

DESAIN DAN IMPLEMENTASI SISTEM MONITORING DAN KONTROL TERINTEGRASI UNTUK MANAJEMEN SMART GRID BERBASIS IOT

SKRIPSI

Oleh :

BRAIN RIZKY HARRIS TARIGAN

218120029



**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS MEDAN AREA MEDAN
2026**

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

Document Accepted 4/8/26

1. Dilarang Mengutip sebagian atau seluruh dokumen ini tanpa mencantumkan sumber
2. Pengutipan hanya untuk keperluan pendidikan, penelitian dan penulisan karya ilmiah
3. Dilarang memperbanyak sebagian atau seluruh karya ini dalam bentuk apapun tanpa izin Universitas Medan Area

Access From (repositori.uma.ac.id)4/8/26

DESAIN DAN IMPLEMENTASI SISTEM MONITORING DAN KONTROL TERINTEGRASI UNTUK MANAJEMEN SMART GRID BERBASIS IOT

SKRIPSI

Diajukan sebagai Salah Satu Syarat untuk Memperoleh
Gelar Sarjana di Fakultas Teknik
Universitas Medan Area

Oleh :

BRAIN RIZKY HARRIS TARIGAN

218120029



**PROGRAM STUDI TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS MEDAN AREA MEDAN
2026**

LEMBAR PENGESAHAN

Judul Skripsi : Desain Dan Implementasi Sistem Monitoring Dan Kontrol Terintegrasi Untuk Manajemen Smart Grid Berbasis IOT

Nama : Brain Rizky Harris Tarigan

NPM : 21.812.0029

Fakultas : Teknik Elektro



Tanggal Lulus: 04 Maret 2026

UNIVERSITAS MEDAN AREA

© Hak Cipta Di Lindungi Undang-Undang

HALAMAN PERNYATAAN

Saya menyatakan bahwa skripsi yang saya susun, sebagai syarat memperoleh gelar sarjana merupakan hasil karya tulis saya sendiri. Adapun bagian-bagian tertentu dalam penulisan skripsi ini yang saya kutip dari hasil karya orang lain telah dituliskan sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan ilmiah.

Saya bersedia menerima sanksi pencabutan gelar akademik yang saya peroleh dan sanksi- sanksi lainnya dengan peraturan yang berlaku, apabila di kemudian hari ditemukan adanya plagiat dalam skripsi ini.



Medan, 26 Januari 2026

Brain Rizky Harris Tarigan
218120029

HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR/SKRIPSI/TESIS UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademik Universitas Medan Area, saya yang bertanda tangan di bawah ini:

Nama : Brain Rizky Harris Tarigan
NPM : 218120029
Program Studi : Teknik Elektro
Fakultas : Teknik
Jenis karya : Tugas Akhir/Skripsi

demikian pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Medan Area Hak Bebas Royalti Noneksklusif (*Non-exclusive Royalty-Free Right*) atas karya ilmiah saya yang berjudul :

“Desain Dan Implementasi Sistem Monitoring Dan Kontrol Terintegrasi Untuk Manajemen Smart Grid Berbasis IOT”

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Medan Area berhak menyimpan, mengalihmedia/format-kan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir/skripsi/tesis saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Medan

Pada tanggal : 26 Januari 2026

Yang menyatakan



(Brain Rizky Harris Tarigan)

ABSTRAK

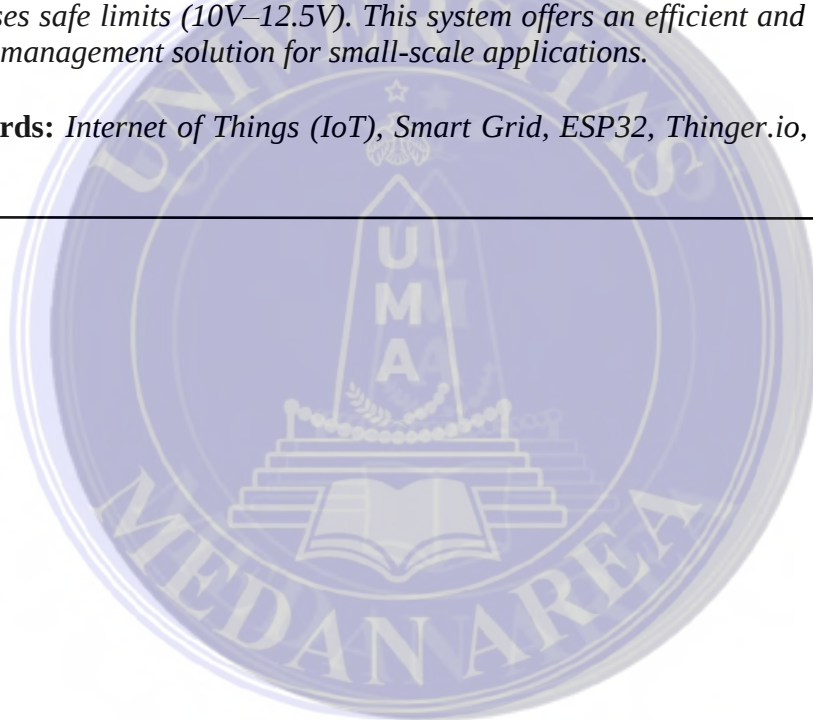
Sistem kelistrikan konvensional menghadapi tantangan dalam hal efisiensi, komunikasi satu arah, dan kurangnya pemantauan *real-time*. *Smart Grid* hadir sebagai solusi dengan mengintegrasikan teknologi informasi untuk menciptakan jaringan yang adaptif. Penelitian ini bertujuan untuk merancang dan mengimplementasikan sistem monitoring dan kontrol terintegrasi berbasis *Internet of Things* (IoT) untuk manajemen *smart grid* skala rumah tangga. Sistem ini menggunakan panel surya 30 Wp sebagai sumber energi, baterai 12V 10Ah sebagai penyimpan daya, serta ESP32 sebagai unit pemroses utama yang terhubung dengan sensor INA219, DS18B20, dan platform *Thingier.io*. Metode penelitian menggunakan pendekatan *Research and Development* (R&D) dengan pengujian fungsionalitas sensor dan logika kontrol. Hasil pengujian menunjukkan bahwa sistem mampu memantau tegangan, arus, daya, dan suhu secara *real-time* dengan tingkat akurasi yang baik (galat tegangan 1,65% dan suhu 4,09%). Selain itu, mekanisme proteksi otomatis berfungsi efektif dengan memutus aliran beban saat suhu baterai melebihi 50°C atau tegangan melewati batas aman (10V–12,5V). Sistem ini menawarkan solusi manajemen energi yang efisien dan transparan untuk aplikasi skala kecil.

Kata Kunci: *Internet of Things* (IoT), *Smart Grid*, ESP32, *Thingier.io*, Sistem Monitoring.

ABSTRACT

Conventional electrical systems face challenges regarding efficiency, one-way communication, and lack of real-time monitoring. Smart Grid emerges as a solution by integrating information technology to create an adaptive network. This study aims to design and implement an integrated Internet of Things (IoT)-based monitoring and control system for household-scale smart grid management. The system utilizes a 30 Wp solar panel as an energy source, a 12V 10Ah battery for storage, and an ESP32 as the main processing unit connected to INA219 sensors, DS18B20, and the Thinger.io platform. The research method employs a Research and Development (R&D) approach with functionality testing of sensors and control logic. The results show that the system is capable of monitoring voltage, current, power, and temperature in real-time with good accuracy (voltage error 1.65% and temperature 4.09%). Furthermore, the automatic protection mechanism effectively cuts off the load when the battery temperature exceeds 50°C or the voltage surpasses safe limits (10V–12.5V). This system offers an efficient and transparent energy management solution for small-scale applications.

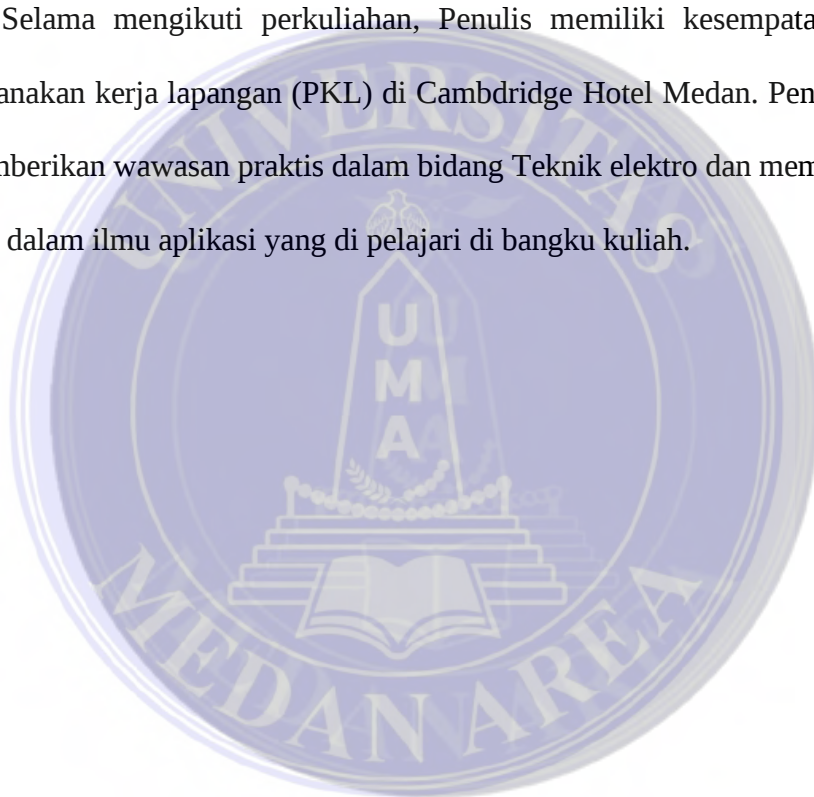
Keywords: *Internet of Things (IoT), Smart Grid, ESP32, Thinger.io, Monitoring System.*



RIWAYAT HIDUP

Penulis dilahirkan di Kampung Muara Bahari tepatnya Tanjung Priok, Jakarta Utara di Pada tanggal 15 Februari 2000 dari ayah Mudahan Tarigan dan ibu Siti Komariah Penulis merupakan putra Pertama dari 5 bersaudara. Tahun 2021 Penulis lulus dari SMK PAB 6 dan pada tahun 2021 terdaftar sebagai mahasiswa Fakultas Teknik Universitas Medan Area.

