

# SISTEM *MONITORING* DAN PROTEKSI *OVERLOAD* DENGAN *LOAD SHEDDING* BERBASIS IOT PADA INSTALASI LISTRIK RUMAH TANGGA

Handi Riva<sup>1\*</sup>, Agus Setiawan<sup>2</sup>

<sup>1,2</sup>Universitas Pamulang; Jl. Witana Harja No. 18b, Pamulang, Kota Tangerang Selatan, Banten 15417; (021) 7412566

## Keywords:

*Internet of Things* (IoT);  
*Monitoring* Daya Listrik;  
Proteksi *Overload*;  
Prioritas Beban.

## Correspondent Email:

hanstech027@gmail.com

**Abstrak.** *Overload* pada instalasi listrik rumah tangga merupakan salah satu penyebab utama gangguan sistem kelistrikan dan potensi bahaya kebakaran akibat beban berlebih. Penelitian ini bertujuan merancang dan mengimplementasikan sistem *monitoring* dan proteksi *overload* berbasis *Internet of Things* (IoT) dengan mekanisme *load shedding* untuk mempertahankan beban prioritas. Sistem menggunakan mikrokontroler ESP32, sensor PZEM-004T sebagai pengukur parameter listrik, modul *relay* sebagai aktuator pemutus beban, serta integrasi aplikasi Blynk dan Telegram untuk *monitoring* dan notifikasi *real-time*. Hasil pengujian komparatif menunjukkan nilai galat rata-rata sebesar  $\pm 0,45\%$  untuk tegangan,  $\pm 0,43\%$  untuk arus, dan  $\pm 0,88\%$  untuk daya aktif dibandingkan pengukuran manual, sehingga berada di bawah batas toleransi  $\pm 5\%$ . Pengujian eksperimental menunjukkan sistem mampu mendeteksi kondisi *overload* pada ambang batas 400 W, memberikan peringatan dini selama 30 detik, dan berhasil melakukan *load shedding* beban *non-prioritas* secara otomatis dengan tingkat keberhasilan 100%. Waktu tunda pengiriman notifikasi Telegram tercatat rata-rata 1–2 detik. Berdasarkan hasil tersebut, sistem dinyatakan akurat, responsif, dan layak diterapkan sebagai proteksi tambahan pada instalasi listrik rumah tangga.



Copyright © [JITET](#) (Jurnal Informatika dan Teknik Elektro Terapan). This article is an open access article distributed under terms and conditions of the Creative Commons Attribution (CC BY NC)

**Abstract.** *Overload conditions in residential electrical installations are one of the primary causes of system disturbances and potential fire hazards due to excessive power consumption. This study proposes the design and implementation of an Internet of Things (IoT)-based monitoring and overload protection system incorporating a load shedding mechanism to maintain priority loads. The system utilizes an ESP32 microcontroller, a PZEM-004T sensor for electrical parameter measurement, relay modules for load control, and integrates Blynk and Telegram applications for real-time monitoring and notifications. Comparative testing results indicate average measurement errors of  $\pm 0.45\%$  for voltage,  $\pm 0.43\%$  for current, and  $\pm 0.88\%$  for active power when compared to manual measurements, all of which are within the acceptable  $\pm 5\%$  tolerance limit. Experimental evaluation demonstrates that the system successfully detects overload conditions at a 400 W threshold, provides a 30-second early warning, and performs automatic load shedding on non-priority loads with a 100% success rate. The average Telegram notification delay is recorded at 1–2 seconds. These results confirm that the proposed system is accurate, responsive, and suitable as an additional protection mechanism for residential electrical installations.*

## 1. PENDAHULUAN

Konsumsi energi listrik pada sektor rumah tangga dan industri kecil terus meningkat seiring bertambahnya penggunaan peralatan elektronik. Namun, banyak pengguna belum memahami batas daya maksimum instalasi listrik yang digunakan, sehingga berpotensi menimbulkan kondisi beban lebih (*overload*). *Overload* dapat menyebabkan gangguan operasional, kerusakan peralatan, pemadaman listrik, hingga risiko kebakaran. Selain itu, sebagian besar instalasi listrik konvensional belum dilengkapi sistem pemantauan dan proteksi otomatis, sehingga tidak mampu memberikan peringatan dini kepada pengguna saat terjadi gangguan kelistrikan [1]. Untuk mengatasi permasalahan tersebut, diperlukan sistem yang mampu memantau konsumsi daya secara *real-time* dan melakukan proteksi otomatis ketika terjadi *overload*. Penerapan mekanisme pelepasan beban berbasis prioritas memungkinkan beban *non-esensial* diputus terlebih dahulu tanpa menghentikan suplai ke beban penting, sehingga kontinuitas layanan listrik tetap terjaga sekaligus melindungi instalasi dari kerusakan akibat kelebihan beban [2]. Pemanfaatan teknologi *Internet of Things* (IoT) memungkinkan integrasi sistem *monitoring*, pengendalian otomatis, dan penyampaian informasi kepada pengguna secara cepat. Notifikasi berbasis Telegram dapat memberikan peringatan dini sehingga respons terhadap kondisi darurat dapat dilakukan lebih efektif [3]. Berdasarkan hal tersebut, penelitian ini bertujuan merancang sistem *monitoring* dan proteksi *overload* berbasis IoT yang mampu mengukur konsumsi daya secara akurat, menerapkan proteksi otomatis dengan konsep prioritas beban, serta memberikan peringatan dini melalui notifikasi Telegram [4]. Beberapa penelitian sebelumnya telah mengintegrasikan mikrokontroler, sensor arus-tegangan, serta *platform* IoT seperti Blynk dan Telegram untuk meningkatkan efisiensi dan keamanan sistem. Namun, sebagian penelitian masih berfokus pada pemantauan tanpa dilengkapi mekanisme proteksi otomatis dan prioritas beban, sehingga diperlukan pengembangan sistem yang lebih komprehensif dalam menjaga keandalan instalasi listrik rumah tangga [5].

