



**REAL TIME SYSTEM AND INTERNET OF THINGS FINAL PROJECT REPORT
DEPARTMENT OF ELECTRICAL ENGINEERING
UNIVERSITAS INDONESIA**

**Where's My Water?
(Smart Plant Watering System)**

GROUP 6

MIRANTI ANGGUNSARI	2106731472
SYAUQI AULIYA M.	2106707201
REICHAN ADHIGUNO	2106703273
DIMAS DANDOSSY W. P.	2206059780

PREFACE

Puji syukur kami panjatkan kehadiran Tuhan Yang Maha Esa atas rahmat dan karunia-Nya sehingga kami dapat menyelesaikan laporan proyek akhir untuk Praktikum Sistem Waktu Nyata dan IOT ini dengan baik dan tepat waktu. Laporan ini disusun sebagai salah satu syarat untuk menyelesaikan praktikum yang telah kami jalani selama satu semester.

Proyek akhir ini berfokus pada implementasi dan pengembangan IOT yang merupakan integrasi erat antara komputasi, jaringan, dan proses fisik. Melalui praktikum ini, kami memperoleh kesempatan untuk menerapkan teori-teori yang telah dipelajari di kelas ke dalam praktik nyata, mengasah kemampuan analitis, serta meningkatkan keterampilan teknis dalam bidang ini.

Kami menyadari bahwa dalam penyusunan laporan ini, banyak pihak yang telah memberikan bantuan dan dukungan, baik secara langsung maupun tidak langsung. Oleh karena itu, kami ingin menyampaikan terima kasih yang sebesar-besarnya kepada dosen pengampu mata kuliah Sistem Waktu Nyata dan IOT, asisten laboratorium, dan teman-teman satu tim atas kerja sama, komitmen, dan kontribusinya dalam menyelesaikan proyek ini.

Kami menyadari bahwa laporan ini masih jauh dari sempurna dan masih banyak kekurangan yang perlu diperbaiki. Oleh karena itu, kritik dan saran yang membangun sangat kami harapkan demi perbaikan dan peningkatan kualitas di masa mendatang.

Akhir kata, kami berharap laporan ini dapat bermanfaat bagi pembaca dan memberikan kontribusi positif bagi perkembangan ilmu pengetahuan di bidang IOT. Semoga Tuhan Yang Maha Esa senantiasa memberikan rahmat dan hidayah-Nya kepada kita semua.

Depok, December 5, 2024

TABLE OF CONTENTS

CHAPTER 1.....	4
INTRODUCTION.....	4
1.1 PROBLEM STATEMENT.....	4
1.3 ACCEPTANCE CRITERIA.....	6
1.4 ROLES AND RESPONSIBILITIES.....	7
1.5 TIMELINE AND MILESTONES.....	9
CHAPTER 2.....	12
IMPLEMENTATION.....	12
2.1 HARDWARE DESIGN AND SCHEMATIC.....	12
2.2 SOFTWARE DEVELOPMENT.....	15
2.2.1. PROGRAM.....	15
2.2.2. MODULE IMPLEMENTATION.....	24
2.3 HARDWARE AND SOFTWARE INTEGRATION.....	25
CHAPTER 3.....	27
TESTING AND EVALUATION.....	27
3.1 TESTING.....	27
3.1.1. Testing Manual Mode.....	27
3.1.2. Testing Automatic with Threshold Mode.....	28
3.1.3. Testing Automatic with Schedule Mode.....	28
3.1.4. Testing Automatic with Interval Mode.....	28
3.2 RESULT.....	29
3.3 EVALUATION.....	30
CHAPTER 4.....	31
CONCLUSION.....	31

CHAPTER 1

INTRODUCTION

1.1 PROBLEM STATEMENT

Indonesia sebagai negara agraris seringkali menghadapi berbagai tantangan dalam mengoptimalkan sektor pertaniannya. Salah satu masalah utama pada aspek ini adalah adanya ketergantungan pada metode tradisional dalam pemeliharaan tanaman, khususnya sistem pengairan atau irigasi. Metode penyiraman manual ini tidak mampu untuk menjamin tingkat kelembaban tanah yang optimal, sehingga berisiko berdampak kepada kualitas tanaman. Ketidakpastian kondisi cuaca, kurangnya akses terhadap teknologi, dan rendahnya efisiensi penggunaan air semakin memperburuk situasi ini, sehingga mempengaruhi produktivitas pertanian dan menyebabkan kerugian.

Di sisi lain, perubahan gaya hidup masyarakat modern seperti urbanisasi semakin memperbesar tantangan bagi individu yang ingin menjaga tanaman mereka untuk tetap sehat, terutama di perkotaan. Dengan berbagai keterbatasan waktu dan keseharian yang padat aktivitas, banyak orang kesulitan dalam memantau kebutuhan air tanaman secara rutin dan mengakibatkan kegagalan pemeliharaan tanaman tersebut.

Melihat permasalahan tersebut, kelompok kami mengusulkan inovasi teknologi yang dapat membantu otomatisasi proses penyiraman dengan memanfaatkan perangkat Internet of Things (IoT), berupa sistem perangkat penyiraman cerdas. Diharapkan solusi ini tidak hanya meningkatkan efisiensi penggunaan air, tetapi juga memberikan kenyamanan bagi pengguna, baik secara individu perkotaan atau dalam skala yang lebih besar.

1.2 PROPOSED SOLUTION

Proyek ini bertujuan untuk mengembangkan sistem penyiraman tanaman otomatis berbasis ESP32 yang terintegrasi dengan sensor kelembaban tanah YL-69 dan aplikasi Blynk. Sistem ini dirancang untuk memantau kelembaban tanah secara

real-time dan mengontrol pompa air secara otomatis atau manual melalui jaringan Wi-Fi. Dengan pembacaan data kelembaban tanah setiap lima detik, sistem dapat merespons kondisi tanah dengan cepat, menyiram tanaman ketika kelembaban berada di bawah batas yang ditentukan. Melalui aplikasi Blynk, pengguna juga dapat memantau data kelembaban tanah, status pompa air, dan mengontrol perangkat dari jarak jauh, memberikan kemudahan serta meningkatkan efisiensi penyiraman tanaman.

Komponen Utama:

- YL-69 Soil Moisture Sensor
Untuk mengukur tingkat kelembaban tanah sebagai input stream utama sistem
- ESP32 Microcontroller
Sebagai pusat *data processing* dari semua sensor, relay, dan penghubung dengan aplikasi Blynk
- Mini Pump Motor DC Submersible
Sebagai penyiram tanaman berdasarkan kode untuk memompa air
- 1-Channel Relay Module
Mengatur koneksi daya listrik ke Mini Pump untuk mekanisme nyala mati sistem

Cara Kerja Sistem:

- Koneksi
ESP32 akan terhubung ke internet menggunakan Wifi kredensial yang diset dalam program, kemudian sistem akan membaca nilai kelembaban tanah melalui moisture sensor YL-69 setiap 5 detik.
- Pengukuran Kelembaban Tanah
Nilai kelembaban tanah akan dipetakan pada skala persentase (0-100%) menggunakan fungsi *map()* yang kemudian dikirimkan ke Blynk melalui Virtual Pin dan akan muncul pada *interface* yang bisa dimonitor oleh pengguna secara real-time melalui perangkat handphone.
- Penyiraman (Memompa Air)
 1. Mode Otomatis
Sistem membaca nilai kelembaban tanah dan menyiram ketika nilainya di bawah ambang batas yang ditentukan

2. Mode Manual

Pompa dapat diaktifkan/dimatikan langsung oleh pengguna melalui aplikasi Blynk melalui mekanisme switch

3. Mode Schedule

Sistem memeriksa waktu internal ESP32 untuk memastikan penyiraman dilakukan pada jadwal yang telah ditentukan oleh pengguna. Pompa akan menyala selama 15 detik setiap(melalui perhitungan kebutuhan air tanaman kecil dan debit air mini pump), kemudian mati secara otomatis sampai hari esoknya.

