

## ASSIGNMENT-4

### B7-1A3E DINESH U

Write code and connections in wokwi for ultrasonic sensor. Whenever distance is less than 100cms, send “alert” to ibm cloud and display in device recent events.

Code;

```
#define ECHO_PIN 2
#define TRIG_PIN 3

void setup() {
  Serial.begin(115200);
  pinMode(LED_BUILTIN, OUTPUT);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
}

float readDistanceCM() {
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  int duration = pulseIn(ECHO_PIN, HIGH);
  return duration * 0.034 / 2;
}

void loop() {
  float distance = readDistanceCM();

  bool isNearby = distance < 100;
```

```

digitalWrite(LED_BUILTIN, isNearby);

Serial.print("Measured distance: ");
Serial.println(readDistanceCM());
if (distance<100)
    Serial.print("alert");
Serial.println();

delay(100);
}

```

## OUTPUT;

The screenshot displays the Wokwi Arduino IDE interface. On the left, the code for 'hc-sr04.ino' is shown, which defines pins for the ultrasonic sensor and implements a loop to measure distance and trigger an LED if the distance is less than 100cm. On the right, the simulation window shows an Arduino Uno connected to an HC-SR04 sensor. The serial monitor displays the output of the code, showing the measured distance and an alert message when the distance is less than 100cm.

```

1
2 #define ECHO_PIN 2
3 #define TRIG_PIN 3
4
5 void setup() {
6   Serial.begin(115200);
7   pinMode(LED_BUILTIN, OUTPUT);
8   pinMode(TRIG_PIN, OUTPUT);
9   pinMode(ECHO_PIN, INPUT);
10 }
11
12 float readDistanceCM() {
13   digitalWrite(TRIG_PIN, LOW);
14   delayMicroseconds(2);
15   digitalWrite(TRIG_PIN, HIGH);
16   delayMicroseconds(10);
17   digitalWrite(TRIG_PIN, LOW);
18   int duration = pulseIn(ECHO_PIN, HIGH);
19   return duration * 0.034 / 2;
20 }
21
22 void loop() {
23   float distance = readDistanceCM();
24
25   bool isNearby = distance < 100;
26   digitalWrite(LED_BUILTIN, isNearby);
27
28   Serial.print("Measured distance: ");
29   Serial.println(readDistanceCM());
30   if (distance<100)
31     Serial.print("alert");

```

Simulation output (Serial Monitor):

```

alert
Measured distance: 73.87
alert
Measured distance: 73.87
alert
Measured distance: 73.87
alert

```