## 2. TINJAUAN PUSTAKA

### 2.1. Konsep *overload* dalam Instalasi Listrik

*Overload* atau beban lebih terjadi ketika arus yang mengalir pada suatu rangkaian melebihi batas kapasitas konduktor atau perangkat. Hal ini dapat disebabkan oleh penambahan beban yang melebihi batas daya terpasang. Jika tidak segera ditangani, *overload* dapat mengakibatkan kerusakan peralatan, gangguan pasokan listrik, hingga bahaya kebakaran [6].

### 2.2. Konsep Prioritas Beban

Pada penelitian ini, sistem proteksi *overload* dirancang dengan pembagian beban prioritas dan *non-prioritas*. Beban prioritas adalah beban yang harus tetap menyala demi keberlangsungan fungsi utama rumah tangga, yaitu penerangan, kenyamanan dasar, dan komunikasi. Oleh karena itu, kategori beban prioritas terdiri atas lampu penerangan utama, kipas angin kecil, dan *router* WiFi/*charger* HP. Sebaliknya, beban *non-prioritas* adalah beban yang relatif lebih besar konsumsinya, bersifat tambahan, dan dapat diputus terlebih dahulu ketika terjadi *overload*. Kategori ini terdiri dari speaker (perangkat hiburan), *rice cooker* (peralatan dapur yang tidak mendesak untuk selalu aktif), serta lampu tambahan atau lampu hias [7].

Dengan konsep ini, sistem mampu menjaga agar kebutuhan vital tetap berjalan, sementara beban tambahan diputus otomatis untuk mencegah gangguan akibat kelebihan daya. Berikut pengkategorian yang saya buat dalam Tabel 2.1 :

Tabel 2. 1. Kategori Beban

No	Kategori	Jenis Beban	Perkiraan Daya (W)	Alasan Pemilihan
1	Prioritas	Lampu penerangan utama (LED)	10–15	Penting untuk penerangan rumah, harus selalu menyala.
2	Prioritas	<i>Rice cooker</i>	100–300	Konsumsi Daya Besar,

				Sebagai Pasokan Konsumsi
3	Prioritas	Charger HP / Router WiFi	5-10	Komunikasi & koneksi internet harus tetap tersedia.
4	Non-Prioritas	Speaker	10-20	Perangkat hiburan, tidak mendesak jika dimatikan.
5	Non-Prioritas	Kipas Angin kecil	10-20	Menjaga kenyamanan penghuni, khususnya di iklim tropis.
6	Non-Prioritas	Alat pengusir Nyamuk	5-10	Menjaga Kenyamanan Ruang dari Serangga

### 2.3. Internet Of Things (IoT)

IoT atau *Internet of Things* merupakan konsep teknologi yang memungkinkan perangkat fisik untuk saling terhubung melalui jaringan internet dan melakukan pertukaran data secara otomatis. Dalam bidang kelistrikan, IoT memungkinkan *real-time monitoring*, pemrosesan data di *cloud*, hingga pengendalian perangkat dari jarak jauh. Implementasi IoT menggunakan kombinasi mikrokontroler (seperti ESP32), sensor, dan *platform* komunikasi seperti Telegram atau Blynk [8].

Penggunaan IoT dalam sistem proteksi daya memberikan keuntungan seperti kemudahan pengawasan dari jarak jauh, pengiriman notifikasi instan, serta kemampuan analisis data historis guna

meningkatkan efisiensi dan keamanan penggunaan energi listrik [9].



Gambar 2. 1. *Internet Of Things* (IoT)

### 2.4. ESP 32

ESP32 merupakan mikrokontroler yang dilengkapi konektivitas WiFi dan *Bluetooth* dengan kemampuan pemrosesan yang baik serta konsumsi daya yang rendah. Pada sistem ini, ESP32 berfungsi sebagai pusat kendali yang mengatur pembacaan data sensor, pengendalian beban melalui *relay*, serta komunikasi data ke *platform* IoT dan Telegram [10]. Dalam penelitian ini, ESP32 berperan untuk menginisialisasi seluruh komponen, membaca data dari sensor PZEM-004T melalui komunikasi UART, memproses data untuk menentukan status proteksi, mengendalikan *relay* berdasarkan prioritas beban, serta mengirimkan data *monitoring* ke aplikasi Blynk dan notifikasi ke Telegram ketika terjadi *overload*.



Gambar 2. 2. ESP 32

### 2.5. PZEM 004T

PZEM-004T merupakan sensor yang digunakan untuk mengukur parameter kelistrikan seperti tegangan, arus, daya aktif, faktor daya, dan energi listrik. Sensor ini berkomunikasi dengan ESP32 melalui antarmuka serial UART dan digunakan untuk pemantauan kondisi listrik secara *real-time*. Dalam sistem ini, PZEM-004T berperan

sebagai komponen utama dalam mendeteksi kondisi *overload* [11].



Gambar 2. 3. Sensor PZEM 004T

### 2.6. Relay 2 Channel

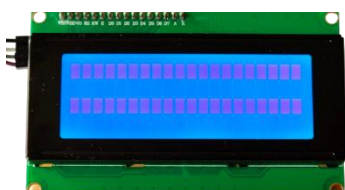
*Relay 2 channel* merupakan modul saklar elektronik yang digunakan untuk mengendalikan dua beban listrik secara terpisah menggunakan sinyal logika dari mikrokontroler. Modul ini umum digunakan pada sistem otomasi dan proteksi listrik karena mampu mengontrol beban AC bertegangan tinggi dengan sinyal kendali berarus kecil. *Relay* yang digunakan bersifat aktif rendah, di mana *relay* akan aktif saat pin *input* diberi logika rendah (*Low*) [12].