Selama mengikuti perkuliahan, Penulis memiliki kesempatan untuk melaksanakan kerja lapangan (PKL) di Cambdridge Hotel Medan. Pengalaman ini memberikan wawasan praktis dalam bidang Teknik elektro dan memperkaya Penulis dalam ilmu aplikasi yang di pelajari di bangku kuliah.



KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Tuhan Yang Maha Esa, karena atas berkat dan karunia-Nya penulis dapat menyelesaikan skripsi dengan judul “Desain dan Implementasi Sistem Monitoring dan Kontrol Terintegrasi untuk Manajemen Smart Grid Berbasis IoT”. Skripsi ini disusun untuk memenuhi salah satu syarat kelulusan Sarjana Teknik Elektro di Universitas Medan Area (UMA).

Skripsi ini dapat terselesaikan dengan baik dan lancar berkat bantuan dari berbagai pihak baik itu berupa materil maupun spiritual yang mendukung. Oleh karena itu, dengan segala kerendahan hati penulis ingin menyampaikan rasa terima kasih kepada:

1. Kedua orang tua saya Mudahan Tarigan dan Ibu Siti Komariah yang telah mengkuliahkan, memberikan doa dan dukungan kasih sayang baik itu jasmani dan rohani.
2. Bapak Prof. Dr. Dadan Ramdan, M.Eng., M.Sc., selaku Rektor Universitas Medan Area.
3. Bapak Dr. Eng. Supriatno, S.T., M.T., selaku Dekan Fakultas Teknik.
4. Bapak Ir. M. Fadlan Siregar, ST., MT, selaku Ketua Prodi Teknik Elektro Universitas Medan Area, yang senantiasa memberikan nasehat dan solusi dalam permasalahan akademik.
5. Bapak Dr. Ir Dina Maizana, MT. selaku Dosen Pembimbing yang telah meluangkan waktu untuk memberikan bimbingan, saran, kritik yang membangun, dan senantiasa sabar dalam penyusunan Tugas Akhir ini sampai selesai

6. Seluruh Dosen Program Studi Teknik Elektro Fakultas Teknik, Universitas Medan Area yang telah membagikan ilmunya selama mengajar mata kuliah dengan tulus ikhlas kepada penulis.
7. Terima kasih kepada Rekan-rekan seperjuangan penulis, khususnya Mahasiswa Teknik Elektro Angkatan 2021 yang sudah penulis anggap sebagai saudara, yang telah banyak memberikan dukungan dan semangat.
8. Secara khusus untuk teman sekelas kami yang telah berpulang lebih dulu, Almarhum/Almarhumah. Meski ragamu tak lagi di sini, kenangan dan semangatmu abadi dalam ingatan. Semoga damai di sisi-Nya.
9. Orang-orang yang sudah seperti keluarga bagi penulis, yang tidak pernah bosan dan senantiasa memberikan bantuan, semangat, dan motivasi yang luar biasa kepada penulis.
10. Untuk seseorang yang spesial, La Tirsha Galarina Br Perangin-Angin terima kasih atas segala dukungan, kesabaran, dan motivasi yang tak henti-hentinya diberikan sehingga penulis tetap semangat menyelesaikan skripsi ini.
11. Terima kasih kepada seluruh rekan-rekan di Cambridge Hotel Medan yang juga selalu memberikan dukungan dan pengalaman berharga.
12. Penulis juga ingin mengucapkan terima kasih kepada semua pihak yang tidak dapat disebutkan satu per satu yang telah membantu penulis selama proses penelitian dan penulisan.

Selama proses penulisan skripsi ini, penulis menyadari bahwa skripsi ini masih mempunyai banyak kekurangan, untuk itu penulis mengharapkan adanya komentar yang membangun agar skripsi ini menjadi lebih sempurna.

Akhirnya, penulis ingin mengucapkan terima kasih kepada semua pihak yang telah membantu penulis menyelesaikan skripsi ini. Penulis berharap skripsi ini bermanfaat bagi siapa saja yang membacanya dan bermanfaat bagi pengembangan ilmu pengetahuan untuk kedepannya.

Medan, Januari 2026
Hormat Saya



Brain Rizky Harris Tarigan
21.812.0029



DAFTAR ISI

LEMBAR PENGESAHAN	ii
HALAMAN PERNYATAAN.....	iii
HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR/SKRIPSI/TESIS UNTUK KEPENTINGAN AKADEMIS.....	iv
ABSTRAK	v
ABSTRACT	vi
RIWAYAT HIDUP	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	xi
DAFTAR TABEL	xiv
DAFTAR GAMBAR.....	xv
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Tujuan Penelitian	3
1.4. Manfaat Penelitian	3
1.5. Batasan Masalah.....	4
1.6. Sistematika Penulisan	4
BAB 1: PENDAHULUAN.....	4
BAB 2: LANDASAN TEORI	5
BAB 3: METODE PENELITIAN	5
BAB 4: HASIL DAN PEMBAHASAN.....	5
BAB 5: KESIMPULAN DAN SARAN.....	5
BAB II TINJAUAN PUSTAKA.....	6
2.1. Konsep Smart Grid.....	6
2.2. Peran IoT dalam Smart Grid	8
2.3. Panel Surya	13
2.3.1. Panel Surya Mono-crystalline.....	14
2.3.2. Panel Surya Poly-crystalline	15
2.4. SCC (Solar charge controller).....	15
2.5. Box Plastik	16
2.6. Modul ESP32	17
2.7. Modul Relay	18

2.8. DC-DC Converter	18
2.9. Modul Pengontrol Pengisian Baterai	19
2.10. Sensor Suhu.....	19
2.10.1. Jenis-Jenis Sensor Suhu	20
2.11. Sensor Tegangan dan Arus.....	22
2.11.1. Jenis-Jenis Sensor Tegangan dan Arus	23
2.12. Baterai	25
2.13. Besi Holow	25
2.14. Kabel	26
2.15. Watt Meter DC	26
2.16. Switch (Saklar DC)	27
2.17. Monitoring	27
2.18. Besaran listrik.....	28
2.19. Software Arduino IDE	30
2.20. Thinger.io	32
2.21. Penelitian Terkait	33
2.22. Kerangka Konseptual	35
BAB III METODE PENELITIAN	36
3.1. Desain Penelitian.....	36
3.2. Alat dan Bahan.....	38
3.3. Perancangan dan Pembuatan Sistem.....	39
3.3.1. Perancangan dan Pembuatan Sistem.....	40
3.3.2. Perancangan dan Pembuatan Sistem.....	42
3.3.3. Flowchart Cara Kerja Keseluruhan Sistem.....	47
3.3.4. Pembuatan Dashboard Thinger.io.....	53
3.4. Prosedur Pengumpulan Data	62
3.4.1. Tahap Persiapan Sistem dan Lingkungan.....	62
3.4.2. Tahap Pengumpulan Data Monitoring.....	63
3.4.3. Tahap Pengumpulan Data Uji Fungsi Kontrol	64
3.5. Lokasi Penelitian.....	65
3.6. Jadwal Penelitian.....	65
BAB IV HASIL DAN PEMBAHASAN	67
4.1. Pendahuluan	67
4.1.1. Gambaran Umum Bab	67
4.2. Hasil Implementasi Sistem.....	68
4.2.1. Spesifikasi Panel Surya yang Digunakan Dalam Sistem	

<i>Smart Grid</i>	68
4.2.2. Implementasi Panel Surya Dalam Sistem	70
4.2.3. Implementasi Perangkat Keras Lainnya Dalam Sistem..	72
4.2.4. Implementasi Perangkat Lunak dan <i>Dashboard</i> Monitoring	75
4.3. Pengujian dan Analisa Sistem.....	79
4.3.1. Pengujian Sistem Smart Grid Hari Pertama	79
4.3.2. Pengujian Sistem <i>Smart Grid</i> Hari Kedua.....	87
4.3.3. Pengujian Akurasi Sensor (Kalibrasi).....	94
4.3.4. Pengujian Fungsi Kontrol dan Proteksi	99
4.4. Pembahasan.....	104
BAB V KESIMPULAN DAN SARAN	106
DAFTAR PUSTAKA	109
LAMPIRAN.....	111



DAFTAR TABEL

Tabel 3. 1. Alat dan Bahan yang akan digunakan.....	38
Tabel 3. 2. jadwal penelitian yang direncanakan selama 1 - 6 bulan:.....	66
Tabel 4. 1. Spesifikasi Panel Surya Technical Data	69
Tabel 4. 2. Hasil pengukuran hari pertama secara monitoring	80
Tabel 4. 3. Hasil pengukuran hari pertama secara manual	81
Tabel 4. 4. Hasil pengukuran hari kedua secara monitoring.....	88
Tabel 4. 5. Hasil pengukuran hari kedua secara manual.....	88
Tabel 4. 6. Hasil uji akurasi sensor tegangan.....	96
Tabel 4. 7. Hasil uji akurasi sensor arus.....	97
Tabel 4. 8. Hasil uji akurasi sensor suhu.....	98
Tabel 4. 9. Hasil Pengujian Respons Proteksi Overheat.....	100
Tabel 4. 10. Hasil Pengujian Respons Proteksi Low Voltage dan High Voltage	102

DAFTAR GAMBAR

Gambar 2. 1. : Konsep Smart Grid.....	8
Gambar 2. 2. : Peran IoT dalam Smart Grid	11
Gambar 2. 3. : Panel surya jenis Mono-crystalline.	14
Gambar 2. 4. : Panel surya jenis Poly-crystalline.	15
Gambar 2. 5. : Solar Charge Controller (SCC).	16
Gambar 2. 6. : Box Plastik	16
Gambar 2. 7. : Modul ESP32.	17
Gambar 2. 8. : Modul Relay.....	18
Gambar 2. 9. : Modul Relay.....	19
Gambar 2. 10. : Modul Pengontrol Pengisian Baterai.	19
Gambar 2. 11. : Sensor Suhu.....	20
Gambar 2. 12. : Sensor Suhu.....	23
Gambar 2. 13. : Baterai.	25
Gambar 2. 14. : Besi Hollow.....	25
Gambar 2. 15. : Kabel.	26
Gambar 2. 16. : Watt Meter DC.....	26
Gambar 2. 17. : Switch (Saklar DC)	27
Gambar 2. 18. : Tampilan interface pada aplikasi Arduino IDE	31
Gambar 2. 19. : Dashboard Thinger.io dalam tampilan web	33
Gambar 3. 1. : Diagram Blok Alir Penelitian	37
Gambar 3. 2. :Desain Sistem.....	41
Gambar 3. 3. : Rangkaian Keseluruhan	44
Gambar 3. 4. : Diagram Blok Sistem Monitoring dan Kontrol Terintegrasi	49

Gambar 3. 5. : Antarmuka pendaftaran pada web thinger.io	55
Gambar 3. 6. : Antarmuka penambahan device untuk monitor	58
Gambar 3. 7. : Antar muka penambahan widget ke dashboard	61
Gambar 4. 1. : Visualisasi Penempatan panel surya dalam sistem	71
Gambar 4. 2. : Visualisasi Penempatan SCC	72
Gambar 4. 3. : Visualisasi kotak kontrol dan komponen yang berada didalamnya	73
Gambar 4. 4. : Visualisasi penempatan baterai pada sistem	75
Gambar 4. 5. : Implementasi penguploadan program Arduino IDE ke ESP32	76
Gambar 4. 6. : Implementasi hasil dashboard monitoring di Thinger.io	78
Gambar 4. 7. : Proses pengambilan data pengujian hari pertama secara manual .	80
Gambar 4. 8. : Grafik perbandingan tegangan pengukuran secara monitoring dan manual	82
Gambar 4. 9. : Grafik perbandingan arus pengukuran secara monitoring dan manual	83
Gambar 4. 10. : Grafik perbandingan daya pengukuran secara monitoring dan manual	84
Gambar 4. 11. : Grafik intensitas cahaya pengukuran secara monitoring dan manual	85
Gambar 4. 12. : Grafik perbandingan suhu baterai pengukuran secara monitoring dan manual	86
Gambar 4. 13. : Proses pengambilan data pengujian hari kedua	87
Gambar 4. 14. : Grafik perbandingan tegangan pengukuran secara monitoring dan manual	89

Gambar 4. 15. : Grafik perbandingan arus pengukuran secara monitoring dan manual	90
Gambar 4. 16. : Grafik perbandingan daya pengukuran secara monitoring dan manual	92
Gambar 4. 17. : Grafik intensitas cahaya pengukuran secara monitoring dan manual	93
Gambar 4. 18. : Grafik perbandingan suhu baterai pengukuran secara monitoring dan manual	93
Gambar 4. 19. : Proses pengujian induksi panas pada sensor untuk memicu proteksi overheat	100
Gambar 4. 20. : Tampilan Dashboard monitoring saat proteksi overheat aktif .	101
Gambar 4. 21. : Kondisi fisik modul proteksi saat memutus arus (Cut-Off) pada pengujian tegangan kritis	103

BAB I

PENDAHULUAN

1.1. Latar Belakang

Permintaan energi listrik di seluruh dunia terus meningkat seiring bertambahnya jumlah penduduk dan kemajuan sektor industri. Namun, sistem kelistrikan konvensional yang saat ini digunakan menghadapi berbagai keterbatasan. Sistem ini umumnya hanya mendukung aliran daya satu arah dari pembangkit pusat ke pengguna akhir, minim integrasi data secara real-time, serta masih bergantung pada penanganan manual saat terjadi gangguan, yang dapat memicu pemadaman berskala besar. Kondisi tersebut menjadikan sistem tradisional kurang mampu menjawab kebutuhan efisiensi, keandalan, dan keberlanjutan energi di era modern.

Sebagai jawaban atas tantangan sistem listrik konvensional, konsep Smart Grid atau jaringan listrik pintar muncul sebagai bentuk evolusi yang lebih modern. Smart Grid menggabungkan teknologi informasi dan komunikasi canggih secara cerdas guna menciptakan jaringan yang lebih adaptif, efisien, dan terpercaya. Inovasi ini memungkinkan adanya komunikasi dua arah antara penyedia dan pengguna energi, serta mendukung pemantauan dan pengendalian sistem kelistrikan secara lebih menyeluruh (Yaqoob et al., 2021).

Salah satu pilar utama dalam penerapan Smart Grid adalah kemampuannya dalam melakukan pemantauan dan pengendalian secara otomatis dan real-time. Di titik inilah peran teknologi Internet of Things (IoT) menjadi sangat vital. Melalui

konektivitas berbagai perangkat seperti sensor, aktuator, dan meter pintar ke jaringan internet, IoT memungkinkan akuisisi serta pertukaran data dalam skala besar. Teknologi ini secara signifikan mengatasi berbagai tantangan yang dihadapi jaringan konvensional, seperti keterbatasan konektivitas, kurangnya otomatisasi, dan kesulitan dalam pelacakan perangkat, sehingga mendorong peningkatan efisiensi, keandalan, serta stabilitas sistem kelistrikan secara menyeluruh (Hassan et al., 2023).

Walaupun banyak studi telah membahas elemen-elemen terpisah dari Smart Grid, tantangan utamanya adalah bagaimana merancang sistem yang benar-benar terintegrasi. Sistem yang terintegrasi menjamin bahwa proses pemantauan dan pengendalian berjalan secara terpadu dalam satu platform yang kohesif, memungkinkan pengambilan keputusan yang lebih efektif. Tanpa integrasi yang solid, keterlambatan dalam komunikasi maupun pemrosesan data bisa menghambat efisiensi manajemen jaringan pintar. Oleh karena itu, keberadaan platform komunikasi yang cepat, stabil, dan terintegrasi menjadi komponen esensial dalam pengembangan sistem kelistrikan cerdas (Kabir et al., 2024).

Oleh karena itu, penelitian ini difokuskan pada perancangan dan implementasi sistem monitoring dan kontrol yang terintegrasi untuk mendukung manajemen Smart Grid berbasis teknologi IoT. Sistem yang dikembangkan dirancang untuk memantau parameter kelistrikan penting secara real-time serta menyediakan fungsi kontrol jarak jauh melalui satu antarmuka pusat. Diharapkan, prototipe sistem ini dapat berperan sebagai model rujukan dalam pengembangan infrastruktur Smart Grid mampu beradaptasi dengan kebutuhan masa depan.

1.2. Rumusan Masalah

Berdasarkan uraian latar belakang yang telah disampaikan, maka perumusan masalah dalam penelitian ini dapat dirinci sebagai berikut :

1. Bagaimana merancang sistem monitoring dan kontrol yang terintegrasi dengan teknologi IoT untuk mendukung manajemen Smart Grid?
2. Bagaimanakah cara penerapan sistem monitoring dan kontrol berbasis IoT dalam konteks Smart Grid skala rumah tangga?
3. Parameter kinerja apa saja yang harus dianalisis untuk menjamin keandalan serta efisiensi dari sistem yang dirancang?

1.3. Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut :

1. Merancang desain sistem monitoring dan kontrol terintegrasi untuk manajemen *Smart Grid* yang terhubung dengan teknologi *Internet of Things* (IoT).
2. Mengimplementasikan purwarupa (*prototype*) sistem manajemen *Smart Grid* yang mampu melakukan pemantauan parameter kelistrikan (seperti tegangan, arus, dan daya) serta pengendalian beban secara *real-time*.
3. Menganalisis kinerja sistem yang telah direalisasikan, meliputi tingkat akurasi pembacaan sensor, kecepatan respon (*delay*) kontrol sistem, dan keandalan transmisi data berbasis IoT.

1.4. Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan berbagai manfaat sebagai berikut :

1. Manfaat Akademik: Berkontribusi terhadap pengembangan

pengetahuan, khususnya dalam ranah teknik elektro dan penerapan teknologi Internet of Things (IoT).

2. Manfaat Praktis: Menawarkan solusi teknologi yang dapat meningkatkan efisiensi serta keandalan dalam pengelolaan Smart Grid.
3. Manfaat Sosial: Mendorong adopsi teknologi yang ramah lingkungan dan berkelanjutan dalam sistem distribusi energi listrik.

1.5. Batasan Masalah

Untuk menjaga fokus penelitian, ruang lingkup penelitian ini dibatasi sebagai berikut :

1. Sistem dirancang khusus untuk skala Smart Grid rumahan dengan beban listrik yang terbatas..
2. Parameter yang dimonitor mencakup tegangan, arus, daya, dan suhu sebagai indikator utama performa sistem.
3. Sistem menggunakan teknologi Internet of Things (IoT) dengan protokol komunikasi berbasis Wi-Fi sebagai media transmisi data.

1.6. Sistematika Penulisan

Adapun sistematika penulisan skripsi ini terdiri dari lima (5) BAB, yang masing-masing mencakup beberapa subbab sebagai berikut :

BAB 1: PENDAHULUAN

Bab ini menjelaskan latar belakang, perumusan masalah, tujuan, manfaat, batasan ruang lingkup penelitian, serta sistematika penulisan skripsi.