4. Mode Interval

Sistem dapat memeriksa input waktu dari pengguna. Pompa akan menyala dengan interval waktu yang telah ditentukan oleh user, kemudian akan mati secara otomatis setelah 15 detik menyala.

1.3 ACCEPTANCE CRITERIA

Where's My Plant dapat dikatakan berhasil berjalan apabila memenuhi syarat-syarat berikut ini:

Kategori	Kriteria
Monitoring Kelembaban Tanah	<ol style="list-style-type: none">1. Sistem mampu membaca nilai kelembaban tanah dengan sensor YL-69 secara real-time setiap 5 detik.2. Nilai kelembaban tanah ditampilkan sebagai persentase (0–100%) pada aplikasi Blynk melalui Virtual Pin.
Mode Penyiraman Otomatis	<ol style="list-style-type: none">1. Sistem menyala otomatis ketika kelembaban tanah berada di bawah ambang batas yang telah ditentukan oleh pengguna pada aplikasi Blynk.2. Pompa air berhenti ketika kelembaban tanah mencapai atau melebihi ambang batas yang ditentukan.

Mode Penyiraman Manual	<ol style="list-style-type: none"> 1. Pengguna dapat mengontrol pompa air secara langsung melalui tombol pada aplikasi Blynk untuk menyalakan atau mematikan pompa sesuai keinginan 2. Status pompa ditampilkan secara real-time pada aplikasi Blynk menggunakan indikator LED virtual.
Mode Penjadwalan (Schedule)	<ol style="list-style-type: none"> 1. Sistem mampu menyiram tanaman selama 15 detik setiap hari pada waktu yang telah ditentukan pengguna melalui aplikasi Blynk. 2. Pengguna dapat mengatur jadwal penyiraman melalui antarmuka aplikasi Blynk, dan sistem menjalankan penyiraman sesuai waktu yang diatur.
Aplikasi Blynk	<ol style="list-style-type: none"> 1. Pengguna dapat melihat data kelembaban tanah, status pompa, dan mengontrol mode penyiraman melalui aplikasi Blynk secara intuitif 2. Aplikasi menampilkan indikator LED virtual yang mencerminkan status pompa secara real-time.

Table 1.3.1. Acceptance Criteria

1.4 ROLES AND RESPONSIBILITIES

Berikut ini pembagian peran dan tugas bagi masing-masing anggota kelompok kami:

Roles	Responsibilities	Person
-------	------------------	--------

IoT Hardware Specialist	<ol style="list-style-type: none"> 1. Bertanggung jawab untuk memilih dan memperoleh komponen-komponen yang dibutuhkan untuk sistem (misalnya sensor YL-69, pompa mini, modul relay, ESP32). 2. Merancang skematik sistem, termasuk diagram sirkuit, wiring, dan koneksi antara komponen. 3. Memastikan integrasi semua komponen hardware berjalan lancar dalam sistem IoT. 4. Menguji komponen-komponen untuk memastikan mereka berfungsi dengan baik dan memenuhi kebutuhan proyek. 5. Memecahkan masalah terkait pengaturan hardware dan memastikan komunikasi antar komponen berjalan dengan baik. 	Miranti Anggunsari
IoT Software Specialist	<ol style="list-style-type: none"> 1. Bertanggung jawab untuk menulis dan mengoptimalkan firmware untuk ESP32, termasuk logika untuk membaca data sensor, mengontrol pompa, dan menangani mode penyiraman manual/otomatis/jadwal. 2. Mengembangkan integrasi aplikasi Blynk untuk memantau tingkat kelembaban tanah dan mengontrol pompa secara jarak jauh. 3. Memastikan ESP32 dapat berkomunikasi dengan aplikasi Blynk melalui Wi-Fi untuk pemantauan dan pengontrolan secara real-time. 4. Menangani masalah software dan memastikan interaksi antara hardware, software, dan aplikasi mobile berjalan lancar. 	Syauqi Auliya M.

	5. Memastikan semua fitur yang diprogram (penyiraman manual, otomatis, penjadwalan, dll.) berfungsi dengan baik.	
Presentation and Reporting	<ol style="list-style-type: none"> 1. Membantu menulis laporan proyek, dengan fokus pada penyajian data dan hasil proyek. Memastikan penjelasan teknis jelas dan terstruktur dengan baik. 2. Membuat dan merancang presentasi PowerPoint, menyoroti aspek-aspek utama dari proyek dan memastikan presentasi tersebut jelas dan profesional. 3. Memberikan bantuan dalam proses debugging dengan mengidentifikasi masalah potensial dan memberikan solusi. 	Reichan Adhiguno
Testing and Design	<ol style="list-style-type: none"> 1. Membantu dengan pengujian sistem secara keseluruhan (hardware dan software) untuk memastikan sistem berfungsi sesuai spesifikasi. 2. Membantu dalam pembuatan laporan proyek, terutama dalam merangkum fase pengujian dan langkah-langkah troubleshooting. 3. Membantu merancang tampilan dan antarmuka pengguna dalam aplikasi Blynk agar lebih intuitif dan mudah digunakan. 	Dimas Dandossi W. P.

Table 1.4.1. Roles and Responsibilities

1.5 TIMELINE AND MILESTONES

Milestone	W1 (20/11 - 26/11)	W2 (27/11 - 03/12)	W3 (4/12 - 10/12)
Brainstorming/ideation			

Initial Software Development						
Hardware Design and Schematic						
Prototype Build and Testing						
Full Feature Software Development						
Software and Hardware Integration						
Testing						
Final Product Assembly and Reporting						

Table 1.5.1. Timeline and Milestone Gantt Chart

Milestone Description

- 1. Brainstorming/Ideation**

Tahapan ini mencakup diskusi awal untuk menentukan ruang lingkup, tujuan, dan pemilihan komponen hardware serta software.

- 2. Initial Software Development**

Pada tahap ini, kamu memulai pengembangan kode dasar untuk ESP32 dan aplikasi Blynk, serta pengujian fungsi dasar menggunakan lingkungan virtual.

- 3. Hardware Design and Schematic**

Penyusunan skematik sistem dan pemilihan komponen hardware yang dibutuhkan. Susunan skematik kami buat dengan menggunakan aplikasi Proteus sebagai dasar penyusunan komponen hardware untuk membentuk sistem yang diharapkan.

- 4. Prototype Build and Testing**

Perakitan prototype hardware dan pengujian integrasi awal dengan software. Perakitan ini didasarkan dengan desain skematik yang sudah dibuat.

- 5. Full Feature Software Development**

Pada tahap ini, kami masuk ke dalam pengembangan dan implementasi fitur lengkap seperti mode penyiraman dan pemantauan melalui Blynk.

6. **Software and Hardware Integration**

Pengintegrasian software dengan hardware untuk memastikan semua komponen berfungsi dengan baik bersama-sama.

7. **Testing**

Pengujian menyeluruh untuk memastikan sistem berfungsi sesuai dengan yang diharapkan.

8. **Final Product Assembly and Report**

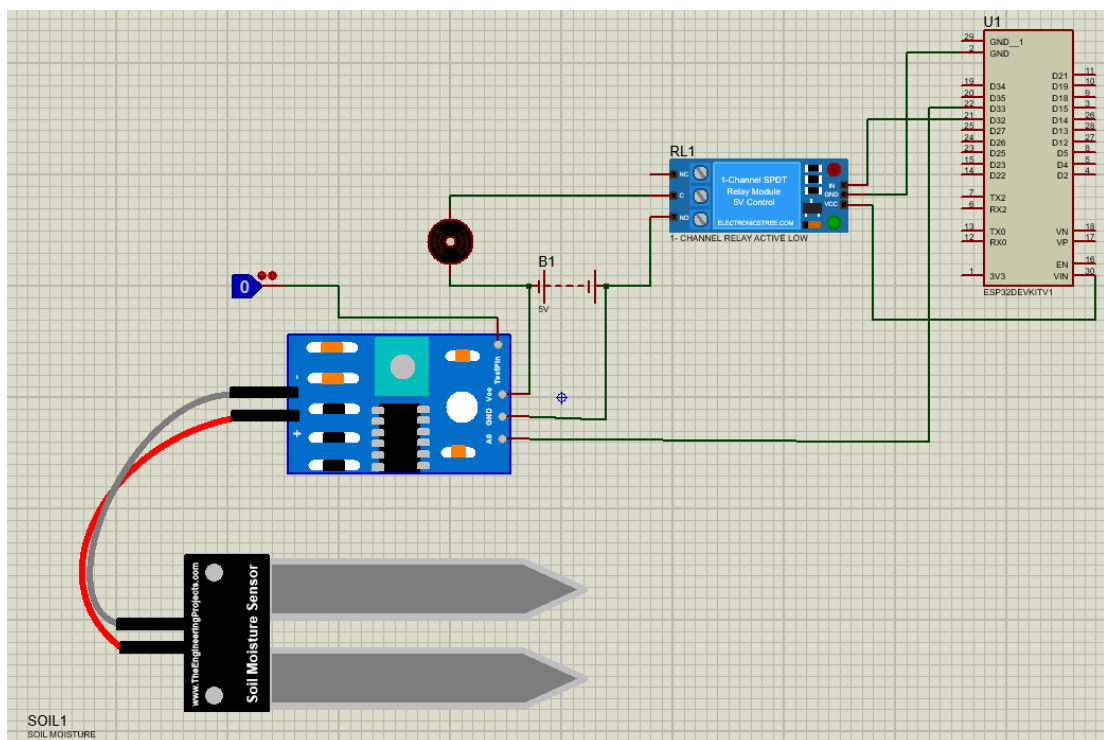
Penyelesaian perakitan akhir dan dokumentasi proyek dalam bentuk laporan dan presentasi.

CHAPTER 2

IMPLEMENTATION

2.1 HARDWARE DESIGN AND SCHEMATIC

Rangkaian *Where's My Water* ini merupakan suatu sistem pemantauan kelembaban tanah yang mampu secara real-time melacak nilai kelembaban tanah di tempat sensor diletakkan. Proyek ini bertujuan untuk memberikan informasi akurat dan instan untuk pemantauan kelembaban tanah sehingga sistem penyiraman dapat aktif sesuai dengan mode yang dipilih oleh user. Untuk menerapkan desain proyek ini, kami menggunakan sejumlah komponen hardware yang mencakup ESP32, sensor YL-69, Relay Module, dan Mini Pump DC.



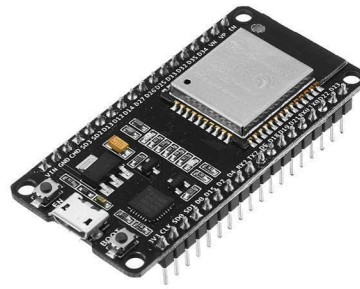
Gambar 2.1.1. Desain rangkaian pada Proteus

Berikut ini penjelasan terperinci untuk masing-masing peranan komponen hardware yang membentuk rangkaian *Where's My Water*.

a. ESP32 Microcontroller

ESP32 pada proyek ini digunakan sebagai pusat kendali dari kerja sistem secara keseluruhan. Relay module untuk mini pump dihubungkan dengan

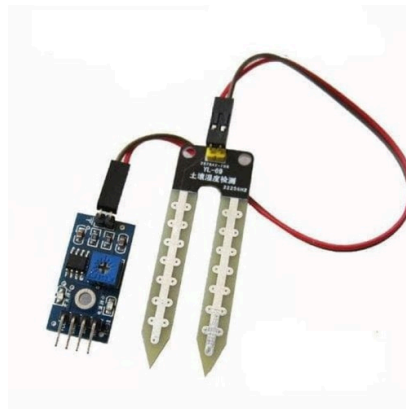
ESP32 melalui pin 32, sedangkan sensor akan dihubungkan melalui pin 33 sebagai input. ESP32 akan menjalankan tiga task, yaitu task sensor, task pompa, dan task interval. Task sensor ini akan digunakan untuk mengatur pembacaan sensor YL-69, sedangkan task pompa akan digunakan untuk mengatur nyala atau matinya pompa. Sedangkan, task interval secara khusus dikonfigurasi untuk mengatur mode interval.



Gambar 2.1.2. ESP32 Microcontroller

b. Soil Moisture Sensor YL-69

Sensor YL-69 dihubungkan ke ESP32 melalui salah satu pin, yaitu pin 33 untuk membaca nilai kelembaban tanah.

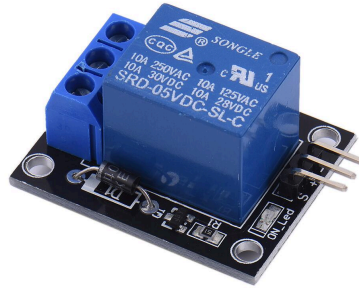


Gambar 2.1.3. Sensor YL-69

c. 1-Channel Relay Module

Relay module ini menjadi penghubung antara mini pump dengan ESP32, di mana kami menggunakan metode koneksi NO (Normally Open). Koneksi NO ini ketika dalam keadaan default atau tidak aktif, akan membuka terminal NO. Yang berarti tidak ada aliran arus listrik antara terminal COM dan terminal NO. Namun, ketika relay aktif, terminal NO akan tertutup sehingga akan

terjadi sambungan antara COM dan NO, sehingga arus listrik dapat mengalir. Kami menggunakan bentuk koneksi ini karena hanya perlu dinyalakan ketika memenuhi kondisi tertentu atau diberi sinyal.