Gambar 2. 4. Relay 2 Channel

### 2.7. LCD I2C 20x4

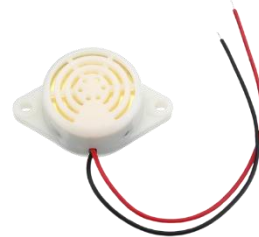
LCD I2C 20×4 digunakan sebagai media tampilan lokal untuk menampilkan informasi kelistrikan seperti tegangan, arus, daya, dan status sistem. Penggunaan antarmuka I2C memungkinkan komunikasi dengan mikrokontroler hanya menggunakan dua jalur data (SDA dan SCL), sehingga lebih menghemat penggunaan pin GPIO. Modul ini bekerja pada tegangan 5V DC dan kompatibel dengan ESP32 maupun Arduino [13].



Gambar 2. 5. LCD I2C 20x4

### 2.8. Buzzer 5 Volt

*Buzzer 5V* merupakan aktuator audio yang digunakan sebagai indikator peringatan pada sistem. Pada penelitian ini, *Buzzer* berfungsi sebagai alarm *overload* yang aktif ketika daya melebihi batas yang ditentukan. *Buzzer* yang digunakan merupakan jenis *Buzzer* aktif, sehingga dapat menghasilkan suara tanpa rangkaian tambahan karena telah dilengkapi osilator internal [14].



Gambar 2. 6. Buzzer 5 VDC

### 2.9. Power supply 5 VDC

*Power supply 5V DC* digunakan sebagai sumber tegangan utama bagi ESP32, sensor PZEM-004T, dan modul *relay*. Catu daya ini dipilih untuk memastikan suplai tegangan yang stabil sehingga seluruh sistem dapat bekerja dengan baik. Tegangan keluaran 5V sesuai dengan kebutuhan modul mikrokontroler dan *relay*, serta aman digunakan untuk rangkaian elektronika berskala kecil [15].



Gambar 2. 7. Power supply 5 VDC

### 2.10. Telegram Bot

Telegram Bot merupakan layanan API dari aplikasi Telegram yang memungkinkan pengiriman pesan otomatis dari sistem ke pengguna. Pada sistem ini, ESP32 dikonfigurasi untuk mengirimkan notifikasi melalui Telegram ketika terjadi kondisi *overload*, pemutusan beban *non-prioritas*, serta saat sistem kembali ke kondisi normal. Fitur ini berfungsi sebagai peringatan dini agar pengguna dapat segera mengetahui kondisi kelistrikan [16].



Gambar 2. 8. *Software Telegram*

### 2.11. Platform Blynk

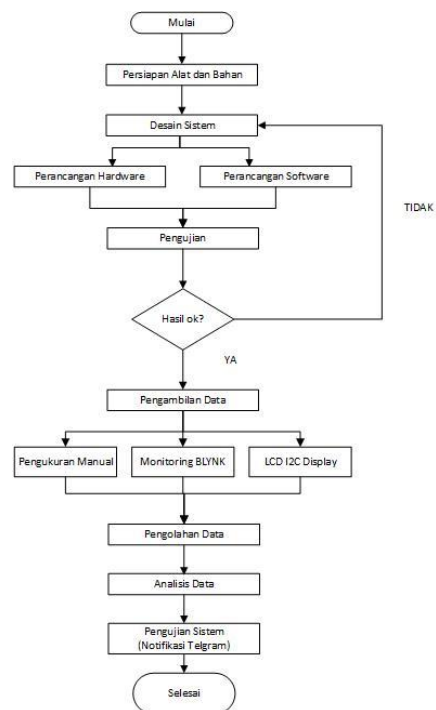
Blynk merupakan *platform* IoT berbasis *cloud* yang menyediakan antarmuka visual untuk *monitoring* perangkat melalui internet. Melalui Blynk, pengguna dapat memantau parameter listrik dan status beban secara *real-time* menggunakan smartphone. Data dari ESP32 dikirim ke *server* Blynk dan ditampilkan dalam bentuk nilai maupun grafik, sehingga memudahkan pengguna dalam memantau kondisi sistem dari jarak jauh [17].



