### **Ultrasonic Sensor-Based Distance Measurement Using ESP32 and Wokwi PlatformIO in Visual Studio Code**

*Candra Wahyu Perdana*

Vocational Faculty, Brawijaya University

Email: candraega28@gmail.com

**Abstract**

This experiment aims to implement a distance measurement system using an ultrasonic sensor (HC-SR04) with an ESP32 microcontroller. The system calculates the distance by measuring the time taken for an ultrasonic pulse to travel to an object and back. The project is developed and simulated using Wokwi PlatformIO in Visual Studio Code. Experimental results show that the system accurately measures distances in centimeters and inches, and proper configuration of the trigger and echo pins is essential for reliable readings.

*Keywords—ESP32, Ultrasonic Sensor, HC-SR04, Distance Measurement, Wokwi PlatformIO, Visual Studio Code*

**1. Introduction**

**1.1 Background**

The Internet of Things (IoT) has enabled the automation of various applications, including smart measurement systems. Ultrasonic sensors such as the HC-SR04 are widely used for distance measurement in robotics, security systems, and industrial automation. The ESP32 microcontroller, with its powerful processing capabilities, allows precise data collection and analysis. By integrating the ultrasonic sensor with ESP32, we can measure distances effectively in both centimeters and inches.

The Wokwi PlatformIO in Visual Studio Code is utilized as the development and simulation environment, enabling real-time testing and debugging of the distance measurement system.

**1.2 Objectives**

* Understand the working principle of the HC-SR04 ultrasonic sensor.
* Implement distance measurement using an ESP32 and display the results via the Serial Monitor.
* Simulate and test the system using Wokwi PlatformIO in Visual Studio Code.

**2. Methodology**

**2.1 Tools & Materials**

**Hardware:**

* ESP32 (simulated using Wokwi PlatformIO)
* HC-SR04 Ultrasonic Sensor
* Jumper Wires

**So**ftware:

* Visual Studio Code with the PlatformIO extension
* Wokwi PlatformIO as the simulation environment
* Arduino IDE (optional, for reference)