Gambar 2.1.4. 1-Channel Relay Module

d. Submersible Mini DC Pump 5V

Pompa ini dihubungkan melalui Relay Module dan akan diatur nyala atau matinya berdasarkan input user maupun secara otomatis diatur dengan berdasarkan logika yang dikonfigurasi pada ESP32.



Gambar 2.1.5. Mini Pump DC

e. Mini Breadboard

Mini Breadboard digunakan untuk secara mudah menghubungkan beberapa perangkat yang memerlukan sumber arus atau VCC dan ground yang.



Gambar 2.1.6. Mini Breadboard

Berdasarkan rancangan skematik yang sudah dibuat, kami berhasil menyusun komponen-komponen hardware tersebut menjadi suatu sistem yang dapat berjalan sesuai dengan harapan. Berikut ini hasil rangkaian secara nyata:

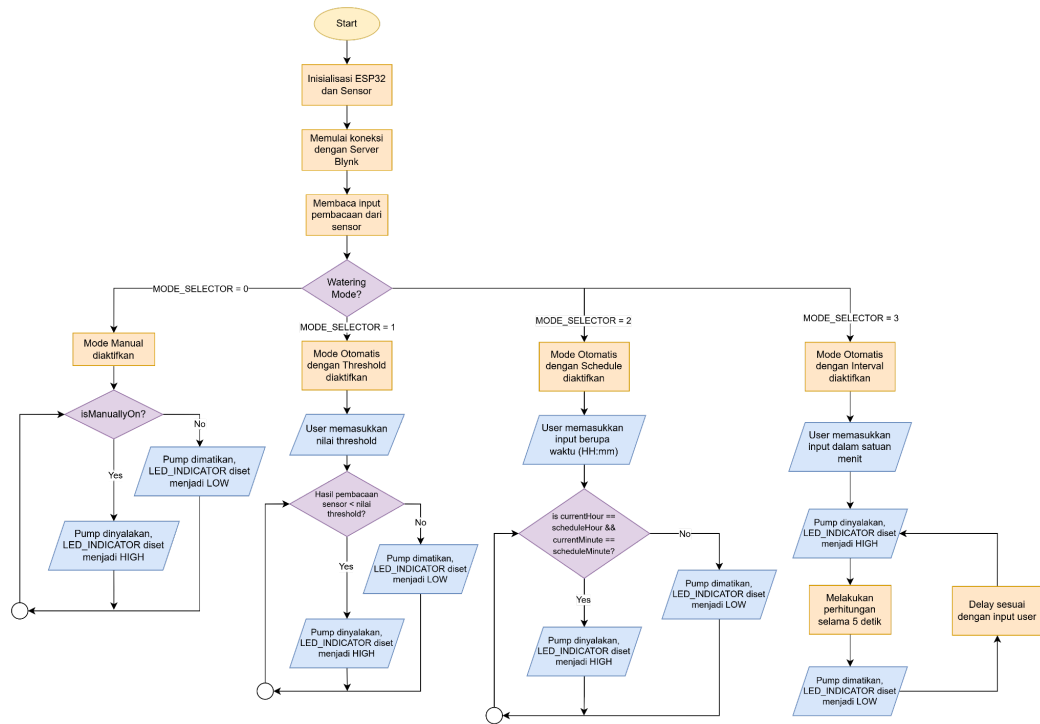


Gambar 2.1.7. Hasil Rangkaian *Where's My Water?*

2.2 SOFTWARE DEVELOPMENT

2.2.1. PROGRAM

Sistem "*Where's My Water*" untuk pemantauan kelembaban tanah dan pengendalian pompa otomatis ini dikembangkan menggunakan platform Blynk, yang menyediakan *interface* pengguna untuk mengontrol dan memantau perangkat berbasis ESP32 secara real-time. Berikut ini kami sertakan flowchart yang dapat memvisualisasikan cara kerja sistem:



Gambar 2.2.1.1. Flowchart

Berikut merupakan rincian elemen dalam program serta fungsionalitas utama perangkat lunak:

Konfigurasi Global

a. Memulai Koneksi dengan Blynk dan Autentikasi

Program dimulai dengan mendefinisikan *BLYNK_TEMPLATE_ID*, *BLYNK_TEMPLATE_NAME*, dan *BLYNK_AUTH_TOKEN*. Konfigurasi ini memastikan komunikasi antara perangkat keras dengan server Blynk.

```

#define BLYNK_TEMPLATE_ID "TMPL66RkUdqmF"           // Blynk
Template ID
#define BLYNK_TEMPLATE_NAME "Where Is My Water"      // Blynk
Template Name
#define BLYNK_AUTH_TOKEN "bslM8vPOcqu33TKggLIcs28WbyaNu7FA" //
Blynk Authentication Token
  
```

b. Library yang digunakan

Program menggunakan library seperti *BlynkSimpleEsp32*, *WiFi*, dan *time.h* untuk mengatur koneksi internet dan waktu berbasis NTP.


```
#include <BlynkSimpleEsp32.h> // Library Blynk
#include <WiFiClient.h>        // Library WiFi Client
#include <WiFi.h>               // Library WiFi
#include <time.h>               // Library untuk waktu NTP
```

c. Virtual Pin pada Blynk

Virtual Pin Blynk digunakan untuk berkomunikasi antara aplikasi Blynk di perangkat pengguna dengan ESP32. Setiap pin virtual memiliki fungsi spesifik, seperti mengaktifkan/mematikan pompa secara manual, mengatur batas kelembaban tanah, menetapkan jadwal penyiraman, dan menampilkan status pompa. Pin-pin ini memungkinkan pengguna untuk mengendalikan sistem secara interaktif melalui aplikasi pada perangkat *mobile*.

```
#define PUMP_BUTTON V0    // Switch pompa manual
#define TRESHOLD_SLIDER V1 // Slider threshold kelembapan
#define MOISTURE_LABEL V2 // Label kelembapan tanah
#define LED_INDICATOR V3  // LED indikator pompa
#define INTERVAL_SLIDER V4 // Slider interval
#define SCHEDULE_INPUT V5 // Input waktu (HH:MM)
#define MODE_SELECTOR V6  // Mode penyiraman
```

d. WiFi Credentials

WiFi Credentials mencakup pengaturan nama jaringan WiFi (SSID) dan kata sandi agar ESP32 dapat terhubung ke jaringan internet dan berkomunikasi dengan server Blynk serta server waktu NTP.

e. Variabel Global

Proyek memanfaatkan variabel global untuk menyimpan data penting yang digunakan oleh berbagai task, antara lain:

- I. Variabel kontrol pompa berfungsi untuk memantau dan mengatur status operasi pompa dalam berbagai mode.
- II. Variabel mode berfungsi untuk menentukan mode kerja sistem, apakah manual, otomatis dengan threshold, berdasarkan jadwal, atau interval.
- III. Variabel input schedule digunakan untuk mencatat waktu jadwal penyiraman serta durasi interval yang diatur pengguna
- IV. Variabel thresholdValue dan sensorPercentageFinal menyimpan data kelembaban tanah untuk *decision-making*.

```

// Variabel Kontrol Pompa
bool isPumpOn = false;
bool isManuallyOn = false;
bool isAutomaticallyOn = false;
bool isScheduledOn = false;
bool isIntervalOn = false;

// Variabel Mode
int currentMode = 1; // Default: automatic with threshold
int thresholdValue = 0;
int sensorPercentage = 0;
int sensorPercentageFinal = 0;

// Variabel Input Schedule dan Interval
int scheduleHour = 0;
int scheduleMinute = 0;
int intervalMinutes = 0;

```

f. BLYNK_WRITE

Berupa mekanisme untuk menerima data yang dikirimkan dari aplikasi Blynk ke ESP32. Setiap input, seperti tombol, slider, atau waktu jadwal, diterima oleh fungsi BLYNK_WRITE, kemudian diproses untuk mengontrol sistem sesuai kebutuhan pengguna.

```

// Input dari Blynk dalam bentuk mode selection
BLYNK_WRITE(MODE_SELECTOR) {
    currentMode = param.asInt();
    Serial.print("Mode changed to ");
    Serial.println(currentMode);
}

// Input dari Blynk dalam bentuk swith untuk mode manual
BLYNK_WRITE(PUMP_BUTTON) {
    isManuallyOn = param.asInt();
    Serial.print("Manual Pump: ");
    Serial.println(isManuallyOn ? "ON" : "OFF");
}

```

```

}

// Input dari Blynk dalam bentuk slider untuk mengatur nilai threshold
BLYNK_WRITE(TRESHOLD_SLIDER) {
    thresholdValue = param.asInt();
    Serial.print("New threshold: ");
    Serial.println(thresholdValue);
}

// Input dari Blynk berupa number input untuk mengatur nilai interval
BLYNK_WRITE(INTERVAL_SLIDER) {
    intervalMinutes = param.asInt();
    //lastIntervalTime = millis();
    Serial.print("New interval: ");
    Serial.print(intervalMinutes);
    Serial.println(" minutes");
}

// Input dari Blynk berupa jam dan menit untuk mengatur jadwal penyiraman
BLYNK_WRITE(SCHEDULE_INPUT) {
    TimeInputParam t(param);
    scheduleHour = t.getStartHour();
    scheduleMinute = t.getStartMinute();
    Serial.print("Schedule set to: ");
    Serial.print(scheduleHour);
    Serial.print(":");
    Serial.println(scheduleMinute);
}

```

FreeRTOS Tasks

a. vTaskInternal

Task ini bertanggung jawab untuk mengontrol pompa dalam mode interval. Task ini akan menyalakan pompa dengan mengatur pin relay ke posisi LOW dan mengaktifkan indikator LED pada aplikasi Blynk. Setelah pompa menyala selama waktu yang ditentukan, fungsi *stopPump* akan dipanggil melalui

mekanisme timer untuk mematikan pompa. Setelah siklus ini selesai, task akan masuk ke fase delay dengan durasi yang ditentukan oleh variabel `intervalMinutes`, sebelum memulai siklus berikutnya.

```
// Task untuk pengaturan interval
void vTaskInterval(void *pvParam) {
    while (1) {
        digitalWrite(RELAY_PIN, LOW);
        Blynk.virtualWrite(LED_INDICATOR, 1);
        Serial.println("Pump ON");
        pumpTimer.setTimeout(1000L, stopPump);
        vTaskDelay(pdMS_TO_TICKS(intervalMinutes * 60000));
    }
}
```

b. `vTaskSensor`

Pada task `vTaskSensor`, sistem bertugas untuk membaca data kelembaban tanah yang diperoleh dari sensor YL69. Setiap 3 detik, sistem melakukan pembacaan analog melalui pin sensor menggunakan fungsi `analogRead(SENSOR_PIN)`. Hasil pembacaan sensor yang berupa nilai analog antara 0 hingga 4095 kemudian dipetakan ke dalam skala persentase 0 hingga 100 menggunakan fungsi `map()`. Hasil pemetaan ini disimpan dalam variabel `sensorPercentage`. Pada sensor YL69, semakin basah tanah, semakin rendah nilai pembacaan sensor dan berlaku sebaliknya. Oleh karena itu, agar lebih intuitif, nilai persentase kelembaban akhir dihitung dengan mengurangi `sensorPercentage` dari 100, menghasilkan nilai `sensorPercentageFinal`. Nilai tersebut kemudian dikirim ke aplikasi Blynk untuk ditampilkan kepada pengguna sebagai persentase kelembaban tanah yang akurat. Selain itu, task ini juga memeriksa mode *threshold*, di mana jika mode aktif dan kelembaban tanah lebih rendah dari ambang batas yang ditentukan (*thresholdValue*), pompa akan diaktifkan secara otomatis. Proses ini berjalan terus-menerus dalam loop, memberikan pembaruan kelembaban tanah setiap 3 detik dan memungkinkan kontrol otomatis berdasarkan kelembaban yang terdeteksi.

```
// Task untuk mengatur sistem pembacaan sensor
void vTaskSensor(void *pvParam) {
```

```

while (1) {
  if (!isManuallyOn) {
    int sensorReading = analogRead(SENSOR_PIN);
    sensorPercentage = map(sensorReading, 0, 4095, 0, 100);
    sensorPercentageFinal = 100 - sensorPercentage;
    Blynk.virtualWrite(MOISTURE_LABEL, sensorPercentageFinal);
    Serial.print("Moisture: ");
    Serial.println(sensorPercentageFinal);

    if (currentMode == 1 && sensorPercentageFinal < thresholdValue) {
      isAutomaticallyOn = true;
    } else {
      isAutomaticallyOn = false;
    }
  }
  // Sensor akan membaca nilai kelembaban tanah setiap 3 detik
  vTaskDelay(pdMS_TO_TICKS(3000));
}
}

```

c. vTaskPump

Task vTaskPump bertanggung jawab untuk mengontrol pompa berdasarkan mode dan waktu yang sedang aktif. Tugas ini berjalan terus-menerus dalam sebuah loop, pertama memeriksa waktu lokal untuk menentukan kapan pompa harus diaktifkan atau dimatikan. Task diperbarui setiap detik, memastikan bahwa status pompa diperiksa dan disesuaikan berdasarkan mode yang aktif.

I. Mode Jadwal (Mode 2)

Pompa akan menyala saat waktu saat ini cocok dengan jadwal yang telah ditentukan (jam dan menit). Pompa akan mati ketika waktu saat ini tidak cocok dengan jadwal. Ketika pompa menyala, indikator LED diaktifkan, dan sistem akan menunggu selama waktu yang ditentukan sebelum mematikan pompa menggunakan fungsi *stopPump*.

II. Mode Interval (Mode 3)

Jika mode ini aktif, tugas interval yang sudah ada (xHandleInterval)

akan dilanjutkan untuk mengatur operasi pompa. Jika tidak, tugas interval akan dihentikan.

III. Mode Kontrol

Pompa dapat dikendalikan secara manual (Mode 0), otomatis berdasarkan pembacaan kelembaban tanah (Mode 1), atau sesuai jadwal (Mode 2). Jika salah satu kondisi ini terpenuhi, pompa akan menyala, dan indikator LED menyala. Pada mode interval, kontrol pompa diserahkan kepada tugas pengaturan interval.

```
// Task untuk mengatur nyala atau matinya pompa
void vTaskPump(void *pvParam) {
    while (1) {
        if (!getLocalTime(&timeinfo)) {
            Serial.println("Failed to obtain time");
            vTaskDelay(pdMS_TO_TICKS(1000));
            continue;
        }

        int currentHour = timeinfo.tm_hour;
        int currentMinute = timeinfo.tm_min;

        // Schedule Mode
        if (currentMode == 2 && currentHour == scheduleHour && currentMinute ==
scheduleMinute && !isPumpOn) {
            digitalWrite(RELAY_PIN, LOW);
            Blynk.virtualWrite(LED_INDICATOR, 1);
            Serial.println("Pump ON");
            isScheduledOn = true;
            isPumpOn = true;
            pumpTimer.setTimeout(1000L, stopPump);
        } else if (currentMode == 2 && currentHour != scheduleHour &&
currentMinute != scheduleMinute) {
            digitalWrite(RELAY_PIN, HIGH);
            Blynk.virtualWrite(LED_INDICATOR, 0);
            Serial.println("Pump OFF");
            isPumpOn = false;
        }
    }
}
```

```

}

// Interval Mode
if (currentMode == 3) {
    vTaskResume(xHandleInterval);

} else if (currentMode != 3) {
    vTaskSuspend(xHandleInterval);
}

// Kontrol Pompa
if ((currentMode == 0 && isManuallyOn) || (currentMode == 1 &&
isAutomaticallyOn) || (currentMode == 2 && isScheduledOn)) {
    digitalWrite(RELAY_PIN, LOW);
    Blynk.virtualWrite(LED_INDICATOR, 1);
    Serial.println("Pump ON");
} else if (currentMode == 3) {
    // Mode Interval diatur dengan task interval yang sudah ada
} else {
    digitalWrite(RELAY_PIN, HIGH);
    Blynk.virtualWrite(LED_INDICATOR, 0);
    Serial.println("Pump OFF");
}

// Update setiap 1 detik
vTaskDelay(pdMS_TO_TICKS(1000));
}
}

```

2.2.2. MODULE IMPLEMENTATION

Berikut merupakan modul-modul praktikum mata kuliah Sistem Waktu Nyata (IoT) yang diimplementasikan dalam proyek “Where’s my Water”, fungsinya dalam program serta bukti penggunaannya.

No	Module	Function	Evidence
1	Introduction to RTOS and Task Scheduling	Mengimplementasikan penjadwalan tugas untuk mengelola berbagai mode kontrol pompa, seperti manual, otomatis, dan terjadwal.	Pada kode, terdapat beberapa tugas yang diatur dengan penjadwalan menggunakan FreeRTOS, seperti vTaskPump, vTaskSensor, dan vTaskInterval.
2	Memory Management and Queue	Menggunakan FreeRTOS untuk pengelolaan memori dan antrian tugas secara efisien.	Penggunaan FreeRTOS untuk multitasking di mana tugas-tugas seperti vTaskPump, vTaskSensor, dan vTaskInterval dijalankan secara bersamaan.
3	Mutex & Semaphore	Tidak diimplementasikan dalam proyek	
4	Software Timer & Hardware Interrupts	Menggunakan timer untuk mengatur interval waktu pompa dan memanfaatkan tugas untuk menunggu waktu tertentu.	Fungsi BlynkTimer digunakan untuk mengatur waktu matinya pompa setelah aktif selama beberapa detik. Timer ini juga memanfaatkan setTimeout.
5	Deadlock & Priority Inversion	Tidak diimplementasikan dalam proyek	
6	Bluetooth & BLE	Tidak diimplementasikan dalam proyek	
7	WiFi, HTTP, &	Menghubungkan perangkat ke	Kode menggunakan

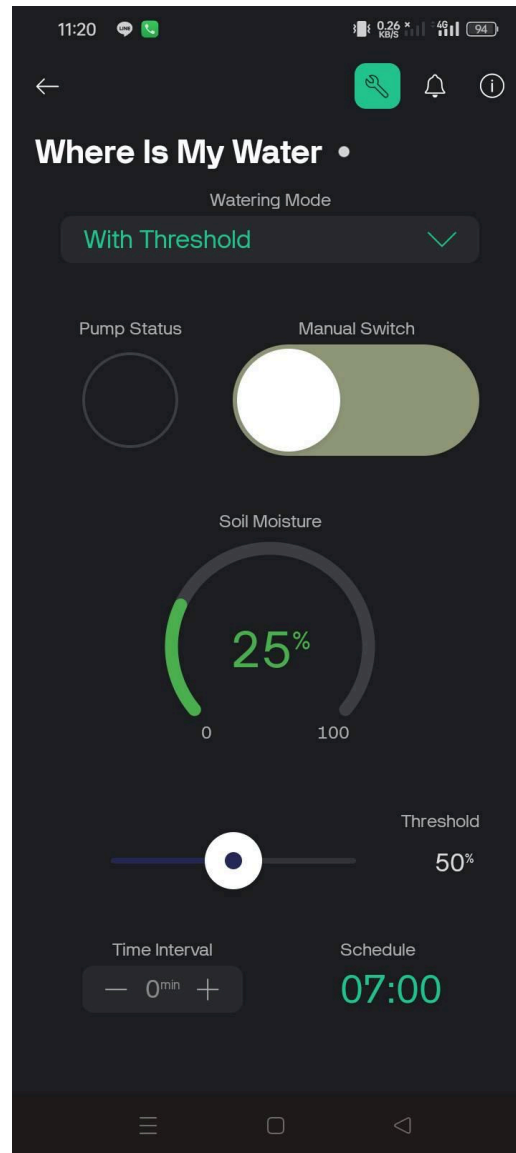
	MQTT	jaringan Wi-Fi dan platform IoT Blynk untuk kontrol dan pemantauan secara remote	pustaka WiFi.h dan BlynkSimpleEsp32.h untuk menghubungkan ESP32 ke jaringan Wi-Fi dan platform Blynk, serta mengirim data.
8	Mesh	Tidak diimplementasikan dalam proyek	
9	IoT Platform - Blynk	Platform untuk mengontrol dan memantau pompa penyiraman secara real-time dari jarak jauh	Modul Blynk diintegrasikan melalui Blynk.begin() untuk autentikasi dan mengirimkan data serta status pompa ke aplikasi Blynk dengan Blynk.virtualWrite().

2.3 HARDWARE AND SOFTWARE INTEGRATION

Berdasarkan rangkaian yang telah dibuat secara fisik maupun yang telah dibuat pada software Proteus, dapat dilihat bahwa sensor YL-69 secara langsung disambungkan dengan ESP32 melalui pin ADC dengan menghubungkan pin A0 pada sensor dengan pin 33 pada ESP32. Pin VCC dan GND pada sensor sendiri dihubungkan dengan breadboard untuk mempermudah pembuatan rangkaian. Selain itu, dalam rangkaian kami terdapat juga relay module yang mengatur masuk atau berhentinya arus listrik ke pompa untuk mengatur pompa itu sendiri. Di mana pin IN pada relay module dihubungkan dengan pin 32 pada ESP32. Tidak seperti pin 33 yang berfungsi sebagai input, pin 32 ini berfungsi sebagai pin output yang akan mengirimkan sinyal dari ESP32 berdasarkan kondisi dan logika yang sedang berjalan ke pompa yang terhubung dengan relay module.

Untuk pengeksekusian program sendiri, kami menggunakan software Arduino Uno untuk mengupload kode yang sudah dibuat ke ESP32 yang telah disambungkan ke komputer dengan menggunakan library DOIT ESP32 DEVKIT V1. Uploading data ini dapat dilakukan menggunakan Arduino IDE karena tersedia berbagai library yang diperlukan oleh sistem untuk membaca dan memproses kode secara real-time.

Untuk user interface sendiri, kami menggunakan Blynk yang sudah dikonfigurasi dengan datastream yang dibutuhkan. Berikut ini UI Blynk yang dapat diakses dengan aplikasi *mobile*.



Gambar 2.3.1. Tampilan console Blynk pada *mobile app*

CHAPTER 3

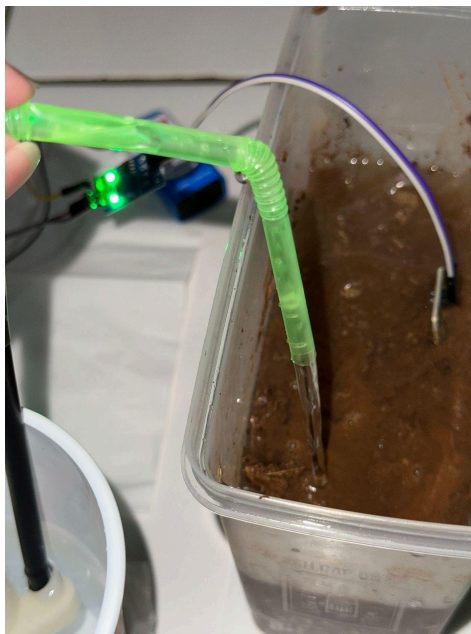
TESTING AND EVALUATION

3.1 TESTING

Setelah fase desain dan implementasi telah berhasil dilakukan, proses developing proyek ini kami lanjutkan dengan fase testing. Di mana, pada fase ini rangkaian dan kode yang telah dibuat akan diuji dengan beberapa kondisi dengan mode yang berbeda-beda untuk memastikan apakah sistem dapat bekerja dengan baik.

Proyek dapat dikatakan berhasil apabila ketika rangkaian dinyalakan, program akan memulai dengan mode default, yaitu automatic with threshold. Di mana user dapat memasukkan nilai threshold yang dibutuhkan, dan sensor akan mengirimkan data kelembaban tanah untuk kemudian dibandingkan dengan nilai threshold. Ketika nilai pembacaan sensor lebih kecil dari nilai threshold, maka pompa akan dinyalakan hingga kelembaban tanah mencapai threshold. Begitu juga sebaliknya.

3.1.1. Testing Manual Mode

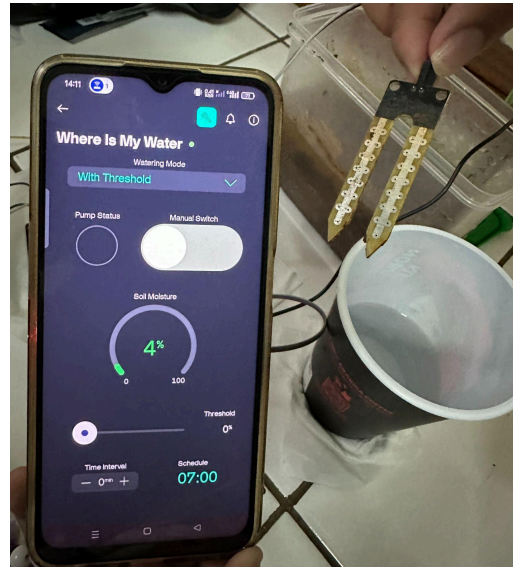


Pada mode manual, pengujian dilakukan dengan memanfaatkan tombol virtual pada aplikasi Blynk (V0) untuk menyalakan dan mematikan pompa secara langsung. Hasil pengujian menunjukkan bahwa sistem merespons perintah dengan cepat, dimana ketika tombol dinyalakan, pompa langsung aktif, dan indikator LED pada aplikasi menyala untuk memberikan status real-time. Sebaliknya, ketika tombol dimatikan,

pompa berhenti dengan segera, dan indikator LED berubah menjadi mati. Mode manual ini memastikan bahwa pengguna memiliki kontrol penuh terhadap pompa air sesuai kebutuhan, terlepas dari nilai kelembaban tanah yang terdeteksi.

3.1.2. Testing Automatic with Threshold Mode

Pada mode ini, nilai threshold kelembaban diatur menggunakan slider (V1) pada aplikasi Blynk. Sistem akan membandingkan nilai kelembaban yang dibaca sensor dengan nilai threshold yang telah ditetapkan. Jika kelembaban tanah lebih kecil dari threshold, pompa akan menyala secara otomatis, dan indikator LED akan menyala.



Pengujian dilakukan dengan mengatur berbagai nilai threshold, seperti pada gambar, serta memastikan pompa berhenti ketika kelembaban tanah mencapai atau melebihi nilai threshold tersebut. Hasil pengujian menunjukkan sistem berjalan dengan baik, di mana pompa hanya aktif ketika kondisi tanah kering sesuai logika threshold.

3.1.3. Testing Automatic with Schedule Mode

Mode jadwal diuji dengan menggunakan input waktu dari aplikasi Blynk (V5), di mana pengguna dapat mengatur jam dan menit untuk menyalakan pompa secara otomatis. Pengujian dilakukan dengan menetapkan waktu tertentu (misalnya pukul 13:00) dan memantau apakah pompa menyala tepat pada waktu yang ditentukan. Hasil pengujian menunjukkan bahwa sistem berhasil mengeksekusi penyalan pompa sesuai jadwal yang ditetapkan, dengan akurasi waktu ± 1 detik. Setelah pompa aktif selama 15 detik, pompa otomatis akan berhenti dan indikator LED kembali mati. Mode ini memastikan penyiraman berjalan sesuai jadwal yang telah diatur pengguna.

3.1.4. Testing Automatic with Interval Mode

Pada mode interval, pengujian dilakukan dengan mengatur interval waktu melalui slider (V4) di aplikasi Blynk, misalnya setiap 10 menit. Sistem diuji untuk memastikan pompa menyala secara berkala sesuai interval yang diatur. Hasil pengujian menunjukkan bahwa pompa berhasil aktif dalam

interval waktu yang telah ditentukan, dan indikator LED pada aplikasi menyala untuk memberi status real-time. Setelah durasi penyiraman selama 5 detik selesai, pompa akan mati dan menunggu interval berikutnya. Mode ini bekerja dengan baik dan cocok untuk situasi di mana penyiraman rutin diperlukan dalam jangka waktu tertentu.

3.2 RESULT

Berdasarkan percobaan yang telah dilakukan, diperoleh hasil sebagai berikut.



Gambar 3.2.1. Hasil testing rangkaian ketika nilai pembacaan sensor melebihi nilai threshold

Berdasarkan dengan hasil testing yang sudah dilakukan, dapat disimpulkan bahwa sistem *Where's My Water* sudah dapat bekerja dengan baik dan efektif, sesuai dengan tujuan dibuatnya sistem ini.

3.3 EVALUATION

Proyek *Where's My Water? (Smart Plant Watering System)* ini bertujuan untuk mendeteksi kelembaban tanah, sehingga dapat memudahkan manusia dalam sistem irigasi otomatis yang efisien dan efektif.

