

SMART SOLUTION FOR RAILWAYS

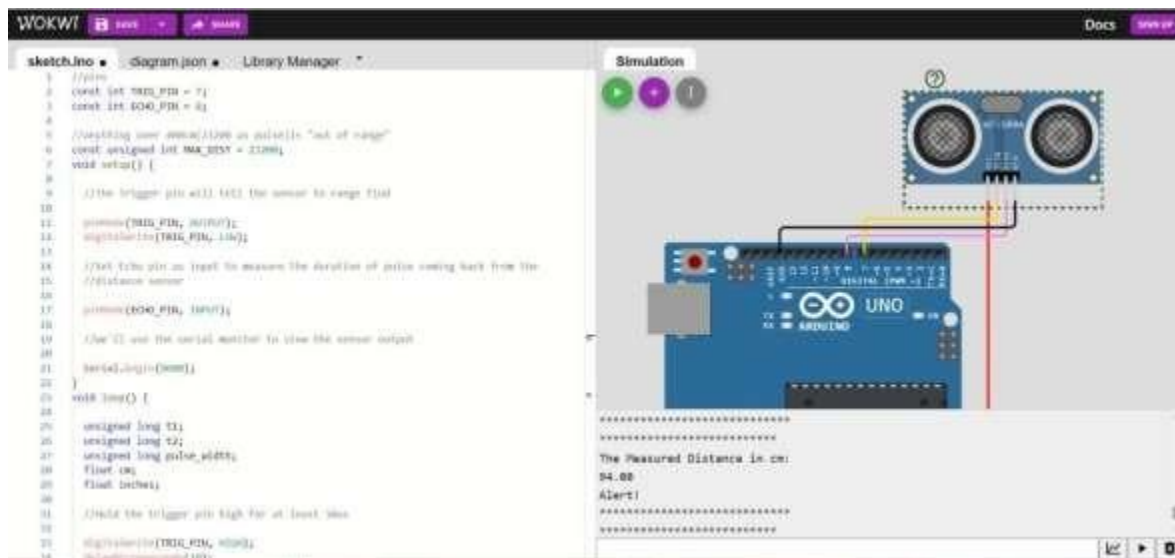
ASSIGNMENT-4

Date	31 October 2022
Team ID	PNT2022TMID37898
Student Name	Manoj Singh.B
Student Roll.No	410819104012

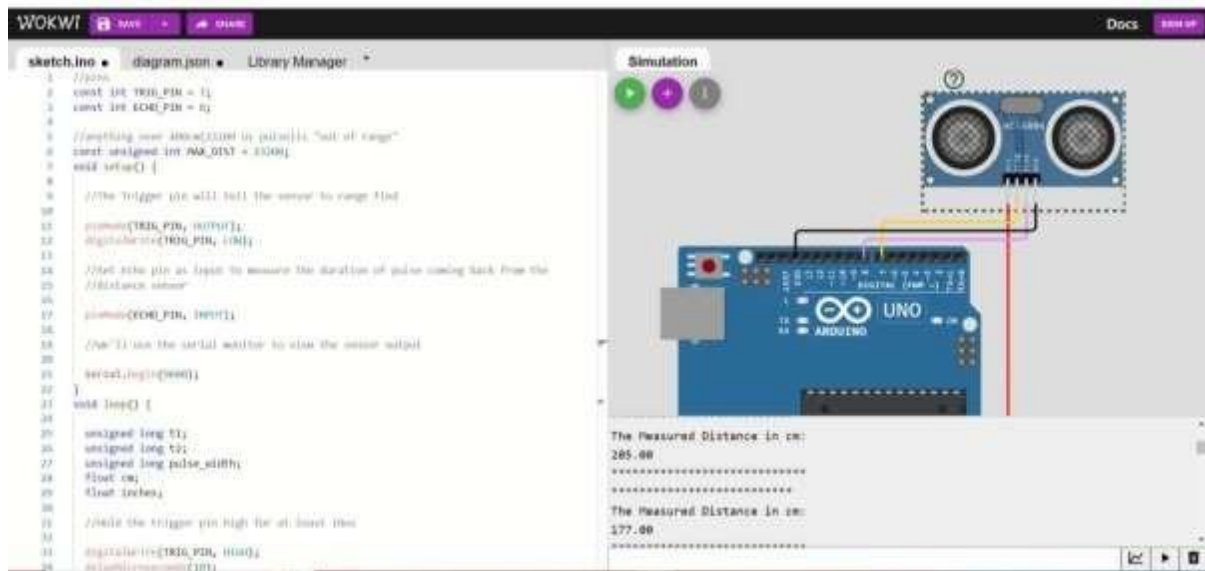
QUESTION:

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.

