**ASSIGNMENT 4**

|  |  |
| --- | --- |
| Assignment Date | 29-10-2022 |
| Student Name | SIVASANKARESAN |
| Team ID | PNT2022TMID17342 |
| Project Name | IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICALTURE |
| Maximum marks | 2 Marks |

**1.Write Code and connections in wokwi for ultrasonic sensor. whatever distance is less than 100 cms send “Alert” to ibm cloud aand 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(“\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*”);

}

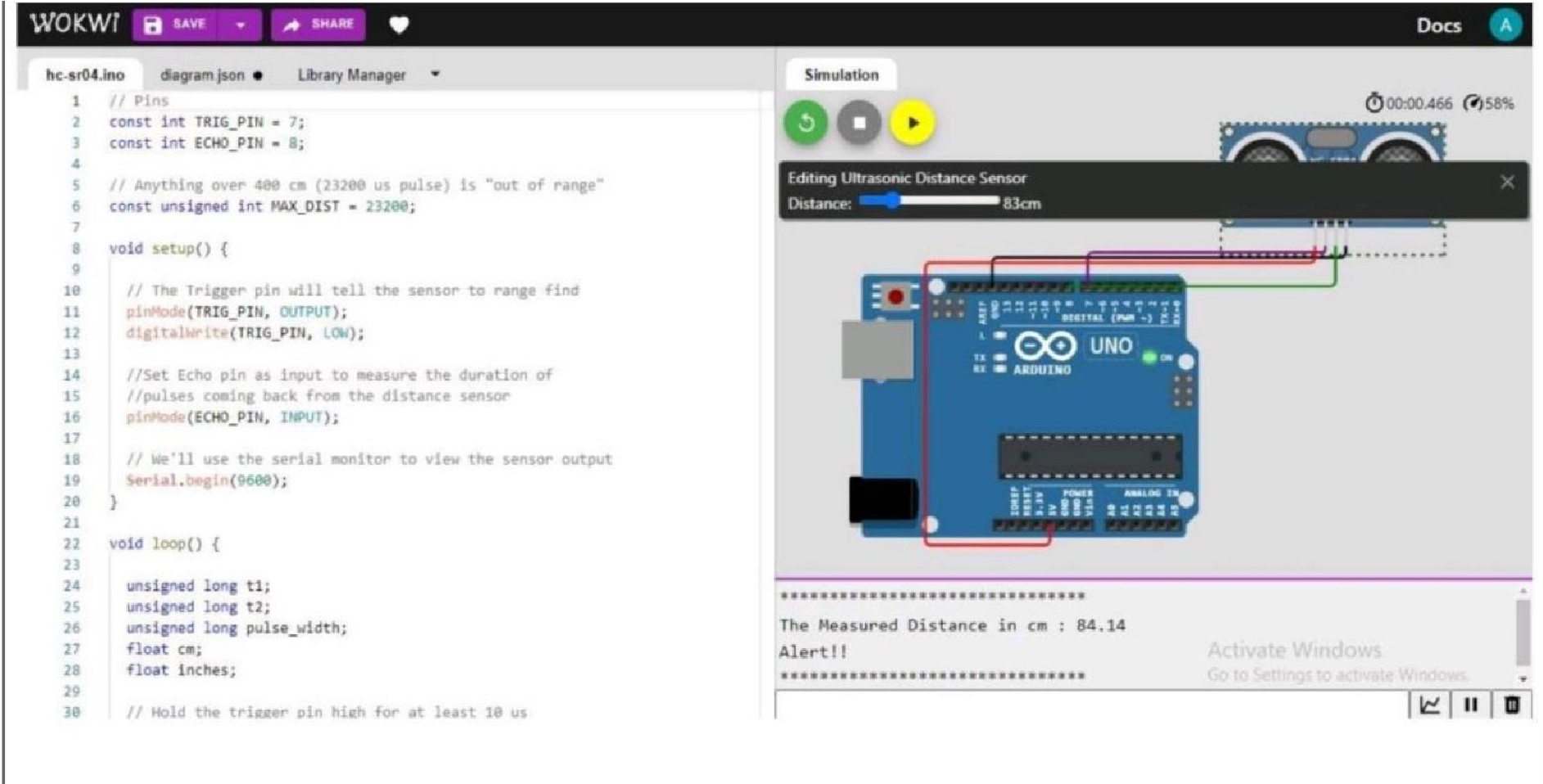
//wait at least 1000ms before next measurement