Berdasarkan pengamatan yang sudah dilakukan, diketahui bahwa sensor kelembaban tanah YL-69 mampu memberikan data kelembaban tanah dalam bentuk persentase yang stabil dan akurat. Di mana sensor akan membaca nilai baru setiap 3 detik untuk memastikan perubahan kelembaban terdeteksi dengan cepat. Selain berdasarkan sensor, ditemukan juga keberhasilan sistem dalam pengamatan cara kerja pompa dengan kondisi-kondisi berikut:

- a. Mode Manual
Switch pada Blynk (V0) dapat digunakan untuk mengontrol pompa secara manual.
- b. Mode Otomatis dengan Threshold
Pompa aktif ketika kelembaban tanah di bawah nilai threshold yang ditetapkan melalui slider.
- c. Mode Otomatis dengan Jadwal
Pompa menyala setiap harinya sesuai waktu yang diatur pada input jadwal.
- d. Mode Otomatis dengan Interval
Pompa menyala secara berkala berdasarkan waktu interval yang dipilih melalui slider.

Selain itu, ditemukan juga bahwa sistem telah berhasil terkoneksi ke WiFi dengan stabil melalui kredensial yang ditetapkan (SSID dan password). Kemudian, data dari sensor dan status pompa dikirim ke aplikasi Blynk real-time. Dan juga respons dari aplikasi Blynk ke ESP32 sangat cepat (benar-benar bekerja secara real-time).

Dengan evaluasi ini, dapat disimpulkan bahwa sistem telah bekerja secara efektif dan efisien. Selain itu, interface Blynk memungkinkan sistem menjadi lebih user-friendly dan stabil. Dengan ini, diharapkan sistem dapat diimplementasikan pada skala kecil hingga menengah untuk kebutuhan penyiraman tanaman otomatis, baik di rumah maupun di lahan pertanian.

CHAPTER 4

CONCLUSION

Where's My Water? yang merupakan suatu sistem yang kami rancang sebagai Smart Plant Watering System. Di dalam rangkaian ini terdapat sensor kelembaban tanah YL-69 yang berfungsi untuk mendeteksi status kelembaban tanah secara real-time. Sensor ini secara langsung mengambil data dari tanah dan mengirimkan data pembacaan tersebut ke ESP32. Berdasarkan data yang telah diterima ini, ESP32 akan melakukan analisis data, dimana pompa akan bekerja sesuai dengan mode yang dipilih oleh user. Dan user dapat dengan mudah memantau sistem kerja dari rangkaian ini beserta dengan nilai kelembaban tanah secara real-time dan status pompa melalui aplikasi *mobile* Blynk.

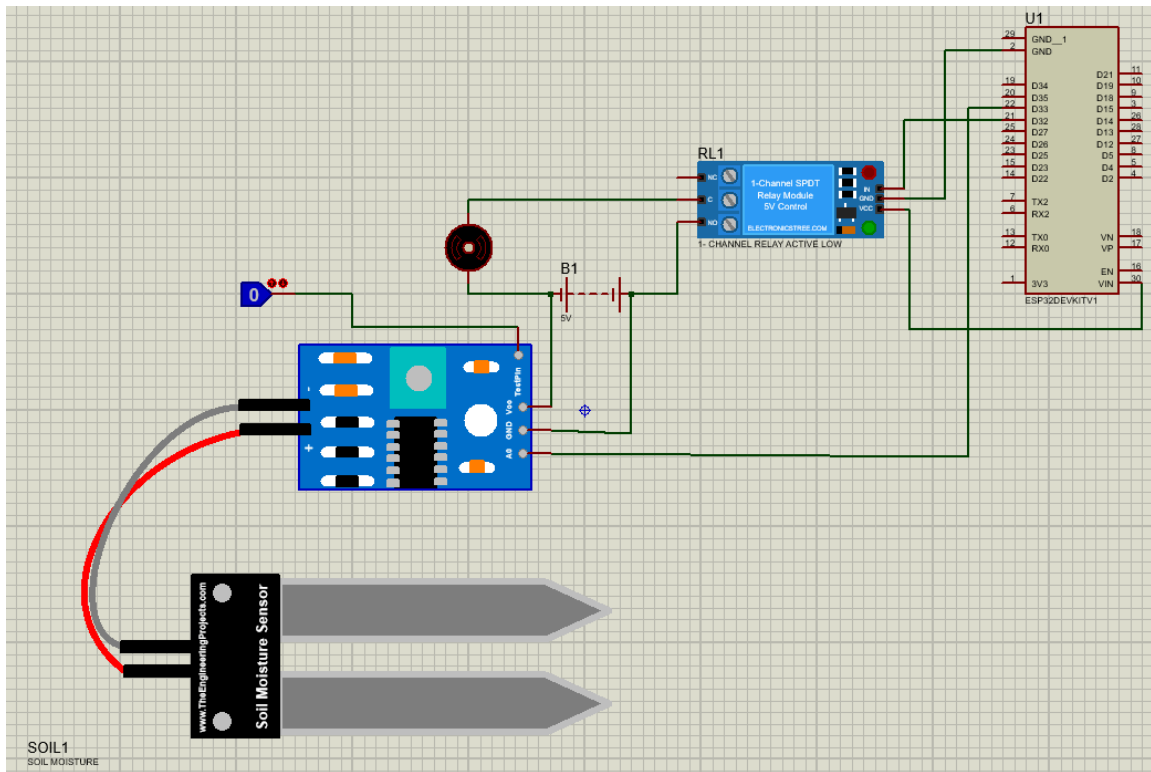
Proyek ini mengimplementasikan beberapa modul yang telah kami pelajari selama praktikum Sistem Waktu Nyata dan IOT dalam satu semester ini, antara lain yaitu modul 1, 2, 4, 7, dan 9. Dengan ini kami harapkan proyek *Where's My Water?* dapat membantu masyarakat, terutama pada aspek pertanian. Kami harap proyek kami dapat dipergunakan dengan baik, kemudian juga kami berharap bahwa *Where's My Water?* dapat dikembangkan menjadi rangkaian yang lebih baik lagi.

REFERENCES

- [1] Digital Laboratory. "Module 1: Introduction to RTOS and Task Scheduling". [Module 1: Introduction to RTOS and Task Scheduling URL](#). [Accessed 2 December, 2024]
- [2] Digital Laboratory. "Module 2: Memory Management and Queue". [Module 2: Memory Management and Queue URL](#). [Accessed 2 December, 2024]
- [3] Digital Laboratory. "Module 4: Software Timer & Hardware Interrupts". [Dasar Teori Modul 4 File](#). [Accessed 3 December, 2024]
- [4] Digital Laboratory. "Module 7: WiFi, HTTP, & MQTT". [Modul 7: WiFi, HTTP, & MQTT URL](#). [Accessed 8 December, 2024]
- [5] Digital Laboratory. "Module 9: IoT Platform - Blynk". [Module 9: IoT Platform - Blynk URL](#). [Accessed 9 December, 2024]

APPENDICES

Appendix A: Project Schematic



Appendix B: Documentation

Dokumentasi dapat diakses pada link di bawah ini.

Testing Documentation.MOV