

LAPORAN PRAKTIKUM INTERNET OF THINGS (IoT)

Fakultas Vokasi, Universitas Brawijaya

Praktik Relay, Button, LED dan Sensor Jarak Ultrasonic Melalui Sistem Simulasi Wokwi

Muhammad Faiza Firdaus

Fakultas Vokasi, Universitas Brawijaya

Email : faizafirdaus@student.ub.ac.id

Abstrak

Praktikum ini bertujuan untuk mencoba perangkat ESP-32 menggunakan sistem simulasi daring Wokwi. Dalam praktikum ini, perancangan sistem dilakukan dengan menggunakan Visual Studio Code dan Wokwi.com untuk mensimulasikan perangkat ESP-32 yang terhubung dengan perangkat tambahan yang dibutuhkan. Implementasi sistem meliputi pemrograman pada file main.cpp untuk membaca data serta menampilkannya melalui terminal. Hasil praktikum menunjukkan bahwa sistem berhasil dijalankan tanpa kendala.

Pendahuluan

1. Latar Belakang

Di era teknologi yang berkembang pesat ini, tak bisa dipungkiri bahwa teknologi telah menjadi bagian dari kehidupan sehari-hari manusia, dan semakin berkembangnya teknologi yang dibuat manusia menandakan semakin banyak alat yang dapat memudahkan kehidupan manusia, salah satunya contohnya ialah perangkat ESP-32.

2. Tujuan Eksperimen

Untuk menerapkan Relay, Button, LED dan Sensor Jarak Ultrasonic ke dalam perangkat ESP-32 melalui sistem simulasi daring Wokwi.

Metodologi

1. Alat dan Bahan

- Perangkat keras berupa :
Laptop
- Perangkat lunak berupa :
Visual Studio Code, Wokwi.com

2. Langkah Implementasi

1. Perancangan

Perancangan sistem dilakukan dengan membuat project baru menggunakan PIO di dalam Visual Studio Code untuk mensimulasikan sistem wokwi.com ke dalam Visual Studio Code. Selanjutnya, melakukan perancangan simulasi ESP-32 melalui wokwi.com untuk merancang simulasi perangkat.

2. Implementasi Sistem

Setelah melakukan perancangan simulasi sistem melalui wokwi.com, langkah pertama adalah memindahkan hasil rancangan dari wokwi.com ke Visual Studio Code, setelah rancangan berhasil di pindahkan, barulah mulai untuk memasukkan mekanisme di dalam main.cpp yang sudah disiapkan di dalam project.

Hasil dan Pembahasan

1. Hasil dan Implementasi

Mekanisme sistem simulasi relay, button dan led yang digunakan dalam praktikum ini adalah :

1. Saat tombol digunakan, ESP-32 akan mengirimkan sinyal ke relay, dan menyalakan lampu secara bersamaan.

Mekanisme sistem simulasi relay, button dan led yang digunakan dalam praktikum ini adalah :

1. Saat jarak di sekitar perangkat berubah, maka perangkat mengirimkan data tersebut melalui terminal, dengan memberi informasi jarak terkini.

Hasilnya, praktikum berjalan sempurna tanpa adanya kendala.

Lampiran

Kode dalam file main.cpp - simulasi relay, button dan led

```
#include <Arduino.h>

// Define pin numbers
const int ButtonPin = 19; // GPIO19 connected to the pushbutton
const int LedPin = 18;    // GPIO18 connected to the LED
const int RelayPin = 23;  // GPIO23 connected to the relay module

void setup() {
    // Set pin modes
    pinMode(ButtonPin, INPUT_PULLUP); // Set the button pin as an input with an internal pull-up
    resistor
    pinMode(LedPin, OUTPUT);           // Set the LED pin as an output
    pinMode(RelayPin, OUTPUT);        // Set the relay pin as an output

    // Initialize the outputs to be OFF
    digitalWrite(LedPin, LOW);
    digitalWrite(RelayPin, LOW);
}

void loop() {
    // Read the state of the button
    int buttonState = digitalRead(ButtonPin);

    // Check if the button is pressed
```

```

// Since the button is wired to pull the pin LOW when pressed, we check for LOW
if (buttonState == LOW) {
  digitalWrite(LedPin, HIGH);    // Turn on the LED
  digitalWrite(RelayPin, HIGH);  // Turn on the relay
} else {
  digitalWrite(LedPin, LOW);     // Turn off the LED
  digitalWrite(RelayPin, LOW);   // Turn off the relay
}
}

```

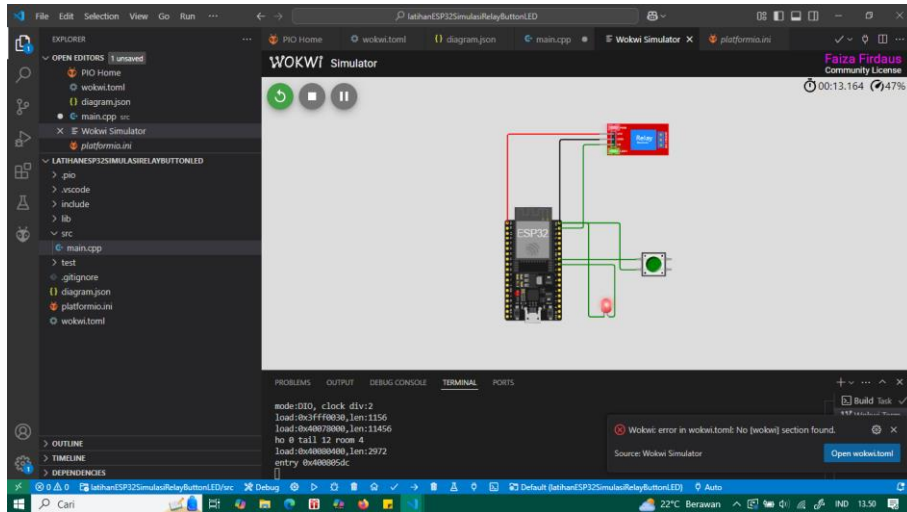
Kode dalam file diagram.json - simulasi relay, button dan led

```

{
  "version": 1,
  "author": "Faiza Firdaus",
  "editor": "wokwi",
  "parts": [
    {
      "type": "board-esp32-devkit-c-v4",
      "id": "esp",
      "top": 0,
      "left": 0,
      "attrs": { "builder": "esp-idf" }
    },
    { "type": "wokwi-relay-module", "id": "relay1", "top": -153.4, "left": 172.8, "attrs": {} },
    {
      "type": "wokwi-pushbutton",
      "id": "btn1",
      "top": 83,
      "left": 240,
      "attrs": { "color": "green", "xray": "1" }
    },
    { "type": "wokwi-led", "id": "led1", "top": 159.6, "left": 167, "attrs": { "color": "red" } }
  ],
  "connections": [
    [ "esp:TX", "$serialMonitor:RX", "", [ ] ],
    [ "esp:RX", "$serialMonitor:TX", "", [ ] ],
    [ "relay1:VCC", "esp:3V3", "red", [ "h0" ] ],
    [ "relay1:GND", "esp:GND.2", "black", [ "h0" ] ],
    [ "esp:19", "btn1:1.1", "green", [ "h0" ] ],
    [ "btn1:2.1", "esp:GND.2", "green", [ "h-28.8", "v-86.2" ] ],
    [ "relay1:IN", "esp:23", "green", [ "h-28.8", "v153.4" ] ],
    [ "led1:C", "esp:GND.2", "green", [ "v0", "h-28.4", "v-172.8" ] ],
    [ "led1:A", "esp:18", "green", [ "v0", "h9.6", "v-96" ] ]
  ],
  "dependencies": {}
}

```

Screenshot hasil simulasi wokwi - simulasi relay, button dan led



Kode dalam file main.cpp - simulasi sensor jarak ultrasonik

```
#include <Arduino.h>

// Define pin numbers
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);
}

```

Kode dalam file diagram.json - simulasi sensor jarak ultrasonic

```

{
  "version": 1,
  "author": "Faiza Firdaus",
  "editor": "wokwi",
  "parts": [
    {
      "type": "board-esp32-devkit-c-v4",
      "id": "esp",
      "top": -9.6,
      "left": 4.84,
      "attrs": { "builder": "esp-idf" }
    },
    { "type": "wokwi-hc-sr04", "id": "ultrasonic1", "top": -36.9, "left": 178.3, "attrs": {} }
  ],
  "connections": [
    [ "esp:TX", "$serialMonitor:RX", "", [] ],
    [ "esp:RX", "$serialMonitor:TX", "", [] ],
    [ "ultrasonic1:VCC", "esp:3V3", "red", [ "v9.6", "h-86.4", "v-124.8", "h-153.6" ] ],
    [ "ultrasonic1:ECHO", "esp:18", "green", [ "v0" ] ],
    [ "ultrasonic1:TRIG", "esp:5", "green", [ "v0" ] ],
    [ "ultrasonic1:GND", "esp:GND.3", "black", [ "v0" ] ]
  ],
  "dependencies": {}
}

```

Screenshot hasil simulasi wokwi - simulasi sensor jarak ultrasonic