Delay(1000);

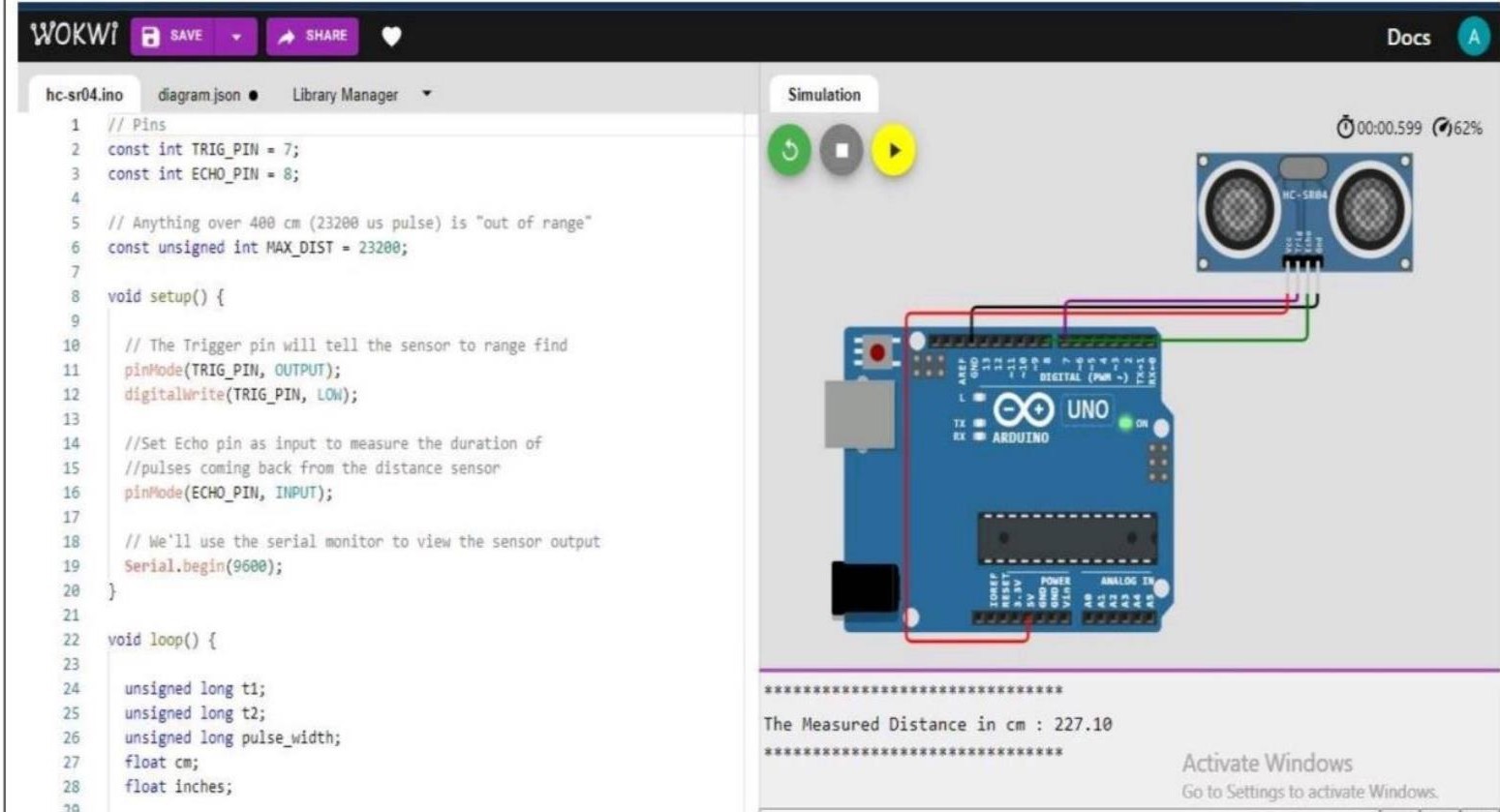
}

**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**