**2.2 Implementation Steps**

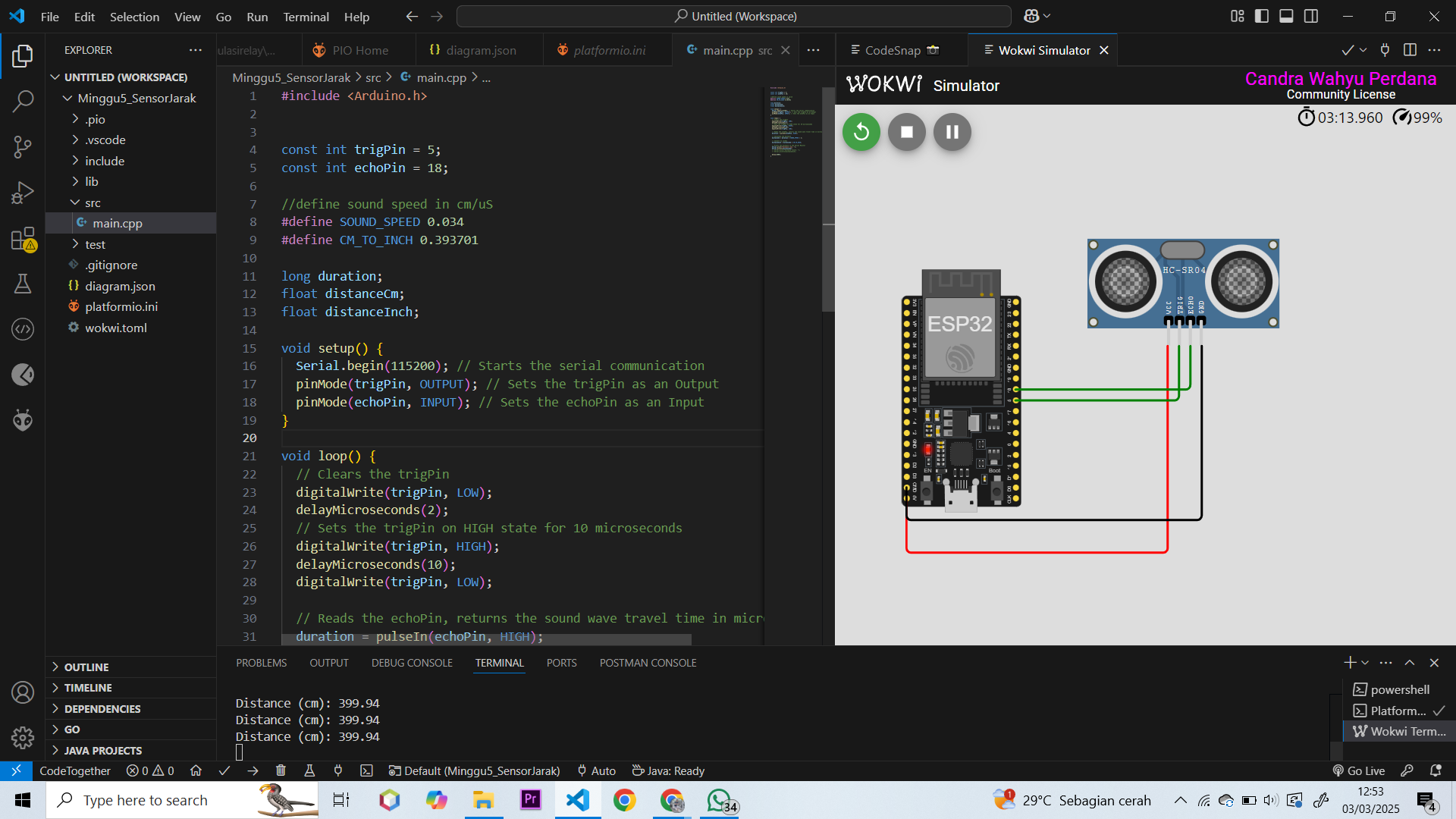
1. **Project Creation:**
   * Open Visual Studio Code and ensure that the PlatformIO extension is installed.
   * Create a new PlatformIO project for the ESP32 board and integrate the simulation using Wokwi PlatformIO.
2. **Circuit Connection:**
   * Connect the **Trig** pin of the HC-SR04 sensor to **GPIO5** on the ESP32.
   * Connect the **Echo** pin to **GPIO18** on the ESP32.
   * Ensure the sensor operates correctly by configuring the GPIO pins properly.
3. **Code Development:**
   * Use the following code to send ultrasonic pulses, measure the duration of the echo, and calculate the distance in both centimeters and inches.
4. **Simulation & Verification:**
   * Run the simulation using Wokwi PlatformIO in Visual Studio Code.
   * Check the Serial Monitor output to ensure the distance is being measured correctly.