BAB 2: LANDASAN TEORI

Menyajikan teori-teori pendukung yang relevan dengan penelitian, termasuk konsep Smart Grid, teknologi Internet of Things (IoT), serta prinsip kerja sistem monitoring dan kontrol..

BAB 3: METODE PENELITIAN

Menjabarkan metode yang digunakan dalam proses perancangan dan implementasi sistem, termasuk pendekatan teknis, perangkat keras, dan perangkat lunak yang digunakan.

BAB 4: HASIL DAN PEMBAHASAN

Menguraikan hasil dari implementasi dan pengujian sistem secara menyeluruh. Bab ini mencakup:

1. Desain dan realisasi prototipe,
2. Hasil pengujian fungsional dan performa sistem,
3. Penyajian data dalam bentuk grafik dan tabel,
4. Analisis mendalam terhadap hasil yang diperoleh, perbandingan dengan teori atau penelitian sebelumnya, serta identifikasi kelebihan dan keterbatasan sistem yang dikembangkan.

BAB 5: KESIMPULAN DAN SARAN

Bab ini menyajikan kesimpulan dari keseluruhan penelitian yang secara ringkas menjawab rumusan masalah. Selain itu, disampaikan pula saran untuk pengembangan sistem di masa mendatang serta rekomendasi untuk arah penelitian lanjutan berdasarkan temuan dan keterbatasan studi ini.

BAB II

TINJAUAN PUSTAKA

2.1. Konsep Smart Grid

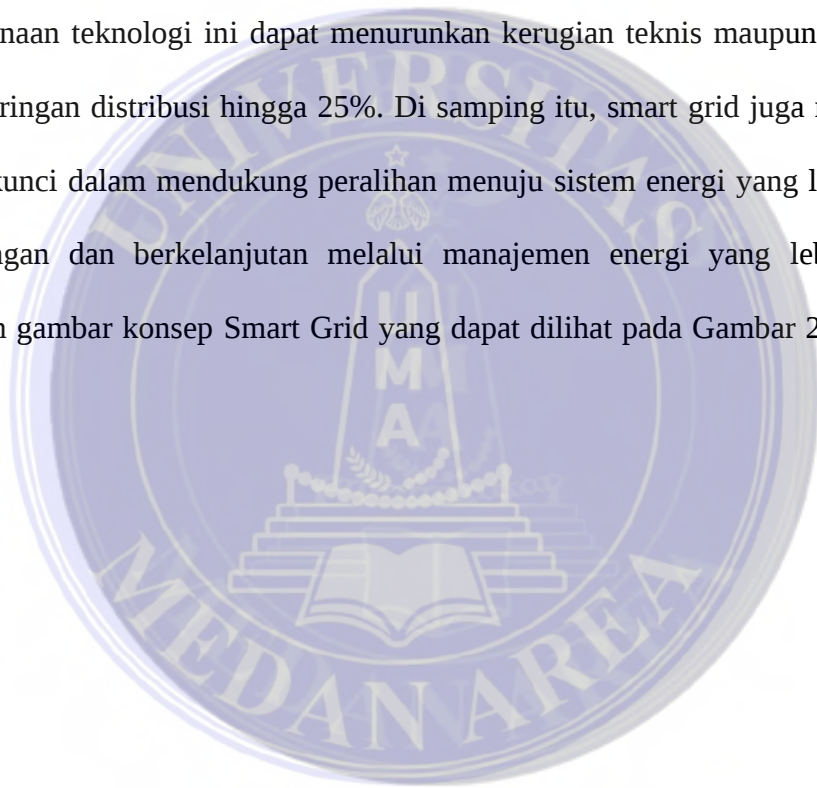
Smart grid, atau jaringan listrik pintar, merupakan transformasi modern dari sistem kelistrikan tradisional. Sistem ini menggabungkan teknologi informasi dan komunikasi (TIK) secara mendalam untuk menciptakan jaringan distribusi listrik yang lebih efisien, andal, fleksibel, dan berkelanjutan. Smart grid adalah jaringan listrik cerdas yang dapat mengintegrasikan tindakan dari semua pengguna yang terhubung pada pembangkit, konsumen dan para pelaku baik dalam rangka penyediaan yang berkelanjutan secara efisien, ekonomis dan pasokan listrik yang aman (IEC, 2016, sebagaimana dikutip dalam Sinaga et al., 2021). Tidak seperti jaringan konvensional yang hanya memungkinkan aliran satu arah, smart grid mendukung komunikasi dan distribusi energi dua arah antara produsen, distributor, dan konsumen. Hal ini memungkinkan manajemen energi yang lebih interaktif dan responsif terhadap kondisi aktual (Yaqoob et al., 2021). Shaukat et al. (2021) mengidentifikasi sejumlah karakteristik utama yang membedakan smart grid dari sistem konvensional, antara lain :

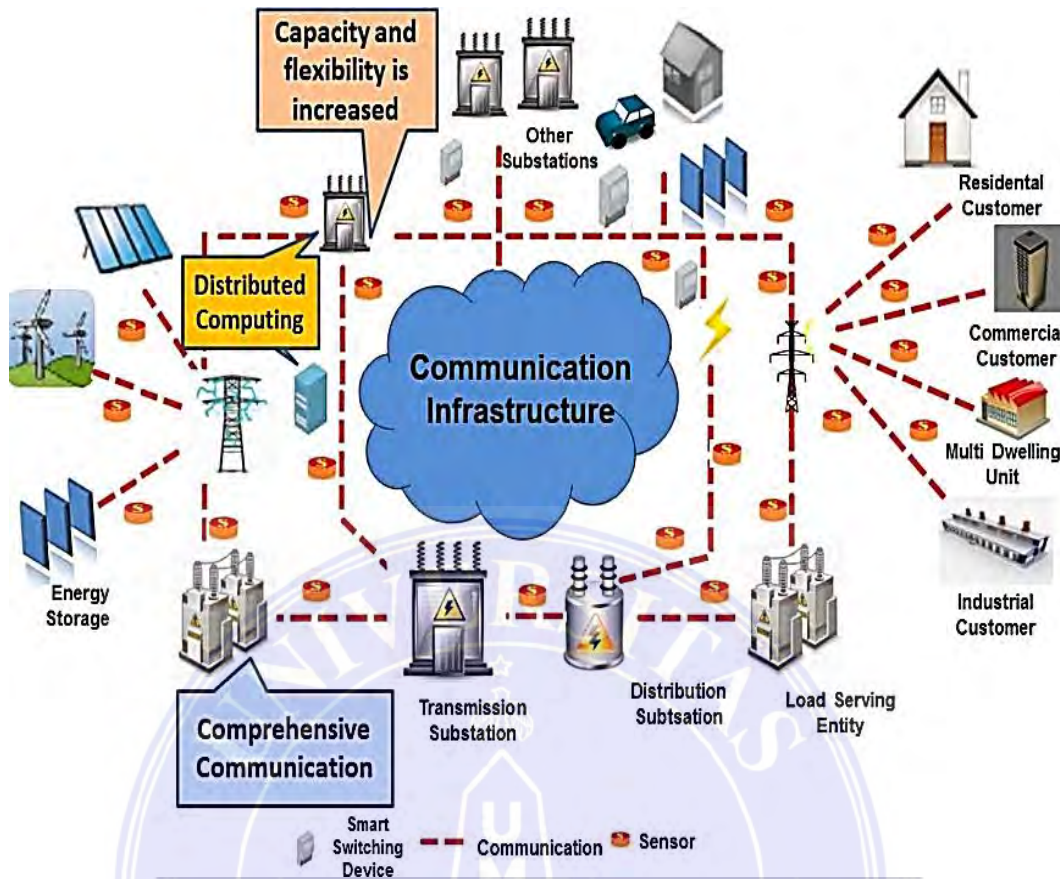
1. Komunikasi Dua Arah: Menyediakan pertukaran data secara real-time antara seluruh pihak dalam sistem, dari pembangkit hingga pengguna akhir.
2. Integrasi Energi Terbarukan: Mampu mengakomodasi sumber energi terdistribusi seperti tenaga surya dan angin ke dalam jaringan dengan cara yang efisien.
3. Manajemen Beban dan Permintaan: Mengoptimalkan distribusi daya

berdasarkan permintaan aktual, membantu mengurangi beban puncak dan meningkatkan efisiensi sistem.

4. Pemulihan Mandiri dan Deteksi Gangguan: Mendeteksi dan menangani gangguan secara otomatis, yang mempercepat pemulihan dan mengurangi durasi pemadaman.

Keunggulan utama dari penerapan smart grid terletak pada kemampuannya dalam meningkatkan efisiensi operasional. Hossain et al. (2021) melaporkan bahwa penggunaan teknologi ini dapat menurunkan kerugian teknis maupun non-teknis pada jaringan distribusi hingga 25%. Di samping itu, smart grid juga memainkan peran kunci dalam mendukung peralihan menuju sistem energi yang lebih ramah lingkungan dan berkelanjutan melalui manajemen energi yang lebih cerdas. Adapun gambar konsep Smart Grid yang dapat dilihat pada Gambar 2.1 dibawah ini:





Gambar 2. 1. : Konsep *Smart Grid*
(Sumber: Gungor et al., 2011)

2.2. Peran IoT dalam Smart Grid

Internet of Things (IoT) menjadi teknologi kunci yang mendasari operasional smart grid modern. Menurut Efendi (2018, sebagaimana dikutip dalam Sandi & Fatma, 2023) Internet of Things ataupun kerap disebut dengan IoT merupakan suatu gagasan dimana seluruh barang yang ada di dunia bisa berkomunikasi antara satu dan yang lain selaku bagian dari perpaduan satu kesatuan sistem yang memakai jaringan internet selaku penghubung. IoT memungkinkan keterhubungan antara berbagai perangkat fisik—dari sensor di lapangan hingga sistem kontrol di pusat data—melalui koneksi internet,

membentuk sebuah ekosistem siber-fisik yang cerdas. Dengan kemampuan ini, IoT mengubah jaringan listrik tradisional menjadi infrastruktur dinamis yang mampu melakukan pemantauan, analisis, dan pengendalian secara real-time. Hal ini secara signifikan meningkatkan efisiensi, keandalan, dan stabilitas sistem kelistrikan (Hassan, Rehmani, & Chen, 2023).

