPUT:

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

The screenshot displays the Wokwi online Arduino IDE interface. On the left, the 'sketch.ino' file contains the following code:

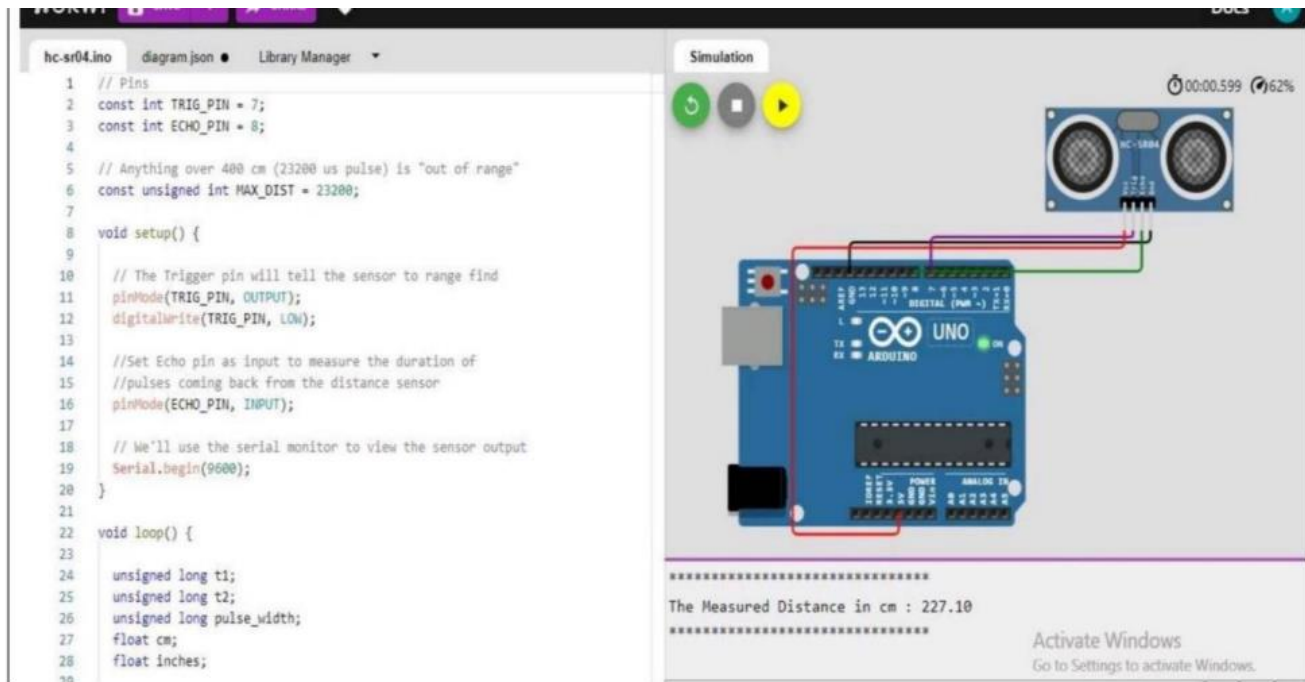
```

1 //pins
2 const int TRIG_PIN=7;
3 const int ECHO_PIN=8;
4
5 //Anything over 400cm (23200 us pulse) is "out of range"
6 const unsigned int MAX_DIST=23200;
7
8 void setup(){
9
10
11 //The trigger pin will tell the sensor to range find
12 pinMode(TRIG_PIN,OUTPUT);
13 digitalWrite(TRIG_PIN,LOW);
14
15 //Set echo pin as input to measure the duration of
16 //pulses coming back from the distance sensor
17 pinMode(ECHO_PIN, INPUT);
18
19 //We'll use the serial monitor to view the sensor output
20 Serial.begin(9600);
21 }
22 void loop(){
23   unsigned long t1;
24   unsigned long t2;
25   unsigned long pulse_width;
26   float cm;
27   float inches;
28   //Hold the trigger pin high for atleast 10 us
29   digitalWrite(TRIG_PIN,HIGH);

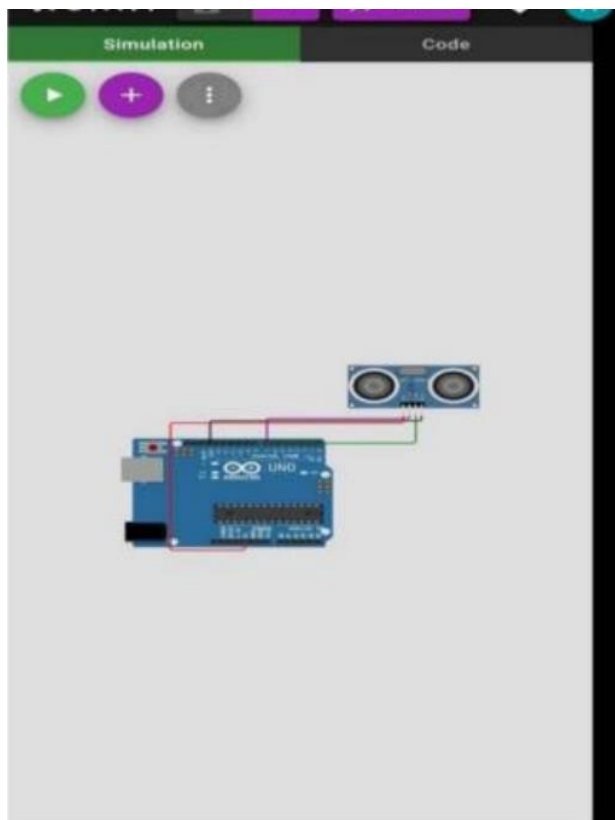
```