Gambar 2. 9. *Software Blynk*

## 3. METODE PENELITIAN

### 3.1. Flowchart Penelitian

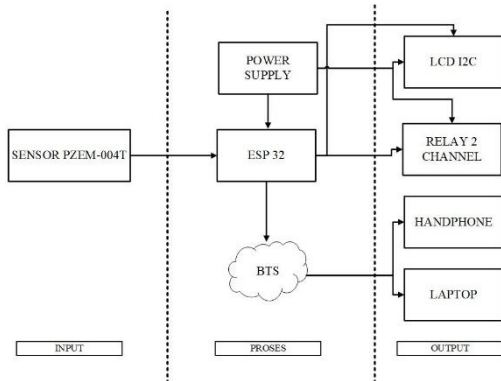


Gambar 3. 1. *Flowchart Penelitian*

Alur penelitian disusun untuk menjelaskan tahapan pengembangan sistem *monitoring* dan proteksi *overload* berbasis IoT secara berurutan, mulai dari persiapan hingga pengujian akhir. Alur ini disajikan dalam bentuk *flowchart* agar proses penelitian lebih mudah dipahami. Penelitian diawali dengan persiapan alat dan bahan yang terdiri dari perangkat keras seperti ESP32, sensor PZEM-004T, *relay*, LCD I2C, dan *power supply*, serta perangkat lunak berupa Arduino IDE, aplikasi Blynk, dan Telegram. Tahap selanjutnya adalah perancangan sistem yang mencakup desain perangkat keras dan perangkat lunak sesuai kebutuhan *monitoring* dan proteksi *overload* dengan konsep prioritas beban. Setelah tahap perancangan, sistem diimplementasikan melalui perakitan perangkat keras ke dalam *box panel* dan pemrograman ESP32. Pengujian awal kemudian dilakukan untuk memastikan seluruh komponen bekerja dengan baik. Jika ditemukan kendala, dilakukan perbaikan hingga sistem berfungsi secara optimal. Setelah sistem stabil, dilakukan pengambilan data melalui pengukuran manual, tampilan LCD, dan aplikasi Blynk. Data tersebut selanjutnya dianalisis untuk menilai akurasi pengukuran, kinerja proteksi *overload*, dan

respon notifikasi. Tahap akhir penelitian adalah pengujian menyeluruh dengan simulasi kondisi *overload* serta penyusunan laporan.

### 3.2. Diagram Blok Sistem



Gambar 3. 2. Blok Diagram Sistem

Diagram blok sistem menunjukkan cara kerja sistem yang terdiri dari tiga bagian utama, yaitu *input*, *proses*, dan *output*. Bagian *input* berupa sensor PZEM-004T yang berfungsi mengukur parameter listrik seperti tegangan, arus, daya, energi, dan faktor daya. Data hasil pengukuran kemudian dikirimkan ke mikrokontroler sebagai masukan sistem. Bagian *proses* dikendalikan oleh ESP32 yang berfungsi mengolah data sensor, membandingkan nilai daya dengan batas yang telah ditentukan, serta menjalankan logika proteksi *overload*. Sistem mendapat suplai dari catu daya 5V DC agar seluruh modul dapat bekerja dengan stabil. ESP32 juga terhubung ke internet untuk mengirimkan data *monitoring* dan notifikasi. Bagian *output* meliputi LCD I2C sebagai tampilan lokal, *relay 2 channel* untuk memutus beban prioritas dan *non-prioritas*, serta aplikasi Blynk dan Telegram sebagai media *monitoring* jarak jauh dan pengiriman notifikasi peringatan.

### 3.3. Daya Listrik

Pada sistem arus bolak-balik (AC) satu fasa yang umum digunakan pada instalasi listrik rumah tangga, perhitungan daya listrik dipengaruhi oleh faktor daya (*power factor*) yang menunjukkan tingkat efisiensi pemanfaatan energi listrik. Daya aktif, yaitu daya yang benar-benar digunakan oleh

beban, dihitung menggunakan persamaan (1):

$$P = V \cdot I \cdot \cos \varphi \quad (1)$$

dengan  $P$  sebagai daya aktif (W),  $V$  sebagai tegangan (V),  $I$  sebagai arus (A), dan  $\cos \varphi$  sebagai faktor daya.

#### 3.3.1. Perhitungan Selisih Pengukuran

Selisih pengukuran digunakan untuk mengetahui perbedaan antara hasil pengukuran sistem *monitoring* dan pengukuran manual sebagai nilai acuan. Perhitungan ini bertujuan untuk mengidentifikasi tingkat penyimpangan data yang dihasilkan oleh sistem berbasis sensor. Selisih pengukuran dihitung menggunakan persamaan (2):

$$\Delta X = X_{alat} - X_{manual} \quad (2)$$

dengan  $\Delta X$  sebagai selisih pengukuran,  $X_{alat}$  sebagai nilai hasil pengukuran sistem (LCD atau Blynk), dan  $X_{manual}$  sebagai nilai hasil pengukuran manual.

#### 3.3.2. Perhitungan Galat Pengukuran

Galat pengukuran digunakan untuk menyatakan tingkat kesalahan relatif hasil pengukuran sistem terhadap nilai acuan dalam bentuk persentase, sehingga memudahkan evaluasi akurasi sistem. Perhitungan galat dilakukan menggunakan persamaan (3) :

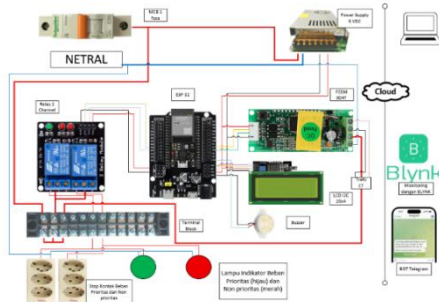
$$Error \% = \frac{(X_{alat} - X_{manual})}{X_{manual}} \times 100 \% \quad (3)$$

dengan  $X_{alat}$  sebagai nilai hasil pengukuran sistem dan  $X_{manual}$  sebagai nilai hasil pengukuran manual.

### 3.4. Perancangan Hardware

Perancangan *hardware* bertujuan membangun sistem *monitoring* dan proteksi *overload* berbasis IoT yang terintegrasi. Seluruh komponen dirangkai berdasarkan *wiring diagram* yang telah dirancang sebelumnya. Sumber listrik AC 220 V dihubungkan ke sensor PZEM-004T untuk pengukuran tegangan dan arus. Data hasil pengukuran dikirimkan ke ESP32 melalui

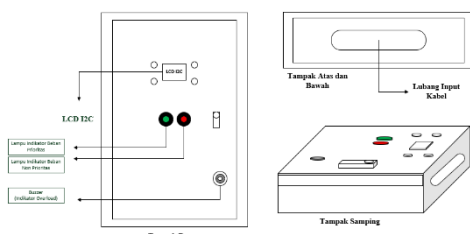
komunikasi serial. Selanjutnya, ESP32 mengendalikan *relay* prioritas dan *non-prioritas*, *Buzzer* sebagai alarm, serta LCD I2C sebagai media tampilan. Sistem juga terhubung ke aplikasi Blynk dan Telegram untuk *monitoring* dan notifikasi.