Menurut Al-Ali et al. (2023), arsitektur IoT dalam sistem smart grid umumnya terdiri atas tiga lapisan utama yang saling terintegrasi, yaitu :

1. Lapisan Persepsi (Perception Layer) :

Lapisan ini merupakan komponen dasar yang langsung berinteraksi dengan dunia fisik. Ia bertugas untuk mengumpulkan data lingkungan dan parameter kelistrikan menggunakan sensor pintar secara real-time. Dalam konteks penelitian ini, digunakan sensor seperti:

- INA219 untuk mengukur arus, dan untuk mengukur tegangan DC.
- Akurasi data dari lapisan ini sangat penting karena menjadi dasar bagi proses-proses selanjutnya dalam sistem.

2. Lapisan Jaringan (Network Layer) :

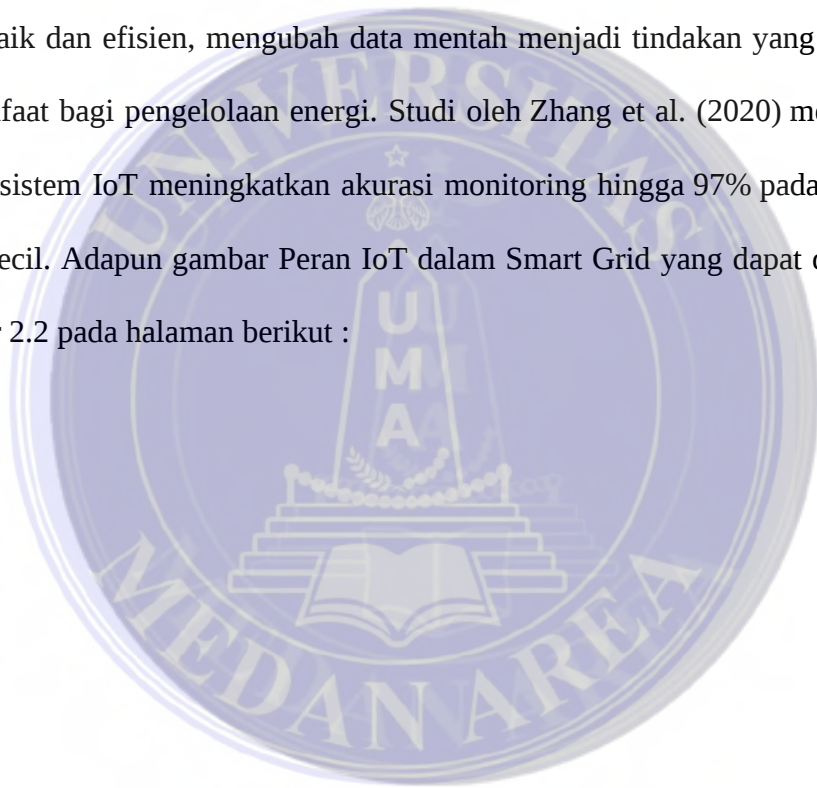
Setelah pengambilan data, lapisan ini mengelola pengiriman data dari sensor ke server atau cloud secara aman dan efisien. Protokol komunikasi nirkabel seperti IOTMP (Internet of Things Message Protocol) pada thinger.io dan Wi-Fi umumnya digunakan karena ringan dan cocok untuk sistem IoT. Pemilihan protokol bergantung pada kebutuhan sistem, seperti latensi rendah, efisiensi energi, dan jangkauan jaringan (Kabir, Dousty, & Pour, 2024).

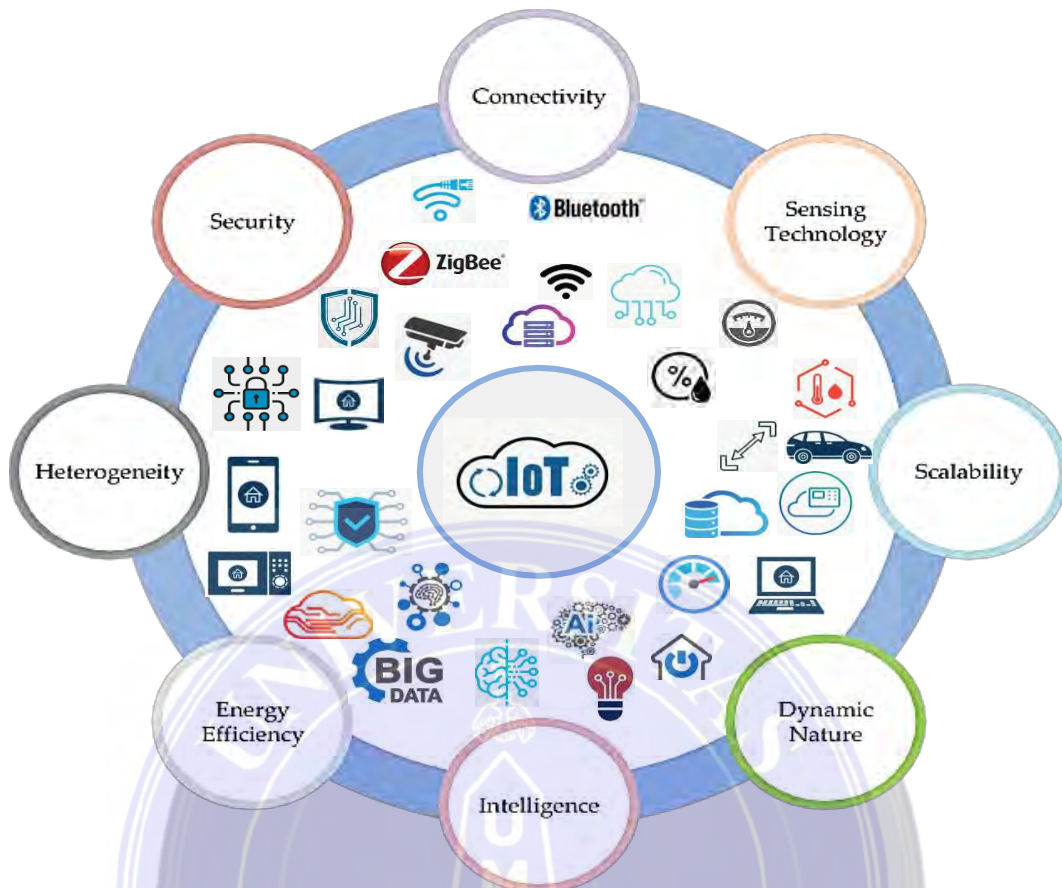
3. Lapisan Aplikasi (Application Layer) :

ini adalah lapisan tertinggi yang berperan dalam pengolahan dan analisis data. Informasi yang dikumpulkan divisualisasikan melalui dashboard interaktif menggunakan platform seperti Thinger.Io. Platform tersebut memiliki fungsi sebagai berikut :

- Memonitoring tegangan, arus, dan suhu
- Mengontrol terhubungnya charger pada baterai

Dengan integrasi dari ketiga lapisan ini, sistem smart grid dapat berjalan lebih baik dan efisien, mengubah data mentah menjadi tindakan yang cerdas dan bermanfaat bagi pengelolaan energi. Studi oleh Zhang et al. (2020) menunjukkan bahwa sistem IoT meningkatkan akurasi monitoring hingga 97% pada smart grid skala kecil. Adapun gambar Peran IoT dalam Smart Grid yang dapat dilihat pada gambar 2.2 pada halaman berikut :



Gambar 2. 2. : Peran IoT dalam *Smart Grid*

(Sumber: Abir et al., 2021)

Gambar ini secara visual menggambarkan ekosistem Internet of Things (IoT) dan elemen-elemen utamanya yang merupakan komponen esensial dalam membangun smart grid. Inti dari peran IoT adalah menjadikan jaringan listrik lebih efisien, andal, dan aman. Berikut penjelasan tiap elemen dalam konteks peran IoT di smart grid:

1. Konektivitas (Connectivity)

Fungsi: Komunikasi dua arah yang berkelanjutan antara pembangkit listrik, gardu induk, dan perangkat pengguna seperti meter pintar.

Ikon Representatif: Wi-Fi, Bluetooth, dan Zigbee – menggambarkan protokol nirkabel yang memungkinkan konektivitas antara jutaan

sensor dan perangkat dalam jaringan.

2. Teknologi Sensor (Sensing Technology)

Fungsi: Sensor bertindak sebagai indera smart grid yang memantau kondisi jaringan—mulai dari aliran listrik, suhu, hingga kondisi lingkungan.

Ikon Representatif: Termometer, ikon kelembapan, dan kamera pengawas – mewakili perangkat pemantauan pada trafo, cuaca, serta aspek keamanan aset kelistrikan.

3. Skalabilitas (Scalability)

Fungsi: Sistem IoT harus mampu tumbuh untuk mengelola jutaan perangkat baru dan volume data besar tanpa penurunan performa.

Ikon Representatif: Ikon cloud dan database – menggambarkan kebutuhan akan infrastruktur penyimpanan dan pengelolaan data berskala besar.

4. Sifat Dinamis (Dynamic Nature)

Fungsi: Memungkinkan respons real-time terhadap fluktuasi permintaan dan pasokan listrik serta integrasi energi terbarukan.

Ikon Representatif: Mobil listrik – melambangkan beban dinamis dan teknologi V2G (*Vehicle-to-Grid*) yang memungkinkan kendaraan menyuplai energi kembali ke jaringan.

5. Kecerdasan (Intelligence)

Fungsi: Mengubah data sensor menjadi wawasan yang dapat ditindaklanjuti menggunakan AI dan *machine learning*.

Ikon Representatif: Otak dan sirkuit – menggambarkan aplikasi

kecerdasan buatan untuk prediksi beban, deteksi gangguan, dan optimasi sistem.

6. Efisiensi Energi (Energy Efficiency)

Fungsi: Memberikan data konsumsi yang rinci, mendukung pengaturan permintaan (*demand response*), dan mengurangi kehilangan daya.

Ikon Representatif: Konsumsi energi yang terukur dan dapat diatur secara dinamis.

7. Heterogenitas (Heterogeneity)

Fungsi: Menyatukan berbagai perangkat dari produsen yang berbeda dengan standar dan protokol yang beragam. Ikon Representatif: Ponsel, laptop, dan server – menunjukkan interoperabilitas sistem IoT terhadap beragam perangkat.

8. Keamanan (Security)

Fungsi: Melindungi jaringan dari ancaman siber melalui enkripsi, autentikasi, dan deteksi intrusi. Ikon Representatif: Perisai dan gembok – simbol pentingnya proteksi terhadap akses ilegal dan serangan digital.

2.3. Panel Surya

Panel surya terbuat dari semikonduktor yang mampu mengubah energi matahari menjadi energi listrik. Panel surya biasa disebut juga sebagai sel *photovoltaic* yang dapat diartikan sebagai “cahaya listrik”. Sel surya juga bergantung pada *photovoltaic* untuk menyerap energi yang berasal dari matahari tersebut. Penggunaannya tergantung pada intensitas cahaya yang mampu diserap panel surya. Daya keluaran panel surya sebanding dengan intensitas cahaya yang diserap. Semakin tinggi intensitas radiasi matahari yang diserap maka semakin

tinggi pula energi listrik yang dihasilkan panel surya.

Panel surya saat ini hadir dalam berbagai bentuk dan jenis yang masing masing memiliki fungsinya masing-masing. Berikut beberapa jenis panel surya yang beredar di pasaran, antara lain:

2.3.1. Panel Surya Mono-crystalline

Mono-crystalline merupakan jenis sel surya yang bahannya berasal dari silisium kristal tunggal yang berbentuk silinder yang diambil dari cairan silisium dan jenis sel surya yang memiliki efisiensi yang paling tinggi yaitu mencapai 1625% (Gunoto & Sofyan, 2020).

Panel surya jenis ini performanya lebih baik dibandingkan dengan jenis Polycrystalline dalam hal saat cuaca mendung, membuatnya menjadi pilihan utam untuk daerah yang memiliki intensitas hujan yang tinggi atau daerah yang sering dilanda oleh hujan.



Gambar 2. 3. : Panel surya jenis *Mono-crystalline*.

(Sumber : [Perbedaan Monocrystalline vs Polycrystalline](#))

2.3.2. Panel Surya Poly-crystalline

Poly-crystalline merupakan jenis sel surya yang materialnya berasal dari susunan Kristal yang acak dengan efisiensi yang dimiliki sebesar 14-16% (Gunoto & Sofyan, 2020).



Gambar 2. 4. : Panel surya jenis *Poly-crystalline*.

(Sumber : [Jual Panel Surya Poly | Distributor Sistem Solar Panel Surya Indonesia](#))

2.4. SCC (Solar charge controller)

Solar charge controller (SCC) khususnya adalah komponen yang digunakan pada sistem PLTS untuk mengatur pengisian baterai menggunakan panel surya untuk optimasi yang lebih baik. Perangkat ini beroperasi dengan mengatur tegangan dan arus pengisian berdasarkan daya yang tersedia dari susunan modul panel surya dan status pengisian baterai SOC (State of Charge).

Secara umum solar charge controller (SCC) yang banyak digunakan pada sistem PLTS saat ini memiliki dua tipe utama, yaitu Pulse Width Modulation (PWM) dan MPPT (Maximum Power Point Tracker). Dalam penelitian ini tipe yang dipakai adalah tipe PWM dimana Sistem kerja dari SCC PWM menggunakan lebar pulse dari on dan offlistrik, sehingga menciptakan seakan-akan sine wave

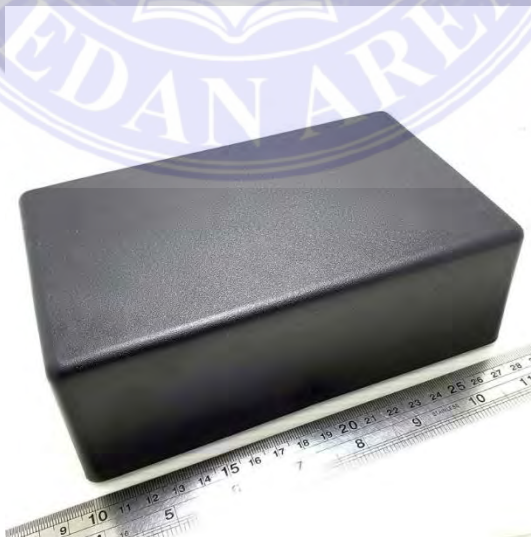
electrical form. Tegangan kerja PWM hanya bisa menyesuaikan dengan tegangan kerja baterai, sehingga jika tegangan yang dihasilkan panel surya di bawah tegangan kerja baterai maka secara otomatis sistem panel surya tidak bisa melakukan pengisian ke baterai (Gunoto & Sofyan, 2020).



Gambar 2. 5. : *Solar Charge Controller (SCC).*
(Gunoto & Sofyan, 2020)

2.5. Box Plastik

Sebagai wadah atau pelindung untuk komponen elektronik seperti mikrokontroler, sensor, dan modul konverter. Ukurannya cukup lega sehingga komponen bisa dipasang dengan rapi sekaligus terlindungi dari debu, kelembapan, maupun kontak fisik yang tidak disengaja.



Gambar 2. 6. : Box Plastik

(Sumber : [Box Kotak Casing Plastik Komponen Project DIY 18X11x6.5 cm](#))

2.6. Modul ESP32

Bertindak sebagai pusat kendali sistem. ESP-32S adalah SoC dengan prosesor dual-core yang dilengkapi Wi-Fi dan Bluetooth. Tugas utamanya mencakup:

- Membaca data dari sensor.
- Memproses data yang diterima.
- Mengirimkan data ke platform IoT (misalnya Thingspeak atau Firebase).
- Menjalankan perintah kontrol untuk mengatur beban melalui relay.

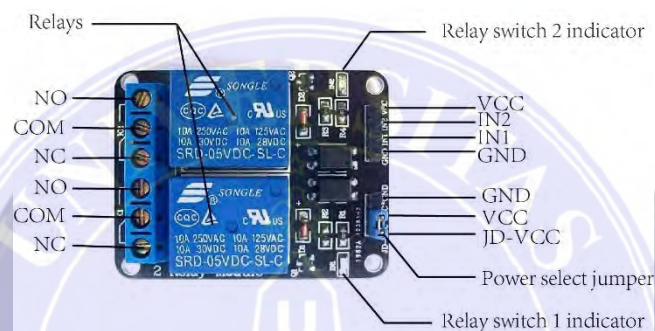


Gambar 2. 7. Modul ESP32.

(Sumber : [JOY-IT Development boards - ESP32 Wi-Fi and Bluetooth module](#) | [Single-board microcontrollers at very favourable prices | reichelt elektronik](#))

2.7. Modul Relay

Modul relay merupakan sebuah board yang tersusun atas satu atau beberapa relay yang dapat dikendalikan melalui mikrokontroler (Noviansyah & Saiyar, 2019, sebagaimana dikutip dalam Rahardjo, 2022). Relay berfungsi sebagai saklar elektronik yang dikendalikan oleh sinyal dari ESP32. Modul 2 channel memungkinkan kontrol dua perangkat berbeda, misalnya kipas dan lampu, secara independen.



Gambar 2. 8. : Modul Relay.

(Sumber : [Learn Coding with Arduino IDE – 2-Channel Relay Module « osoyoo.com](https://www.osoyoo.com/learn-coding-with-arduino-ide-2-channel-relay-module))

2.8. DC-DC Converter

Mengubah tegangan DC dari sumber (12 V) menjadi tegangan yang lebih rendah (misalnya 5 V atau 3,3 V) untuk kebutuhan mikrokontroler. LM2596 dipilih

karena efisiensinya yang tinggi sehingga energi tidak banyak terbuang.



© Photo by ElectroPeak

Gambar 2. 9. : DC-DC Converter

(Sumber : [LM2596 DC-DC Step Down Power Supply Module -Buck Converter](#))

2.9. Modul Pengontrol Pengisian Baterai

Fungsi: Mengatur proses pengisian dan pengosongan baterai dengan batas tegangan yang bisa diprogram. Contohnya, pengisian berhenti saat 14,4 V dan pemutusan beban terjadi pada 10,5 V, sehingga baterai tetap aman dan awet.



Gambar 2. 10. : Modul Pengontrol Pengisian Baterai.

(Sumber : [XH-M604 DC 6-60v Battery Charger Controller](#))

2.10. Sensor Suhu

Sensor DS18B20 adalah sensor suhu digital berbasis protokol komunikasi

1-Wire. Keunggulan utama sensor ini adalah kemampuannya untuk beroperasi dengan hanya menggunakan satu jalur data (selain *ground*) untuk berkomunikasi dengan mikrokontroler. Sensor ini juga memiliki fitur *waterproof* (tahan air) sehingga ideal untuk memantau suhu baterai atau lingkungan yang lembap dalam sistem *Smart Grid*..



Gambar 2. 11. : Sensor Suhu.

