LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya



**Praktik Simulasi Relay, Button & LED**  
*Bilal Al Ihsan*

Fakultas Vokasi, Universitas Brawijaya

Email : bilalihsan@student.ub.ac.id

**Abstrak**

Sensor jarak ultrasonik digunakan untuk mengukur jarak antara sensor dan objek yang ada di sekitarnya menggunakan gelombang suara. Sensor ini bekerja dengan mengirimkan gelombang suara melalui Trigger Pin dan mengukur waktu yang dibutuhkan gelombang suara untuk kembali setelah memantul dari objek. Sensor HC-SR04 adalah salah satu sensor ultrasonik yang banyak digunakan dalam berbagai aplikasi, seperti robotika dan sistem pengukuran jarak.

Pada praktikum ini, ESP32 akan digunakan untuk mensimulasikan sensor ultrasonik HC-SR04 dalam pengukuran jarak, sepenuhnya tanpa perangkat fisik, menggunakan simulasi di Wokwi dengan VS Code dan PlatformIO. Praktikum ini bertujuan untuk memahami cara kerja sensor ultrasonik dan cara menghubungkannya dengan mikrokontroler untuk mendapatkan hasil pengukuran jarak.

Kata kunci-- *ESP32, Ultasonic, ,* *HC-SR04, PlatformIO, Wokwi*

**1. Introduction**

**1.1 Latar belakang**

Sensor jarak adalah komponen yang sering digunakan dalam berbagai aplikasi, seperti robotika, kendaraan otomatis, dan sistem pengukuran jarak. Salah satu jenis sensor jarak yang umum digunakan adalah sensor ultrasonik, yang mengukur jarak dengan menggunakan gelombang suara. Sensor ini bekerja dengan mengirimkan gelombang suara dan mengukur waktu yang dibutuhkan gelombang tersebut untuk kembali setelah memantul dari objek.

Pada praktikum ini, kita akan mensimulasikan penggunaan sensor ultrasonik dengan ESP32 menggunakan Wokwi di VS Code dengan PlatformIO, tanpa memerlukan perangkat fisik. Simulasi ini akan mengukur jarak dan menampilkan hasilnya di Serial Monitor.

.

**1.2 Tujuan eksperimen**

Praktikum ini bertujuan untuk:

1. Memahami cara kerja sensor ultrasonik dalam pengukuran jarak.
2. Menggunakan ESP32 untuk membaca data dari sensor ultrasonik.
3. Mengimplementasikan pemrograman menggunakan PlatformIO untuk menghubungkan sensor ultrasonik ke ESP32.
4. Menjalankan simulasi virtual di Wokwi untuk memvisualisasikan proses pengukuran jarak.

**2. Methodology**

**2.1 Tools & Materials**

Dalam praktikum ini, alat dan bahan yang digunakan meliputi:

Alat:

1. ESP32 Development Board (simulasi di Wokwi)
2. VS Code dengan PlatformIO
3. Wokwi sebagai simulator rangkaian

Bahan:

1. Sensor Ultrasonik HC-SR04 (simulasi di Wokwi)

**2.2 Implementation Steps (Langkah Implementasi**

1. Setup Proyek di PlatformIO (VS Code)

* Install PlatformIO Extension di VS Code jika belum terpasang.
* Buat proyek baru di PlatformIO dengan memilih board ESP32 dan pastikan Wokwi sudah terintegrasi dalam proyek tersebut.
* Buat file konfigurasi wokwi.toml untuk memilih sensor ultrasonik dan pin yang digunakan.

2. Konfigurasi Simulasi di Wokwi

* Sensor Ultrasonik HC-SR04 dihubungkan ke ESP32 sebagai berikut:
* VCC → 5V
* GND → GND
* Trig → GPIO 5 ESP32
* Echo → GPIO 18 ESP32

3. Pemrograman ESP32

* Program menggunakan PlatformIO untuk mengaktifkan Trigger (Trig) pada sensor ultrasonik dan membaca hasil dari Echo untuk menghitung jarak berdasarkan waktu yang dibutuhkan gelombang ultrasonik untuk kembali.
* Gunakan Serial Monitor untuk menampilkan hasil pengukuran jarak dalam satuan cm.

4. Menjalankan Simulasi

* Jalankan simulasi di Wokwi dan pantau hasilnya di Serial Monitor.
* Pastikan hasil pengukuran jarak yang ditampilkan sesuai dengan jarak virtual yang ditentukan dalam simulasi.

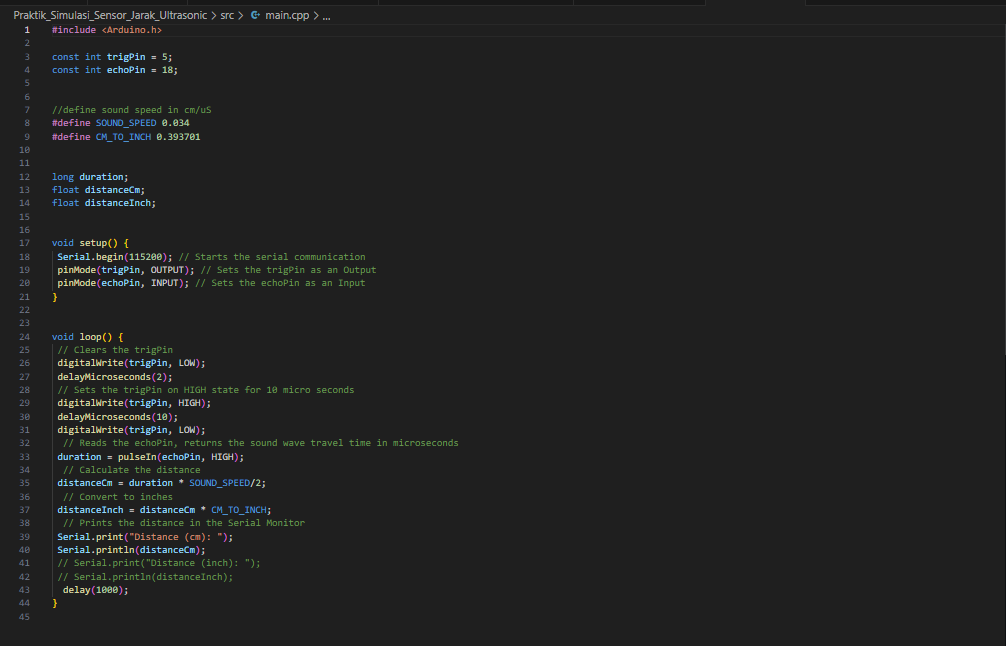
**3. Results and Discussion**

Hasil pengujian menunjukkan bahwa:

1. ESP32 berhasil mengukur jarak dengan akurasi yang baik dan mengonversi waktu yang dibutuhkan gelombang ultrasonik untuk kembali menjadi jarak.
2. Simulasi berhasil menunjukkan hasil yang konsisten, dengan perbedaan jarak yang jelas terlihat antara setiap percobaan.
3. Program ini dapat dikembangkan lebih lanjut untuk mengintegrasikan sensor lainnya atau menghubungkannya ke platform IoT untuk pemantauan jarak secara real-time.

**4. Appendix**

Kode Program



Pengujian Program

