

ASSIGNMENT- 4

CODE AND CONNECTION IN WOKWI FOR THE ULTRASONIC SENSORS

CODE:

```
#include "Arduino.h"
#include "HCSR04.h"
```

```
UltraSonicDistanceSensor::UltraSonicDistanceSensor(
    byte triggerPin, byte echoPin, unsigned short maxDistanceCm, unsigned
    long maxTimeoutMicroSec) {
    this->triggerPin = triggerPin;
    this->echoPin = echoPin;
    this->maxDistanceCm = maxDistanceCm;
    this->maxTimeoutMicroSec = maxTimeoutMicroSec;
    pinMode(triggerPin, OUTPUT);
    pinMode(echoPin, INPUT);
}
```

```
float UltraSonicDistanceSensor::measureDistanceCm() {
    //Using the approximate formula  $19.307^{\circ}\text{C}$  results in roughly 343m/s which
    is the commonly used value for air.
    return measureDistanceCm(19.307);
}
```

```
float UltraSonicDistanceSensor::measureDistanceCm(float temperature) {
    unsigned long maxDistanceDurationMicroSec;
```

```
digitalWrite(triggerPin, LOW);
delayMicroseconds(2);
```

```

digitalWrite(triggerPin, HIGH);
delayMicroseconds(10);
digitalWrite(triggerPin, LOW);
float speedOfSoundInCmPerMicroSec = 0.03313 + 0.0000606 * temperature;
maxDistanceDurationMicroSec = 2.5 * maxDistanceCm /
speedOfSoundInCmPerMicroSec;
if (maxTimeoutMicroSec > 0) {
    maxDistanceDurationMicroSec = min(maxDistanceDurationMicroSec,
maxTimeoutMicroSec);
}

unsigned long durationMicroSec = pulseIn(echoPin, HIGH,
maxDistanceDurationMicroSec); // can't measure beyond max distance

float distanceCm = durationMicroSec / 2.0 *
speedOfSoundInCmPerMicroSec;
if (distanceCm == 0 || distanceCm > maxDistanceCm) {
    return -1.0 ;
} else {
    return distanceCm;
}
}

```