Gambar 3. 3. Wiring Diagram Sistem

### 3.4.1. Desain Alat

Seluruh rangkaian ditempatkan di dalam box panel berukuran 40 × 30 × 18 cm yang berfungsi sebagai pelindung sekaligus penataan sistem. LCD I2C dipasang pada bagian depan panel untuk menampilkan parameter listrik dan status sistem secara *real-time*. Lampu indikator hijau dan merah digunakan untuk menunjukkan kondisi beban prioritas dan *non-prioritas*. *Buzzer* 5V dipasang sebagai alarm suara yang aktif selama 30 detik ketika terjadi *overload*. Jalur kabel dirapikan menggunakan kabel *duct* untuk meningkatkan keamanan dan kerapian instalasi. Desain panel ini memudahkan pemantauan, aman digunakan, dan mudah dalam perawatan.



Gambar 3. 4. Desain Panel

### 3.5. Perancangan Software

Perancangan *Software* pada penelitian ini dilakukan menggunakan Arduino IDE sebagai *platform* pemrograman utama untuk mikrokontroler ESP32. Tahap ini bertujuan untuk membuat logika program

yang mengatur seluruh fungsi sistem, mulai dari pembacaan sensor, pengiriman data IoT, hingga pengendalian proteksi *overload*.

```

1 #define BLYNK_TEMPLATE_ID "TMPL6WZV5_0Ry"
2 #define BLYNK_TEMPLATE_NAME "SKRIPSI RANCANGAN IOT HANDI RIVA"
3 #define BLYNK_AUTH_TOKEN "o2F-CPaXSP9xzHEn01WlvIA1j6abmN2K"
4
5 #include <WiFi.h>
6 #include <WiFiClientSecure.h>
7 #include <UniversalTelegramBot.h>
8 #include <BlynkSimpleEsp32.h>
9
10 #include <Wire.h>
11 #include <LiquidCrystal_I2C.h>
12 #include <PZEM004Tv30.h>
13 #include <Preferences.h>
14 #include <time.h>
    
```

Gambar 3. 5. Inisialisasi Library Token Blynk pada Software IDE

*Monitoring* melalui Blynk IoT merupakan bagian penting dari sistem ini karena memungkinkan pengguna untuk melihat kondisi listrik secara *real-time* melalui *smartphone*.



Gambar 3. 6. Tampilan Dashboard Blynk pada Ponsel

## 4. HASIL DAN PEMBAHASAN

### 4.1. Hasil Produk

Sistem *Monitoring* dan proteksi *overload* berbasis IoT ini dirancang dengan mengintegrasikan perangkat keras (*hardware*) dan perangkat lunak (*Software*) dalam satu kesatuan fungsional. Pada sisi *hardware*, sistem menggunakan sensor PZEM-004T sebagai akuisisi data listrik, ESP32 sebagai pusat kendali, *relay* 2 channel sebagai aktuator pemutus beban, LCD I2C 20×4 sebagai tampilan lokal, *Buzzer* sebagai peringatan audio, serta panel indikator merah dan hijau sebagai indikator status beban prioritas maupun *non-prioritas*.



Gambar 4. 1. Hasil produk

#### 4.2. Penyajian Data

Data pengukuran diperoleh melalui pengukuran manual, tampilan LCD I2C, dan aplikasi Blynk. Data tersebut diolah untuk menentukan nilai rata-rata, selisih, dan *error* pengukuran. Pengolahan data bertujuan mengevaluasi akurasi sensor PZEM-004T dibandingkan alat ukur acuan. Parameter yang dianalisis meliputi tegangan, arus, dan daya, yang menjadi dasar penilaian kelayakan sistem *monitoring* dan proteksi *overload* pada instalasi rumah tangga.

##### 4.2.1. Selisih Nilai Daya

Hasil pengukuran daya menggunakan LCD I2C dan aplikasi Blynk menunjukkan nilai rata-rata sebesar 378,8 W. Selisih rata-rata antara hasil pengukuran manual dan sistem *monitoring* adalah sebesar 3,3 W. Selisih ini tergolong kecil jika dibandingkan dengan total daya beban dan masih berada dalam batas toleransi *error* yang dapat diterima.

Tabel 4. 1. Selisih Nilai Daya

No	Manual (W)	LC D I2C (W)	Blynk (W)	Selisih (LCD - Manual)	Selisih (Blynk - Manual)
1	372,8	377,0	377,0	4,2	4,2
2	373,6	374,4	374,4	0,8	0,8
3	374,8	379,2	379,2	4,4	4,4

4	378,8	381,1	381,1	2,3	2,3
5	377,5	380,4	380,4	2,9	2,9
6	374,0	377,8	377,8	3,8	3,8
7	375,6	380,7	380,7	5,1	5,1
8	374,1	377,4	377,4	3,3	3,3
9	379,6	379,9	379,9	0,3	0,3
10	374,0	380,0	380,0	6	6
<b>Rata-rata</b>	<b>375,5</b>	<b>378,8</b>	<b>378,8</b>	<b>3,3</b>	<b>3,3</b>

Grafik perbandingan nilai daya menunjukkan bahwa daya aktif yang ditampilkan oleh LCD I2C dan aplikasi Blynk secara umum lebih tinggi dibandingkan hasil perhitungan manual. Pola grafik menunjukkan konsistensi tren kenaikan dan penurunan daya pada seluruh percobaan, meskipun terdapat selisih pada setiap titik pengukuran. Perbedaan tersebut dipengaruhi oleh metode perhitungan daya aktif pada sistem yang dilakukan secara langsung oleh sensor PZEM-004T.

Gambar 4. 2. Grafik Selisih Nilai Daya



**4.2.2. Galat Daya pada Alat Ukur**

Berdasarkan hasil perhitungan ulang daya aktif manual yang telah disesuaikan menggunakan faktor daya ( $\cos \phi$ ) hasil pembacaan sensor PZEM-004T, diperoleh nilai galat daya antara pengukuran manual dan sistem (LCD I2C serta Blynk) pada rentang 0,08% hingga 1,60%. Rata-rata galat daya tercatat sebesar 0,88%, yang menunjukkan bahwa perbedaan antara daya aktif manual dan daya aktif sistem masih relatif kecil dibandingkan nilai daya yang diukur (sekitar 375–380 W). Nilai galat terbesar terjadi pada percobaan ke-10 sebesar 1,60%, sedangkan nilai galat terkecil terjadi pada percobaan ke-9 sebesar 0,08%. Perbedaan ini dipengaruhi oleh beberapa faktor seperti fluktuasi tegangan dan arus selama proses pengukuran, perbedaan waktu sampling antara alat ukur manual dan sistem digital, dan karakteristik perhitungan internal daya aktif pada sensor PZEM-004T. Seluruh nilai galat masih berada jauh di bawah batas toleransi  $\pm 5\%$  yang umum digunakan dalam evaluasi sistem pengukuran instalasi listrik satu fasa.

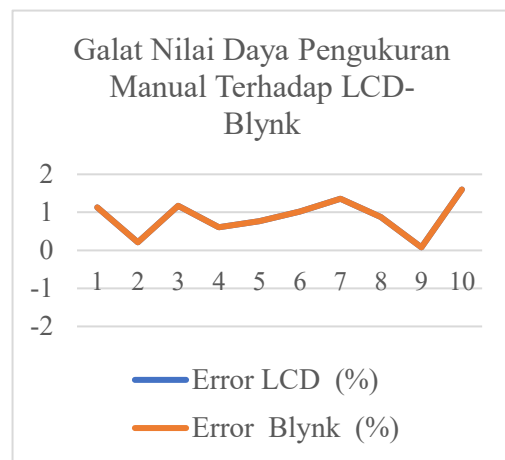
Tabel 4. 2. Galat Daya pada Alat Ukur

No	Manual (V)	LCD I2C (V)	Blynk (V)	Error LCD (%)	Error Blynk (%)
1	372,8	377,0	377,0	1,13	1,13
2	373,6	374,4	374,4	0,21	0,21
3	374,8	379,2	379,2	1,17	1,17
4	378,8	381,1	381,1	0,61	0,61
5	377,5	380,4	380,4	0,77	0,77
6	374,0	377,8	377,8	1,02	1,02
7	375,6	380,7	380,7	1,36	1,36

8	374,1	377,4	377,4	0,88	0,88
9	379,6	379,9	379,9	0,08	0,08
10	374,0	380,0	380,0	1,60	1,60
<b>Rata-rata</b>	<b>375,5</b>	<b>378,8</b>	<b>378,8</b>	<b>0,88</b>	<b>0,88</b>

Meskipun terdapat variasi nilai galat pada setiap percobaan, seluruh nilai galat masih berada dalam batas toleransi yang dapat diterima untuk sistem *monitoring* satu fasa. Nilai rata-rata galat sebesar 0,88% menunjukkan bahwa sistem *monitoring* berbasis IoT yang dirancang memiliki tingkat akurasi yang baik serta konsisten terhadap pengukuran manual.

Gambar 4. 3. Grafik Galat Daya Terhadap Alat ukur



**4.3. Pengujian Sistem**

Pengujian sistem meliputi uji *overload*, uji cabut beban, uji notifikasi Telegram, dan uji proteksi *relay*. Hasil pengujian menunjukkan bahwa sistem mampu mendeteksi *overload* secara konsisten pada *threshold* 400 W, memberikan peringatan dini, serta memutus beban *non-prioritas* tepat setelah waktu tunda 30 detik. Sistem juga berhasil membatalkan proteksi ketika beban dikurangi sebelum waktu tunda berakhir, serta mengirimkan notifikasi Telegram dengan *delay* 0–1 detik. Seluruh pengujian

menunjukkan tingkat keberhasilan proteksi sebesar 100%.

Tabel 4. 3. Pengujian *Overload*

Trial	Total Daya Saat <i>Overload</i> (W)	Thres hold (W)	T start <i>Overload</i>	T relay Trip (Non-Pri OFF)	Trip Delay (s)
1	456.1 W	400	02:33 :58	02:34 :28	0
2	483.2 W	400	02:35 :09	02:35 :39	0
3	404.1 W	400	02:36 :12	02:36 :42	0
4	420.2 W	400	02:37 :14	02:37 :44	0
5	401.1 W	400	02:39 :02	02:39 :32	0
6	401.1 W	400	02:40 :12	02:40 :42	0
7	400.1 W	400	02:41 :17	02:41 :47	0
8	449.8 W	400	02:42 :26	02:42 :56	0
9	407.5 W	400	02:43 :24	02:43 :54	0
10	444.6 W	400	02:44 :23	02:44 :53	0

Tabel 4. 4. Pengujian Uji Cabut Beban

Trial uji cabut beban	Total Daya Saat Mengirim Notif <i>Overload</i> ( )	Total Daya Saat Beban Dicabut ( )	T start <i>overload</i>	T Reco ver normal	Durasi <i>Overload</i> (s)
1	449.4	395.2	02:4 5:38	02:4 5:47	30

