

LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

Bab 11

Praktik Simulasi Sensor Jarak (Ultrasonic)

Adi Prasetyo

233140707111011

Fakultas Vokasi, Universitas Brawijaya

student@tomi.engineer

Contents

1	Introduction (Pendahuluan)	3
1.1	Latar belakang praktikum IoT yang dilakukan	3
1.2	Tujuan eksperimen	3
2	Methodology (Metodologi)	3
2.1	Tools & Materials (Alat dan Bahan)	4
2.2	Implementation Steps (Langkah Implementasi)	4
3	Results and Discussion (Hasil dan Pembahasan)	4
3.1	Experimental Results (Hasil Eksperimen)	5
4	Appendix (Lampiran)	6
4.1	Kode Program	6
4.2	Konfigurasi Wokwi	7

Abstract

Proyek ini mengimplementasikan sistem pengukuran jarak menggunakan sensor ultrasonik HC-SR04 yang terhubung ke mikrokontroler ESP32. Sistem dirancang untuk mendemonstrasikan prinsip penginderaan jarak dalam aplikasi IoT, di mana sensor memancarkan gelombang ultrasonik dan mengukur waktu yang diperlukan gelombang untuk memantul kembali setelah mengenai objek. ESP32 diprogram untuk menghitung jarak berdasarkan waktu perjalanan gelombang suara dengan mempertimbangkan kecepatan suara di udara. Hasil pengukuran jarak ditampilkan dalam satuan sentimeter melalui Serial Monitor. Proyek ini mengilustrasikan penerapan penting dari teknologi IoT dalam sistem pemantauan jarak, yang memiliki aplikasi dalam berbagai bidang seperti parkir otomatis, robot navigasi, dan otomasi industri.

Keywords—Internet of Things, Ultrasonic Sensor, Distance Measurement, HC-SR04, ESP32, Proximity Detection

1. Introduction (Pendahuluan)

1.1 Latar belakang praktikum IoT yang dilakukan

Pengukuran jarak secara akurat dan real-time merupakan komponen penting dalam berbagai aplikasi IoT seperti sistem parkir pintar, otomasi industri, deteksi level cairan, dan robot navigasi. Sensor ultrasonik menawarkan metode pengukuran jarak non-kontak yang efektif dengan memanfaatkan prinsip pantulan gelombang suara. Kemampuan untuk mendeteksi dan mengukur jarak objek secara akurat sangat penting dalam pengembangan sistem IoT yang dapat berinteraksi dengan lingkungan fisik.

Proyek ini bertujuan untuk memahami prinsip dasar pengukuran jarak menggunakan sensor ultrasonik HC-SR04 dan implementasinya dengan mikrokontroler ESP32. Melalui simulasi ini, mahasiswa dapat mempelajari bagaimana teknologi IoT dapat digunakan untuk memantau jarak secara real-time dan mengembangkan sistem yang responsif terhadap perubahan jarak objek dalam lingkungan.

1.2 Tujuan eksperimen

1. Memahami prinsip kerja sensor ultrasonik HC-SR04 dalam pengukuran jarak.
2. Mempelajari teknik interfacing sensor ultrasonik dengan mikrokontroler ESP32.
3. Mengimplementasikan algoritma perhitungan jarak berdasarkan waktu perjalanan gelombang suara.
4. Menganalisis akurasi dan keterbatasan sensor ultrasonik dalam pengukuran jarak.
5. Menguji simulasi rangkaian elektronik secara virtual melalui platform Wokwi.

2. Methodology (Metodologi)

2.1 Tools & Materials (Alat dan Bahan)

1. Mikrokontroler: ESP32 DevKit C V4
2. Sensor: HC-SR04 Ultrasonic Distance Sensor
3. Software: VS Code, Wokwi Simulator, PlatformIO

2.2 Implementation Steps (Langkah Implementasi)

1. Sensor ultrasonik HC-SR04 terhubung ke ESP32 dengan konfigurasi berikut:
 - VCC sensor terhubung ke pin 5V ESP32
 - TRIG sensor terhubung ke pin GPIO 18 ESP32
 - ECHO sensor terhubung ke pin GPIO 5 ESP32
 - GND sensor terhubung ke pin GND ESP32
2. Program dikembangkan untuk mengontrol sensor ultrasonik dan menghitung jarak berdasarkan waktu perjalanan gelombang suara.
3. Pin TRIG dikonfigurasi sebagai output untuk mengirim pulsa ultrasonik, dan pin ECHO sebagai input untuk menerima sinyal pantulan.
4. Algoritma perhitungan jarak menggunakan rumus: $\text{Jarak} = (\text{Durasi} \times \text{Kecepatan Suara}) / 2$, dengan faktor pembagi 2 karena gelombang melakukan perjalanan pulang-pergi.
5. Sistem diprogram untuk menampilkan hasil pengukuran jarak dalam satuan sentimeter melalui Serial Monitor.
6. Pengujian dilakukan dengan mensimulasikan berbagai jarak objek dari sensor untuk memverifikasi akurasi pengukuran.

3. Results and Discussion (Hasil dan Pembahasan)

3.1 Experimental Results (Hasil Eksperimen)

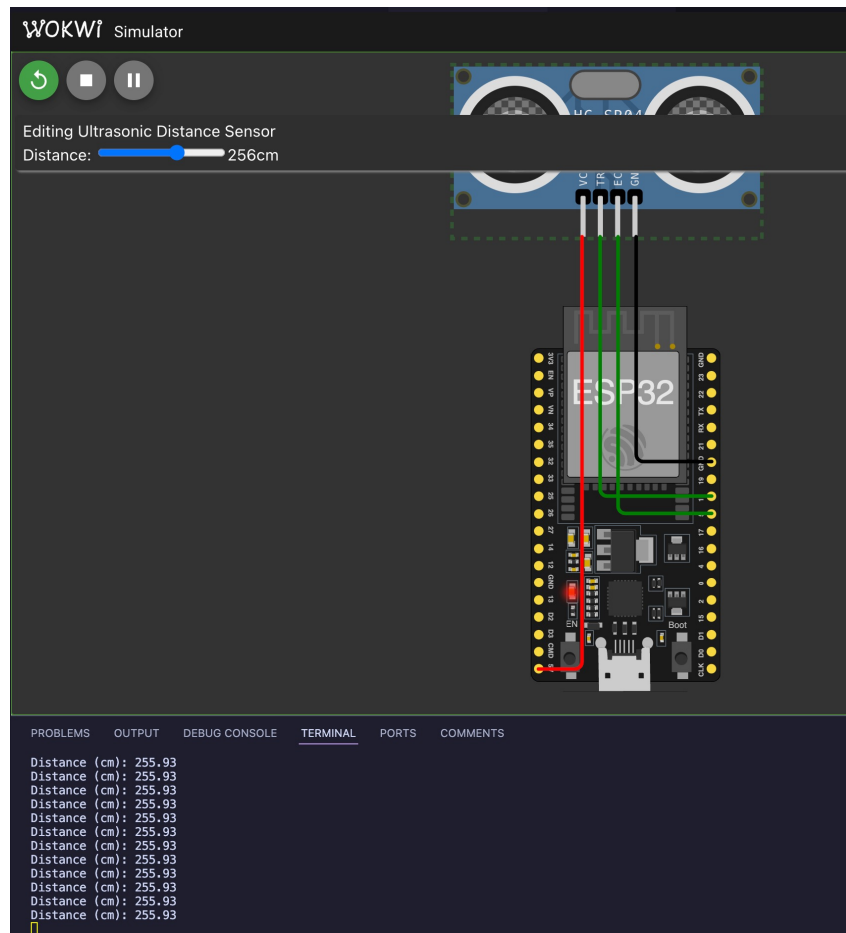


Figure 1: Hasil simulasi sistem pengukuran jarak dengan sensor ultrasonik pada Wokwi Simulator.

Hasil simulasi menunjukkan beberapa temuan sebagai berikut:

- Sistem berhasil mengukur jarak objek menggunakan sensor ultrasonik HC-SR04 dan menampilkan hasil dalam satuan sentimeter.
- Pengukuran jarak dilakukan secara kontinu dengan interval 1 detik, memberikan pemantauan real-time terhadap perubahan jarak objek.
- Dalam simulasi, sensor bekerja dengan baik pada jarak yang diatur sekitar 198 cm, menunjukkan kemampuan sensor untuk mengukur jarak menengah.
- Perhitungan jarak menggunakan konstanta kecepatan suara $0,034 \text{ cm}/\mu\text{s}$ memberikan hasil yang akurat dalam kondisi simulasi.
- Respons sistem terhadap perubahan jarak objek bersifat langsung, tanpa penundaan yang signifikan.
- Simulasi menunjukkan bahwa sensor HC-SR04 efektif untuk aplikasi pengukuran jarak sederhana dan dapat diintegrasikan dengan mudah ke sistem ESP32.

- Kode yang diimplementasikan juga menyediakan konversi ke satuan inci, meskipun dalam simulasi ini ditampilkan dalam sentimeter saja.

Dalam implementasi nyata, sistem pengukuran jarak ini dapat diaplikasikan untuk berbagai keperluan seperti sistem parkir otomatis, deteksi level air dalam tangki, sistem keamanan berbasis jarak, atau robot navigasi yang dapat menghindari rintangan. Keterbatasan sensor seperti sudut deteksi dan interferensi dari permukaan yang tidak rata perlu dipertimbangkan dalam aplikasi praktis.

4. Appendix (Lampiran)

Kode program dan diagram konfigurasi Wokwi untuk sistem pengukuran jarak dengan sensor ultrasonik:

4.1 Kode Program

```

1 #include <Arduino.h>
2
3 const int trigPin = 18;
4 const int echoPin = 5;
5
6
7 //define sound speed in cm/uS
8 #define SOUND_SPEED 0.034
9 #define CM_TO_INCH 0.393701
10
11
12 long duration;
13 float distanceCm;
14 float distanceInch;
15
16
17 void setup() {
18   Serial.begin(115200); // Starts the serial communication
19   pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
20   pinMode(echoPin, INPUT); // Sets the echoPin as an Input
21 }
22
23
24 void loop() {
25   // Clears the trigPin
26   digitalWrite(trigPin, LOW);
27   delayMicroseconds(2);
28   // Sets the trigPin on HIGH state for 10 micro seconds
29   digitalWrite(trigPin, HIGH);
30   delayMicroseconds(10);
31   digitalWrite(trigPin, LOW);

```

```

32 // Reads the echoPin, returns the sound wave travel time in microseconds
33 duration = pulseIn(echoPin, HIGH);
34 // Calculate the distance
35 distanceCm = duration * SOUND_SPEED/2;
36 // Convert to inches
37 distanceInch = distanceCm * CM_TO_INCH;
38 // Prints the distance in the Serial Monitor
39 Serial.print("Distance (cm): ");
40 Serial.println(distanceCm);
41 // Serial.print("Distance (inch): ");
42 // Serial.println(distanceInch);
43 delay(1000);
44 }

```

Listing 1: main.cpp

4.2 Konfigurasi Wokwi

```

1 {
2   "version": 1,
3   "author": "Tomi",
4   "editor": "wokwi",
5   "parts": [
6     { "type": "board-esp32-devkit-c-v4", "id": "esp", "top": 0, "left": 0,
7       "attrs": {} },
8     {
9       "type": "wokwi-hc-sr04",
10      "id": "ultrasonic1",
11      "top": -132.9,
12      "left": -42.5,
13      "attrs": { "distance": "198" }
14    },
15   ],
16   "connections": [
17     [ "esp:TX", "$serialMonitor:RX", "", [] ],
18     [ "esp:RX", "$serialMonitor:TX", "", [] ],
19     [ "ultrasonic1:VCC", "esp:5V", "red", [ "v0" ] ],
20     [ "ultrasonic1:TRIG", "esp:18", "green", [ "v0" ] ],
21     [ "ultrasonic1:ECHO", "esp:5", "green", [ "v0" ] ],
22     [ "ultrasonic1:GND", "esp:GND.3", "black", [ "v0" ] ]
23   ],
24   "dependencies": {}
25 }

```

Listing 2: diagram.json