LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya



**Projeck Simulasi Sensor Jarak**

**(Ultrasonic) mengunakan ESP32**

*Andrian Alfini*

Fakultas Vokasi, Universitas Brawijaya

Email : andrianalfian2003@gmail.com

**Abstrak**

Sensor ultrasonik merupakan salah satu jenis sensor yang sering digunakan untuk mengukur jarak suatu objek dengan prinsip pemantulan gelombang suara. Pada proyek ini, dilakukan simulasi pengukuran jarak menggunakan sensor ultrasonik HC-SR04 yang dihubungkan dengan ESP32. Hasil pengukuran akan ditampilkan secara real-time Hasil pengujian menunjukkan bahwa sistem dapat mengukur jarak.

**Kata Kunci:** *ESP32,* *Sensor Ultrasonik HC-SR04 , Wokwi, PlatformIO, vscode.*

**1.Pendahuluan**

* 1. **Latar belakang**

Sensor ultrasonik telah menjadi salah satu komponen penting dalam berbagai aplikasi, terutama dalam sistem otomatisasi, Teknologi ini dapat menjadi dasar dalam pengembangan sistem yang lebih kompleks, seperti kendaraan otonom, alat bantu navigasi, dan sistem penghindaran rintangan.

**1.2 Tujuan eksperimen**

1. Memahami penggunaan Wokwi sebagai simulator mikrokontroler.
2. Menggunakan PlatformIO di VS Code untuk pemrograman ESP32.
3. Mengimplementasikan Sensor Ultrasonik HC-SR04 pada mikrokontroler ESP32 ntuk mengukur jarak objek secara real-time.

**2. Methodology (Metodologi)**

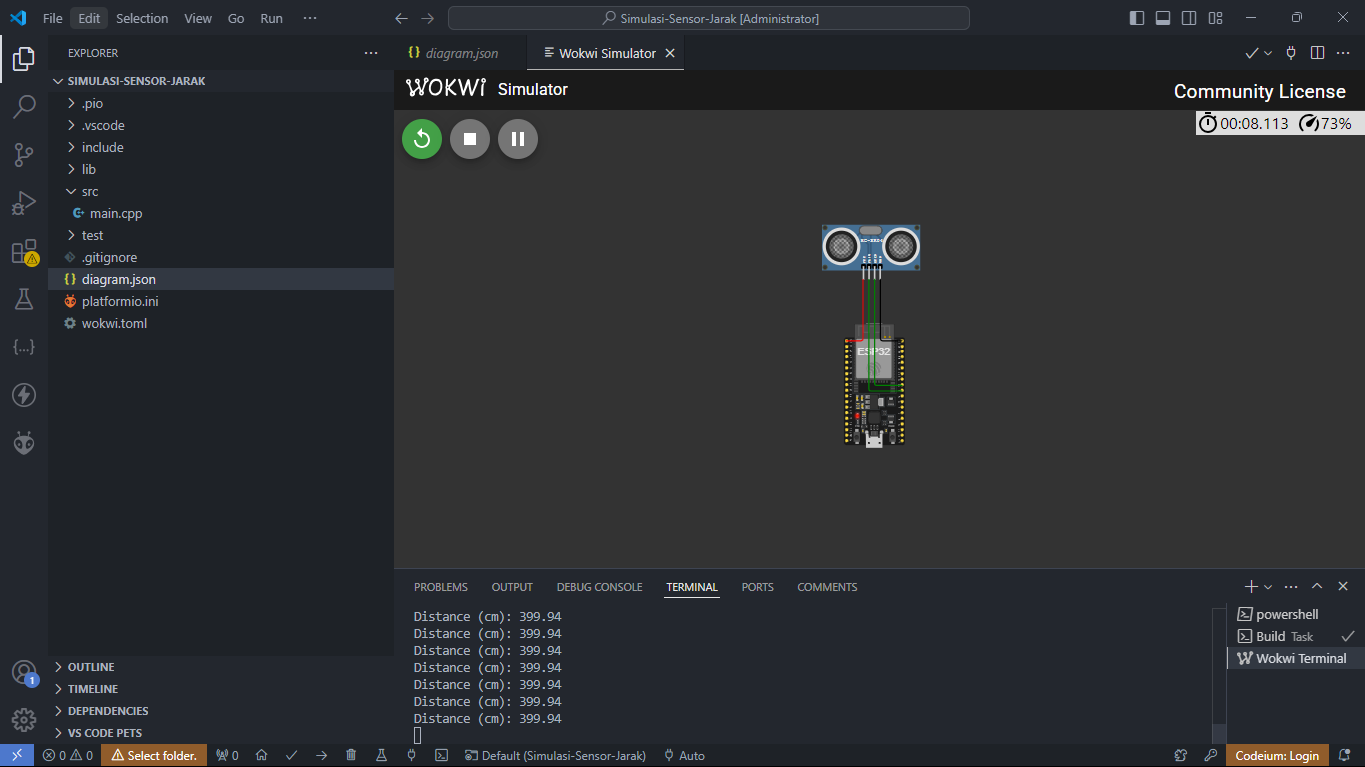
**2.1 Tools & Materials (Alat dan Bahan)**

* ESP32 (simulasi)
* Sensor Ultrasonic HC-SR04
* Wokwi Simulator
* PlatformIO di VS Code dan codingan
  1. **Implementation Steps (Langkah Implementasi)**

1. Membuat proyek baru di Wokwi dan PlatformIO.
2. Menghubungkan Sensor Ultrasonic HC-SR04 ke ESP32 dalam Wokwi.
3. Menulis kode program untuk sensor jarak
4. Menjalankan simulasi dan memastikan data ditampilkan pada serial monitor.

**3. Results and Discussion (Hasil dan Pembahasan)**

**3.1 Experimental Results**



**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 micro seconds

 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);

}