CASE 1: Distance less than 100cm → It Alerts



CASE 2: Distance more than 100cm → It won't Alert

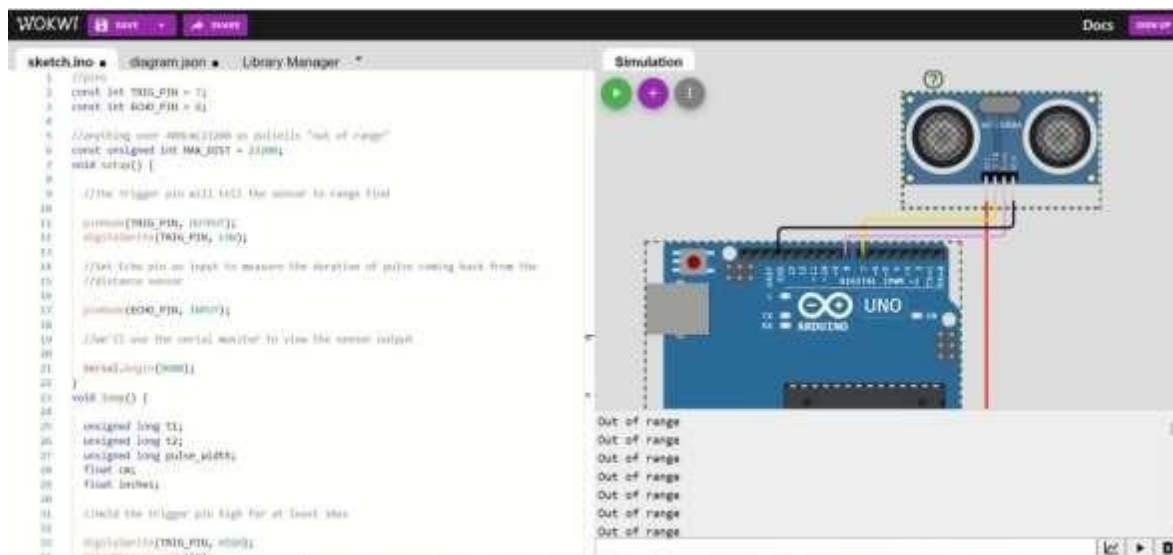


The screenshot shows the Wokwi simulation environment. On the left, the sketch.ino file contains the following code:

```
1 //pin
2 const int TRIG_PIN = 7;
3 const int ECHO_PIN = 6;
4
5 //anything over 4000/2000 is pulse is "out of range"
6 const unsigned int MAX_DIST = 1000;
7 void setup() {
8
9   //The trigger pin will tell the sensor to range find
10
11   pinMode(TRIG_PIN, OUTPUT);
12   digitalWrite(TRIG_PIN, LOW);
13
14   //Set echo pin as input to measure the duration of pulse coming back from the
15   //distance sensor
16
17   pinMode(ECHO_PIN, INPUT);
18
19   //We'll use the serial monitor to view the sensor output
20   Serial.begin(9600);
21 }
22
23 void loop() {
24
25   unsigned long t1;
26   unsigned long t2;
27   unsigned long pulse_width;
28   float cm;
29   float inches;
30
31   //Make the trigger pin high for at least 10us
32
33   digitalWrite(TRIG_PIN, HIGH);
34   delayMicroseconds(10);
```

On the right, the simulation window shows the Arduino Uno board connected to the HC-SR04 sensor. The sensor's output is displayed as "The Measured Distance in cm: 285.00".

CASE 3: Beyond limits → Out of Range



The screenshot shows the Wokwi simulation environment. On the left, the sketch.ino file contains the following code:

```
1 //pin
2 const int TRIG_PIN = 7;
3 const int ECHO_PIN = 6;
4
5 //anything over 4000/2000 is pulse is "out of range"
6 const unsigned int MAX_DIST = 1000;
7 void setup() {
8
9   //The trigger pin will tell the sensor to range find
10
11   pinMode(TRIG_PIN, OUTPUT);
12   digitalWrite(TRIG_PIN, LOW);
13
14   //Set echo pin as input to measure the duration of pulse coming back from the
15   //distance sensor
16
17   pinMode(ECHO_PIN, INPUT);
18
19   //We'll use the serial monitor to view the sensor output
20   Serial.begin(9600);
21 }
22
23 void loop() {
24
25   unsigned long t1;
26   unsigned long t2;
27   unsigned long pulse_width;
28   float cm;
29   float inches;
30
31   //Make the trigger pin high for at least 10us
32
33   digitalWrite(TRIG_PIN, HIGH);
34   delayMicroseconds(10);
```

On the right, the simulation window shows the Arduino Uno board connected to the HC-SR04 sensor. The sensor's output is displayed as "Out of range".

CODING:

```
const int TRIG_PIN = 7;
const int ECHO_PIN = 8;

const unsigned int MAX_DIST = 23200;
void setup() {

    pinMode(TRIG_PIN, OUTPUT);
    digitalWrite(TRIG_PIN, LOW);

    pinMode(ECHO_PIN, INPUT);

    Serial.begin(9600);
}
void loop() {

    unsigned long t1;
    unsigned long t2;
    unsigned long pulse_width;
    float cm;
    float inches;
    digitalWrite(TRIG_PIN, HIGH);
    delayMicroseconds(10);
    digitalWrite(TRIG_PIN, LOW);
    while (digitalRead(ECHO_PIN) == 0)
t1 = micros();
    while (digitalRead(ECHO_PIN) == 1);
    t2 = micros();
    pulse_width = t2 - t1;
    cm = pulse_width / 58;
    inches = pulse_width / 148.0;

    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.println("*****");  
}  
  
//wait at least 1000ms before next measurement  
  
delay(1000);  
}
```

CIRCUIT:

