

**PERANCANGAN SISTEM MONITORING GUNA  
DETEKSI GANGGUAN LAMPU LALU LINTAS BERBASIS  
*INTERNET OF THINGS* MENGGUNAKAN ESP8266,  
PZEM004T, DAN PLATFORM BLYNK**

**SKRIPSI**



Oleh:  
**Ana Jefri Hamzah**  
**2203025003**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI DAN INFORMATIKA  
UNIVERSITAS MUHAMMADIYAH PROF. DR. HAMKA  
JAKARTA  
2026**

**PERANCANGAN SISTEM MONITORING GUNA DETEKSI  
GANGGUAN LAMPU LALU LINTAS BERBASIS *INTERNET  
OF THINGS* MENGGUNAKAN ESP8266, PZEM004T, DAN  
PLATFORM BLYNK**

**SKRIPSI**

Disusun untuk Memenuhi Persyaratan Kelulusan Sarjana Teknik Elektro



Oleh:  
**Ana Jefri Hamzah**  
**2203025003**

**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INDUSTRI DAN INFORMATIKA  
UNIVERSITAS MUHAMMADIYAH PROF. DR. HAMKA  
JAKARTA  
2026**

## HALAMAN PERSETUJUAN

### HALAMAN PERSETUJUAN

PERANCANGAN SISTEM MONITORING GUNA DETEKSI GANGGUAN  
LAMPU LALU LINTAS BERBASIS *INTERNET OF THINGS*  
MENGUNAKAN ESP8266, PZEM004T, DAN PLATFORM BLYNK

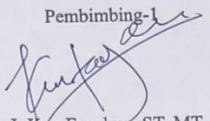
SKRIPSI

Dibuat untuk Memenuhi Persyaratan Kelulusan Sarjana Teknik Elektro

Oleh:  
Ana Jefri Hamzah  
2203025003

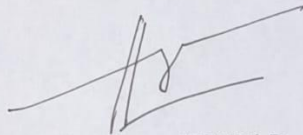
Telah diperiksa dan disetujui untuk diajukan ke Sidang Ujian Skripsi  
Program Studi Teknik Informatika Fakultas Teknologi Industri dan Informatika  
UHAMKA  
Tanggal, 19 Juli 2026

Pembimbing-1



Ir. Kun Fayakun, ST, MT  
NIDN. 0305125701

Mengetahui,  
Ketua Program Studi Teknik Elektro



Ir. Harry Ramza, ST, MT, Ph.D  
NIDN. 0303097006

## HALAMAN PENGESAHAN

### HALAMAN PENGESAHAN

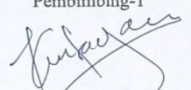
PERANCANGAN SISTEM MONITORING GUNA DETEKSI GANGGUAN  
LAMPU LALU LINTAS BERBASIS *INTERNET OF THINGS*  
MENGUNAKAN ESP8266, PZEM004T, DAN PLATFORM BLYNK

SKRIPSI

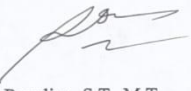
Oleh:  
Ana Jefri Hamzah  
2203025003

Telah diuji dan dinyatakan lulus dalam Sidang Ujian Skripsi  
Program Studi Teknik Informatika Fakultas Teknologi Industri dan Informatika  
UHAMKA  
Tanggal, 6 Agustus 2026

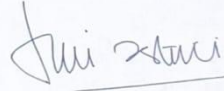
Pembimbing-1

  
Ir. Kun Fayakun, ST, MT  
NIDN. 0305125701

Penguji-1

  
Rosalina, S.T., M.T.  
NIDN. 0304017001

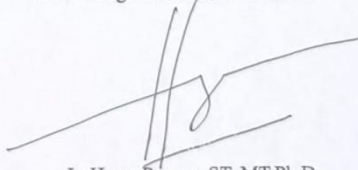
Penguji-2

  
Dr. Dwi Astuti Cahyasiwi, S.T., M.T.  
NIDN. 0323027401

Mengesahkan,  
Dekan  
Fakultas Teknologi Industri dan  
Informatika UHAMKA

  
Dr. Dan Mugisidi, ST., M.Si  
NIDN. 0301126901

Mengetahui,  
Ketua Program Studi Teknik Elektro

  
Ir. Harry Ramza, ST, MT, Ph.D  
NIDN. 0303097006

## PERNYATAAN KEASLIAN

Saya, yang membuat pernyataan

Nama : Ana Jefri Hamzah

NIM : 2203025003

Judul skripsi : Perancangan Sistem Monitoring Guna Deteksi Gangguan Lampu Lalu Lintas Berbasis *Internet of Things* Menggunakan ESP8266, PZEM004T, dan Platform Blynk

Menyatakan bahwa, skripsi ini merupakan karya saya sendiri (ASLI) dan isi dalam skripsi ini tidak terdapat karya yang pernah diajukan oleh orang lain untuk memperoleh gelar akademis di suatu institusi pendidikan tinggi mana pun, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis dan/atau diterbitkan oleh orang lain, KECUALI yang secara tertulis diacu dalam naskah ini dan disebutkan dalam Daftar Referensi.

Segala sesuatu yang terkait dengan naskah dan karya yang telah dibuat adalah menjadi tanggung jawab saya pribadi.



## KATA PENGANTAR

Puji syukur kehadiran Allah SWT atas segala rahmat, hidayah, dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul "**Perancangan Sistem Monitoring Guna Deteksi Gangguan Lampu Lalu Lintas Berbasis Internet of Things Menggunakan ESP8266, PZEM004T, dan Platform Blynk**" dengan baik. Skripsi ini disusun untuk memenuhi salah satu persyaratan dalam memperoleh gelar Sarjana Teknik pada Program Studi Teknik Elektro, Fakultas Teknologi Industri dan Informatika, Universitas Muhammadiyah Prof. Dr. Hamka.

Penulis menyadari bahwa keberhasilan dalam penyusunan skripsi ini tidak lepas dari bimbingan, arahan, dukungan, serta bantuan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis ingin menyampaikan rasa hormat dan terima kasih yang sebesar-besarnya kepada:

1. **Prof. Dr. Gunawan Suryoputro, M.Hum.**, selaku Rektor Universitas Muhammadiyah Prof. Dr. Hamka.
2. **Dr. Ir. Dan Mugisidi, ST., M.Si. IPP** selaku Dekan Fakultas Teknologi Industri dan Informatika UHAMKA.
3. **Ir. Harry Ramza, ST, MT, Ph.D.**, selaku Ketua Program Studi Teknik Elektro UHAMKA yang telah memberikan arahan dan dukungan kepada penulis.
4. **Ir. Kun Fayakun, ST, MT.**, selaku Dosen Pembimbing yang telah meluangkan waktu, memberikan bimbingan, masukan yang sangat berharga, serta motivasi yang luar biasa selama proses penyusunan skripsi ini.
5. Seluruh staf dosen dan karyawan Program Studi Teknik Elektro UHAMKA yang telah membekali penulis dengan ilmu pengetahuan selama masa perkuliahan.
6. Kedua orang tua dan keluarga besar tercinta yang senantiasa memberikan doa yang tidak pernah putus, dukungan moril, maupun materil hingga penulis dapat menyelesaikan studi ini.

7. Rekan-rekan mahasiswa angkatan 2022 serta semua pihak yang tidak dapat penulis sebutkan satu per satu, terima kasih atas bantuan, diskusi, dan semangat kebersamaan selama ini.
8. Untuk Nisyatul Kharimah, Calon Istri Tercinta, Terima kasih kepada Nisyatul Kharimah, seseorang yang selalu memberikan dukungan, perhatian, semangat, dan doa selama proses penyelesaian skripsi ini. Terima kasih telah menjadi tempat berbagi cerita, mendengarkan keluh kesah, dan memberikan semangat ketika saya merasa lelah dalam menjalani proses pendidikan ini.
9. Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan dikarenakan keterbatasan yang penulis miliki. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun demi perbaikan di masa yang akan datang. Akhir kata, semoga skripsi ini dapat memberikan manfaat serta kontribusi positif bagi perkembangan ilmu pengetahuan, khususnya di bidang Teknik Elektro dan teknologi *Internet of Things* (IoT).

Jakarta, 18 Juli 2026

Penulis,

**Ana Jefri Hamzah**

NIM. 2203025003

## **PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademika Universitas Muhammadiyah Prof. DR. HAMKA (UHAMKA), saya yang bertandatangan di bawah ini:

Nama : Ana Jefri Hamzah

NIM : 2203025003

Judul skripsi : Teknik Elektro

Menyetujui, memberikan Hak Bebas Royalti Noneksklusif (*non-exclusive royalty free right*) kepada Universitas Muhammadiyah Prof. DR. HAMKA (UHAMKA) atas karya ilmiah saya beserta perangkat yang ada (jika diperlukan) yang berjudul:

Perancangan Sistem Monitoring Guna Deteksi Gangguan Lampu Lalu Lintas  
Berdasarkan *Internet of Things* Menggunakan ESP8266, PZEM004T, dan Platform  
Blynk

Hak Bebas Royalti Noneksklusif ini Universitas Muhammadiyah Prof. DR. HAMKA berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (database), merawat, dan memublikasikan skripsi saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Segala sesuatu yang terkait dengan naskah dan karya yang telah dibuat adalah menjadi tanggung jawab saya pribadi.

Jakarta, 18 Juli 2026

Ana Jefri Hamzah

## ABSTRAK

### **Perancangan Sistem Monitoring Guna Deteksi Gangguan Lampu Lalu Lintas Berbasis *Internet of Things* Menggunakan ESP8266, PZEM004T, dan Platform Blynk**

Ana Jefri Hamzah

Lampu lalu lintas merupakan salah satu perangkat penting dalam mengatur kelancaran arus kendaraan di persimpangan jalan. Gangguan pada lampu lalu lintas dapat menyebabkan kemacetan, meningkatkan risiko kecelakaan, serta mengganggu kelancaran lalu lintas. Selain itu, proses monitoring kondisi lampu masih banyak dilakukan secara manual sehingga keterlambatan dalam mengetahui adanya gangguan sering terjadi. Penelitian ini bertujuan untuk merancang sistem monitoring dan deteksi gangguan lampu lalu lintas berbasis *Internet of Things* menggunakan NodeMCU ESP8266, sensor PZEM004T, sensor LDR, AC Dimmer, dan platform Blynk. Sensor LDR digunakan untuk mendeteksi intensitas cahaya lingkungan sebagai dasar pengaturan tingkat kecerahan lampu melalui AC Dimmer, sedangkan sensor PZEM004T digunakan untuk membaca parameter kelistrikan berupa tegangan, arus, daya, energi, frekuensi, dan faktor daya. Seluruh data diproses oleh NodeMCU ESP8266 dan dikirimkan ke platform Blynk sehingga kondisi lampu dapat dipantau secara *real-time* melalui smartphone. Berdasarkan hasil pengujian, sistem berhasil mengintegrasikan fungsi monitoring parameter kelistrikan dengan pengaturan intensitas cahaya dalam satu sistem. Parameter tegangan dan frekuensi berhasil terbaca pada ketiga sensor PZEM004T, sedangkan parameter arus, daya, dan faktor daya hanya berhasil terbaca pada sensor yang menggunakan beban lampu pijar. Meskipun demikian, sistem tetap mampu menampilkan informasi kondisi lampu, status gangguan, nilai intensitas cahaya, serta tingkat kecerahan lampu melalui platform Blynk secara *real-time*. Dengan demikian, sistem yang dikembangkan mampu mendukung proses monitoring dan deteksi gangguan lampu lalu lintas secara lebih efektif melalui integrasi fungsi monitoring dan pengaturan intensitas cahaya berbasis *Internet of Things*.

**Kata kunci:** *Internet of Things, NodeMCU ESP8266, PZEM004T, Lampu Lalu Lintas, Sensor LDR, Blynk.*

## ABSTRACT

### **Design of an Internet of Things-Based Traffic Light Monitoring for Fault Detection System Using ESP8266, PZEM004T, and the Blynk Platform**

Ana Jefri Hamzah

Traffic lights are essential devices for regulating vehicle flow at road intersections. Failures in traffic lights can lead to traffic congestion, increase the risk of accidents, and disrupt traffic flow. In addition, traffic light monitoring is still commonly performed manually, resulting in delays in detecting failures. This research aims to design an Internet of Things-based traffic light monitoring and fault detection system using the NodeMCU ESP8266, PZEM004T sensor, LDR sensor, AC Dimmer, and the Blynk platform. The LDR sensor is used to detect ambient light intensity as the basis for automatically adjusting the brightness of the traffic lights through the AC Dimmer, while the PZEM004T sensor measures electrical parameters including voltage, current, power, energy, frequency, and power factor. All data are processed by the NodeMCU ESP8266 and transmitted to the Blynk platform, enabling real-time monitoring via a smartphone. The experimental results show that the proposed system successfully integrates electrical parameter monitoring with automatic light intensity control into a single system. Voltage and frequency were successfully measured by all three PZEM004T sensors, while current, power, and power factor were successfully measured only on the sensor connected to an incandescent lamp load. Nevertheless, the system was able to display traffic light conditions, fault status, ambient light intensity, and lamp brightness levels through the Blynk platform in real time. Therefore, the developed system can support a more effective traffic light monitoring and fault detection process by integrating monitoring and automatic light intensity control within an Internet of Things framework.

**Keywords:** *Internet of Things, NodeMCU ESP8266, PZEM004T, Traffic Light, LDR Sensor, Blynk.*

## DAFTAR ISI

|                                                                       |      |
|-----------------------------------------------------------------------|------|
| HALAMAN PERSETUJUAN.....                                              | iii  |
| HALAMAN PENGESAHAN.....                                               | iv   |
| PERNYATAAN KEASLIAN.....                                              | v    |
| KATA PENGANTAR.....                                                   | vi   |
| PERSETUJUAN PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN<br>AKADEMIS..... | viii |
| ABSTRAK.....                                                          | ix   |
| ABSTRACT.....                                                         | x    |
| DAFTAR ISI.....                                                       | xi   |
| DAFTAR TABEL.....                                                     | xiv  |
| DAFTAR GAMBAR.....                                                    | xv   |
| DAFTAR LAMPIRAN.....                                                  | xvii |
| BAB I.....                                                            | 1    |
| PENDAHULUAN.....                                                      | 1    |
| 1.1    Latar Belakang Masalah.....                                    | 1    |
| 1.2    Perumusan Masalah.....                                         | 3    |
| 1.3    Batasan Masalah.....                                           | 3    |
| 1.4    Tujuan Penelitian.....                                         | 3    |
| 1.5    Manfaat Penelitian.....                                        | 4    |
| 1.6    Sistematika Penelitian.....                                    | 4    |
| BAB II.....                                                           | 7    |
| DASAR TEORI.....                                                      | 7    |
| 2.1    Dasar Teori.....                                               | 7    |
| 2.1.1    Lalu Lintas.....                                             | 7    |

|                            |                                                    |    |
|----------------------------|----------------------------------------------------|----|
| 2.1.2                      | Lampu Lalu Lintas .....                            | 8  |
| 2.1.3                      | Sistem Monitoring.....                             | 9  |
| 2.1.4                      | Gangguan Kelistrikan pada Lampu Lalu Lintas .....  | 10 |
| 2.1.5                      | <i>Internet of Things</i> (IoT).....               | 11 |
| 2.1.6                      | Mikrokontroler NodeMCU ESP8266 .....               | 12 |
| 2.1.7                      | Sensor PZEM004T.....                               | 14 |
| 2.1.8                      | Sensor <i>Light Dependent Resistor</i> (LDR) ..... | 17 |
| 2.1.9                      | AC Dimmer.....                                     | 19 |
| 2.1.10                     | Blynk.....                                         | 21 |
| 2.2                        | Penelitian yang Relevan.....                       | 23 |
| 2.3                        | Kerangka Teori.....                                | 25 |
| BAB III .....              |                                                    | 24 |
| METODOLOGI .....           |                                                    | 24 |
| 3.1                        | Alur Perancangan .....                             | 24 |
| 3.2                        | Identifikasi Kebutuhan .....                       | 26 |
| 3.2.1                      | <i>Hardware</i> (Perangkat Keras).....             | 26 |
| 3.2.2                      | <i>Software</i> (Perangkat Lunak).....             | 28 |
| 3.3                        | Metode Perancangan .....                           | 30 |
| 3.3.1                      | Gambaran Umum Sistem .....                         | 30 |
| 3.3.2                      | <i>Flowchart</i> Program .....                     | 31 |
| 3.3.3                      | Cara Kerja Sistem .....                            | 32 |
| BAB IV .....               |                                                    | 34 |
| HASIL DAN PEMBAHASAN ..... |                                                    | 34 |
| 4.1                        | Perancangan .....                                  | 34 |
| 4.2                        | Hasil Pengujian .....                              | 35 |

|                     |                                    |    |
|---------------------|------------------------------------|----|
| 4.2.1               | Pengujian LCD 16x2 I2C.....        | 35 |
| 4.2.2               | Pengujian Sensor LDR.....          | 36 |
| 4.2.3               | Pengujian AC Dimmer Robotdyn ..... | 38 |
| 4.2.4               | Pengujian Sensor PZEM .....        | 39 |
| 4.2.5               | Pengujian Koneksi Wi-fi.....       | 45 |
| 4.2.6               | Pengujian Blynk.....               | 46 |
| 4.2.7               | Pengujian Sistem.....              | 48 |
| 4.3                 | Pembahasan.....                    | 51 |
| BAB V.....          |                                    | 54 |
| SIMPULAN .....      |                                    | 54 |
| 5.1                 | Kesimpulan .....                   | 54 |
| 5.2                 | Saran.....                         | 55 |
| DAFTAR PUSTAKA..... |                                    | 56 |
| LAMPIRAN.....       |                                    | 59 |

## DAFTAR TABEL

|                                                                                                            |    |
|------------------------------------------------------------------------------------------------------------|----|
| <b>Tabel 2.1</b> Datasheet NodeMCU ESP8266 .....                                                           | 14 |
| <b>Tabel 2.2</b> Datasheet Sensor PZEM004T.....                                                            | 16 |
| <b>Tabel 3.1</b> Alat dan Material (Perangkat Keras) .....                                                 | 27 |
| <b>Tabel 4.1</b> Hasil Pembacaan Sensor LDR.....                                                           | 37 |
| <b>Tabel 4.2</b> Hasil Pengujian Akurasi Pembacaan Parameter Kelistrikan pada Beban Lampu LED 5 W. ....    | 42 |
| <b>Tabel 4.3</b> Hasil Pengujian Akurasi Pembacaan Parameter Kelistrikan pada Beban Lampu Pijar 10 W. .... | 43 |
| <b>Tabel 4.4</b> Hasil Pengujian Akurasi Pembacaan Parameter Kelistrikan pada Beban Lampu Pijar 15 W. .... | 43 |
| <b>Tabel 4.5</b> Hasil Pengujian Akurasi Pembacaan Parameter Kelistrikan pada Beban Lampu Pijar 25 W. .... | 44 |
| <b>Tabel 4.6</b> Hasil Pengujian Latensi Pengiriman Data ke Platform Blynk.....                            | 47 |
| <b>Tabel 4.7</b> Hasil Pengujian Deteksi Gangguan pada Sistem Monitoring <i>Traffic Light</i> .....        | 50 |

## DAFTAR GAMBAR

|                                                                                      |    |
|--------------------------------------------------------------------------------------|----|
| <b>Gambar 2.1</b> Lampu Lalu Lintas.....                                             | 8  |
| <b>Gambar 2.2</b> <i>Internet of Things</i> .....                                    | 11 |
| <b>Gambar 2.3</b> NodeMCU ESP8266.....                                               | 13 |
| <b>Gambar 2.4</b> Sensor PZEM004T .....                                              | 15 |
| <b>Gambar 2.5</b> Sensor LDR .....                                                   | 17 |
| <b>Gambar 2.6</b> AC Dimmer .....                                                    | 20 |
| <b>Gambar 2.7</b> Rangkaian <i>Zero Crossing Detector</i> .....                      | 20 |
| <b>Gambar 2.8</b> Tampilan Dashboard Platform Blynk .....                            | 21 |
| <b>Gambar 2.9</b> Tampilan Platform Blynk pada Handphone.....                        | 22 |
| <b>Gambar 2.10</b> Pemrograman untuk terhubung ke Blynk .....                        | 23 |
| <b>Gambar 3.1</b> Diagram Alur Perencanaan .....                                     | 24 |
| <b>Gambar 3.2</b> Tampilan Layar Arduino IDE .....                                   | 29 |
| <b>Gambar 3.3</b> Blok Diagram Sistem.....                                           | 30 |
| <b>Gambar 3.4</b> Flowchart Pemrograman.....                                         | 31 |
| <b>Gambar 4.1</b> <i>Wiring Diagram</i> Sistem Monitoring <i>Traffic Light</i> ..... | 34 |
| <b>Gambar 4.2</b> Hasil Pengujian LCD 16x2 I2C .....                                 | 36 |
| <b>Gambar 4.3</b> Hasil Pengujian Sensor LDR .....                                   | 37 |
| <b>Gambar 4.4</b> Hasil Pengujian AC Dimmer Robotdyn .....                           | 39 |
| <b>Gambar 4.5</b> Hasil Pemberian Alamat Baru pada Sensor PZEM004T .....             | 40 |
| <b>Gambar 4.6</b> Hasil Pengujian Pembacaan Sensor PZEM004T dengan AVO meter .....   | 41 |
| <b>Gambar 4.7</b> Hasil Pengujian Koneksi Wi-fi .....                                | 46 |
| <b>Gambar 4.8</b> Hasil Pengujian Koneksi ke Blynk .....                             | 47 |
| <b>Gambar 4.9</b> Hasil Pengujian Sistem Monitoring <i>Traffic Light</i> .....       | 48 |

|                                                               |    |
|---------------------------------------------------------------|----|
| <b>Gambar 4.10</b> Tampilan Layar Monitoring pada Blynk ..... | 49 |
| <b>Gambar 4.11</b> Wiring Diagram Akhir Sistem .....          | 52 |

## DAFTAR LAMPIRAN

|                  |    |
|------------------|----|
| Lampiran 1 ..... | 59 |
|------------------|----|

# **BAB I**

## **PENDAHULUAN**

### **1.1 Latar Belakang Masalah**

Berdasarkan data BPS, banyaknya kendaraan bermotor di Indonesia pada tahun 2025 telah melampaui 166 juta unit. Peningkatan tersebut menunjukkan bahwa aktivitas transportasi darat terus berkembang setiap tahun. Dampaknya, sistem pengaturan lalu lintas yang andal sangat penting mengingat meningkatnya volume lalu lintas di ruas-ruas jalan, terutama di persimpangan. Dalam kondisi tersebut, berfungsinya rambu lalu lintas dengan baik sangat penting untuk pengaturan arus lalu lintas.

Namun, lampu lalu lintas tidak selalu berada dalam kondisi tanpa kendala. Gangguan seperti lampu yang padam, suplai listrik terputus, atau kerusakan pada komponen tertentu masih dapat terjadi. Jika gangguan tersebut terjadi pada persimpangan dengan volume kendaraan yang tinggi, dampaknya dapat langsung dirasakan oleh pengguna jalan. Pada kondisi tertentu, situasi ini dapat memicu terjadinya kecelakaan karena setiap pengendara berusaha melewati persimpangan secara bersamaan.

Dalam praktiknya, informasi mengenai gangguan lampu lalu lintas sering kali baru diketahui setelah adanya laporan dari masyarakat atau ketika petugas menemukan gangguan saat berada di lapangan. Kondisi ini menyebabkan waktu yang dibutuhkan untuk mengetahui adanya gangguan menjadi lebih lama karena belum tersedia sistem yang dapat memberikan informasi secara otomatis kepada petugas.

Meskipun laporan mengenai gangguan telah diterima, proses penanganan tetap membutuhkan waktu karena petugas harus melakukan verifikasi sebelum melakukan perbaikan. Oleh karena itu, semakin cepat informasi mengenai gangguan diterima, semakin cepat pula proses tindak lanjut dapat dilakukan.

Dari permasalahan tersebut terlihat bahwa kebutuhan utama bukan hanya memperbaiki lampu lalu lintas yang mengalami gangguan, tetapi juga menyediakan sistem yang bisa memberikan informasi mengenai keadaan lampu dengan cepat

serta bisa diakses dari jarak jauh. Dengan adanya informasi yang diterima lebih awal, petugas dapat segera mengetahui lokasi dan kondisi gangguan sehingga proses tindak lanjut dapat dipersiapkan lebih cepat dibandingkan apabila hanya mengandalkan laporan masyarakat atau pemeriksaan langsung di lapangan.

Kemajuan dalam teknologi IoT telah membuka peluang untuk mengatasi kesulitan-kesulitan ini. Melalui teknologi ini, kondisi suatu perangkat dapat dipantau secara jarak jauh dan informasi dapat dikirim secara langsung kepada pengguna melalui jaringan internet. Penerapan IoT pada sistem lampu lalu lintas diharapkan dapat membantu petugas mengetahui adanya gangguan lebih cepat tanpa harus menunggu laporan dari masyarakat ataupun melakukan pengecekan secara langsung ke lapangan.

Beberapa studi sebelumnya telah menerapkan teknologi Internet of Things dalam pengelolaan lampu lalu lintas. Sistem pemantauan kerusakan lampu lalu lintas menggunakan ESP32 dan platform Blynk agar kondisi lampu dapat diawasi secara seketika (Asry, Lutfi, & Ahmad, 2024). Di sisi lain, (Kusumo & Salam, 2023) merancang sebuah sistem kontrol dan pemantauan lampu lalu lintas yang berbasis ESP8266 memungkinkan manajemen lampu dilakukan melalui aplikasi Blynk. Penelitian lainnya menunjukkan bahwa sensor LDR dapat digunakan dalam sistem berbasis IOT untuk menyesuaikan pencahayaan secara otomatis sesuai dengan intensitas cahaya di sekitarnya (Irsyada, Haq, Rohmah, & Saputra, 2022).

Beberapa penelitian sebelumnya telah menerapkan teknologi IoT untuk monitoring maupun pengendalian lampu lalu lintas. Namun, sistem yang dikembangkan umumnya masih berfokus pada satu fungsi. Pada penelitian ini, sistem monitoring tidak sebatas dipakai guna mendeteksi gangguan lampu lalu lintas, namun pula dilengkapi dengan pengaturan intensitas cahaya menggunakan sensor LDR. Penambahan fungsi tersebut diharapkan dapat membuat sistem tidak hanya memudahkan proses monitoring, tetapi juga lebih efisien dalam penggunaan energi sesuai dengan kondisi pencahayaan di sekitar.

## 1.2 Perumusan Masalah

Berlandaskan informasi yang sudah dijabarkan sebelumnya, rumusan masalah pada kajian ini disusun selaku panduan dalam pelaksanaan penelitian. Berikut adalah rumusan masalah yang diajukan

1. Bagaimana menyusun sistem monitoring dan deteksi gangguan lampu lalu lintas berbasis *IOT* memakai ESP8266, sensor PZEM004T, dan platform Blynk?
2. Bagaimana kinerja sistem dalam mendeteksi gangguan lampu lalu lintas berdasarkan parameter kelistrikan yang diperoleh dari sensor PZEM004T?
3. Bagaimana mengintegrasikan fungsi monitoring kondisi lampu lalu lintas dengan pengaturan intensitas cahaya berbasis sensor LDR menjadi satu sistem IoT?

## 1.3 Batasan Masalah

Supaya diskusi tetap fokus serta sejalan dengan sasaran penelitian, batasan ruang lingkup studi ini ditetapkan pada beberapa hal berikut..

1. Gangguan yang dideteksi pada penelitian ini dibatasi pada kondisi yang dapat diidentifikasi melalui perubahan parameter kelistrikan.
2. Penelitian dilaksanakan dalam bentuk prototipe dan tidak diterapkan secara langsung pada sinyal lalu lintas di jalan umum.
3. Sistem diyakini berfungsi dalam situasi di mana koneksi internet tersedia dan dalam keadaan stabil.

## 1.4 Tujuan Penelitian

Studi berikut harapannya mampu memberi keuntungan untuk sejumlah stakeholder yang terlibat dalam ranah penelitian. Manfaat dari study berikut dijelaskan berikut ini:

1. Menyusun sistem monitoring serta deteksi gangguan lampu lalu lintas berbasis *IOT* memakai ESP8266, sensor PZEM004T, dan platform Blynk?
2. Mengetahui kinerja sistem untuk mendeteksi gangguan lampu lalu lintas berlandaskan parameter kelistrikan yang diperoleh dari sensor PZEM004T?

3. Mengintegrasikan fungsi monitoring kondisi lampu lalu lintas dengan pengaturan intensitas cahaya berbasis sensor LDR menjadi satu sistem IoT?

### 1.5 Manfaat Penelitian

Para penulis di bidang ini diperkirakan akan memperoleh berbagai manfaat dari penelitian ini. Maka manfaat penelitian berikut dijelaskan berikut ini:

1. Bagi Mahasiswa :

Dapat menjadi tambahan referensi bagi mahasiswa, khususnya di bidang Teknik Elektro, dalam mengembangkan studi terkait sistem monitoring berbasis *IOT* pada lampu lalu lintas maupun sistem kelistrikan lainnya.

2. Bagi Perguruan Tinggi :

Dapat menjadi salah satu referensi ilmiah yang mendukung kegiatan pembelajaran serta pengembangan penelitian di lingkungan perguruan tinggi, utamanya di bidang *IOT*, sistem monitoring, serta otomasi.

3. Bagi Instansi Terkait :

Dapat berfungsi sebagai panduan awal dalam perancangan sistem pemantauan lampu lalu lintas yang menggunakan Internet of Things, sehingga dapat mendukung proses pengawasan keadaan lampu dengan lebih cepat dan efektif.

### 1.6 Sistematika Penelitian

Lima bab yang saling terkait dalam skripsi ini memastikan bahwasannya topiknya terstruktur dengan baik dan mudah dipahami. Berikut ini adalah garis besar skripsi ini:

## BAB I PENDAHULUAN

Bab berikut menjelaskan latar belakang penelitian, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, serta sistematika penulisan. Bab ini menjadi dasar yang menjelaskan alasan serta tujuan dilaksanakannya penelitian.

## **BAB II DASAR TEORI**

Bagian berikut berisi teori-teori yang mendorong penelitian, mencakup konsep IoT, lampu lalu lintas, ESP8266, sensor PZEM004T, sensor LDR, modul AC Dimmer, platform Blynk, serta teori lain yang berkaitan dengan sistem monitoring dan deteksi gangguan. Disisi lain, selanjutnya, bab ini menyajikan gambaran umum mengenai penelitian-penelitian sebelumnya yang menjadi landasan bagi penelitian ini.

## **BAB III METODOLOGI PENELITIAN**

Bagian berikut memaparkan tahapan penelitian, diawali dari rancangan sistem, kebutuhan perangkat keras serta perangkat lunak, diagram blok sistem, diagram alir (flowchart), perancangan prototype, hingga metode pengujian yang digunakan untuk mengevaluasi kinerja sistem.

## **BAB IV HASIL DAN PEMBAHASAN**

Bagian berikut menampilkan perolehan implementasi prototype serta hasil pengujian sistem. Penjelasan dilakukan pada kinerja sistem monitoring, deteksi gangguan lampu lalu lintas berdasarkan parameter kelistrikan, pengiriman data ke platform Blynk, serta pengaturan intensitas cahaya lampu memakai sensor LDR dan modul AC Dimmer.

## **BAB V SIMPULAN**

Bab ini menyajikan ringkasan yang diambil dari temuan studi serta rekomendasi yang bisa digunakan sebagai referensi untuk pengembangan riset di masa mendatang.

## **BAB II**

### **DASAR TEORI**

#### **2.1 Dasar Teori**

##### **2.1.1 Lalu Lintas**

Istilah “lalu lintas” merujuk pada arus kendaraan, orang, ataupun barang apa pun yang bergerak di sepanjang suatu rute. Sesuai dengan UU No 22 Tahun 2009 mengenai Lalu Lintas dan Angkutan Jalan, lalu lintas dijelaskan selaku pergerakan kendaraan serta orang di area lalu lintas jalan serta fasilitas terkait. Pelaksanaan lalu lintas mempunyai tujuan guna menyediakan layanan transportasi yang aman, nyaman, teratur, lancar, serta terintegrasi, sehingga dapat mendukung aktivitas masyarakat.

Seiring meningkatnya jumlah kendaraan bermotor di Indonesia, pengelolaan lalu lintas menjadi semakin penting, terutama di dekat persimpangan dan di sepanjang ruas jalan yang padat lalu lintasnya. Pengaturan arus kendaraan pada lokasi tersebut memerlukan sistem pengendalian yang mampu mengurangi potensi konflik antar pengguna jalan sekaligus menjaga kelancaran mobilitas.

Salah satu infrastruktur yang penting untuk mendukung kelancaran transportasi adalah Alat Pemberi Isyarat Lalu Lintas (APILL). Alat ini berguna mengatur arus kendaraan dan pejalan kaki di persimpangan melalui sinyal lampu, sehingga risiko bentrokan antara arus lalu lintas dapat diminimalkan. Dalam implementasinya, APILL memerlukan perawatan secara rutin agar tetap beroperasi dengan optimal. Namun, proses penanganan kerusakan sering kali sangat bergantung pada hasil pemeriksaan serta laporan dari masyarakat, yang berisiko memperlambat penanganan jika terjadi gangguan. Karena itu, diperlukan suatu sistem yang dapat mempercepat pemantauan status APILL, sehingga informasi tentang gangguan dapat dengan cepat diterima oleh petugas (Abror & Haratama, 2024).

### 2.1.2 Lampu Lalu Lintas

Pada persimpangan jalan, kendaraan dari berbagai arah bergerak secara bersamaan sehingga diperlukan suatu sistem yang dapat mengatur giliran kendaraan untuk melintas. Salah satu perangkat yang digunakan adalah Alat Pemberi Isyarat Lalu Lintas (APILL). Berlandaskan Peraturan Menteri Perhubungan No. 49 Tahun 2014, APILL merupakan perlengkapan jalan yang memberikan isyarat kepada pengguna jalan melalui lampu berwarna merah, kuning, dan hijau guna mengatur pergerakan lalu lintas.

Keberadaan APILL mempunyai fungsi penting untuk menjaga kelancaran serta keselamatan lalu lintas, terutama pada persimpangan dengan volume kendaraan yang tinggi. Penelitian mengenai sistem pemeliharaan APILL di Kota Surabaya menunjukkan bahwa kondisi APILL perlu dipantau secara berkala agar tetap berfungsi dengan baik. Penelitian tersebut juga menjelaskan bahwa informasi mengenai kerusakan masih banyak diperoleh dari hasil inspeksi maupun laporan masyarakat, sehingga kecepatan penanganan sangat dipengaruhi oleh proses penyampaian informasi (Abror & Haratama, 2024).



**Gambar 2.1** Lampu Lalu Lintas

(Sumber: <https://www.detik.com/jatim/berita/d-7365008/kenapa-lampu-lalu-lintas-ada-3-warna-ini-sejarahny>)

Pada APILL, setiap warna lampu memiliki arti yang berbeda. Lampu merah menunjukkan bahwa kendaraan harus berhenti sebelum melewati garis henti. Lampu kuning menandakan bahwa fase lampu akan berubah sehingga pengemudi perlu lebih waspada dan mempersiapkan diri untuk berhenti atau melanjutkan

perjalanan sesuai kondisi. Sementara itu, lampu hijau memberikan izin kepada kendaraan untuk melintas apabila kondisi persimpangan aman.

### **2.1.3 Sistem Monitoring**

Perubahan dalam teknologi informasi telah mengubah metode pengawasan suatu sistem. Dulu, status suatu perangkat hanya bisa diketahui melalui observasi langsung di lokasi, tetapi sekarang proses ini bisa dilakukan dari jarak jauh dengan dukungan jaringan komunikasi. Data yang didapat juga dapat ditampilkan secara langsung agar pengguna dapat mengetahui keadaan suatu sistem tanpa perlu selalu berada di tempat tersebut. Inovasi ini menyebabkan sistem pemantauan mulai banyak digunakan di berbagai sektor, seperti industri, kesehatan, pertanian, transportasi, dan sistem kelistrikan (Kaur & Sharma, 2023).

Sistem pemantauan adalah sebuah sistem yang digunakan untuk mengumpulkan, memproses, dan menampilkan informasi mengenai kondisi suatu objek secara terus-menerus atau dalam interval tertentu. Data ini didapatkan dari sensor yang terpasang dalam objek yang diobservasi, selanjutnya dikirim ke perangkat pengolah melalui saluran komunikasi untuk kemudian ditampilkan kepada pengguna. Keberadaan sistem pemantauan memungkinkan deteksi perubahan kondisi pada suatu perangkat dengan lebih cepat, sehingga mempermudah pengawasan serta membantu dalam proses pengambilan keputusan (Witczak & Szymoniak, 2024).

Umumnya, sistem pemantauan terdiri dari beberapa komponen penting, yaitu objek yang diamati, sensor yang berfungsi untuk mengumpulkan data, perangkat yang mengolah data, jalur komunikasi, serta antarmuka pengguna sebagai sarana untuk menyampaikan informasi. Berkat integrasi yang erat antara semua komponen ini, pengguna bisa dengan cepat menerima dan melihat data yang dikumpulkan dari sensor-sensor tersebut. Dengan kemajuan IoT, sistem pemantauan kini tidak hanya dapat menunjukkan data secara langsung, tetapi juga memungkinkan pengguna untuk mendapatkan informasi dari berbagai lokasi selama terhubung ke jaringan internet. Berbagai studi mengindikasikan bahwa implementasi sistem pemantauan yang berbasis IoT dapat meningkatkan efisiensi

proses pemantauan karena informasi bisa diterima lebih cepat dibandingkan metode pemeriksaan secara manual (Atmasani, 2026).

#### **2.1.4 Gangguan Kelistrikan pada Lampu Lalu Lintas**

Lampu lalu lintas merupakan salah satu perangkat kelistrikan yang bekerja secara terus-menerus guna mengatur arus kendaraan di persimpangan jalan. Karena beroperasi hampir selama 24 jam, setiap komponen kelistrikan pada sistem tersebut memiliki kemungkinan mengalami gangguan seiring dengan lamanya waktu penggunaan maupun pengaruh dari lingkungan sekitar. Apabila gangguan tersebut tidak segera diketahui, gangguan pada fungsi lampu lalu lintas dapat menyebabkan kemacetan lalu lintas atau bahkan meningkatkan risiko terjadinya kecelakaan.

Gangguan kelistrikan pada lampu lalu lintas dapat disebabkan oleh beberapa faktor, seperti terputusnya suplai tegangan, lampu yang mengalami kerusakan, sambungan kabel yang kurang baik, maupun gangguan pada sumber listrik. Kondisi tersebut menyebabkan lampu tidak dapat bekerja sebagaimana mestinya, baik dalam bentuk lampu yang padam, menyala tidak normal, maupun tidak menyala sama sekali. Oleh karena itu, pemeriksaan terhadap kondisi kelistrikan menjadi bagian penting dalam pemeliharaan lampu lalu lintas.

Penelitian mengenai sistem pemeliharaan APILL menunjukkan bahwa kondisi lampu lalu lintas perlu dipantau secara berkala agar tetap berfungsi dengan baik (Abror & Haratama, 2024). Namun, proses identifikasi gangguan masih banyak dilakukan melalui inspeksi lapangan maupun laporan dari masyarakat sehingga waktu penanganan bergantung pada kecepatan informasi yang diterima petugas. Di sisi lain, penelitian mengenai kegagalan sistem lampu lalu lintas menjelaskan bahwa gangguan kelistrikan, seperti hilangnya tegangan suplai maupun kerusakan pada beban, dapat menyebabkan lampu tidak bekerja sebagaimana mestinya (Rosiana & Herwana, 2024). Temuan tersebut menunjukkan bahwa informasi mengenai kondisi kelistrikan jadi suatu aspek penting untuk mendorong proses pemeliharaan lampu lalu lintas.

### 2.1.5 *Internet of Things (IoT)*

Perkembangan Teknologi komunikasi dan jaringan internet telah menghadirkan transformasi besar dalam banyak aspek kehidupan, termasuk dalam cara individu memantau dan mengontrol sebuah perangkat. Sebelumnya, pemantauan harus dilakukan secara langsung di tempat, namun kini berbagai perangkat dapat saling bertukar data dengan jaringan internet, memungkinkan informasi diperoleh dengan lebih cepat serta mudah. Kemajuan ini telah mendorong munculnya berbagai sistem yang dapat beroperasi secara otomatis atau menyediakan informasi secara real-time. Dengan semakin meningkatnya permintaan akan akses informasi yang cepat, penerapan IoT juga terus berkembang serta diterapkan di sejumlah sektor, seperti industri, pertanian, kesehatan, transportasi, hingga sistem kelistrikan.



**Gambar 2.2** *Internet of Things*

(Sumber: <https://diskominfo.badungkab.go.id/artikel/18205-pentingnya-internet-of-things-iot->)

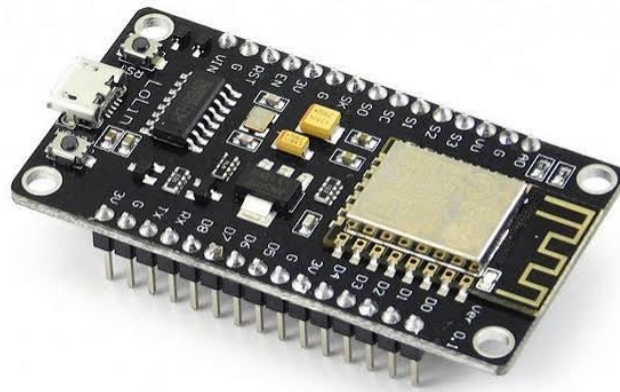
IoT adalah sebuah ide yang mengkorelasikan berbagai macam alat fisik ke jaringan internet, memungkinkan alat tersebut untuk saling bertukar informasi, mengirim data, dan menerima instruksi tanpa perlu seringnya interaksi dari penggunanya. Secara umum, suatu sistem IoT tersusun atas sensor yang bertugas untuk mengumpulkan data, mikrokontroler yang berguna memproses informasi, saluran komunikasi untuk pengiriman data, serta platform yang digunakan untuk

menyimpan dan menunjukkan hasil pengamatan kepada pengguna. Melalui penggabungan ini, data dari perangkat dapat diakses secara langsung menggunakan komputer atau smartphone selagi terhubung dengan internet (Ikha & Yuhendri, 2022).

Dalam beberapa tahun terakhir, penerapan IoT semakin mudah dijumpai pada berbagai sistem monitoring dan otomasi (Andika, Permana, & Attamimi, 2022). Hal tersebut didukung oleh semakin luasnya akses internet, berkembangnya teknologi sensor, serta hadirnya mikrokontroler yang memiliki kemampuan komunikasi data dengan biaya yang relatif terjangkau. Perkembangan tersebut membuat IoT tidak lagi terbatas pada kebutuhan industri berskala besar, tetapi juga banyak dimanfaatkan dalam penelitian maupun pengembangan berbagai sistem berbasis monitoring. Selain mampu menyajikan informasi dengan *real-time*, IoT juga memberi kemudahan pada proses penyimpanan data dan membantu pengguna dalam melakukan pengawasan terhadap suatu sistem dari berbagai lokasi.

#### **2.1.6 Mikrokontroler NodeMCU ESP8266**

Perkembangan teknologi IoT mendorong permintaan terhadap mikrokontroler yang tidak hanya mampu mengolah informasi tetapi juga dapat berinteraksi melalui internet. Situasi ini memicu peningkatan berbagai papan pengembangan dengan modul WiFi yang terintegrasi, yang semakin banyak digunakan dalam sistem pemantauan dan otomatisasi. Salah satu perangkat yang populer adalah NodeMCU ESP8266. Dengan ukuran yang kompak, alat ini juga sangat mudah untuk diprogram dan memiliki dukungan dari komunitas pengguna yang besar, sehingga sering digunakan dalam penelitian dan pengembangan prototipe (Irsyada, Haq, Rohmah, & Saputra, 2022).



**Gambar 2.3** NodeMCU ESP8266

(Sumber: <https://bisaioti.com/esp8266-modul-wi-fi-serbaguna-untuk-pengembangan-iot/>)

NodeMCU ESP8266 merupakan papan pengembang berbasis chip ESP8266EX yang dikembangkan untuk mempermudah implementasi sistem berbasis IoT. Berbeda dengan mikrokontroler konvensional yang memerlukan modul komunikasi tambahan, NodeMCU ESP8266 telah dilengkapi modul WiFi IEEE 802.11 b/g/n sampai bisa terhubung langsung ke jaringan internet. Selain itu, board ini kompatibel dengan Arduino IDE sehingga proses pemrograman menjadi lebih sederhana dan mudah dipelajari, bahkan bagi pengguna yang baru mengembangkan sistem berbasis mikrokontroler.

Secara umum, NodeMCU ESP8266 memiliki banyak antarmuka yang memungkinkannya berkomunikasi dengan perangkat lain. Ini termasuk General Purpose Input Output (GPIO), Serial Peripheral Interface (SPI), Universal Asynchronous Receiver Transmitter (UART), Pulse Width Modulation (PWM), dan Inter-Integrated Circuit (I2C). Dengan antarmuka yang tersedia, NodeMCU ESP8266 dapat dihubungkan dengan beragam macam sensor serta aktuator. Ini memungkinkannya digunakan pada berbagai aplikasi berbasis IoT. Selain itu, board ini menggunakan prosesor 32-bit Tensilica L106, yang memiliki frekuensi kerja hingga 80 MHz. Frekuensi ini dapat ditingkatkan menjadi 160 MHz jika diperlukan oleh aplikasi

**Tabel 2.1** Datasheet NodeMCU ESP8266

|                          |                 |
|--------------------------|-----------------|
| <b>Microcontroller</b>   | ESP-8266 32-bit |
| <b>Clock speed</b>       | 80 MHz          |
| <b>USB to Serial</b>     | CP2102          |
| <b>Operating Voltage</b> | 3.3 V           |
| <b>Input Voltage</b>     | 4.5V-10V        |
| <b>Flash Memory/SRAM</b> | 4 MB/ 64 KB     |
| <b>Digital I/O Pins</b>  | 11              |
| <b>Analog In Pins</b>    | 1               |
| <b>ADC Range</b>         | 0-3.3V          |
| <b>UART/SPI/I2C</b>      | 1/1/1           |
| <b>Wifi Built-In</b>     | 802.11 b/g/n    |

Sumber: <https://www.make-it.ca/nodemcu-details-specifications/>

NodeMCU ESP8266 sangat disukai karena berbagai spesifikasinya; itu juga murah, tidak menggunakan banyak daya, dan mudah mendapatkan instruksi dan pustaka pemrograman. Menurut penelitian di Indonesia, Sistem pemantauan berbasis IoT mengandalkan NodeMCU ESP8266 sebagai pengontrol utamanya karena kemampuannya untuk membaca data dari sensor dan mengirimkan data tersebut secara real-time melalui internet (Wantudi, Triayudi, & Benrahman, 2023). NodeMCU ESP8266 masih jadi salah satu mikrokontroler yang paling populer untuk dipakai pada pembuatan sistem pengawasan dan otomasi berukuran kecil hingga menengah. Ini karena banyaknya manfaatnya.

#### 2.1.7 Sensor PZEM004T

Data mengenai kondisi beban merupakan komponen penting dalam setiap sistem pemantauan kelistrikan. Parameter kelistrikan, termasuk tegangan, arus, daya, energi listrik, frekuensi, serta faktor daya, memungkinkan pengguna untuk memantau kondisi beban. Perubahan nilai pada parameter tersebut dapat menjadi indikasi awal adanya gangguan pada sistem, misalnya penurunan tegangan, hilangnya suplai listrik, atau sesuatu yang tidak berfungsi dengan baik. Maka, untuk transmisi data secara real-time ke sistem pemantauan, diperlukan sebuah sensor

yang mampu mengukur sifat-sifat listrik secara akurat. Penelitian mengenai monitoring parameter kelistrikan juga menunjukkan bahwa pengamatan terhadap besaran-besaran listrik secara *real-time* dapat membantu proses pengawasan dan pengambilan keputusan pada suatu sistem monitoring (Surya, et al., 2023).

PZEM004T adalah sensor yang banyak digunakan untuk kebutuhan ini karena merupakan modul pengukuran parameter kelistrikan AC yang dapat mengukur berbagai besaran listrik dalam satu perangkat. Parameter yang dapat diukur termasuk tegangan (V), arus (A), daya aktif (W), energi listrik (kWh), frekuensi (Hz), serta faktor daya (PF). Untuk membuat mikrokontroler dapat mengolah dan menampilkan hasil pengukuran secara real-time, komunikasi serial digunakan untuk mengirimkan data ke mikrokontroler. PZEM004T memiliki kemampuan ini, sehingga banyak digunakan dalam sistem monitoring energi listrik dan aplikasi IoT (Syahri & Bintoro, 2023).



**Gambar 2.4** Sensor PZEM004T

(Sumber: <https://www.nn-digital.com/blog/2019/07/10/mengenal-pzem-004t-modul-elektronik-untuk-alat-pengukuran-listrik/>)

PZEM004T bekerja dengan mengukur tegangan dari sumber listrik AC secara langsung. Di sisi lain, pengukuran arus dilakukan dengan menggunakan transformasi arus (CT) yang dipasang pada penghantar beban. Rangkaian internal modul kemudian memproses data hasil pengukuran untuk menghitung faktor daya, daya, energi, dan frekuensi. Kemudian, data ini dikirim ke mikrokontroler melalui antarmuka komunikasi UART, dan sistem pengawasan dapat memprosesnya lebih

lanjut. Mekanisme ini memungkinkan pencapaian berbagai parameter kelistrikan secara bersamaan tanpa menggunakan sensor terpisah untuk setiap besaran listrik.

Untuk mengetahui tingkat akurasi sensor PZEM004T, hasil pengukuran dibandingkan dengan hasil pengukuran menggunakan AVO meter sebagai alat ukur referensi. Persentase error digunakan untuk mengetahui selisih antara hasil pengukuran sensor PZEM004T dengan hasil pengukuran AVO meter.

$$Error (\%) = \frac{Nilai\ PZEM - Nilai\ AVO}{Nilai\ AVO} \times 100\% \quad (2.1)$$

Keterangan:

Error (%) = persentase kesalahan pengukuran.

Nilai PZEM = hasil pengukuran tegangan sensor PZEM004T.

Nilai AVO = hasil pengukuran menggunakan AVO meter sebagai alat ukur referensi.

**Tabel 2.2** Datasheet Sensor PZEM004T

| Parameter          | Nilai                         |
|--------------------|-------------------------------|
| <b>Tegangan</b>    | 80-260 VAC                    |
| <b>Arus</b>        | 0-100 A (dengan CT eksternal) |
| <b>Daya aktif</b>  | ~23 kW                        |
| <b>Energi</b>      | 0-9999,99 kWh                 |
| <b>Frekuensi</b>   | 45-65 Hz                      |
| <b>Faktor daya</b> | 0,00-1,00                     |
| <b>Komunikasi</b>  | UART (9600 bps)               |

Sumber: <https://elektrologi.iptek.web.id/wp-content/uploads/2023/02/PZEM-004T-V3.0-Datasheet-User-Manual.pdf>

PZEM004T memiliki beberapa keunggulan, di antaranya mampu mengukur beberapa parameter kelistrikan secara bersamaan, akurasi perangkat ini cukup baik, kompatibel dengan berbagai jenis mikrokontroler, dan mendukung koneksi serial dasar. Selain itu, penggunaan current transformer membuat proses pengukuran arus menjadi lebih aman karena tidak memerlukan pemutusan penghantar utama.

Berbagai keunggulan tersebut menjadikan PZEM004T banyak digunakan pada penelitian maupun pengembangan sistem monitoring energi listrik berbasis *IoT* (Muslihi, 2025).

### 2.1.8 Sensor *Light Dependent Resistor* (LDR)

Intensitas cahaya ialah suatu faktor lingkungan yang bisa memengaruhi kebutuhan pencahayaan pada berbagai sistem penerangan. Pada siang hari, cahaya matahari umumnya telah mampu memberikan pencahayaan yang cukup sehingga kebutuhan cahaya tambahan menjadi lebih rendah. Sebaliknya, di malam hari ataupun saat keadaan lingkungan gelap, diperlukan pencahayaan yang lebih tinggi agar objek tetap terlihat dengan jelas. Maka, dibutuhkan sensor yang bisa mendeteksi perubahan intensitas cahaya dengan otomatis sehingga sistem pencahayaan dapat menyesuaikan kondisi lingkungan. Pemanfaatan sensor cahaya pada sistem otomatis juga telah banyak diterapkan pada berbagai penelitian, terutama pada sistem penerangan berbasis *Internet of Things* dan otomasi (Firmansyah, Azami, Mubarak , & Mubaroqan, 2022).

Di antara berbagai sensor yang digunakan untuk mengukur intensitas cahaya, resistor bergantung cahaya (LDR) menonjol. Resistansi resistor bergantung cahaya (LDR) bervariasi sesuai dengan intensitas cahaya. Pada resistor yang bergantung pada cahaya (LDR), resistansi akan menurun seiring meningkatnya intensitas cahaya dan meningkat seiring menurunnya intensitas cahaya. Berkat karakteristik ini, LDR banyak digunakan sebagai sensor cahaya. Biaya implementasi yang rendah, kemudahan integrasi dengan mikrokontroler, serta desain rangkaian yang sederhana menjadi alasan di balik hal ini.



**Gambar 2.5** Sensor LDR

(Sumber: <https://siman.elektro.poltekba.ac.id/details/ldr-light-sensor-module-25>)

Prinsip kerja LDR didasarkan pada perubahan konduktivitas material semikonduktor akibat paparan cahaya. Semakin banyak cahaya yang mencapai permukaan sensor, resistansi listriknya semakin menurun karena semakin banyak elektron yang dapat mengalir melalui bahan tersebut. Sebaliknya, ketika intensitas cahaya menurun, semakin sedikit elektron yang dapat mengalir bebas, sehingga menyebabkan peningkatan resistansi. Perubahan nilai hambatan tersebut kemudian dimanfaatkan oleh mikrokontroler sebagai informasi untuk dapat membedakan antara terang dan gelap, sehingga menjadikannya pilihan yang tepat untuk sistem pengendalian pencahayaan otomatis.

LDR sangat bagus karena harganya murah, konsumsi dayanya rendah, mudah dihubungkan ke berbagai jenis mikrokontroler, dan memiliki desain yang sederhana. Selain itu, sensor ini juga mempunyai respons yang cukup baik pada perubahan intensitas cahaya sehingga masih banyak digunakan pada sistem penerangan otomatis, lampu jalan, *smart home*, maupun berbagai aplikasi berbasis *Internet of Things* (Pambudi, Supriyadi, Kurniadi, Fazryansyah, & Stefanie, 2025).

Pada penelitian ini, sensor LDR digunakan untuk membaca kondisi pencahayaan lingkungan, yang menentukan mode operasi lampu. Sensor LDR mengubah nilai resistansinya berdasarkan intensitas cahaya yang diterima.

NodeMCU ESP8266 memiliki satu pin analog yang digunakan untuk membaca nilai sensor LDR. Karena NodeMCU ESP8266 menggunakan ADC 10-bit, nilai yang dibaca berupa nilai ADC, yang berkisar antara 0 dan 1023. Nilai ADC ini kemudian dapat diubah menjadi tegangan keluaran sensor LDR dengan menggunakan persamaan berikut:

$$V_{LDR} = \frac{ADC}{1023} \times 3,3 \quad (2.2)$$

Keterangan:

$V_{LDR}$  = tegangan keluaran sensor LDR (V)

$ADC$  = nilai pembacaan ADC sensor

3,3 = tegangan sumber sensor (V)

1023 = nilai maksimum ADC 10-bit ESP8266

Setelah nilai tegangan diperoleh, resistansi LDR dapat dihitung menggunakan persamaan pembagi tegangan:

$$R_{LDR} = R_{fixed} \left( \frac{V_{Ldr}}{3,3 - V_{LDR}} \right) \quad (2.3)$$

Nilai resistansi yang diperoleh selanjutnya digunakan untuk menentukan intensitas cahaya dalam satuan lux berdasarkan karakteristik LDR. Pada perhitungan ini digunakan parameter dan . Nilai menunjukkan resistansi LDR pada kondisi pencahayaan 10 lux, sedangkan merupakan konstanta karakteristik LDR. Intensitas cahaya dihitung menggunakan persamaan:

$$Lux = \left( \frac{RL10 \times 1000 \times 10^\gamma}{R_{LDR}} \right)^{\frac{1}{\gamma}} \quad (2.4)$$

Dengan:

$Lux$  = intensitas cahaya (lux)

$RL10 = 50 \text{ k}\Omega$

$\gamma = 0,7$

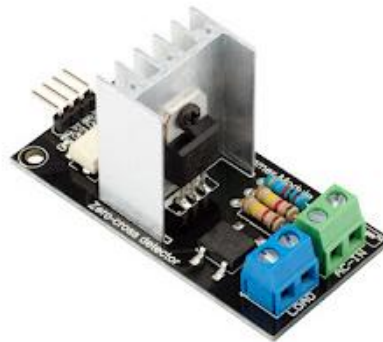
$R_{LDR}$  = resistansi LDR ( $\Omega$ )

( Sumber : <https://forum.arduino.cc/t/ldr-reading-to-lux-conversion/138411>)

### 2.1.9 AC Dimmer

Pengaturan tingkat kecerahan pada lampu AC tidak dapat dilakukan dengan cara mengubah besar tegangan sumber secara langsung. Sebaliknya, pengaturan dilakukan dengan mengendalikan jumlah daya listrik yang diterima oleh beban pada setiap siklus gelombang AC. Metode ini memungkinkan intensitas cahaya lampu berubah secara bertahap tanpa memengaruhi frekuensi maupun tegangan sumber listrik. Suatu perangkat yang banyak dipakai buat tujuan ini ialah modul AC Dimmer, terutama pada sistem pencahayaan otomatis dan berbagai aplikasi kendali berbasis mikrokontroler (Lia, Rihartanto, & Dwi, 2026).

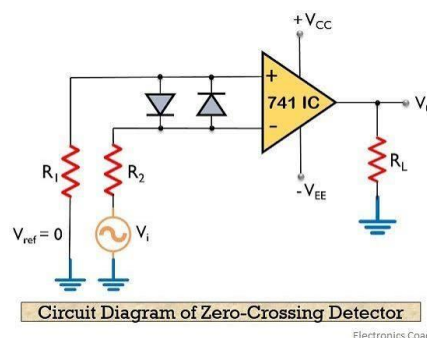
RobotDyn



**Gambar 2.6** AC Dimmer

(Sumber: <https://www.sinauprogramming.com/2020/10/kontrol-keluaran-sinyal-ac-dengan-ac.html>)

Modul AC Dimmer bekerja menggunakan metode *phase-angle control* dengan memanfaatkan TRIAC sebagai komponen utama pengendali daya. Pada metode ini, TRIAC tidak langsung menghantarkan arus ketika gelombang AC melewati titik nol (*zero crossing*), tetapi diberi jeda waktu tertentu sebelum diaktifkan. Besarnya jeda tersebut menentukan lama TRIAC menghantarkan arus dalam setiap siklus gelombang AC. Semakin kecil jeda penyalan, semakin besar daya yang diterima beban sehingga lampu akan menyala lebih terang. Sebaliknya, semakin besar jeda penyalan, daya yang diterima beban menjadi lebih kecil sehingga intensitas cahaya lampu akan berkurang.



**Gambar 2.7** Rangkaian *Zero Crossing Detector*

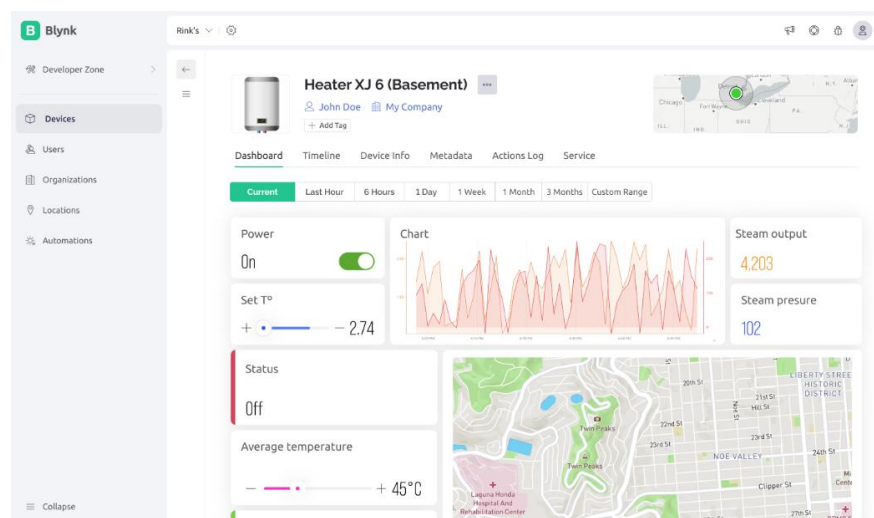
(Sumber: <https://electronicscoach.com/zero-crossing-detector.html>)

Agar proses pengaturan dapat dilakukan secara akurat, modul AC Dimmer umumnya dilengkapi dengan rangkaian *zero crossing detector*. Rangkaian berikut berfungsi mendeteksi ketika gelombang AC melewati titik nol dan mengirimkan sinyal tersebut ke mikrokontroler sebagai acuan dalam menentukan waktu

penyalan TRIAC. Selanjutnya, mikrokontroler mengirimkan sinyal pemicu sesuai nilai yang telah ditentukan sehingga tingkat kecerahan lampu dapat diatur secara bertahap sesuai kebutuhan. Dengan mekanisme tersebut, modul AC Dimmer dapat menghasilkan pengaturan pencahayaan yang lebih stabil dan halus dibandingkan metode penyaklaran konvensional.

### 2.1.10 Blynk

Pada system *IoT*, data yang diperoleh dari sensor tidak sebatas diproses oleh mikrokontroler, namun pula perlu dikirim kepada pengguna agar bisa dipantau dari lokasi yang berbeda melalui jaringan internet. Untuk mendukung proses tersebut, dibutuhkan sebuah platform yang mampu menerima data dari perangkat, menyimpannya pada server, data tersebut disajikan dengan cara yang mudah dimengerti. Salah satu platform yang banyak digunakan untuk kebutuhan tersebut adalah Blynk. Dengan memanfaatkan kemampuan komunikasi data real-time-nya, platform ini memungkinkan pemantauan status perangkat melalui dasbor web ataupun ponsel pintar.

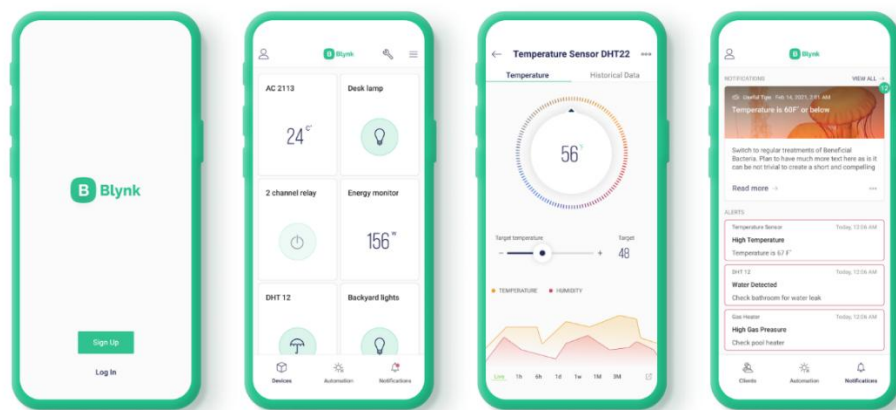


**Gambar 2.8** Tampilan Dashboard Platform Blynk

(Sumber: <https://docs.blynk.io/en>)

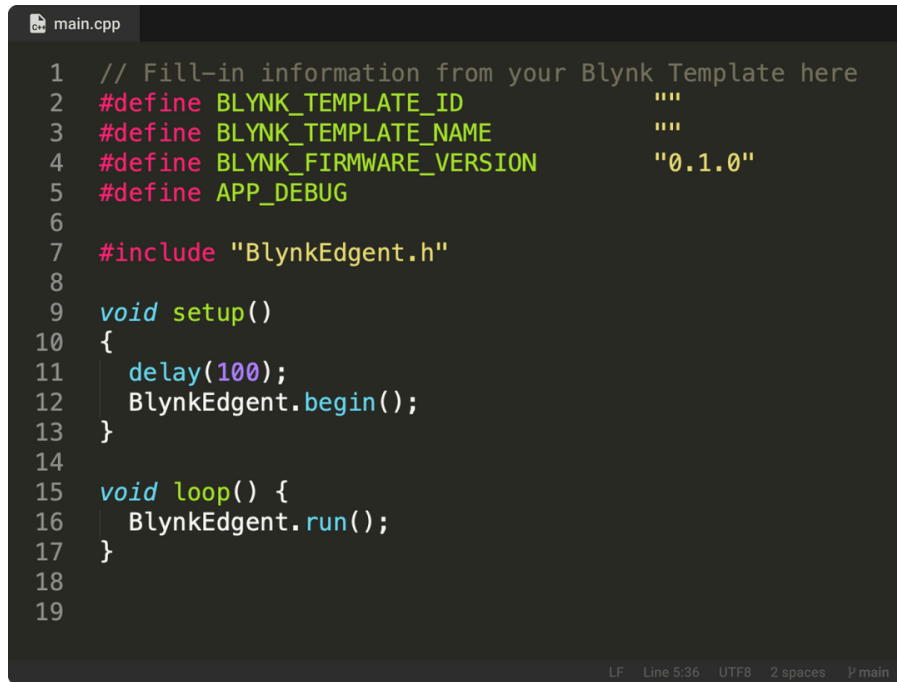
Blynk merupakan platform *Internet of Things* yang menyediakan layanan cloud, aplikasi seluler, serta dashboard berbasis web untuk membangun sistem monitoring dan kendali jarak jauh. Platform berikut mendorong beragam jenis

mikrokontroler, misalnya Arduino, NodeMCU ESP8266, ESP32, dan Raspberry Pi. Selain menyajikan data sensor dengan real-time, Blynk juga menyediakan berbagai komponen antarmuka (*widget*) yang dapat disesuaikan dengan kebutuhan pengguna sehingga proses pembuatan sistem monitoring menjadi lebih mudah tanpa harus mengembangkan aplikasi secara mandiri.



**Gambar 2.9** Tampilan Platform Blynk pada Handphone  
(Sumber: <https://docs.blynk.io/en>)

Selama proses komunikasi data, NodeMCU ESP8266 mengirimkan data yang dibaca sensor ke server Blynk melalui jaringan Wi-Fi. Kemudian, data berikut dikirim ke aplikasi Blynk, yang memungkinkan pengguna melalui smartphone guna melihat kondisi perangkat secara langsung. Selain melakukan fungsi pemantauan, Blynk memungkinkan pengiriman notifikasi, penyimpanan data, dan komunikasi dua arah. Dengan demikian, pengguna dapat mengirimkan perintah ke mikrokontroler dari aplikasi apabila diperlukan. Karena itulah, Blynk terintegrasi ke dalam berbagai macam sistem pemantauan dan otomatisasi IoT.



```

1 // Fill-in information from your Blynk Template here
2 #define BLYNK_TEMPLATE_ID          ""
3 #define BLYNK_TEMPLATE_NAME        ""
4 #define BLYNK_FIRMWARE_VERSION     "0.1.0"
5 #define APP_DEBUG
6
7 #include "BlynkEdgent.h"
8
9 void setup()
10 {
11     delay(100);
12     BlynkEdgent.begin();
13 }
14
15 void loop() {
16     BlynkEdgent.run();
17 }
18
19

```

**Gambar 2.10** Pemrograman untuk terhubung ke Blynk

(Sumber: <https://docs.blynk.io/en>)

Kemudahan konfigurasi, dukungan layanan cloud, serta kompatibilitas dengan berbagai jenis mikrokontroler menjadikan Blynk sebagai suatu platform yang banyak dimanfaatkan untuk pengembangan prototipe IoT. Dengan antarmuka yang sederhana, pengguna bisa memantau parameter sistem secara real-time tanpa harus ada di lokasi perangkat, sehingga Blynk sesuai digunakan pada berbagai aplikasi monitoring maupun kendali jarak jauh (Herlina, Syahbana, Gunawan, & Rizqi, 2022).

## 2.2 Penelitian yang Relevan

Perkembangan teknologi IoT sudah mendorong sejumlah penelitian mengenai sistem monitoring, termasuk pada bidang lampu lalu lintas dan monitoring kelistrikan. Beberapa penelitian terdahulu digunakan sebagai referensi untuk mengetahui metode yang telah diterapkan, hasil yang diperoleh, serta kekurangan yang masih dapat dikembangkan pada penelitian ini.

Sebuah sistem berbasis ESP32 dan platform Blynk untuk memantau gangguan pada lampu lalu lintas telah dikembangkan, yang memungkinkan pemantauan kondisi

lampu dengan real-time (Rosiana & Herwana, 2024). Berdasarkan hasil pengujian, secara keseluruhan, sistem ini dapat melaporkan kondisi lampu sesuai rencana. Di sisi lain, parameter listrik belum dimanfaatkan sebagai dasar untuk mendeteksi masalah atau mengatur intensitas cahaya, karena penelitian ini masih berfokus pada fungsi pemantauan.

Pengendalian dan *monitoring traffic light* juga telah diterapkan melalui jaringan Internet menggunakan ESP8266 dan platform Blynk (Abdillah & Kopravi, 2024). Sistem tersebut memungkinkan proses pemantauan serta pengendalian lampu dilakukan dari jarak jauh. Namun, penelitian tersebut belum memanfaatkan sensor untuk mendeteksi gangguan berdasarkan kondisi kelistrikan lampu lalu lintas.

Pengaturan intensitas cahaya dengan otomatis berlandaskan keadaan lingkungan telah diterapkan menggunakan sensor LDR pada sistem pencahayaan berbasis *Internet of Things* (Irsyada, Haq, Rohmah, & Saputra, 2022). Sensor LDR dimanfaatkan untuk mendeteksi perubahan intensitas cahaya sehingga tingkat kecerahan lampu dapat menyesuaikan kondisi siang maupun malam. Akan tetapi, namun, hingga saat ini, teknologi ini baru digunakan untuk penerangan umum dan belum diterapkan pada lampu lalu lintas.

Monitoring parameter kelistrikan pada lampu menggunakan sensor PZEM004T telah diterapkan pada sistem berbasis ESP8266 dan platform Blynk (Safira, Elsi, & Haryanto, 2025). Agar lebih mudah memantau kondisi listrik lampu-lampu tersebut, sistem yang dirancang ini dapat menampilkan daya, energi, arus, serta tegangan di ponsel pintar secara real time. Namun, penelitian tersebut belum mengimplementasikan sistem pada lampu lalu lintas maupun mengintegrasikan fungsi deteksi gangguan dan pengaturan intensitas cahaya.

Berdasarkan beberapa penelitian terdahulu, bisa diketahui bahwasannya teknologi *IoT* sudah banyak diaplikasikan pada sistem monitoring lampu ataupun sistem kelistrikan. Namun, penelitian yang telah dilakukan umumnya masih berfokus pada fungsi monitoring, pengendalian, atau pengaturan intensitas cahaya secara terpisah. Selain itu, pemanfaatan parameter kelistrikan sebagai dasar deteksi gangguan pada lampu lalu lintas yang dipadukan dengan pengaturan intensitas cahaya masih belum banyak dikembangkan. Maka, studi ini mengintegrasikan

fungsi monitoring kondisi kelistrikan, deteksi gangguan, serta pengaturan intensitas cahaya lampu lalu lintas dalam satu sistem berbasis *Internet of Things*.

### **2.3 Kerangka Teori**

Berdasarkan prinsip yang dijelaskan pada bab sebelumnya, sistem yang dikembangkan ini terdiri dari beberapa komponen yang saling terkait. Parameter listrik, termasuk daya, energi, tegangan, arus, frekuensi, dan faktor daya, dapat diukur menggunakan sensor PZEM004T. Data tersebut kemudian diproses oleh NodeMCU ESP8266, yang berfungsi sebagai unit pemrosesan pusat sistem.

Selain membaca parameter kelistrikan, sistem juga memanfaatkan sensor LDR guna mendeteksi intensitas cahaya di lingkungan sekitar. Informasi tersebut digunakan sebagai dasar pengaturan tingkat kecerahan lampu melalui modul AC Dimmer sehingga intensitas cahaya lampu dapat menyesuaikan kondisi siang maupun malam.

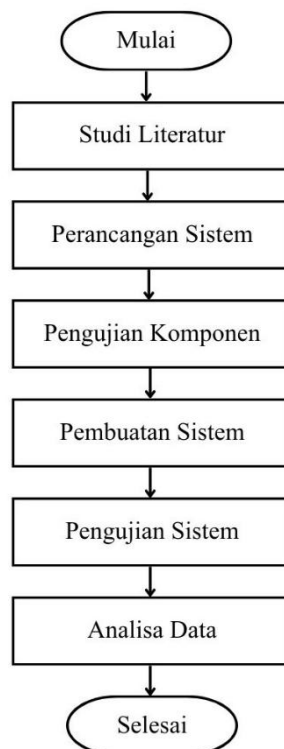
Setelah hasil pengukuran ditransmisikan ke platform Blynk dengan jaringan internet, pengguna bisa menggunakan smartphone untuk memantau keadaan lampu lalu lintas dengan real-time. Dengan integrasi semua komponen ini, diharapkan sistem pemantauan akan bisa memberi informasi terkait keadaan kelistrikan lampu lalu lintas serta mengubah intensitas pencahayaan berdasarkan kondisi lingkungan.

## BAB III

### METODOLOGI

#### 3.1 Alur Perancangan

Alur perancangan adalah tahapan penelitian yang disusun secara sistematis untuk membantu dalam proses perancangan, pembuatan, dan pengujian sistem. Tujuan dari menyusun alur perancangan adalah agar setiap tahapan dapat dilakukan secara terstruktur sehingga hasil yang diperoleh sesuai dengan tujuan penelitian. Gambar 3.1 menunjukkan diagram alur perancangan penelitian.



**Gambar 3.1** Diagram Alur Perencanaan

Berikut ini merupakan penjelasan dari diagram alur perencanaan:

1. Studi Literatur

Untuk melakukan penelitian ini, jurnal akademik, buku, standar, dan datasheet komponen dikumpulkan. Referensi ini digunakan untuk mendapatkan dasar teori tentang sistem pengawasan, Internet of Things (IoT), NodeMCU ESP8266, sensor PZEM004T, sensor LDR, platform

Blynk, AC Dimmer, dan standar lampu lalu lintas di Indonesia. Hasil penelitian menjadi dasar untuk menentukan konsep dan desain sistem.

## 2. Perencanaan Sistem

Tahap perancangan sistem dilakukan dengan menyusun konsep kerja sistem berdasarkan hasil studi literatur. Pada tahap ini ditentukan arsitektur sistem, pemilihan komponen yang digunakan, hubungan antarperangkat, serta alur komunikasi data dari sensor hingga aplikasi monitoring. Selain itu, dilakukan pula penyusunan diagram alur perencanaan, flowchart program, dan rancangan perangkat keras sebagai acuan dalam proses implementasi sistem.

## 3. Pengujian Komponen

Sebelum sistem dirakit secara keseluruhan, setiap komponen terlebih dahulu diuji untuk memastikan dapat berfungsi dengan baik sesuai spesifikasinya. Pengujian dilakukan terhadap NodeMCU ESP8266, sensor PZEM004T, sensor LDR, AC Dimmer, relay, serta komunikasi dengan platform Blynk. Tahap ini bertujuan untuk meminimalkan kesalahan pada saat proses integrasi sehingga setiap komponen yang digunakan telah dipastikan bekerja dengan baik.

## 4. Pembuatan Sistem

Tahap pembuatan sistem dilakukan dengan mengintegrasikan seluruh komponen perangkat keras dan perangkat lunak sesuai dengan rancangan yang telah dibuat. Proses ini meliputi perakitan rangkaian, pemasangan setiap komponen, serta pemrograman NodeMCU ESP8266 menggunakan Arduino IDE. Setelah seluruh komponen terhubung, sistem siap untuk dilakukan pengujian secara keseluruhan.

## 5. Pengujian Sistem

Pengujian sistem dilakukan untuk mengetahui kinerja sistem secara keseluruhan setelah seluruh komponen berhasil diintegrasikan. Pengujian dilakukan dengan menjalankan sistem sesuai kondisi operasi yang dirancang untuk memastikan setiap fungsi dapat bekerja secara terpadu, mulai dari proses pembacaan data, pengolahan data, pengaturan intensitas

lampu, hingga monitoring melalui aplikasi Blynk. Hasil pengujian ini menjadi dasar dalam mengevaluasi tingkat keberhasilan sistem yang telah dikembangkan.

#### 6. Analisa Data

Tahap terakhir adalah menganalisis data yang diperoleh selama proses pengujian sistem. Analisis dilakukan terhadap hasil pembacaan parameter kelistrikan, respons sensor LDR terhadap perubahan intensitas cahaya, kinerja AC Dimmer dalam mengatur tingkat kecerahan lampu, serta keberhasilan proses monitoring melalui aplikasi Blynk. Hasil analisis tersebut selanjutnya digunakan sebagai dasar dalam pembahasan hasil penelitian dan penarikan kesimpulan.

### 3.2 Identifikasi Kebutuhan

Sebelum proses perancangan dan pembuatan sistem dilakukan, terlebih dahulu perlu diidentifikasi berbagai kebutuhan yang digunakan selama penelitian. Identifikasi ini bertujuan untuk memastikan seluruh perangkat yang dibutuhkan telah tersedia sehingga proses perancangan, implementasi, dan pengujian sistem dapat berjalan dengan baik. Kebutuhan dalam penelitian ini terdiri atas perangkat keras (*hardware*) dan perangkat lunak (*software*) yang masing-masing memiliki fungsi berbeda dalam mendukung pengembangan sistem.

#### 3.2.1 *Hardware* (Perangkat Keras)

Perangkat keras adalah komponen penting dalam membangun sistem monitoring. Ini karena seluruh proses pembacaan, pengolahan, dan pengiriman data dilakukan oleh bagian yang saling terhubung. Akibatnya, perangkat keras yang dipilih disesuaikan dengan kebutuhan sistem yang akan dibangun. Tabel 3.1 menunjukkan perangkat keras yang digunakan dalam penelitian ini.

**Tabel 3.1** Alat dan Material (Perangkat Keras)

| <b>No</b> | <b>Nama<br/>Komponen</b> | <b>Jumlah</b> | <b>Fungsi</b>                                                                                                                |
|-----------|--------------------------|---------------|------------------------------------------------------------------------------------------------------------------------------|
| <b>1</b>  | NodeMCU<br>ESP8266       | 1 buah        | Sebagai mikrokontroler yang mengendalikan seluruh proses kerja sistem.                                                       |
| <b>2</b>  | Sensor<br>PZEM004T<br>V3 | 3 buah        | Menghitung semua parameter kelistrikan lampu lalu lintas, termasuk tegangan, arus, daya, energi, frekuensi, dan faktor daya. |
| <b>3</b>  | Sensor LDR               | 1 buah        | Mendeteksi intensitas cahaya di lingkungan sebagai dasar pengaturan tingkat kecerahan lampu.                                 |
| <b>4</b>  | AC Dimmer                | 1 buah        | Mengatur tingkat kecerahan lampu AC berdasarkan hasil pembacaan sensor LDR.                                                  |
| <b>5</b>  | Lampu AC<br>220V         | 3 buah        | Sebagai prototipe lampu lalu lintas yang terdiri dari lampu merah, kuning, dan hijau.                                        |
| <b>6</b>  | Power Supply             | 1 buah        | Menyuplai daya untuk mengoperasikan seluruh rangkaian sistem.                                                                |
| <b>7</b>  | LCD 16x2                 | 1 buah        | Menampilkan informasi hasil pembacaan parameter kelistrikan secara langsung pada alat.                                       |
| <b>8</b>  | Breadboard<br>mini       | 1 buah        | Sebagai media perakitan sementara untuk menghubungkan komponen elektronik.                                                   |
| <b>9</b>  | Kabel Jumper             | Secukupnya    | Menghubungkan antar komponen dalam rangkaian.                                                                                |

Perangkat keras tersebut saling terhubung sehingga membentuk satu sistem monitoring yang mampu membaca parameter kelistrikan pada setiap lampu lalu lintas, mengatur intensitas cahaya sesuai kondisi lingkungan, serta menampilkan hasil monitoring baik melalui LCD maupun aplikasi Blynk.

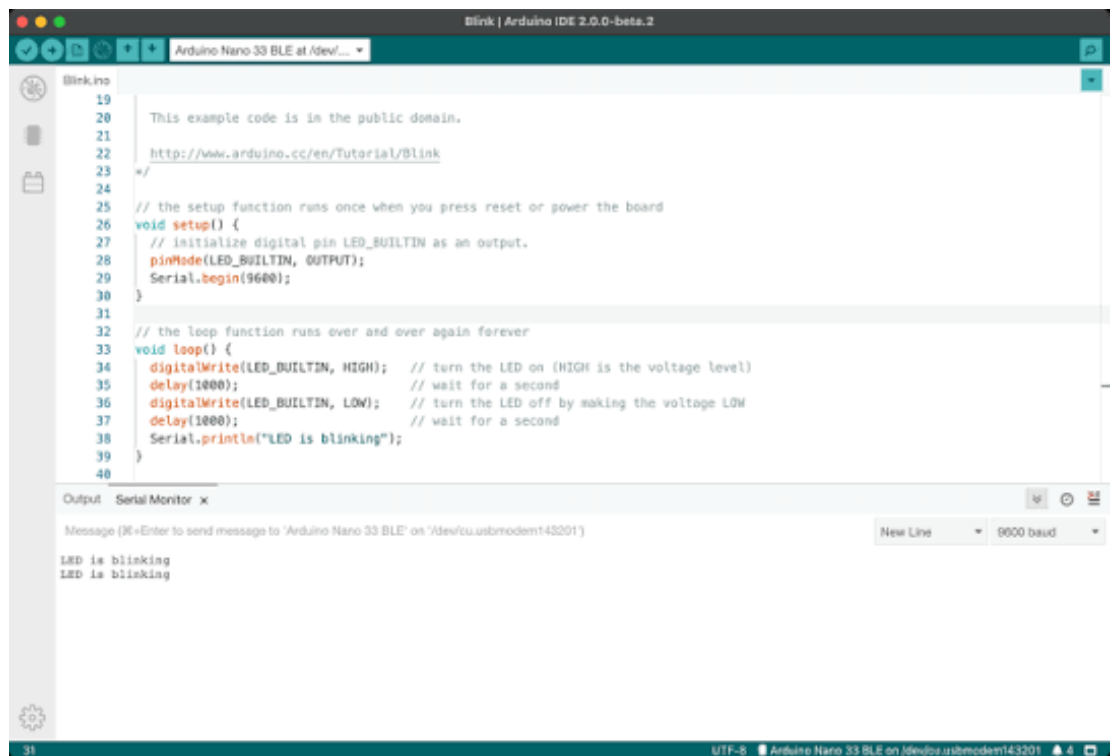
### **3.2.2 Software (Perangkat Lunak)**

Perangkat lunak yang digunakan pada penelitian ini berperan dalam proses pemrograman mikrokontroler serta monitoring sistem yang telah dirancang. Pemilihan perangkat lunak disesuaikan dengan kebutuhan sistem sehingga proses pengembangan, pengujian, dan pemantauan dapat dilakukan dengan lebih mudah. Adapun perangkat lunak yang digunakan pada penelitian ini terdiri atas Arduino IDE dan Blynk.

#### **1. Arduino IDE**

Arduino IDE sebagai perangkat lunak utama untuk menyusun program pada NodeMCU ESP8266. Software ini dipilih karena mampu mendukung proses pengembangan sistem secara lengkap, mulai dari penulisan program, pengelolaan library, proses kompilasi, hingga pengunggahan program ke mikrokontroler. Selain itu, Arduino IDE juga mendukung berbagai board mikrokontroler dan memiliki komunitas pengguna yang luas sehingga memudahkan proses pengembangan maupun penyelesaian kendala yang ditemukan selama pembuatan sistem monitoring dan deteksi gangguan lampu lalu lintas.

Seluruh logika kerja sistem dirancang dan ditulis menggunakan Arduino IDE. Program yang dibuat mencakup proses pembacaan data dari tiga sensor PZEM004T, pembacaan nilai sensor LDR, pengaturan tingkat kecerahan lampu melalui AC Dimmer, komunikasi dengan LCD 16×2, serta pengiriman data menuju aplikasi Blynk melalui jaringan Wi-Fi. Seluruh proses tersebut dijalankan secara terintegrasi sehingga NodeMCU ESP8266 dapat mengendalikan sistem sesuai dengan rancangan yang telah dibuat.



**Gambar 3.2** Tampilan Layar Arduino IDE

(Sumber: <https://www.arduino.cc/pro/software-arduino-pro-ide/>)

## 2. Blynk

Setelah data dari setiap sensor berhasil diproses oleh NodeMCU ESP8266, informasi tersebut perlu ditampilkan agar kondisi sistem dapat dipantau oleh pengguna tanpa harus berada di lokasi alat. Untuk memenuhi kebutuhan tersebut, penelitian ini memanfaatkan aplikasi Blynk sebagai media monitoring yang dijalankan pada smartphone Android. Penggunaan aplikasi ini membuat hasil pembacaan sistem dapat diakses selama NodeMCU ESP8266 dan *smartphone* sama-sama terhubung ke jaringan internet.

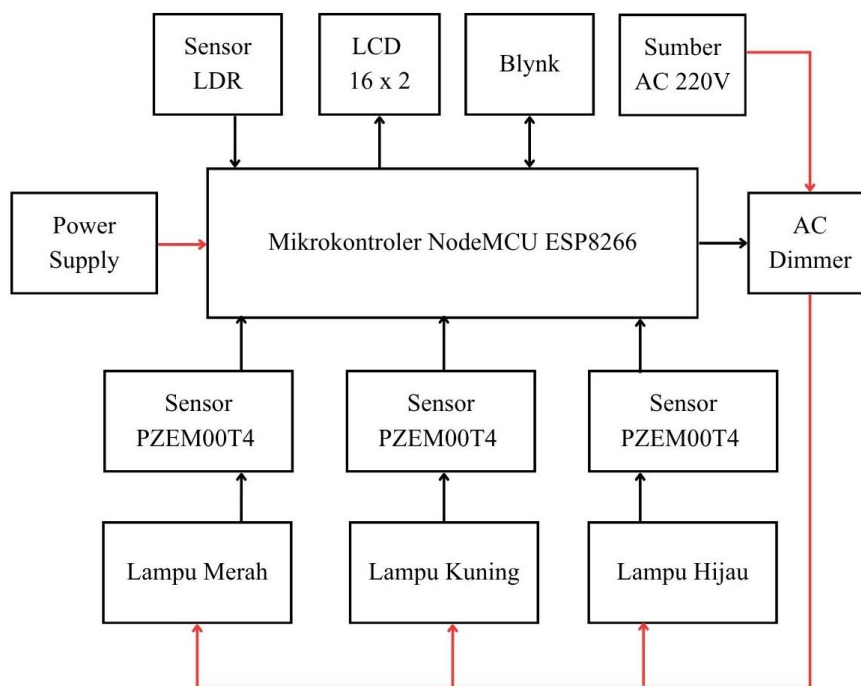
Tampilan pada aplikasi Blynk dirancang agar seluruh parameter penting dapat dipantau dengan mudah. Data yang ditampilkan meliputi tegangan, arus, daya, energi listrik, frekuensi, faktor daya dari masing-masing sensor PZEM004T, serta informasi kondisi sistem sesuai dengan rancangan antarmuka yang dibuat. Setiap perubahan data yang dibaca oleh sensor akan dikirim secara

otomatis oleh NodeMCU ESP8266 ke server Blynk, kemudian diperbarui pada aplikasi sehingga pengguna memperoleh informasi secara real-time.

### 3.3 Metode Perancangan

#### 3.3.1 Gambaran Umum Sistem

Pada penelitian ini, sistem monitoring dan deteksi gangguan lampu lalu lintas berbasis Internet of Things dirancang dengan menggabungkan berbagai bagian yang saling terhubung. Sistem memiliki kemampuan untuk memantau kondisi kelistrikan pada setiap lampu lalu lintas, mengidentifikasi perubahan intensitas cahaya di lingkungan sekitar, dan menampilkan hasil pemantauan secara real-time pada smartphone melalui aplikasi Blynk. Gambar 3.2 menunjukkan gambar sistem yang dirancang secara keseluruhan, dengan jalur data ditunjukkan dengan garis hitam dan jalur daya ditunjukkan dengan garis merah.



**Gambar 3.3** Blok Diagram Sistem

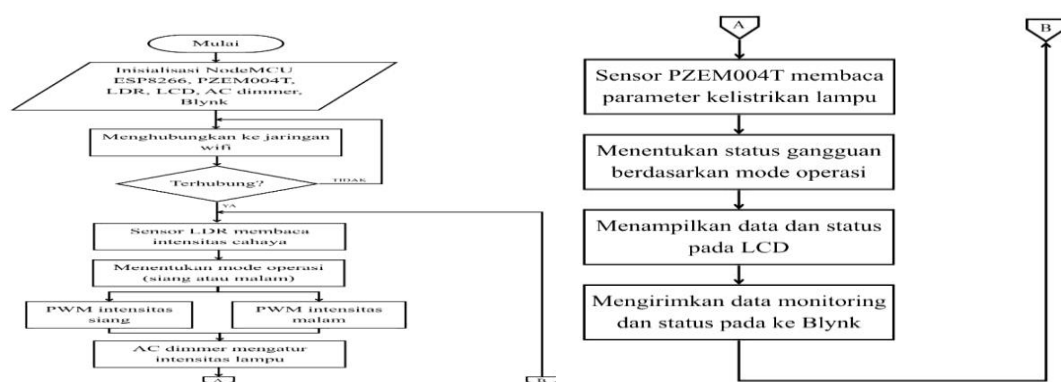
NodeMCU ESP8266 biasanya berfungsi sebagai pusat pengendali yang menghubungkan semua komponen sistem. Ada tiga sensor PZEM004T yang digunakan untuk membaca parameter kelistrikan lampu lalu lintas: lampu merah,

lampu kuning, dan lampu hijau. NodeMCU ESP8266 kemudian menerima dan mengolah data pengukuran. Sebaliknya, sensor LDR digunakan untuk mengukur intensitas cahaya di lingkungan sekitar dan digunakan sebagai referensi untuk mengubah tingkat kecerahan lampu melalui modul AC Dimmer. Hasil pembacaan sensor ditampilkan pada LCD 16×2, sehingga perangkat dapat secara langsung mengetahui kondisi sistem.

NodeMCU ESP8266 mengirimkan semua data yang dibaca sensor ke platform Blynk melalui jaringan Wi-Fi, yang ditampilkan pada LCD. Pengguna dapat melihat nilai tegangan, arus, daya, energi listrik, frekuensi, faktor daya, dan kondisi lampu lalu lintas secara real-time melalui aplikasi Blynk di smartphone mereka. Sehingga proses pemantauan tidak perlu dilakukan secara langsung di lokasi pemasangan alat, pengguna dapat segera mendapatkan informasi jika sistem mendeteksi gangguan atau parameter kelistrikan yang berada di luar batas yang telah ditentukan.

### 3.3.2 Flowchart Program

Flowchart program menggambarkan urutan proses yang dijalankan oleh NodeMCU ESP8266 pada sistem monitoring dan deteksi gangguan lampu lalu lintas. Diagram alir ini menunjukkan hubungan antarproses mulai dari inisialisasi perangkat, pengaturan intensitas lampu, pembacaan parameter kelistrikan, hingga pengiriman hasil monitoring ke aplikasi Blynk. Flowchart program yang digunakan pada penelitian ini ditunjukkan pada Gambar 3.4.



Gambar 3.4 Flowchart Pemrograman

Setelah sistem diaktifkan, NodeMCU ESP8266 menginisialisasi semua perangkat yang digunakan, termasuk sensor PZEM004T, sensor LDR, LCD 16×2, AC Dimmer, dan platform Blynk. Setelah ini selesai, sistem akan menghubungkan NodeMCU ke jaringan Wi-Fi. Jika koneksi belum berhasil, NodeMCU akan melakukan percobaan hingga berhasil terhubung, memungkinkan proses monitoring secara real-time.

Setelah koneksi berhasil, sensor LDR membaca intensitas cahaya lingkungan untuk menentukan mode operasi lampu, yaitu mode siang atau mode malam. Mode operasi ini menjadi dasar dalam menentukan nilai PWM yang diberikan kepada AC Dimmer untuk mengatur tingkat kecerahan lampu. Proses ini dilakukan terlebih dahulu karena perubahan intensitas lampu akan memengaruhi parameter kelistrikan yang nantinya dibaca oleh sensor PZEM004T.

Setelah intensitas lampu sesuai dengan mode operasinya, tiga sensor PZEM004T membaca parameter kelistrikan pada masing-masing lampu lalu lintas. Data hasil pembacaan kemudian dianalisis berdasarkan mode operasi yang sedang digunakan untuk menentukan status kondisi lampu. Selanjutnya, nilai parameter beserta status gangguan ditampilkan pada LCD sebagai media monitoring lokal dan dikirimkan ke aplikasi Blynk sehingga pengguna dapat memantau kondisi lampu lalu lintas secara *real-time*. Setelah seluruh proses selesai, sistem kembali mengulangi pembacaan sensor LDR sehingga monitoring dan deteksi gangguan berlangsung secara terus-menerus.

### 3.3.3 Cara Kerja Sistem

Cara kerja sistem monitoring dan deteksi gangguan lampu lalu lintas dimulai ketika NodeMCU ESP8266 berhasil melakukan inisialisasi seluruh perangkat dan terhubung ke jaringan Wi-Fi. Setelah sistem aktif, proses monitoring dilakukan secara terus-menerus (*loop*) sehingga setiap perubahan kondisi lampu dapat dipantau secara real-time melalui LCD maupun aplikasi Blynk.

Tahap pertama dalam proses monitoring adalah pembacaan intensitas cahaya menggunakan sensor LDR. Data dari sensor ini digunakan untuk menentukan mode operasi lampu, yaitu mode siang atau mode malam. Berdasarkan

mode operasi tersebut, NodeMCU menghasilkan sinyal PWM yang dikirimkan ke AC Dimmer untuk mengatur tingkat kecerahan lampu. Urutan ini dipilih karena perubahan intensitas lampu akan memengaruhi nilai parameter kelistrikan yang akan dibaca pada tahap berikutnya.

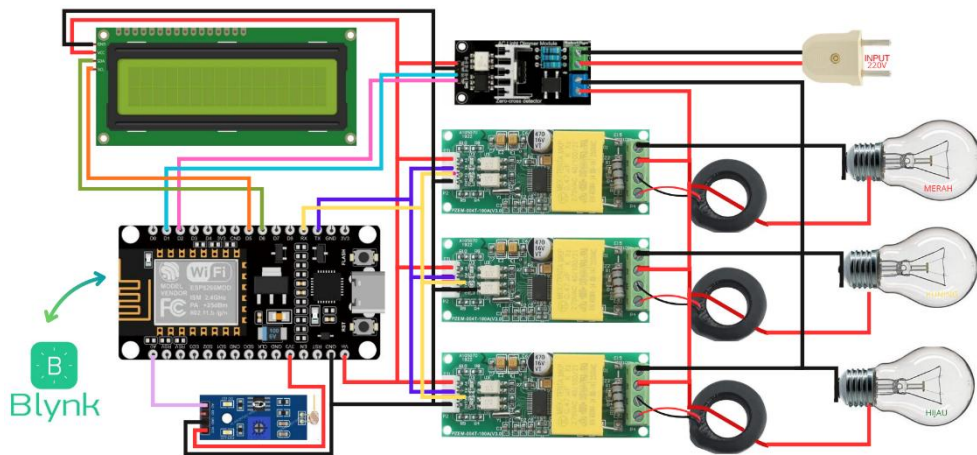
Setelah intensitas lampu sesuai dengan mode operasinya, tiga sensor PZEM004T membaca parameter kelistrikan masing-masing lampu lalu lintas. Parameter ini mencakup tegangan, arus, daya, energi, frekuensi, dan faktor daya. NodeMCU memproses semua data ini dengan membandingkannya dengan rentang parameter yang telah ditentukan untuk setiap mode operasi. Dengan cara ini, sistem dapat mempertahankan perbedaan konsumsi daya antara mode siang dan mode malam sambil membedakan kondisi normal dan kondisi gangguan. Hasil analisis selanjutnya ditampilkan pada LCD sebagai media monitoring lokal dan dikirimkan ke aplikasi Blynk dengan menggunakan jaringan internet. Kedua nilai parameter kelistrikan dan status kondisi lampu ditampilkan, sehingga pengguna dapat melihat secara langsung apabila terjadi gangguan pada salah satu lampu lalu lintas. Selama sistem berfungsi, semua proses berulang.

## BAB IV

### HASIL DAN PEMBAHASAN

#### 4.1 Perancangan

Wiring diagram merupakan rancangan hubungan antar komponen yang digunakan sebagai acuan dalam proses perakitan sistem monitoring dan deteksi gangguan lampu lalu lintas. Pada tahap ini, setiap komponen dihubungkan sesuai dengan fungsi dan kebutuhan komunikasi agar seluruh sistem dapat bekerja sebagai satu kesatuan. Wiring diagram yang digunakan pada penelitian ini ditunjukkan pada Gambar 4.1.



**Gambar 4.1** *Wiring Diagram Sistem Monitoring Traffic Light*

Berdasarkan Gambar 4.1, NodeMCU ESP8266 menjadi pusat pengendali yang menghubungkan seluruh komponen pada sistem. Seluruh data yang berasal dari sensor akan diproses oleh NodeMCU, kemudian hasilnya digunakan untuk mengatur intensitas lampu serta menampilkan informasi hasil monitoring. Oleh karena itu, hampir seluruh komponen pada sistem terhubung secara langsung dengan NodeMCU sebagai pusat pengolahan data.

Sensor LDR dihubungkan ke NodeMCU untuk membaca intensitas cahaya di sekitar lampu lalu lintas. Data yang diperoleh tidak langsung digunakan untuk proses monitoring, tetapi terlebih dahulu digunakan untuk menentukan mode operasi lampu, yaitu mode siang atau mode malam. Setelah mode operasi ditentukan, NodeMCU mengirimkan sinyal PWM ke modul AC Dimmer sehingga

intensitas cahaya lampu dapat disesuaikan dengan kondisi lingkungan. Urutan ini dipilih karena perubahan tingkat kecerahan lampu akan memengaruhi besarnya arus dan daya yang nantinya dibaca oleh sensor PZEM004T.

Proses monitoring dilakukan menggunakan tiga buah sensor PZEM004T V3 yang masing-masing dipasang pada lampu merah, lampu kuning, dan lampu hijau. Penggunaan satu sensor pada setiap lampu bertujuan agar kondisi masing-masing lampu dapat dipantau secara terpisah. Dengan cara tersebut, apabila salah satu lampu mengalami gangguan, sistem dapat langsung mengetahui lampu yang bermasalah tanpa memengaruhi proses monitoring pada dua lampu lainnya.

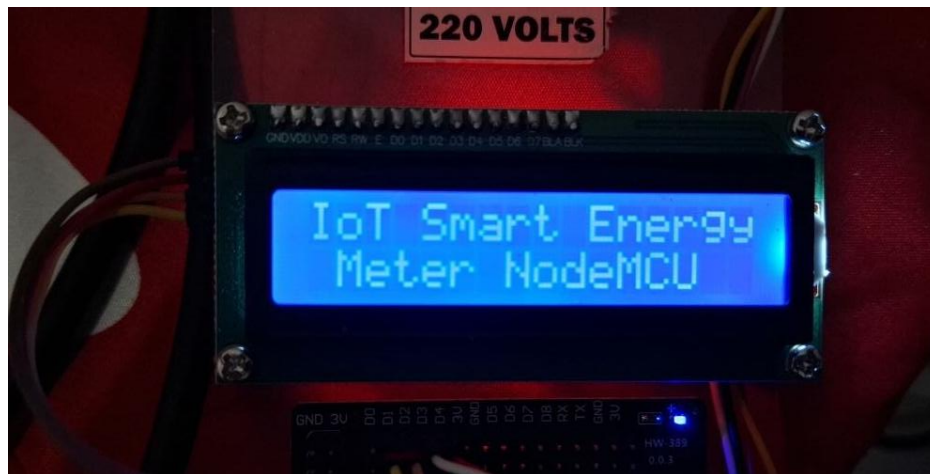
NodeMCU kemudian memproses semua data yang dibaca sensor untuk menentukan kondisi setiap lampu menggunakan parameter yang telah ditetapkan. Selanjutnya, data ditampilkan pada LCD 16×2 sebagai media pemantauan lokal dan dikirim ke aplikasi Blynk melalui jaringan Wi-Fi. Dengan demikian, smartphone dapat memantau kondisi lampu lalu lintas secara real-time.

## **4.2 Hasil Tes**

Setelah proses perancangan sistem selesai, tahap berikutnya adalah menguji kinerja setiap bagian dan sistem secara keseluruhan. Pengujian dilakukan secara bertahap, dimulai dengan pengujian LCD, sensor LDR, AC Dimmer, sensor PZEM004T, sistem monitoring, dan platform Blynk. Tujuan dari pengujian ini adalah untuk mengetahui apakah seluruh komponen telah bekerja sesuai dengan rencana dan mampu saling terintegrasi untuk menjalankan fungsi pemantauan dan deteksi gangguan lampu lalu lintas.

### **4.2.1 Pengujian LCD 16x2 I2C**

Pengujian LCD 16×2 I2C dilakukan untuk mengetahui apakah modul LCD dapat berkomunikasi dengan NodeMCU ESP8266 serta mampu menampilkan informasi yang dikirimkan oleh sistem. Selain itu, pengujian ini juga bertujuan untuk memastikan bahwa LCD dapat digunakan sebagai media monitoring lokal sehingga pengguna dapat melihat kondisi sistem secara langsung tanpa harus membuka aplikasi Blynk.



**Gambar 4.2** Hasil Pengujian LCD 16x2 I2C

Hasil pengujian yang ditunjukkan pada Gambar 4.2 menunjukkan bahwa LCD 16×2 I2C berhasil menampilkan tampilan awal sistem berupa tulisan "IoT Smart Energy Meter NodeMCU". Ini menunjukkan bahwa proses komunikasi antara NodeMCU ESP8266 dan LCD telah berjalan dengan baik melalui antarmuka I2C. Selama pengujian, tampilan LCD juga dapat diperbarui sesuai dengan data yang dikirimkan oleh NodeMCU, sehingga modul LCD dapat digunakan sebagai media pemantauan lokal untuk menampilkan.

#### **4.2.2 Pengujian Sensor LDR**

Pengujian sensor LDR dilakukan untuk mengetahui kemampuan sensor untuk membaca perubahan intensitas cahaya di lingkungan sekitar. Hasil pembacaan sensor digunakan untuk menentukan mode operasi sistem, yaitu siang atau malam, yang akan memengaruhi pengaturan intensitas cahaya melalui AC Dimmer. Hasil pengujian sensor LDR ditunjukkan pada Gambar 4.3 dan Tabel 4.1.



(a) Siang



(b) Malam

**Gambar 4.3** Hasil Pengujian Sensor LDR

Berdasarkan hasil pengujian pada Tabel 4.1, nilai ADC yang dihasilkan sensor LDR berubah mengikuti intensitas cahaya yang diterima. Pada pukul 12.00, nilai ADC berada pada angka 68 dengan hasil konversi sekitar 4343,17 lux yang menunjukkan kondisi lingkungan paling terang selama pengujian. Sebaliknya, ketika kondisi lingkungan semakin gelap, nilai ADC mengalami peningkatan hingga mencapai 1023 pada pukul 21.00 dengan intensitas cahaya yang mendekati 0 lux. Hasil tersebut menunjukkan bahwa sensor LDR dapat memberikan perubahan nilai ADC sesuai dengan perubahan kondisi pencahayaan lingkungan.

**Tabel 4.1** Hasil Pembacaan Sensor LDR

| Pukul | Nilai ADC LDR | Intensitas Cahaya (Lux) |
|-------|---------------|-------------------------|
| 09.00 | 827           | 12,74                   |
| 12.00 | 68            | 4343,17                 |
| 15.00 | 910           | 5,06                    |
| 18.00 | 991           | 0,74                    |
| 21.00 | 1023          | $\approx 0$             |

Nilai ADC sensor LDR kemudian dikonversi menjadi intensitas cahaya dalam satuan lux. Ini dilakukan secara bertahap, pertama mengubah nilai ADC menjadi tegangan, kemudian menghitung resistansi LDR, dan kemudian mengubah nilai resistansi tersebut menjadi intensitas cahaya berdasarkan karakteristik LDR.

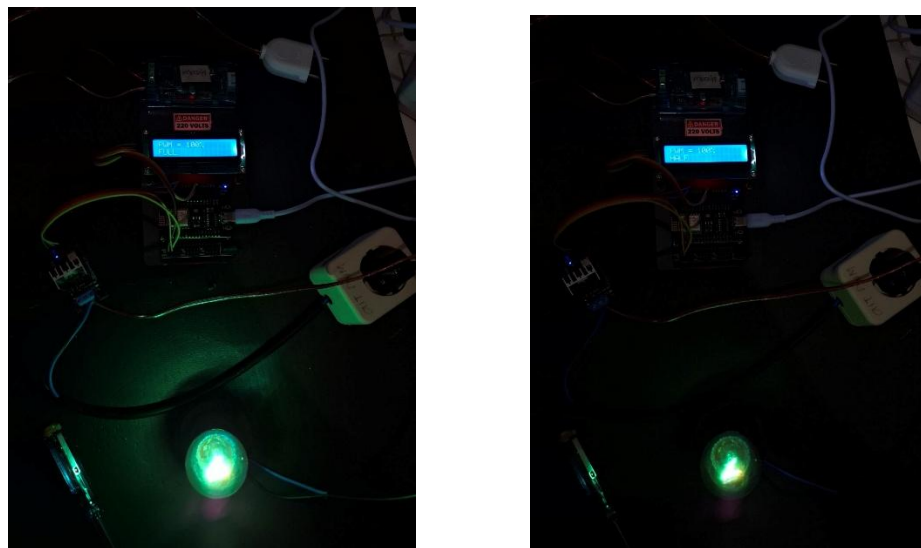
Konversi nilai ADC menjadi tegangan dilakukan menggunakan Persamaan (2.2), dan nilai resistansi LDR dihitung menggunakan Persamaan (2.3), sedangkan konversi nilai resistansi menjadi intensitas cahaya dilakukan menggunakan Persamaan (2.4).

Nilai ADC yang berubah dan hasil konversi intensitas cahaya digunakan sebagai dasar untuk menentukan mode operasi sistem. Untuk mencegah perubahan intensitas lampu yang terlalu sering yang disebabkan oleh perubahan cahaya sesaat, seperti mendung atau bayangan, sistem menggunakan rentang nilai tertentu sebagai acuan perpindahan mode. Oleh karena itu, perubahan yang terjadi dalam waktu singkat dalam kondisi pencahayaan tidak langsung menyebabkan perubahan intensitas lampu.

Hasil pembacaan sensor LDR adalah langkah pertama dalam menjalankan sistem. Sebelum sensor PZEM004T melakukan pembacaan parameter kelistrikan, mode operasi yang ditentukan berdasarkan kondisi pencahayaan digunakan untuk mengatur intensitas lampu melalui AC Dimmer. Proses pembacaan parameter kelistrikan dan deteksi gangguan dilakukan berdasarkan kondisi operasi lampu yang sedang digunakan. Dengan cara ini, intensitas lampu dapat diatur secara lebih stabil, dan hasil pengawasan PZEM004T dapat disesuaikan dengan kondisi operasi lampu.

#### **4.2.3 Pengujian AC Dimmer Robotdyn**

Pengujian AC Dimmer RobotDyn dilakukan untuk mengetahui kemampuan modul dalam mengatur intensitas cahaya lampu AC berdasarkan nilai PWM yang diberikan oleh NodeMCU ESP8266. Pada pengujian ini digunakan lampu AC 220 V sebagai beban, sedangkan nilai PWM diubah menjadi dua kondisi, yaitu PWM 50% dan PWM 100%. Hasil pengujian ditunjukkan pada Gambar 4.6.



(a) PWM 100%

(b) PWM 50%

**Gambar 4.4** Hasil Pengujian AC Dimmer Robotdyn

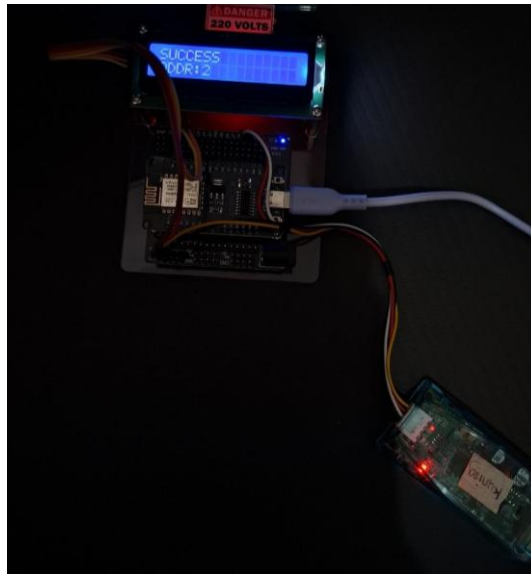
Berdasarkan hasil pengujian pada Gambar 4.6, perubahan nilai PWM memberikan pengaruh terhadap tingkat kecerahan lampu. Pada saat PWM diatur sebesar 50%, lampu menyala dengan intensitas yang lebih rendah dibandingkan kondisi PWM 100%. Sebaliknya, ketika nilai PWM dinaikkan menjadi 100%, lampu menghasilkan cahaya yang lebih terang karena AC Dimmer memberikan daya yang lebih besar ke beban. Hal ini menunjukkan bahwa AC Dimmer RobotDyn mampu mengatur intensitas lampu sesuai nilai PWM yang dikirimkan oleh NodeMCU.

Pengujian AC Dimmer tidak hanya bertujuan untuk mengatur tingkat kecerahan lampu, tetapi juga menjadi dasar dalam proses pembacaan parameter kelistrikan oleh sensor PZEM004T. Perubahan intensitas lampu akan memengaruhi besarnya arus dan daya yang dikonsumsi, sehingga pengaturan PWM dilakukan terlebih dahulu sebelum proses monitoring dilakukan.

#### 4.2.4 Pengujian Sensor PZEM

Pengujian sensor PZEM004T dilakukan untuk memastikan bahwa setiap sensor dapat berkomunikasi dengan NodeMCU ESP8266 serta mampu membaca parameter kelistrikan sesuai dengan kondisi lampu yang dimonitor. Pada penelitian ini digunakan tiga buah sensor PZEM004T yang masing-masing dipasang pada lampu merah, lampu kuning, dan lampu hijau.

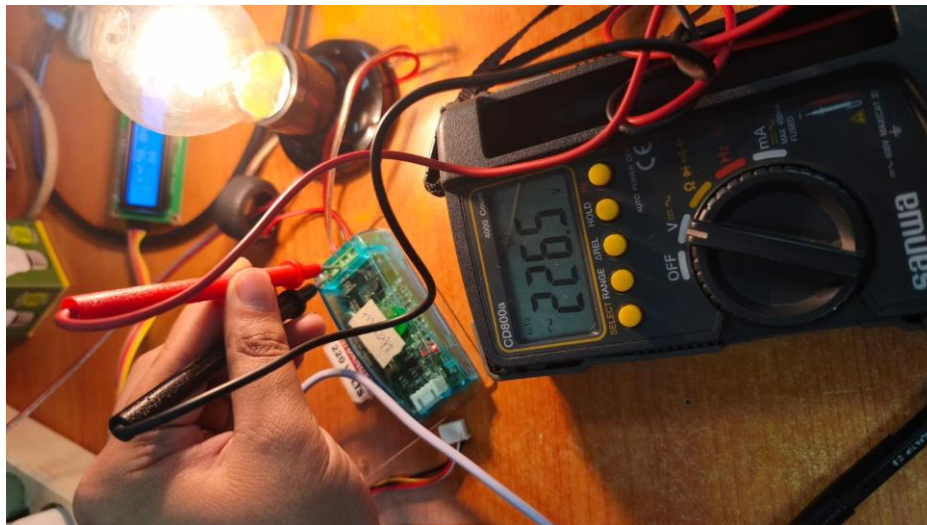
Karena ketiga sensor PZEM004T menggunakan komunikasi Modbus RTU pada jalur serial yang sama, setiap sensor harus memiliki *address* yang berbeda. Apabila seluruh sensor menggunakan *address* bawaan yang sama, NodeMCU tidak dapat membedakan sensor mana yang sedang mengirimkan data sehingga pembacaan parameter kelistrikan menjadi tidak valid.



**Gambar 4.5** Hasil Pemberian Alamat Baru pada Sensor PZEM004T

Berdasarkan Gambar 4.4, proses pemberian alamat berhasil dilakukan sehingga masing-masing sensor memiliki identitas komunikasi yang berbeda. Dengan adanya *address* yang berbeda, NodeMCU dapat membaca data dari setiap sensor secara bergantian sesuai dengan lampu yang dimonitor.

Selanjutnya dilakukan pengujian pembacaan parameter kelistrikan untuk mengetahui apakah sensor mampu mengukur tegangan dan mengirimkan hasil pembacaan ke NodeMCU.



**Gambar 4.6** Hasil Pengujian Pembacaan Sensor PZEM004T dengan AVO meter

Setelah pengujian komunikasi NodeMCU dan sensor PZEM004T selesai, langkah selanjutnya adalah menguji kemampuan sensor untuk mengukur parameter kelistrikan seperti tegangan, arus, dan daya. Tujuan dari pengujian ini adalah untuk mengetahui seberapa akurat pengukuran sensor PZEM004T dengan membandingkannya dengan AVO meter sebagai alat ukur referensi. Selain itu, pengujian dilakukan untuk memastikan bahwa data yang dibaca sensor dapat diterima dan diproklamasikan.

Sebagai contoh, pada pengujian pertama parameter arus dengan beban lampu LED 5 W, hasil pengukuran menggunakan AVO meter sebesar 20 mA, sedangkan hasil pengukuran sensor PZEM004T sebesar 22 mA. Perhitungan persentase error adalah sebagai berikut:

$$Error = \frac{22 - 20}{20} \times 100\%$$

$$Error = \frac{2}{20} \times 100\% = 10\%$$

Berdasarkan perhitungan tersebut, diperoleh nilai error pengukuran arus sebesar 10%. Persentase error dihitung menggunakan Persamaan (2.1).

**Tabel 4.2** Hasil Pengujian Akurasi Pembacaan Parameter Kelistrikan pada Beban Lampu LED 5 W.

| Uji Coba- | Parameter Listrik | Hasil Pengukuran AVO meter | Hasil Pengukuran Sensor PZEM004T | Error (%) |
|-----------|-------------------|----------------------------|----------------------------------|-----------|
| 1         | Tegangan (V)      | 226                        | 226                              | 0         |
|           | Arus (mA)         | 20                         | 22                               | 10        |
|           | Daya (Watt)       | 4,4                        | 4,8                              | 10        |
| 2         | Tegangan (V)      | 226                        | 22                               | 0,4       |
|           | Arus (mA)         | 22                         | 21                               | 4,5       |
|           | Daya (Watt)       | 4,8                        | 4,6                              | 4,1       |
| 3         | Tegangan (V)      | 227                        | 226                              | 0,4       |
|           | Arus (mA)         | 21                         | 22                               | 4,7       |
|           | Daya (Watt)       | 4,6                        | 4,8                              | 4,3       |
| 4         | Tegangan (V)      | 228                        | 228                              | 0         |
|           | Arus (mA)         | 21                         | 21                               | 0         |
|           | Daya (Watt)       | 4,7                        | 4,7                              | 0         |

Berdasarkan Tabel 4.2, sensor PZEM004T mampu mengukur parameter kelistrikan pada beban lampu LED 5 W dengan baik. Hasil pengukuran tegangan menunjukkan nilai yang hampir sama dengan hasil pengukuran menggunakan AVO meter, yaitu berada pada kisaran 226–228 V dengan nilai error sebesar 0–0,4%. Hal ini menunjukkan bahwa sensor memiliki tingkat akurasi yang tinggi dalam mengukur tegangan.

Pada parameter arus dan daya diperoleh nilai error yang lebih besar, yaitu berkisar antara 0–10%. Nilai tersebut masih dapat diterima karena arus yang diukur berada pada orde miliampere sehingga selisih pembacaan sebesar 1–2 mA saja dapat menghasilkan persentase error yang relatif besar. Selain itu, penggunaan lampu LED yang memiliki rangkaian driver internal juga dapat menyebabkan bentuk gelombang arus tidak sepenuhnya sinusoidal sehingga memengaruhi hasil pengukuran sensor maupun alat ukur referensi. Secara keseluruhan, hasil pengujian menunjukkan bahwa sensor PZEM004T memiliki kemampuan yang baik untuk membaca parameter kelistrikan pada beban berdaya rendah.

**Tabel 4.3** Hasil Pengujian Akurasi Pembacaan Parameter Kelistrikan pada Beban Lampu Pijar 10 W.

| No | Mode Operasi | Parameter Listrik | Hasil Pengukuran AVO meter | Hasil Pengukuran Sensor PZEM004T | Error (%) |
|----|--------------|-------------------|----------------------------|----------------------------------|-----------|
| 1  | PWM 25%      | Tegangan (V)      | 117                        | 118                              | 0,85      |
|    |              | Arus (mA)         | 19                         | 20                               | 5,26      |
|    |              | Daya (Watt)       | 2,3                        | 2,4                              | 4,35      |
| 2  | PWM 50%      | Tegangan (V)      | 164                        | 165                              | 0,61      |
|    |              | Arus (mA)         | 31                         | 33                               | 6,45      |
|    |              | Daya (Watt)       | 5,3                        | 5,5                              | 3,77      |
| 3  | PWM 75%      | Tegangan (V)      | 198                        | 197                              | 0,51      |
|    |              | Arus (mA)         | 40                         | 42                               | 5,00      |
|    |              | Daya (Watt)       | 8,0                        | 8,2                              | 2,50      |
| 4  | PWM 100%     | Tegangan (V)      | 226                        | 227                              | 0,44      |
|    |              | Arus (mA)         | 44                         | 46                               | 4,55      |
|    |              | Daya (Watt)       | 9,9                        | 10,2                             | 3,03      |

Berdasarkan Tabel 4.3, hasil pengujian menunjukkan bahwa peningkatan nilai PWM menyebabkan tegangan efektif yang diterima lampu semakin besar. Pada kondisi PWM 25% tegangan yang terukur sebesar 117–118 V, kemudian meningkat menjadi sekitar 226–227 V pada PWM 100%. Seiring meningkatnya tegangan, nilai arus dan daya yang terukur juga mengalami peningkatan dari sekitar 20 mA dan 2,4 W menjadi 46 mA dan 10,2 W.

Hasil perbandingan dengan AVO meter menunjukkan bahwa sensor PZEM004T memiliki nilai error tegangan kurang dari 1%, sedangkan error arus dan daya masing-masing berada pada kisaran 4,55–6,45% dan 2,50–4,35%. Perbedaan tersebut masih tergolong kecil dan menunjukkan bahwa sensor mampu mengukur parameter kelistrikan dengan cukup baik pada berbagai tingkat PWM.

**Tabel 4.4** Hasil Pengujian Akurasi Pembacaan Parameter Kelistrikan pada Beban Lampu Pijar 15 W.

| No | Mode Operasi | Parameter Listrik | Hasil Pengukuran AVO meter | Hasil Pengukuran Sensor PZEM004T | Error (%) |
|----|--------------|-------------------|----------------------------|----------------------------------|-----------|
| 1  | PWM 25%      | Tegangan (V)      | 116                        | 117                              | 0,86      |
|    |              | Arus (mA)         | 28                         | 29                               | 3,57      |
|    |              | Daya (Watt)       | 3,5                        | 3,6                              | 2,86      |
| 2  | PWM 50%      | Tegangan (V)      | 166                        | 167                              | 0,60      |
|    |              | Arus (mA)         | 47                         | 49                               | 4,26      |

|   |             |              |      |      |      |
|---|-------------|--------------|------|------|------|
|   |             | Daya (Watt)  | 7,8  | 8,0  | 2,56 |
| 3 | PWM<br>75%  | Tegangan (V) | 197  | 198  | 0,51 |
|   |             | Arus (mA)    | 60   | 62   | 3,33 |
|   |             | Daya (Watt)  | 11,7 | 11,9 | 1,71 |
| 4 | PWM<br>100% | Tegangan (V) | 227  | 226  | 0,44 |
|   |             | Arus (mA)    | 66   | 68   | 3,03 |
|   |             | Daya (Watt)  | 15,0 | 15,3 | 2,00 |

Berdasarkan Tabel 4.4, pola hasil pengukuran menunjukkan kecenderungan yang sama dengan pengujian sebelumnya. Semakin besar nilai PWM yang diberikan oleh AC dimmer, maka semakin tinggi tegangan efektif yang diterima beban sehingga arus dan daya yang dikonsumsi lampu juga meningkat. Pada kondisi PWM 100%, daya yang terbaca oleh sensor mencapai sekitar 15,3 W yang mendekati daya nominal lampu.

Nilai error tegangan berada pada rentang 0,44–0,86%, sedangkan error arus dan daya masing-masing berada pada kisaran 3,03–4,26% dan 1,71–2,86%. Nilai error tersebut menunjukkan bahwa sensor tetap memiliki tingkat akurasi yang baik meskipun beban yang digunakan lebih besar dibandingkan pengujian sebelumnya.

**Tabel 4.5** Hasil Pengujian Akurasi Pembacaan Parameter Kelistrikan pada Beban Lampu Pijar 25 W.

| No | Mode Operasi | Parameter Listrik | Hasil Pengukuran AVO meter | Hasil Pengukuran Sensor PZEM004T | Error (%) |
|----|--------------|-------------------|----------------------------|----------------------------------|-----------|
| 1  | PWM<br>25%   | Tegangan (V)      | 118                        | 119                              | 0,85      |
|    |              | Arus (mA)         | 47                         | 49                               | 4,26      |
|    |              | Daya (Watt)       | 5,6                        | 5,8                              | 3,57      |
| 2  | PWM<br>50%   | Tegangan (V)      | 165                        | 166                              | 0,61      |
|    |              | Arus (mA)         | 77                         | 80                               | 3,90      |
|    |              | Daya (Watt)       | 12,8                       | 13,1                             | 2,34      |
| 3  | PWM<br>75%   | Tegangan (V)      | 196                        | 197                              | 0,51      |
|    |              | Arus (mA)         | 100                        | 103                              | 3,00      |
|    |              | Daya (Watt)       | 19,5                       | 19,9                             | 2,05      |
| 4  | PWM<br>100%  | Tegangan (V)      | 225                        | 226                              | 0,44      |
|    |              | Arus (mA)         | 111                        | 114                              | 2,70      |
|    |              | Daya (Watt)       | 24,9                       | 25,4                             | 2,01      |

Berdasarkan Tabel 4.5, peningkatan nilai PWM kembali menghasilkan kenaikan tegangan efektif, arus, dan daya pada lampu pijar 25 W. Pada kondisi PWM 25%, daya yang terukur masih berada pada kisaran 5,8 W, sedangkan pada PWM 100% meningkat hingga sekitar 25,4 W yang mendekati daya nominal lampu.

Hasil pengujian juga menunjukkan bahwa nilai error tegangan tetap berada di bawah 1%, sedangkan error arus dan daya masing-masing berkisar antara 2,70–4,26% dan 2,01–3,57%. Dibandingkan dengan pengujian pada lampu 10 W, nilai error arus cenderung sedikit lebih kecil. Hal ini disebabkan oleh besarnya arus yang diukur sehingga selisih pembacaan antara sensor dan AVO meter menjadi relatif lebih kecil jika dinyatakan dalam bentuk persentase.

Berdasarkan seluruh hasil pengujian pada Tabel 4.2 hingga Tabel 4.5, sensor PZEM004T mampu mengukur parameter tegangan, arus, dan daya dengan tingkat akurasi yang baik pada berbagai jenis beban dan variasi PWM. Tegangan yang diukur memiliki nilai error kurang dari 1% pada seluruh pengujian, sedangkan error arus dan daya masih berada pada kisaran sekitar 2–10%. Nilai tersebut masih dapat diterima untuk aplikasi monitoring berbasis *Internet of Things*.

Hasil pengujian juga menunjukkan bahwa perubahan nilai PWM pada dimmer AC memengaruhi jumlah tegangan efektif yang diterima lampu. Semakin tinggi nilai PWM yang diberikan, semakin besar tegangan efektif, arus, dan daya yang diukur. Hal ini menunjukkan bahwa sistem pengaturan intensitas cahaya yang menggunakan dimmer AC bekerja sesuai dengan prinsip pengendalian sudut penyalan (*control phase angle*). Selain itu, sensor PZEM004T memiliki kemampuan untuk secara konsisten memantau perubahan parameter kelistrikan tersebut.

#### **4.2.5 Pengujian Koneksi Wi-fi**

Sebelum memulai proses pengiriman data ke platform Blynk, pengujian koneksi Wi-Fi dilakukan untuk memastikan bahwa NodeMCU ESP8266 terhubung ke jaringan internet. Pengujian ini sangat penting karena semua proses pengawasan

dilakukan secara real-time, sehingga koneksi internet yang stabil adalah kunci untuk sistem yang beroperasi dengan baik..



**Gambar 4.7** Hasil Pengujian Koneksi Wi-fi

Hasil pengujian NodeMCU ESP8266 dapat dilihat pada Gambar 4.7. Mereka menunjukkan bahwa NodeMCU ESP8266 berhasil terhubung ke jaringan Wi-Fi dan berkoneksi dengan server Blynk. Sistem dapat mengirimkan data hasil pembacaan sensor secara berkala setelah koneksi berhasil tanpa mengalami kegagalan komunikasi selama proses pengujian. Hal ini menunjukkan bahwa aplikasi Blynk dan koneksi internet yang digunakan telah mendukung proses pemantauan real-time..

#### **4.2.6 Pengujian Blynk**

Pengujian Blynk dilakukan untuk memastikan bahwa NodeMCU ESP8266 berhasil terhubung dengan platform Blynk sehingga data hasil monitoring dapat dikirimkan melalui jaringan internet. Pengujian dilakukan setelah NodeMCU berhasil terkoneksi dengan jaringan Wi-Fi, kemudian sistem melakukan proses autentikasi menggunakan *Authentication Token* yang telah dikonfigurasi pada program.



**Gambar 4.8** Hasil Pengujian Koneksi ke Blynk

Sebelum pengujian tampilan pemantauan pada aplikasi Blynk, dilakukan pengujian kecepatan pengiriman data dari NodeMCU ESP8266 ke platform Blynk. Tujuan dari pengujian ini adalah untuk mengetahui seberapa cepat server Blynk menerima data hasil pembacaan sensor, sehingga informasi yang ditampilkan pada aplikasi dapat dilihat dalam kondisi sistem dalam waktu nyata. Dengan hasil seperti yang ditunjukkan pada Tabel 4.6, pengujian dilakukan sepuluh kali dalam waktu sekitar satu jam.

**Tabel 4.6** Hasil Pengujian Latensi Pengiriman Data ke Platform Blynk

| No | Waktu Pengujian | Waktu Data Dikirim | Waktu Data Diterima di Blynk | Latensi (detik) |
|----|-----------------|--------------------|------------------------------|-----------------|
| 1  | 09.00           | 09:00:05           | 09:00:06                     | 1               |
| 2  | 09.07           | 09:07:18           | 09:07:20                     | 2               |
| 3  | 09.14           | 09:14:26           | 09:14:27                     | 1               |
| 4  | 09.21           | 09:21:12           | 09:21:15                     | 3               |
| 5  | 09.28           | 09:28:34           | 09:28:36                     | 2               |
| 6  | 09.35           | 09:35:08           | 09:35:12                     | 4               |
| 7  | 09.42           | 09:42:16           | 09:42:18                     | 2               |
| 8  | 09.49           | 09:49:25           | 09:49:30                     | 5               |
| 9  | 09.56           | 09:56:10           | 09:56:12                     | 2               |
| 10 | 10.00           | 10:00:21           | 10:00:22                     | 1               |

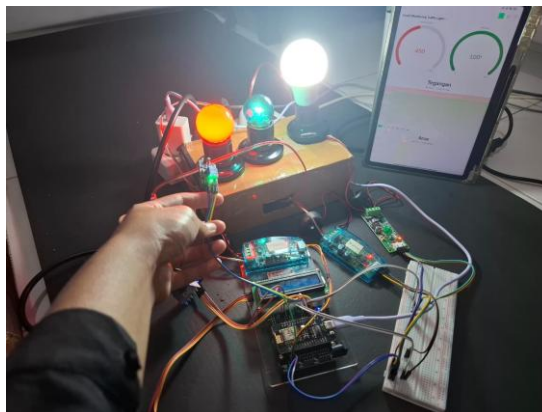
Hasil pengujian menunjukkan bahwa sistem membutuhkan waktu sekitar 1 hingga 5 detik untuk mengirimkan data dari NodeMCU ESP8266 ke platform Blynk, dengan latensi rata-rata 2,3 detik. Ini menunjukkan bahwa platform Blynk berhasil menerima semua data yang dibaca sensor dan menjalankan proses pemantauan dalam waktu nyata, seperti yang ditunjukkan dalam Tabel 4.6.

Perbedaan waktu latensi pada setiap pengujian dipengaruhi oleh beberapa faktor, di antaranya kualitas koneksi jaringan WiFi yang digunakan, waktu pemrosesan data pada NodeMCU ESP8266, proses komunikasi serial antara mikrokontroler dengan sensor PZEM004T, serta waktu respons server Blynk dalam menerima dan menampilkan data. Selain itu, proses pembacaan beberapa parameter kelistrikan secara berurutan, seperti tegangan, arus, daya, energi, frekuensi, dan faktor daya, juga memerlukan waktu pemrosesan sebelum data dikirimkan ke platform Blynk.

Meskipun latensi berubah setiap pengujian, waktu pengiriman data yang dihasilkan untuk aplikasi monitoring berbasis Internet of Things masih dianggap cepat dan stabil. Dengan latensi rata-rata 2,3 detik, sistem yang dirancang dapat menyajikan informasi secara berkala untuk membantu proses pemantauan online. Hasil pengujian menunjukkan bahwa NodeMCU dan platform Blynk berkomunikasi dengan baik, yang berarti sistem siap digunakan selama tahap pengujian.

#### 4.2.7 Pengujian Sistem

Pengujian dilakukan pada sistem untuk memastikan bahwa seluruh komponen yang telah dirancang dapat bekerja bersama untuk melakukan fungsi yang diharapkan. Gambar 4.9 menunjukkan hasil pengujian sistem saat NodeMCU ESP8266 mengintegrasikan pembacaan sensor LDR, pengaturan intensitas lampu dengan AC Dimmer, pembacaan parameter kelistrikan oleh sensor PZEM004T, dan pengiriman data monitoring ke platform Blynk.



**Gambar 4.9** Hasil Pengujian Sistem Monitoring *Traffic Light*

Hasil pengujian menunjukkan bahwa sistem mampu menjalankan proses monitoring seperti yang direncanakan. Sensor LDR mengidentifikasi intensitas cahaya di sekitarnya dan memilih mode operasi lampu. Kemudian, AC Dimmer mengubah tingkat kecerahan lampu sesuai dengan mode tersebut. Selanjutnya, parameter kelistrikan dibaca oleh sensor PZEM004T, dan NodeMCU mengirimkan semua data pemantauan ke aplikasi Blynk. Dengan demikian, smartphone dapat memantau secara real-time.



**Gambar 4.10** Tampilan Layar Monitoring pada Blynk

Pada tampilan Blynk terlihat bahwa parameter tegangan dan frekuensi berhasil terbaca pada ketiga sensor PZEM004T. Sementara itu, parameter arus hanya berhasil terbaca pada sensor yang terhubung dengan beban lampu pijar. Oleh karena itu, nilai daya (*power*) dan faktor daya (*power factor*) juga hanya dapat ditampilkan pada sensor tersebut karena kedua parameter tersebut dihitung

berdasarkan hasil pembacaan arus. Selain menampilkan parameter kelistrikan, aplikasi Blynk juga menampilkan nilai intensitas cahaya dari sensor LDR, persentase PWM lampu, serta status gangguan pada masing-masing lampu sehingga kondisi sistem dapat dipantau secara langsung melalui smartphone.

Untuk memastikan bahwa fitur deteksi gangguan telah bekerja sesuai dengan perancangan sistem, dilakukan pengujian pada beberapa skenario operasi dengan variasi kondisi pencahayaan yang memengaruhi nilai PWM lampu. Pengujian dilakukan pada kondisi siang, senja, dan malam, kemudian diberikan beberapa skenario seperti kondisi normal, gangguan arus akibat lampu padam atau terputus, serta gangguan tegangan akibat penurunan tegangan suplai. Hasil pengujian deteksi gangguan sistem ditunjukkan pada Tabel 4.7.

**Tabel 4.7** Hasil Pengujian Deteksi Gangguan pada Sistem Monitoring *Traffic Light*

| No | Kondisi | PWM  | Lux   | Skenario       | Tegangan | Arus   | Status Sistem |
|----|---------|------|-------|----------------|----------|--------|---------------|
| 1  | Siang   | 100% | 4.280 | Normal         | 226 V    | 111 mA | Normal        |
| 2  | Siang   | 100% | 4.350 | Lampu putus    | 226 V    | 0 mA   | Current Fault |
| 3  | Siang   | 100% | 4.190 | Tegangan turun | 185 V    | 90 mA  | Voltage Fault |
| 4  | Senja   | 80%  | 8,2   | Normal         | 198 V    | 98 mA  | Normal        |
| 5  | Senja   | 80%  | 7,6   | Lampu putus    | 198 V    | 0 mA   | Current Fault |
| 6  | Senja   | 80%  | 6,9   | Tegangan turun | 170 V    | 82 mA  | Voltage Fault |
| 7  | Malam   | 60%  | 0,8   | Normal         | 165 V    | 80 mA  | Normal        |
| 8  | Malam   | 60%  | 0,6   | Lampu putus    | 165 V    | 0 mA   | Current Fault |
| 9  | Malam   | 60%  | 0,4   | Tegangan turun | 150 V    | 68 mA  | Voltage Fault |

Berdasarkan Tabel 4.7, pengujian sistem dilakukan pada tiga kondisi pencahayaan, yaitu siang, senja, dan malam dengan nilai intensitas cahaya yang berbeda. Pada kondisi siang, intensitas cahaya berada pada kisaran 4.190–4.350 lux sehingga sistem mengatur PWM sebesar 100%. Pada kondisi senja, nilai intensitas cahaya berada pada kisaran 6,9–8,2 lux dengan PWM sebesar 80%, sedangkan pada kondisi malam nilai intensitas cahaya berada pada kisaran 0,4–0,8 lux dengan PWM

sebesar 60%. Perbedaan nilai lux tersebut menunjukkan bahwa sensor LDR dapat digunakan untuk menentukan kondisi pencahayaan yang selanjutnya menjadi acuan dalam pengaturan intensitas lampu melalui AC Dimmer.

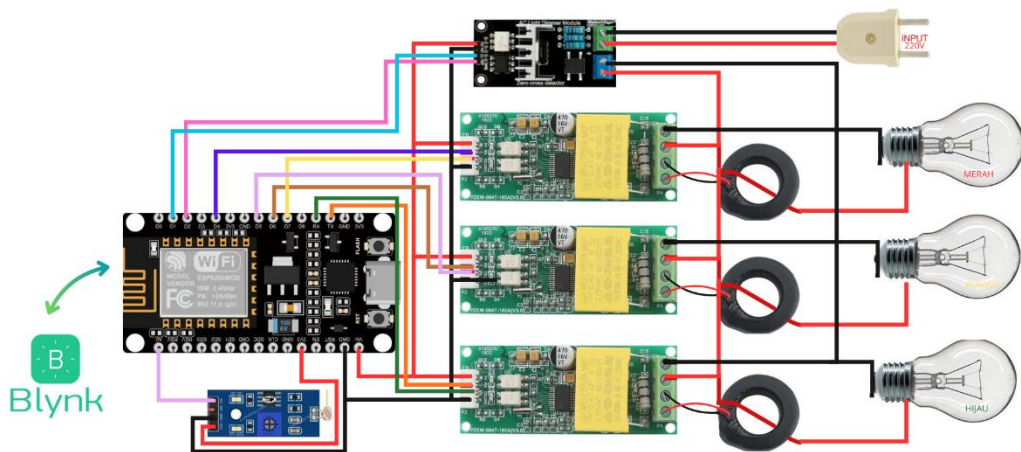
Pada kondisi normal, aplikasi Blynk menampilkan status *Normal* karena nilai tegangan dan arus masih berada dalam rentang yang telah ditentukan. Ketika lampu dipadamkan atau diputus sehingga arus tidak terbaca oleh sensor PZEM004T, sistem berhasil memberikan notifikasi *Current Fault*. Sementara itu, ketika tegangan suplai diturunkan hingga berada di bawah batas yang telah ditentukan, sistem memberikan notifikasi *Voltage Fault*. Hasil tersebut menunjukkan bahwa algoritma deteksi gangguan yang diimplementasikan pada NodeMCU ESP8266 mampu mengidentifikasi kondisi abnormal berdasarkan parameter kelistrikan yang dibaca sensor PZEM004T, sehingga informasi gangguan dapat ditampilkan secara otomatis pada aplikasi Blynk.

Secara keseluruhan, hasil pengujian menunjukkan bahwa sistem telah mampu menjalankan fungsi monitoring dan pengiriman data secara *real-time*. Seluruh informasi yang diperoleh dari sensor dapat ditampilkan pada dashboard Blynk sehingga memudahkan pengguna dalam memantau kondisi setiap lampu lalu lintas.

#### 4.3 Pembahasan

Berdasarkan hasil perancangan dan pengujian yang telah dilakukan, sistem monitoring dan deteksi gangguan lampu lalu lintas berbasis Internet of Things menggunakan NodeMCU ESP8266, sensor PZEM004T, sensor LDR, AC Dimmer, dan platform Blynk telah berhasil direalisasikan. Seluruh komponen dapat bekerja secara terintegrasi mulai dari proses pembacaan intensitas cahaya, pengaturan tingkat kecerahan lampu, monitoring parameter kelistrikan, hingga pengiriman data ke aplikasi Blynk secara *real-time*. Hal tersebut menunjukkan bahwa rancangan sistem yang dibuat telah mampu memenuhi tujuan utama penelitian, yaitu menggabungkan fungsi monitoring dan pengendalian lampu lalu lintas dalam satu sistem IoT.

Selama proses implementasi, terdapat perubahan pada rancangan awal sistem. Pada tahap perancangan, ketiga sensor PZEM004T direncanakan menggunakan satu jalur komunikasi serial dengan metode perubahan *address* pada masing-masing sensor. Metode tersebut telah berhasil dilakukan sehingga setiap sensor memiliki alamat yang berbeda. Namun, setelah diterapkan pada NodeMCU ESP8266, komunikasi dengan ketiga sensor belum dapat berjalan secara stabil sehingga pembacaan data secara bersamaan tidak dapat dilakukan. Oleh karena itu, konfigurasi komunikasi diubah dengan menggunakan pin komunikasi yang berbeda untuk setiap sensor PZEM004T. Perubahan tersebut mengharuskan pin yang semula digunakan oleh LCD dialihkan untuk komunikasi sensor sehingga LCD tidak lagi digunakan pada implementasi akhir. Sebagai penggantinya, seluruh informasi monitoring ditampilkan melalui platform Blynk yang dapat diakses menggunakan smartphone. Meskipun terjadi perubahan dari rancangan awal, fungsi utama sistem tetap dapat berjalan dengan baik bahkan menghasilkan komunikasi yang lebih stabil.



**Gambar 4.11** Wiring Diagram Akhir Sistem

Hasil pengujian juga menunjukkan bahwa sensor PZEM004T mampu membaca parameter tegangan dan frekuensi pada ketiga lampu lalu lintas. Namun, pembacaan arus hanya berhasil pada sensor yang terhubung dengan beban lampu pijar sehingga nilai daya (*power*) dan faktor daya (*power factor*) juga hanya dapat ditampilkan pada sensor tersebut. Meskipun demikian, sistem tetap dapat

melakukan proses monitoring kondisi lampu dan memberikan informasi status gangguan melalui aplikasi Blynk berdasarkan parameter kelistrikan yang berhasil diperoleh. Dengan demikian, sistem yang dikembangkan telah mampu menjalankan fungsi monitoring kondisi lampu lalu lintas sesuai dengan tujuan penelitian.

Selain fungsi monitoring, penelitian ini juga berhasil mengintegrasikan sensor LDR dengan AC Dimmer untuk mengatur intensitas cahaya lampu secara otomatis berdasarkan kondisi lingkungan. Sensor LDR digunakan untuk menentukan mode operasi siang atau malam sehingga sistem dapat memberikan nilai PWM yang berbeda sesuai kondisi pencahayaan. Setelah intensitas lampu diatur, sensor PZEM004T melakukan pembacaan parameter kelistrikan sesuai kondisi operasi yang sedang berlangsung. Seluruh data tersebut kemudian dikirimkan ke platform Blynk sehingga pengguna dapat memantau nilai intensitas cahaya, parameter kelistrikan, serta status gangguan setiap lampu secara *real-time*. Integrasi tersebut menjadi salah satu keunggulan sistem karena proses pengaturan intensitas dan monitoring tidak berjalan secara terpisah, melainkan saling mendukung dalam satu sistem yang terintegrasi.

Berdasarkan diskusi sebelumnya, kita dapat mengetahui bahwa sistem yang dibuat telah mampu menyelesaikan masalah penelitian. Sistem ini menggunakan platform Blynk, AC Dimmer, sensor LDR, sensor PZEM004T, dan NodeMCU ESP8266 sebagai media pengawasan. Selain itu, sistem memiliki kemampuan untuk melakukan monitoring parameter kelistrikan untuk mengidentifikasi gangguan lampu dan mengintegrasikan pengaturan intensitas cahaya secara otomatis ke dalam sistem *Internet of Things* (IoT).

## BAB V

### SIMPULAN

#### 5.1 Kesimpulan

Berdasarkan hasil perancangan, pengujian, dan pembahasan yang telah dilakukan, maka dapat diambil beberapa kesimpulan sebagai berikut.

1. Sistem monitoring dan deteksi gangguan lampu lalu lintas berbasis IoT berhasil direalisasikan menggunakan NodeMCU ESP8266 dan dapat menjalankan proses monitoring kondisi lampu serta parameter kelistrikan melalui platform Blynk. Hasil implementasi menunjukkan bahwa sistem dapat membaca parameter kelistrikan dari ketiga sensor PZEM004T dan menampilkan hasil monitoring pada smartphone secara *real-time*.
2. Berdasarkan hasil pengujian, ketiga sensor PZEM004T berhasil membaca parameter kelistrikan berupa tegangan, arus, daya, frekuensi, dan faktor daya pada masing-masing lampu. Data hasil pembacaan tersebut dapat digunakan oleh sistem untuk menentukan kondisi normal maupun gangguan pada lampu berdasarkan parameter kelistrikan yang diperoleh, kemudian hasil kondisi tersebut ditampilkan melalui platform Blynk.
3. Integrasi monitoring kondisi lampu dengan pengaturan intensitas cahaya berhasil diterapkan dalam satu sistem IoT. Sensor LDR dapat membaca kondisi pencahayaan lingkungan untuk menentukan mode siang dan malam, kemudian AC Dimmer mengatur intensitas lampu sesuai dengan kondisi tersebut. Hasil pembacaan LDR, pengaturan intensitas lampu, parameter kelistrikan, dan status gangguan dapat dipantau melalui platform Blynk secara *real-time* menggunakan smartphone.

## 5.2 Saran

Berikut ini adalah beberapa rekomendasi yang dapat digunakan untuk penelitian lanjutan berdasarkan temuan penelitian:

1. Penelitian selanjutnya disarankan menggunakan mikrokontroler dengan jumlah antarmuka komunikasi serial yang lebih banyak, seperti ESP32, sehingga proses komunikasi dengan beberapa sensor PZEM004T dapat dilakukan secara lebih stabil tanpa memerlukan penyesuaian konfigurasi pin seperti pada ESP8266.
2. Pengembangan selanjutnya dapat menggunakan lampu lalu lintas yang memiliki karakteristik beban sesuai dengan kondisi sebenarnya sehingga seluruh parameter kelistrikan, seperti arus, daya, dan faktor daya, dapat terbaca dengan lebih optimal pada setiap lampu.
3. Sistem dapat dikembangkan dengan menambahkan fitur notifikasi otomatis ke Telegram, WhatsApp, atau platform lainnya. Ini akan memungkinkan petugas untuk mendapatkan informasi segera ketika terjadi gangguan lampu lalu lintas tanpa membuka aplikasi pemantauan..
4. Pengembangan berikutnya juga dapat dilakukan dengan menerapkan sistem pada lampu lalu lintas sebenarnya serta menambahkan fitur penyimpanan data (*data logging*) untuk menganalisis riwayat kondisi lampu dan mendukung proses pemeliharaan secara berkala.
5. Penelitian selanjutnya disarankan mengembangkan metode deteksi gangguan dengan menambahkan klasifikasi jenis gangguan berdasarkan perubahan parameter kelistrikan, seperti tegangan, arus, dan daya. Dengan demikian, sistem tidak hanya memberikan informasi bahwa terjadi gangguan, tetapi juga dapat mengidentifikasi jenis gangguan yang terjadi sehingga dapat membantu petugas dalam menentukan tindakan pemeriksaan dan perbaikan secara lebih cepat.

## DAFTAR PUSTAKA

- Abdillah, F., & Koprari, M. (2024). Implementasi Sistem Kontrol dan Monitoring Lampu Jarak Jauh Menggunakan Blynk API dan Micropython. *The Indonesian Journal of Computer Science*, 8196-8205.
- Abror, A. H., & Haratama, K. R. (2024). Studi Sistem Pemeliharaan (APILL) Alat Isyarat Pemberi Lalu Lintas (Studi Kasus: Kota Surabaya). *Mitrans : Jurnal Media Publikasi Terapan Transportasi*, 355-360.
- Andika, J., Permana, E., & Attamimi, S. (2022). Perancangan Sistem Otomatisasi dan Monitoring Perangkat Perawatan Tanaman Hias Berbasis Internet of Things. *Jurnal Teknologi Elektro*, 100-107.
- Asry, A. I., Lutfi, A. U., & Ahmad, R. (2024). Monitoring System for Traffic Light Lamp Damage Using Blynk Application Based on IoT ESP32. *Jurnal Teknologi Transportasi dan Logistik*, 59-66.
- Atmasani, D. (2026). Studi Literatur: Penerapan Internet of Things pada Sistem Monitoring dan Kontrol Berbasis Elektronika. *Jurnal Media Elektronika*, 23-33.
- Firmansyah, M. R., Azami, M. T., Mubarak, A., & Mubaroqan, R. (2022). Sensor Lampu Otomatis Berbasis Light Dependent Resistor (LDR). *Jurnal Eletronika, Kelistrikan, Control, Robot, Power, Telekomunikasi, Komputer, AI*, 14-16.
- Herlina, A., Syahbana, M. I., Gunawan, M. A., & Rizqi, M. M. (2022). Sistem Kendali Lampu Berbasis Iot Menggunakan Aplikasi Blynk 2.0. *Jurnal Inovasi dan Sains Teknik Elektro*, 61-66.
- Ikha, A. F., & Yuhendri, M. (2022). Monitoring Pemakaian Energi Listrik Berbasis Internet of Things. *JTEIN: Jurnal Teknik Elektro Indonesia*, 257-266.
- Irsyada, R., Haq, M. A., Rohmah, A., & Saputra, P. A. (2022). Implementasi NodeMCU ESP8266 dan Sensor Cahaya Pada Lampu Berbasis Internet Of Things. *Jurnal Ilmiah Sistem Informasi dan Ilmu Komputer*, 22-32.

- Kaur, S., & Sharma, S. (2023). IoT-Enabled Air Pollution Monitoring Systems: Technologies, Solutions, and Challenges. *Recent Developments in Electronics and Communication Systems*, 347-352.
- Kusumo, B., & Salam, F. (2023). Rancang Bangun Sistem Kontrol dan Alat Monitoring Traffic Light Menggunakan ESP8266 Berbasis IoT. *Jurnal Elektro*, 247-256.
- Lia, Rihartanto, & Dwi, T. M. (2026). Sistem Kontrol Intensitas Cahaya Berbasis ESP32 Menggunakan AC Light Dimmer dan Telegram. *Jurnal Sains Informatika Terapan (JSIT)*, 511-516.
- Muslihi, M. T. (2025). Pengembangan dan Evaluasi Sistem Monitoring Konsumsi Daya Listrik Berbasis IoT dengan Sensor PZEM-004T dan ESP8266. *Jurnal Fasilkom*, 77-83.
- Pambudi, A. P., Supriyadi, A. P., Kurniadi, A. R., Fazryansyah, M. R., & Stefanie, A. (2025). Implementasi Sistem Pencahayaan Otomatis dan Monitoring Berbasis IoT pada Area Parkir Menggunakan Sensor LDR dan PIR. *jurnal Informatika dan Teknik Elektro Terapan*, 2119-2126.
- Putra, H., Eteruddin, H., Monice, & Aliansyah, A. N. (2022). Analisis Sistem Monitoring Daya Listrik Menggunakan Internet of Things( IoT ) Pada Gedung Teknik Elektro Politeknik Negeri Bengkalis. *Jurnal Fokus Elektroda*, 207-211.
- Rosiana, & Herwana, M. R. (2024). *TrafficEase IoT : Sistem Monitoring Kerusakan pada Lampu Lalu Lintas Berbasis Internet-of-Things*. Yogyakarta: Universitas Islam Indonesia.
- Safira, V. R., Elsi, Z. R., & Haryanto, D. (2025). Monitoring Arus Listrik pada Lampu Berbasis Internet of Things Menggunakan Aplikasi Blynk. *SENTRI : Jurnal Riset Ilmiah*, 3145-3151.
- Surya, I., Kustija, J., Pawinanto, R. E., Pramudita, R., Rizqulloh, M. A., Wahyudin, D., & Haritman, E. (2023). Sistem monitoring beban listrik dan perbaikan faktor daya menggunakan PZEM004T dan dashboard Adafruit berbasis IoT. *Jurnal Ilmiah Telekomunikasi, Elektronika, dan Listrik Tenaga*, 235-246.

- Syahri, A., & Bintoro, A. (2023). Monitoring dan Controlling Daya Berbasis Arduino Uno Menggunakan Sensor PZEM-004T. *Jurnal Energi Elektrik*, 43-51.
- UU No 22. (2009). *tentang Lalu Lintas dan Angakatan Jalan*. <https://peraturan.bpk.go.id/Details/38654/uu-no-22-tahun-2009>.
- Wantudi, A., Triayudi, A., & Benrahman. (2023). Sistem Monitoring Motion, Smoke, Listrik, Suhu, dan Kelembapan pada Data Center Menggunakan NodeMCU ESP8266. *Jurnal Ilmiah Penelitian dan Pembelajaran Informatika*, 106-114.
- Witczak, D., & Szymoniak, S. (2024). Review of Monitoring and Control Systems Based on Internet. *Applied Science*, 1-27.

## LAMPIRAN

### *Lampiran 1*

```
#define BLYNK_TEMPLATE_ID "TMPL6vHNAw8vM"
#define BLYNK_TEMPLATE_NAME "Smart monitoring traffic light"
#define BLYNK_AUTH_TOKEN "zeENteCGUqwEzFWcY_bnl6sk6DGIHnth"

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <PZEM004Tv30.h>
#include <SoftwareSerial.h>
#include <RBDdimmer.h>

char ssid[] = "Redmi Pad";
char pass[] = "akulapar7";

// =====
// AC DIMMER
// =====
#define zeroCrossPin 5
#define dimPin 4

dimmerLamp dimmer(dimPin, zeroCrossPin);

// =====
// PZEM 1
// RED
// Address 0x10
// =====

#define RX1 D7
#define TX1 D4

SoftwareSerial pzemSerial1(RX1, TX1);
PZEM004Tv30 pzem1(pzemSerial1, 0x10);

// =====
// PZEM 2
// YELLOW
// Address 0x11
// =====

#define RX2 D5
#define TX2 D6
```

```
SoftwareSerial pzemSerial2(RX2, TX2);
PZEM004Tv30 pzem2(pzemSerial2, 0x11);
```

```
// =====
// PZEM 3
// GREEN
// RX/TX ESP8266
// Address 0x03
// =====
```

```
PZEM004Tv30 pzem3(Serial, 0x03);
```

```
// =====
// DATA SENSOR
// =====
```

```
int ldrValue;
int pwmValue;
```

```
float voltage[3];
float current[3];
float power[3];
float frequency[3];
float pf[3];
float energy[3];
```

```
// STATUS
```

```
String redStatus = "NORMAL";
String yellowStatus = "NORMAL";
String greenStatus = "NORMAL";
```

```
// TIMER
```

```
unsigned long sensorTimer;
unsigned long blynkTimer;
```

```
const unsigned long faultDelay = 10000;
```

```
unsigned long redVoltTimer;
unsigned long redCurrTimer;
```

```
unsigned long yellowVoltTimer;
unsigned long yellowCurrTimer;
```

```
unsigned long greenVoltTimer;
```

```

unsigned long greenCurrTimer;

void readLDR()
{
    ldrValue = analogRead(A0);
}

void readPZEM()
{
    voltage[0] = pzem1.voltage();
    current[0] = pzem1.current();
    power[0] = pzem1.power();
    energy[0] = pzem1.energy();
    frequency[0] = pzem1.frequency();
    pf[0] = pzem1.pf();

    voltage[1] = pzem2.voltage();
    current[1] = pzem2.current();
    power[1] = pzem2.power();
    energy[1] = pzem2.energy();
    frequency[1] = pzem2.frequency();
    pf[1] = pzem2.pf();

    voltage[2] = pzem3.voltage();
    current[2] = pzem3.current();
    power[2] = pzem3.power();
    energy[2] = pzem3.energy();
    frequency[2] = pzem3.frequency();
    pf[2] = pzem3.pf();

    for(int i=0;i<3;i++)
    {
        if(isnan(voltage[i])) voltage[i]=0;
        if(isnan(current[i])) current[i]=0;
        if(isnan(power[i])) power[i]=0;
        if(isnan(energy[i])) energy[i]=0;
        if(isnan(frequency[i])) frequency[i]=0;
        if(isnan(pf[i])) pf[i]=0;
    }
}

void updateStatus()
{
    //=====

```

```

// RED
//=====
if(voltage[0] < 200)
{
    if(redVoltTimer == 0)
        redVoltTimer = millis();

    if(millis() - redVoltTimer >= faultDelay)
        redStatus = "VOLT FAULT";
}
else
{
    redVoltTimer = 0;
    redStatus = "NORMAL";
}

if(current[0] < 0.01)
{
    if(redCurrTimer == 0)
        redCurrTimer = millis();

    if(millis() - redCurrTimer >= faultDelay)
        redStatus = "CURR FAULT";
}
else
{
    redCurrTimer = 0;
}

//=====
// YELLOW
//=====
if(voltage[1] < 200)
{
    if(yellowVoltTimer == 0)
        yellowVoltTimer = millis();

    if(millis() - yellowVoltTimer >= faultDelay)
        yellowStatus = "VOLT FAULT";
}
else
{
    yellowVoltTimer = 0;
    yellowStatus = "NORMAL";
}

```

```

if(current[1] < 0.01)
{
    if(yellowCurrTimer == 0)
        yellowCurrTimer = millis();

    if(millis() - yellowCurrTimer >= faultDelay)
        yellowStatus = "CURR FAULT";
}
else
{
    yellowCurrTimer = 0;
}

//=====
// GREEN
//=====
if(voltage[2] < 200)
{
    if(greenVoltTimer == 0)
        greenVoltTimer = millis();

    if(millis() - greenVoltTimer >= faultDelay)
        greenStatus = "VOLT FAULT";
}
else
{
    greenVoltTimer = 0;
    greenStatus = "NORMAL";
}

if(current[2] < 0.01)
{
    if(greenCurrTimer == 0)
        greenCurrTimer = millis();

    if(millis() - greenCurrTimer >= faultDelay)
        greenStatus = "CURR FAULT";
}
else
{
    greenCurrTimer = 0;
}
}

void sendToBlynk()
{

```

```

    Blynk.virtualWrite(V0, ldrValue);

    Blynk.virtualWrite(V1, pwmValue);

    Blynk.virtualWrite(V2, voltage[0]);
    Blynk.virtualWrite(V3, voltage[1]);
    Blynk.virtualWrite(V4, voltage[2]);

    Blynk.virtualWrite(V5, current[0]*1000);
    Blynk.virtualWrite(V6, current[1]*1000);
    Blynk.virtualWrite(V7, current[2]*1000);

    Blynk.virtualWrite(V8, power[0]);
    Blynk.virtualWrite(V9, power[1]);
    Blynk.virtualWrite(V10, power[2]);

    Blynk.virtualWrite(V11, frequency[0]);
    Blynk.virtualWrite(V12, frequency[1]);
    Blynk.virtualWrite(V13, frequency[2]);

    Blynk.virtualWrite(V14, pf[0]);
    Blynk.virtualWrite(V15, pf[1]);
    Blynk.virtualWrite(V16, pf[2]);

    Blynk.virtualWrite(V17, redStatus);
    Blynk.virtualWrite(V18, yellowStatus);
    Blynk.virtualWrite(V19, greenStatus);
}

void setup() {
    Serial.begin(9600);

    // PZEM
    pzemSerial1.begin(9600);
    pzemSerial2.begin(9600);

    // WiFi
    WiFi.begin(ssid, pass);

    unsigned long t = millis();

    while (WiFi.status() != WL_CONNECTED && millis() - t < 10000) {
        delay(100);
    }

    // Blynk

```

```

Blynk.config(BLYNK_AUTH_TOKEN);
Blynk.connect(5000);

// Dimmer
dimmer.begin(NORMAL_MODE, ON);
dimmer.setPower(100);

sensorTimer = millis();
blynkTimer = millis();
}

void updatePWM()
{
  if(ldrValue >= 801)
  {
    pwmValue = 60;
  }
  else if(ldrValue >= 501)
  {
    pwmValue = 80;
  }
  else
  {
    pwmValue = 100;
  }

  dimmer.setPower(pwmValue);
}

void loop() {

  Blynk.run();
  if (millis() - sensorTimer >= 1000) {

    sensorTimer = millis();
    readLDR();
    updatePWM();
    readPZEM();
    updateStatus();
  }

  if (millis() - blynkTimer >= 2000) {

    blynkTimer = millis();

    if (Blynk.connected()) {

```

```
        sendToBlynk();  
    }  
}  
}
```