(Sumber : [DS18B20 Temperature Sensor Arduino Tutorial \(4 Examples\)](#))

2.10.1. Jenis-Jenis Sensor Suhu

Sensor suhu dikategorikan berdasarkan cara mereka mendeteksi perubahan panas, baik melalui kontak fisik maupun tanpa kontak.

- Sensor Suhu Digital (DS18B20)
 - o Karakteristik: Menggunakan protokol 1-Wire yang memungkinkan transmisi data melalui satu kabel data tunggal. Sensor ini sering hadir dalam bentuk *waterproof* (tahan air).
 - o Cara Kerja: Mengonversi suhu langsung menjadi sinyal digital di dalam komponen sensor menggunakan *chip* silikon. Data tersebut ditransmisikan ke mikrokontroler secara serial tanpa

memerlukan konversi analog-ke-digital tambahan pada mikrokontroler.

- Termistor (Thermal Resistor)
 - o Karakteristik: Komponen pasif yang nilai hambatan listriknya berubah secara signifikan terhadap perubahan suhu. Terdiri dari dua jenis: NTC (*Negative Temperature Coefficient* - hambatan turun saat suhu naik) dan PTC (*Positive Temperature Coefficient* - hambatan naik saat suhu naik).
 - o Cara Kerja: Perubahan suhu menyebabkan perubahan resistansi pada bahan semikonduktor atau keramik di dalamnya. Perubahan ini diukur sebagai variasi tegangan dalam rangkaian pembagi tegangan analog.
- RTD (Resistance Temperature Detector)
 - o Karakteristik: Menggunakan bahan logam murni (seperti Platina/Pt100) yang memiliki hubungan suhu dan hambatan yang sangat linier dan stabil.
 - o Cara Kerja: Bekerja dengan prinsip bahwa hambatan listrik logam meningkat seiring dengan peningkatan suhu. RTD biasanya digunakan pada aplikasi industri yang membutuhkan akurasi sangat tinggi.
- Termokopel (Thermocouple)

- o Karakteristik: Sensor yang terdiri dari dua jenis logam berbeda yang digabung pada salah satu ujungnya. Sangat tahan lama dan mampu mengukur suhu yang sangat ekstrem.
- o Cara Kerja: Berdasarkan efek Seebeck, di mana perbedaan suhu antara ujung sambungan dan ujung terbuka menghasilkan tegangan listrik kecil (orde milivolt) yang sebanding dengan perbedaan suhu tersebut.

2.11. Sensor Tegangan dan Arus

Mengukur tegangan, arus, serta menghitung daya listrik secara real-time. Data dikirim ke ESP32 melalui protokol I2C untuk mendukung manajemen energi. INA219 adalah modul sensor dwifungsi (tegangan dan arus) yang berkomunikasi menggunakan protokol I2C. Sensor ini sangat populer dalam proyek *monitoring* daya DC karena mampu mengukur:

- Tegangan Bus (V): Tegangan sumber (hingga 26V).
- Arus (A): Arus yang mengalir melalui *shunt resistor* (hingga $\pm 3.2A$ untuk modul standar).
- Daya (W): Kalkulasi daya secara otomatis oleh *chip* sensor.



Gambar 2. 12. : Sensor Tegangan dan Arus.

(Sumber : [Belajar Modul INA219 Sensor Arus, Tegangan & Daya dengan Arduino - NN Digital | Belajar Arduino, dan Teknologi Informasi Lainnya](#))

2.11.1. Jenis-Jenis Sensor Tegangan dan Arus

Sensor ini digunakan untuk memantau parameter daya listrik, baik pada sistem arus searah (DC) maupun arus bolak-balik (AC).

- Sensor Daya Digital (INA219)
 - o Karakteristik: Modul sensor dwifungsi yang memantau tegangan dan arus secara simultan melalui antarmuka I2C.
 - o Cara Kerja: Mendeteksi tegangan jatuh (*voltage drop*) pada resistor shunt bernilai sangat kecil (0.1 ohm) yang dipasang seri dengan beban. *Chip* internal mengalokasikan arus dan daya secara otomatis dan mengirimkan data digital ke mikrokontroler.

- Sensor Arus Berbasis Efek Hall (ACS712)
 - o Karakteristik: Sensor analog yang mengukur arus AC maupun DC dengan isolasi listrik yang baik antara sirkuit beban dan sirkuit sensor.
 - o Cara Kerja: Arus yang mengalir melalui jalur tembaga di dalam sensor menghasilkan medan magnet. Sensor Efek Hall mendeteksi medan magnet ini dan mengubahnya menjadi tegangan analog yang linier dengan besaran arus.
- Sensor Tegangan Analog (Voltage Divider)
 - o Karakteristik: Rangkaian sederhana yang terdiri dari dua resistor untuk menurunkan tegangan tinggi agar dapat dibaca oleh pin Analog-to-Digital Converter (ADC) mikrokontroler.
 - o Cara Kerja: Menggunakan prinsip pembagi tegangan (Hukum Ohm). Tegangan *output* adalah sebagian kecil dari tegangan *input*, yang dihitung berdasarkan rasio nilai kedua resistor tersebut.
- Transformator Arus (Current Transformer - CT)
 - o Karakteristik: Digunakan khusus untuk mengukur arus AC pada kabel bertegangan tinggi tanpa memutus rangkaian (non-invasif).
 - o Cara Kerja: Bekerja berdasarkan prinsip induksi elektromagnetik. Arus pada kabel utama bertindak sebagai kumparan primer yang menginduksi arus yang lebih kecil pada kumparan sekunder sensor, yang kemudian dikonversi menjadi tegangan untuk dibaca oleh

pengontrol.

2.12. Baterai

Sebagai penyimpan energi. Kapasitas 10 Ah berarti baterai mampu memasok 1 A selama 10 jam, atau 10 A selama 1 jam.

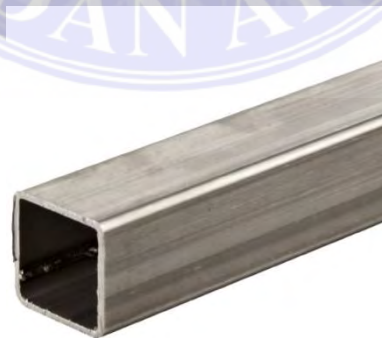


Gambar 2. 13. : Baterai.

(Sumber : [12V 10Ah Akku, AGM Bleiakku, B.B. Battery SHR10-12, 151x65x94 \(lxbxh\), Pol T2 Faston 250 \(6,3 mm\)](#))

2.13. Besi Hallow

Material kerangka/dudukan panel surya serta box elektronik. Ketebalan 3 mm dipilih agar kokoh, tahan cuaca, dan mampu menopang beban.



Gambar 2. 14. : Besi *Hollow*.

(Sumber : [Hollow Besi ukuran 40x80 3mm | Toko Bangunan Online](#))

2.14. Kabel

Kabel merupakan komponen vital dalam sistem monitoring dan kontrol smart grid berbasis IoT karena berfungsi sebagai media penghantar energi listrik maupun sinyal data. Pemilihan jenis dan ukuran kabel harus disesuaikan dengan besar arus yang akan dilewatkan agar sistem tetap efisien dan aman.



Gambar 2. 15. : Kabel.

(Sumber : [Jual Kabel Nyaf 1 X 6mm Di Seller Mitra Mandiri Cable - Mangga Besar, Kota Jakarta Barat | Blibli](#))

2.15. Watt Meter DC

Digunakan sebagai alat ukur tambahan untuk menguji kinerja sistem. Mampu memantau tegangan, arus, dan daya pada berbagai kondisi hingga 60 V dan 150 A.



Gambar 2. 16. : Watt Meter DC.

(Sumber : <https://indonesian.alibaba.com/product-detail/150A-Digital-DC-Inline-4-8V-1600706281758.html>)

2.16. Switch (Saklar DC)

Saklar manual utama untuk memutus atau menyambung daya sistem secara langsung.



Gambar 2. 17. : *Switch* (Saklar DC)

(Sumber : [12V DC 20Amp T85 SPST ON-OFF Mini Rocker Switch - Daier](#))

2.17. Monitoring

Monitoring adalah proses pengawasan dan pemantauan secara terusmenerus terhadap suatu sistem, aktivitas, atau kondisi dengan tujuan mengumpulkan informasi yang relevan mengenai perkembangan atau perubahan yang sedang berlangsung. Hal ini melibatkan pengumpulan data secara periodik atau berkelanjutan, analisis data dan tindakan yang diperlukan berdasarkan hasil pemantauan. Penerapan pengawasan dapat diterapkan dalam berbagai konteks, termasuk lingkungan, teknologi informasi, layanan kesehatan, keamanan, produksi industri, dan banyak bidang lainnya. Contoh pemantauan mencakup pemantauan lingkungan untuk memantau kualitas udara atau air, pemantauan jaringan komputer untuk memantau kinerja dan keamanan sistem, dan pemantauan kesehatan pasien di lingkungan klinis.

2.18. Besaran listrik

Besaran merupakan segala sesuatu yang dapat dihitung, diukur, serta dinyatakan dalam bentuk nilai atau angka. Dalam bidang kelistrikan, besaran pokok yang sering digunakan antara lain tegangan, arus, dan daya listrik.

1. Tegangan

Menurut Alexander dan Sadiku (2021), tegangan atau beda potensial adalah energi yang dibutuhkan untuk memindahkan satu satuan muatan listrik melalui suatu elemen. Tegangan dinyatakan dalam satuan Volt (V), di mana 1 Volt setara dengan 1 Joule per Coulomb. Berdasarkan standar yang berlaku, sistem tegangan umumnya diklasifikasikan menjadi tegangan rendah, menengah, dan tinggi.

