**Flood detection / flood monitoring system using esp32**

**Program:-**

**#include <Arduino.h>**

**// Pin definitions**

**#define TRIG\_PIN 5**

**#define ECHO\_PIN 18**

**#define LED1\_PIN 2**

**#define LED2\_PIN 4**

**// Function to get distance from the HC-SR04 sensor**

**long getDistance() {**

**// Clear the trigPin by setting it LOW**

**digitalWrite(TRIG\_PIN, LOW);**

**delayMicroseconds(2);**

**// Trigger the sensor by setting the trigPin HIGH for 10 microseconds**

**digitalWrite(TRIG\_PIN, HIGH);**

**delayMicroseconds(10);**

**digitalWrite(TRIG\_PIN, LOW);**

**// Read the echoPin, and return the sound wave travel time in microseconds**

**long duration = pulseIn(ECHO\_PIN, HIGH);**

**// Calculate the distance (in cm) based on the speed of sound (340 m/s)**

**long distance = duration \* 0.034 / 2;**

**return distance;**

**}**

**void setup() {**

**// Initialize serial communication at 115200 bits per second**

**Serial.begin(115200);**

**// Configure pin modes**

**pinMode(TRIG\_PIN, OUTPUT);**

**pinMode(ECHO\_PIN, INPUT);**

**pinMode(LED1\_PIN, OUTPUT);**

**pinMode(LED2\_PIN, OUTPUT);**

**}**

**void loop() {**

**// Get the distance from the HC-SR04 sensor**

**long distance = getDistance();**

**Serial.print("Distance: ");**

**Serial.print(distance);**

**Serial.println(" cm");**

**// Define distance thresholds for flood warning**

**const long SAFE\_DISTANCE = 100; // Safe distance in cm**

**const long WARNING\_DISTANCE = 50; // Warning distance in cm**

**// Control LEDs based on distance**

**if (distance > SAFE\_DISTANCE) {**

**// Safe distance**

**digitalWrite(LED1\_PIN, LOW); // Green LED off**

**digitalWrite(LED2\_PIN, LOW); // Red LED off**

**} else if (distance > WARNING\_DISTANCE && distance <= SAFE\_DISTANCE) {**

**// Warning distance**

**digitalWrite(LED1\_PIN, HIGH); // Green LED on**

**digitalWrite(LED2\_PIN, LOW); // Red LED off**

**} else {**

**// Flood distance**

**digitalWrite(LED1\_PIN, LOW); // Green LED off**

**digitalWrite(LED2\_PIN, HIGH); // Red LED on**

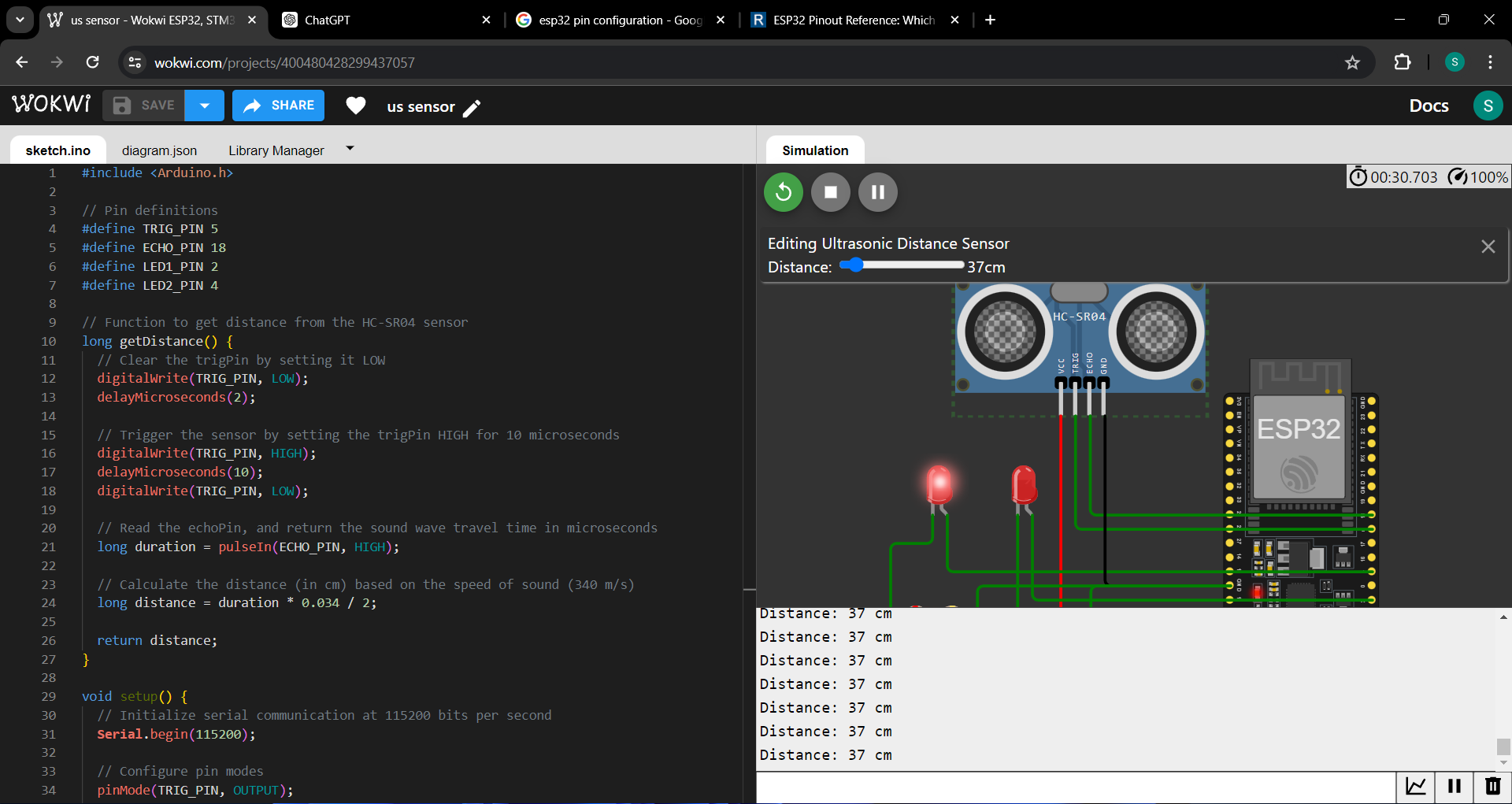
**}**

**// Wait for 500 milliseconds before the next loop**

**delay(500);**

**}**

**Working:-**

****