2	453.7	399.6	02:4 6:26	02:4 6:36	30
3	451.4	394.3	02:4 6:56	02:4 7:08	30
4	449.2	397.3	02:4 7:28	02:4 7:38	30
5	407.9	396.6	02:4 7:52	02:4 8:06	30
6	428.1	396.2	02:4 8:26	02:4 8:41	30
7	408.0	399.2	02:4 8:54	02:4 9:04	30
8	400.9	393.7	02:4 9:20	02:4 9:35	30
9	426.5	395.7	02:4 9:54	02:5 0:08	30
10	438.4	394.8	02:5 0:26	02:5 0:36	30

Tabel 4. 5. Pengujian Notifikasi telegram

Trial notifikasi telegram	Event Type (Warning/Recover/Proteksi)	T notif sent	T notif received	Delay (s)
1	Warning (Overload)	02:3 3:58	02:3 3:59	1
2	Warning (Overload)	02:3 5:09	02:3 5:10	1
3	Warning (Overload)	02:3 6:12	02:3 6:13	1
4	Warning (Overload)	02:3 7:14	02:3 7:15	1
5	Recover (Daya kembali Normal)	02:4 5:47	02:4 5:48	1
6	Recover (Daya kembali Normal)	02:4 6:36	02:4 6:37	1
7	Recover (Daya kembali Normal)	02:4 7:08	02:4 7:09	1

8	Proteksi ( <i>Relay Non-Pri OFF</i> )	02:3 4:28	02:3 4:28	0
9	Proteksi ( <i>Relay Non-Pri OFF</i> )	02:3 5:39	02:3 5:39	0
10	Proteksi ( <i>Relay Non-Pri OFF</i> )	02:3 6:42	02:3 6:42	0

Tabel 4. 6. Uji Proteksi

Uji Pro teksi	Kon disi Prot eksi	Day a Saat Ove rloa d (W)	W akt u Tri p (s)	De la y Tr ip (s)	Stat us Rela y Prio rita s	Stat us Rela y Non Prio rita s
1		408. 4 W		0	ON	OFF
2		404. 5 W		0	ON	OFF
3		406. 1 W		0	ON	OFF
4		403. 1 W		0	ON	OFF
5	<i>Ove rloa d &gt;40 0 W</i>	410. 7 W	30 (s)	0	ON	OFF
6		411. 7 W		0	ON	OFF
7		406. 1 W		0	ON	OFF
8		406. 9 W		0	ON	OFF
9		406. 4 W		0	ON	OFF
10		403. 3 W		0	ON	OFF

Tabel 4. 7. Ringkasan Performa

N o	Paramete r	Nilai Rata -rata	Erro r (%)	Keteranga n
1	Tegangan	234, 5	0,45	Memenuhi PUIL ±5%

2	Arus	1,62	0,45	Akurasi baik (kelas 1-2)
3	Daya	378, 8	0,88	Akurasi baik
4	<i>Delay Notifikasi Telegram</i>	1-2 detik		Respons cepat
5	Waktu Proteksi <i>Trip</i>	30 detik		Proteksi bekerja normal
6	Akurasi Proteksi	100 %		<i>Recovery</i> berhasil

## 5. KESIMPULAN

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan, sistem *monitoring* dan proteksi *overload* berbasis *Internet of Things* (IoT) pada instalasi listrik rumah tangga berhasil dikembangkan sesuai dengan tujuan penelitian. Sistem ini dirancang untuk memantau parameter kelistrikan secara *real-time*, memberikan peringatan dini, serta melakukan proteksi otomatis dengan konsep prioritas beban guna meningkatkan keamanan dan keandalan instalasi listrik rumah tangga. Adapun kesimpulan yang diperoleh dari penelitian ini adalah sebagai berikut :

- a. Sistem *monitoring* dan proteksi *overload* berbasis IoT berhasil direalisasikan menggunakan mikrokontroler ESP32, sensor PZEM-004T, *relay*, LCD I2C, *Buzzer*, serta integrasi aplikasi Blynk dan Telegram Bot sebagai media *monitoring* dan notifikasi.
- b. Sistem mampu memantau parameter kelistrikan berupa tegangan, arus, daya, energi listrik, faktor daya, dan frekuensi secara *real-time* baik melalui tampilan lokal maupun *monitoring* jarak jauh.
- c. Hasil pengujian akurasi menunjukkan bahwa sistem memiliki rata-rata galat pengukuran sebesar 0,45% untuk tegangan, 0,45% untuk arus, dan 0,88% untuk daya, di mana seluruh nilai tersebut berada di bawah batas toleransi

yang ditetapkan oleh Persyaratan Umum Instalasi Listrik (PUIL).

- d. Mekanisme proteksi *overload* berjalan sesuai dengan logika yang dirancang, yaitu memberikan peringatan dini selama 30 detik ketika daya melebihi ambang batas 400 W, disertai aktivasi *Buzzer* dan pengiriman notifikasi melalui Telegram serta tampilan status pada LCD dan aplikasi Blynk.
- e. Apabila beban tidak dikurangi dalam waktu peringatan, sistem secara otomatis melakukan *load shedding* dengan memutus beban *non-prioritas*, sementara beban *prioritas* tetap beroperasi. Hasil pengujian menunjukkan bahwa mekanisme proteksi ini berjalan konsisten dengan tingkat keberhasilan sebesar 100%.
- f. Sistem mampu merespons secara adaptif ketika beban dikurangi sebelum waktu proteksi berakhir, ditandai dengan kembalinya sistem ke kondisi normal, penghentian *Buzzer*, tetap aktifnya *relay non-prioritas*, serta pengiriman notifikasi pemulihan melalui Telegram.
- g. Integrasi aplikasi Blynk dan Telegram Bot berjalan dengan baik, di mana Blynk mampu menampilkan data *monitoring* dalam bentuk nilai dan grafik secara *real-time*, sedangkan Telegram Bot berhasil mengirimkan notifikasi peringatan, proteksi, pemulihan, serta laporan pemantauan berkala dengan waktu tunda rata-rata 1–2 detik.