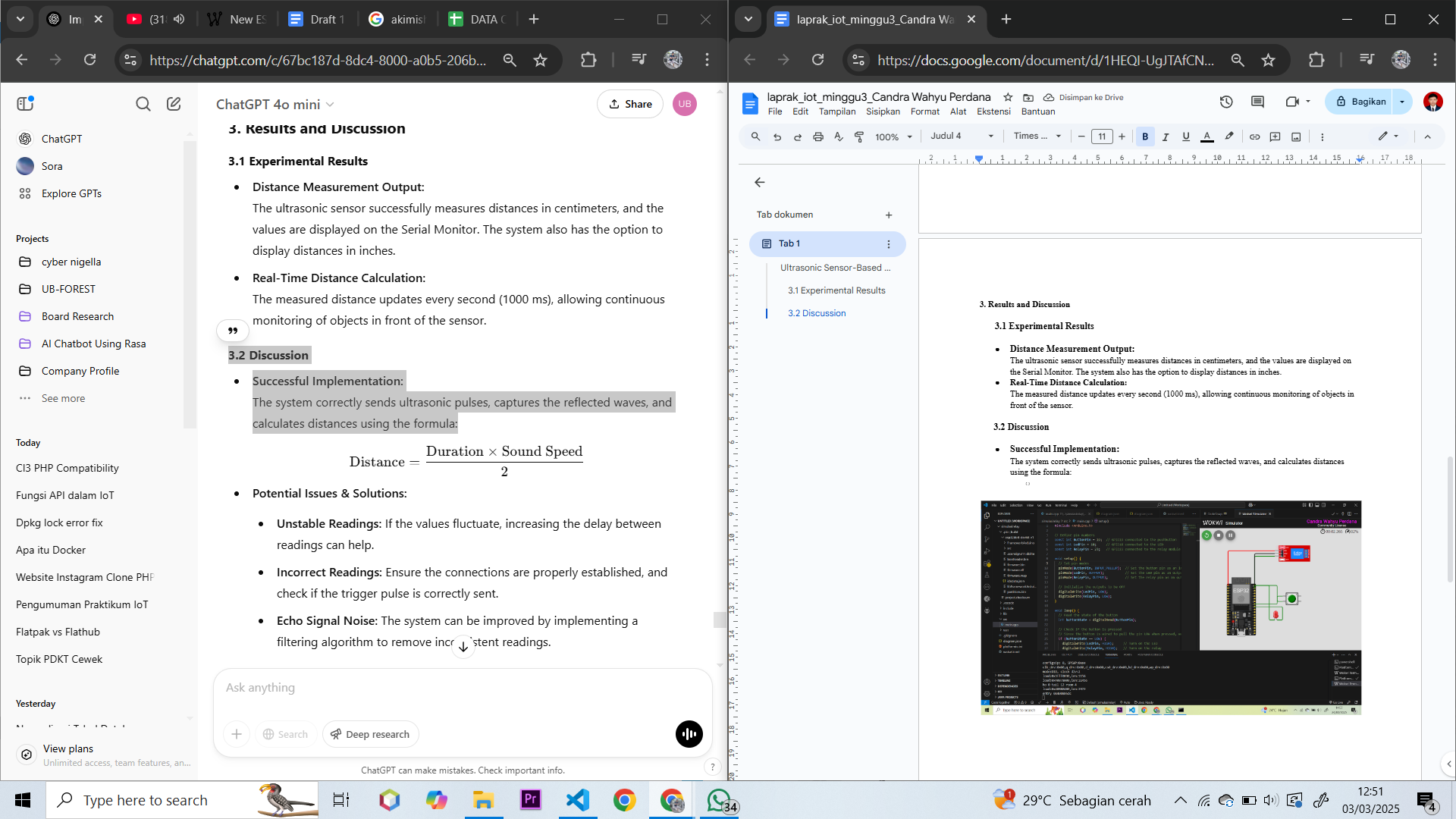
**3. Results and Discussion**

#### **3.1 Experimental Results**

* **Distance Measurement Output:**The ultrasonic sensor successfully measures distances in centimeters, and the values are displayed on the Serial Monitor. The system also has the option to display distances in inches.
* **Real-Time Distance Calculation:**The measured distance updates every second (1000 ms), allowing continuous monitoring of objects in front of the sensor.



#### **3.2 Discussion**

* **Successful Implementation:**The system correctly sends ultrasonic pulses, captures the reflected waves, and calculates distances using the formula: 
* **Potential Issues & Solutions:**
  + **Unstable Readings:** If the values fluctuate, increasing the delay between readings can help.
  + **Incorrect Readings:** Ensure the connections are properly established, and check if the trigger pulse is correctly sent.
  + **Echo Signal Noise:** The system can be improved by implementing a filtering algorithm to remove inconsistent readings.

**4. Appendix**

#include <Arduino.h>

const int trigPin = 5;

const int echoPin = 18;

//define sound speed in cm/uS

#define SOUND\_SPEED 0.034

#define CM\_TO\_INCH 0.393701

long duration;

float distanceCm;

float distanceInch;

void setup() {

Serial.begin(115200); // Starts the serial communication

pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output

pinMode(echoPin, INPUT); // Sets the echoPin as an Input

}

void loop() {

// Clears the trigPin

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

// Sets the trigPin on HIGH state for 10 microseconds

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds

duration = pulseIn(echoPin, HIGH);

// Calculate the distance

distanceCm = duration \* SOUND\_SPEED / 2;

// Convert to inches

distanceInch = distanceCm \* CM\_TO\_INCH;

// Prints the distance in the Serial Monitor

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

Serial.println(distanceCm);

// Serial.print("Distance (inch): ");

// Serial.println(distanceInch);

delay(1000);

}

This report describes the implementation of an ultrasonic distance measurement system using an ESP32 microcontroller and an HC-SR04 sensor, with simulation and development carried out in Wokwi PlatformIO in Visual Studio Code. The system successfully calculates distances using the time-of-flight principle and outputs accurate readings on the Serial Monitor. Future improvements could include noise filtering and real-world testing with additional components like an LCD display or wireless data transmission.