On the right, the 'Simulation' window shows an Arduino Uno board connected to an HC-SR04 ultrasonic sensor. The sensor's Vcc is connected to the 5V pin, Gnd to the GND pin, Trig to pin 7, and Echo to pin 8. A digital readout at the bottom of the simulation window displays: "The Measured Distance in cm : 201.79". The top right of the simulation window shows a timer at 01:11.599 and a battery level at 98%.

**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
5 // Anything over 400 cm (1200 in) will be "out of range"
6 const unsigned int MAX_DIST = 2000;
7
8 void setup() {
9
10   // The Trigger and Echo pins of the sensor to digital pins
11   pinMode(TRIG_PIN, OUTPUT);
12   pinMode(ECHO_PIN, INPUT);
13
14   // Set the 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 see 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
31   digitalWrite(TRIG_PIN, HIGH);
32   delayMicroseconds(10);
33   digitalWrite(TRIG_PIN, LOW);
34
35   // Wait for pulse on echo pin
36   while (digitalRead(ECHO_PIN) == 0);
37
38   // Measure how long the echo pin was held high (pulse width)
39   // Note: the Arduino's counter will overflow after ~55 ms
40   t1 = micros();
41   while (digitalRead(ECHO_PIN) == 1);
42   t2 = micros();
43   pulse_width = t2 - t1;
44
45   // Calculate distance in centimeters and inches. The constants
46   // are based on the speed of sound, and calculated from the assumed speed
47   // of sound in air at sea level (340 m/s).
48   cm = pulse_width / 59.0;
49   inches = pulse_width / 148.0;
50
51   // Print out results
52   if (pulse_width < MAX_DIST) {
53     Serial.println("Not at range");
54   } else {
55     Serial.println("*****");
56     Serial.print("The Measured Distance in cm : ");
57     Serial.println(cm);
58
59     if (cm > 100) {
60       Serial.println("Alert it !!");
61     }
62   }
63   Serial.println("*****");
64
65   // Wait at least 100ms before next measurement
66   delay(100);
67 }

```