#### DAFTAR PUSTAKA

- [1] F. T. Saroha and Y. Prabowo, "PENGEMBANGAN SISTEM OTOMATIS DAN KEAMANAN RUMAH BERBASIS *INTERNET OF THINGS* ( IOT ) Development of automated systems and home security based on the *Internet of Things* ( IOT )," vol. 3, no. September, pp. 1011–1020, 2024.
- [2] M. Z. Hasan and E. Junianto, "Sistem *Monitoring* dan Kontrol Peralatan Listrik Berbasis IoT Menggunakan Aplikasi Blynk," *eProsiding Tek. Inform. ...*, vol. 4, no. 2, pp. 401–413, 2023, [Online]. Available: <http://eprosiding.ars.ac.id/index.php/pti/article/view/1075>
- [3] S. Ma'shumah, E. K. Pramartaningthyas, and A. G. Rokhim, "Implementasi *Internet of Things* (IoT) Pada Sistem *Monitoring* Dan Notifikasi Pemakaian Listrik Rumah Tangga Berbasis Aplikasi Blynk," *Power Elektron. J. Orang Elektro*, vol. 12, no. 3, pp. 144–149, 2024, doi: 10.30591/polekro.v12i3.5282.
- [4] A. Setiawan, I. Istiadi, and G. Priyandoko, "Pengendali Dan Pemantau Arus Tegangan Pada Terminal Listrik Rumah Tangga Berbasis IoT," *JOINTECS (Journal Inf. Technol. Comput. Sci.)*, vol. 8, no. 1, p. 27, 2023, doi: 10.31328/jointecs.v8i1.4633.
- [5] J. W. Simatupang and M. Aziz, "Portable *Real-time Monitoring* System for PT PLN ULP Cikarang Distribution Substation Based on IoT and PZM 004T Sensor," *J. Energy Electr. Eng.*, vol. 6, no. 1, 2024, [Online]. Available: <https://jurnal.unsil.ac.id/index.php/jeece/article/view/12758>
- [6] S. N. Indonesia and B. S. Nasional, "Persyaratan Umum Instalasi Listrik 2011," vol. 2011, no. PUIL, 2011.
- [7] M. Z. Z. Muhtadi, S. Ranolat, and Pujiyanto, "Sistem *Monitoring* Energi Listrik 3 Fase Berbasis IoT dan Firebase," *Elposys J. Sist. Kelistrikan*, vol. 12, no. 2, pp. 132–137, 2025, doi: 10.33795/elposys.v12i2.7206.
- [8] M. A. Ulum and S. I. Haryudo, "Perancangan Sistem *Monitoring* Kecepatan Putar Motor DC Berbasis *Internet of Things* Menggunakan Aplikasi BLYNK," *J. Tek. Elektro*, vol. 9, no. 1, pp. 855–862, 2020.
- [9] M. F. Budairi and H. Istiqlaliyah, "Analisis Efisiensi Kebutuhan Daya Listrik Pada Alat Penggoreng Keripik Buah Serbaguna Dengan Sistem Vacuum Frying," *Semin. Nas. Inov. Teknol.*, vol. 1, pp. 56–61, 2021.
- [10] M. F. Pela and R. Pramudita, "Sistem *Monitoring* Penggunaan Daya Listrik Berbasis *Internet of Things* Pada Rumah Dengan Menggunakan Aplikasi Blynk," *Infotech J. Technol. Inf.*, vol. 7, no. 1, pp. 47–54, 2021, doi: 10.37365/jti.v7i1.106.
- [11] M. A. Yahya, A. B. Yunanda, M. Sidqon, A. Kridoyono, T. Informatika, and J. Timur, "DENGAN NOTIFIKASI LOKASI MELALUI TELEGRAM," vol. 13, no. 3, pp. 229–240, 2024.
- [12] D. Biometrik and D. A. N. K. Kunci, "RANCANG BANGUN KEAMANAN SEPEDA MOTOR IOT," vol. 13, no. 3, pp. 241–254, 1945.
- [13] A. Sirajudin *et al.*, "OTOMASI PEMETAAN PERANGKAT ACTIVE DAN PASSIVE NETWORK PADA," vol. 13, no. 3.

- [14] A. Winarno, A. M. Ibad, and D. T. Wahyudi, "PROTOTIPE ALAT PERINGATAN DAN PINTU AIR MIKROKONTROLER ESP32 BERBASIS *INTERNET OF THINGS* ( IoT )," vol. 13, no. 3.
- [15] M. Somantri, M. R. Fauzan, and I. Surya, "Optimization of IoT-based *monitoring* system for automatic *power factor* correction using PZEM-004T sensor," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 39, no. 2, p. 860, 2025, doi: 10.11591/ijeecs.v39.i2.pp860-873.
- [16] M. F. Evizal, Muhammad Inas Farras Tsamarah, and Daniel Rosadi, "Design and Development of a Human Blood Type Detection System Based on the *Internet of Things*," *Int. J. Sci. Res.*, vol. 4, no. 2, pp. 49–55, 2025, doi: 10.25299/ijsr.2024.22271.
- [17] M. I. Rosli and M. R. Ahmad, "*Internet of Things Monitoring* System of a Modeled Cleanroom," *Elektr. J. Electr. Eng.*, vol. 20, no. 3, pp. 7–14, 2021, doi: 10.11113/elektrika.v20n3.283.