Secara matematis, hubungan antara tegangan, arus, dan hambatan dirumuskan dalam Hukum Ohm sebagai berikut:

$$V = I \times R$$

Dimana :

V = Tegangan Listrik (Volt/V)

I = Arus Listrik (Ampere/A)

R = Hambatan atau Tahanan Listrik (Ohm)

2. Arus

Alexander dan Sadiku (2021) mendefinisikan arus listrik sebagai laju aliran muatan listrik terhadap waktu. Arus hanya dapat mengalir apabila terdapat rangkaian tertutup dan perbedaan potensial. Bahan yang mampu menghantarkan arus listrik dengan baik disebut konduktor.

Satuan arus listrik adalah Ampere (A) dan secara matematis dapat

dinyatakan dengan persamaan:

$$I = \frac{V}{R}$$

Dimana :

I = Arus Listrik (Ampere/A)

V = Tegangan Listrik (Volt/V)

R = Hambatan atau Tahanan (Ohm)

Arus listrik Dibagi menjadi dua bagian yaitu:

Arus listrik searah (DC).

Arus listrik bolak balik (AC).

3. Daya

Daya listrik merupakan laju energi listrik yang diserap atau disalurkan oleh suatu elemen dalam satuan waktu. Menurut Alexander dan Sadiku (2021), daya menunjukkan besarnya usaha atau energi yang dilakukan dalam interval waktu tertentu. Satuan daya listrik adalah Watt (W).

Daya listrik adalah banyaknya energi listrik yang digunakan dalam satu detik. Satuan daya listrik adalah watt, 1 watt 1 Joule/detik. Secara matematis persamaan daya listrik dapat direpresentasikan sebagai:

$$P = \frac{W}{t}$$

Dimana:

P = Daya listrik (Watt/W)

W = Energi listrik (Joule/J)

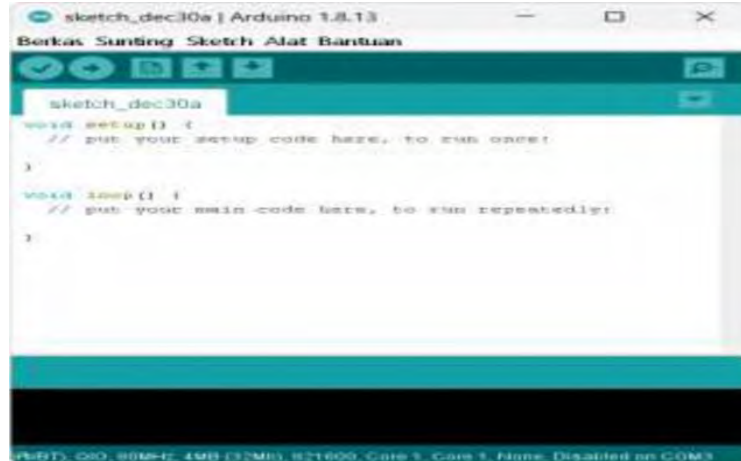
t = Waktu (sekon atau detik/s)

2.19. Software Arduino IDE

Arduino IDE (Integrated Development Environment) adalah perangkat lunak yang digunakan untuk memprogram pada Arduino, atau dengan kata lain Arduino IDE adalah perangkat lunak yang mendukung pemrograman untuk papan Arduino. Arduino IDE dapat diunduh secara gratis dari situs resmi Arduino IDE. Arduino IDE ini berguna sebagai editor teks untuk membuat, mengedit dan memvalidasi kode program juga dapat digunakan untuk diunggah ke papan Arduino. Kode program yang digunakan pada Arduino disebut "sketch" Arduino, atau juga dikenal sebagai kode sumber Arduino, dengan ekstensi file sumber code.ino.

1. Bagian-bagian Arduino IDE

Editor pemrograman sering kali memiliki fungsi potong/tempel dan fungsi cari/ganti teks, seperti halnya Arduino IDE. Dalam deskripsi, aplikasi memberikan pesan umpan balik saat menyimpan dan mengekspor dan di mana kesalahan ditampilkan. Konsol log menampilkan teks log aktivitas Arduino IDE, termasuk pesan kesalahan lengkap dan informasi lainnya, dapat dilihat dalam Gambar 2.15. Sudut kanan bawah menunjukkan port serial yang digunakan. Tombol Toolbar berisi ikon pintasan untuk menguji dan mengunduh program, membuat, membuka, dan menyimpan sketsa, serta membuka monitor serial.



Gambar 2. 18. : Tampilan interface pada aplikasi Arduino IDE

2. Sketch Arduino

Pada Arduino bahasa pemrograman yang digunakan adalah C/C++.

Program Arduino dibagi menjadi 3 bagian utama yaitu: Struktur, Nilai (berisi variabel dan konstanta) dan yang terakhir adalah function.

a. Struktur

Struktur kode pada Arduino berisi fungsi `setup()` dan `loop()`.

- `Setup()`
fungsi ini dipanggil pertama kali ketika menjalankan sebuah sketch. digunakan sebagai tempat inisialisasi variable, pin mode, penggunaan library dan lainnya. fungsi ini dijalankan sekali ketika board dinyalakan atau di reset ulang.
- `Loop()`

Setelah membuat fungsi `setup()` sebagai tempat inisialisasi variabel dan menetapkan nilai maka selanjutnya adalah membuat fungsi `loop()` seperti namanya fungsi ini akan melakukan perulangan berturut-turut, memungkinkan program untuk

mengubah dan menanggapi. digunakan untuk mengontrol board Arduino.

b. Nilai

Berisi variabel atau konstanta tergantung pada type data yang didukung oleh Arduino.

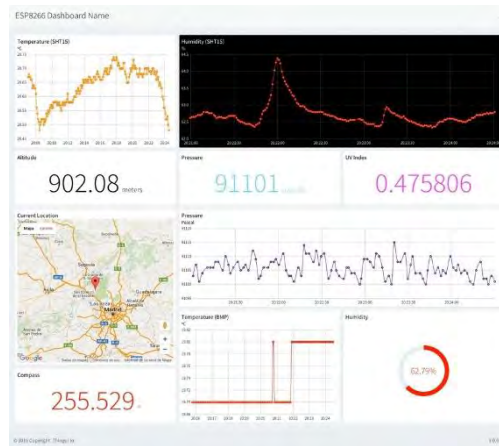
c. Function

Menyegmentasikan kode ke dalam fungsi memungkinkan pemrogram membuat potongan kode modular yang melakukan tugas yang ditentukan, kemudian kembali ke kode sumber tempat fungsi tersebut dipanggil.

2.20. Thinger.io

Thinger.io adalah platform Internet of Things (IOT) yang dirancang untuk memudahkan pengguna dalam menghubungkan, berkomunikasi, dan memelihara perangkat IOT dengan cara yang efisien melalui antarmuka yang mudah digunakan dan intuitif.

Platform ini menawarkan berbagai fitur yang memungkinkan pengumpulan data secara real-time, integrasi perangkat yang dinamis, serta analisis dan visualisasi data. Hal ini menjadikannya pilihan yang menarik dan fleksibel untuk berbagai aplikasi IOT modern, mulai dari pemeliharaan lingkungan, otomatisasi rumah, hingga sistem manajemen energi. Dengan antarmuka yang mudah digunakan dan skalabilitas yang tinggi, Thinger.io merupakan solusi yang tepat bagi para calon peneliti dan pelajar (Rizki Andika, 2022)



Gambar 2. 19. : *Dashboard Thingier.io dalam tampilan web*

(Sumber : <https://docs.thinger.io/features/dashboards>)

2.21. Penelitian Terkait

Berbagai penelitian sebelumnya telah mengulas aspek-aspek penting dalam pengembangan sistem monitoring dan kontrol berbasis IoT untuk smart grid. Studi-studi tersebut memberikan fondasi yang solid dalam memahami tren terkini serta mengidentifikasi celah penelitian yang dapat dijadikan pijakan bagi penelitian ini.

Sejumlah studi fokus pada aspek komunikasi dan deteksi gangguan. Misalnya, Rahman et al. (2021) mengembangkan sistem pemantauan smart grid menggunakan teknologi LoRaWAN, yang terbukti mampu mendeteksi gangguan 50% lebih cepat dibandingkan metode konvensional dengan tingkat akurasi sekitar $\pm 1,5\%$. Meskipun teknologi ini unggul dalam jangkauan luas, penelitian tersebut tidak menekankan pada integrasi kontrol beban otomatis, khususnya untuk skala rumahan yang menuntut latensi rendah.

Penelitian lainnya mengangkat sisi optimasi kontrol. Li et al. (2020) menerapkan algoritma reinforcement learning dalam pengelolaan energi, yang

berhasil meningkatkan efisiensi sistem hingga 18% selama beban puncak. Namun, fokus utama mereka adalah pengembangan algoritma, sementara detail implementasi perangkat keras serta pemanfaatan platform IoT untuk akuisisi data di lapangan kurang dijelaskan.

Aspek keamanan sistem IoT juga menjadi perhatian sejumlah peneliti. Andrianto et al. (2022) mengeksplorasi integrasi antara IoT dan blockchain untuk menjaga keamanan data pada smart grid, merekomendasikan penggunaan enkripsi AES-256 serta autentikasi dua faktor. Walau signifikan, studi ini bersifat teoretis dan tidak melibatkan uji coba prototipe secara fisik.

Dalam konteks penerapan praktis di skala rumah tangga, Javaid et al. (2023) mengembangkan sistem manajemen energi berbasis IoT untuk residensial, yang mampu mengoptimalkan konsumsi listrik melalui penjadwalan beban yang cerdas. Di sisi lain, Astudillo-Salinas et al. (2022) berhasil membangun sistem akuisisi data berbasis Thinger.io, yang bersifat open-source dan berbiaya rendah—membuktikan efektivitas platform tersebut untuk pengembangan sistem IoT yang efisien.

Berdasarkan kajian tersebut, dapat disimpulkan bahwa kebanyakan penelitian terdahulu berfokus pada satu aspek tertentu—baik itu algoritma optimasi, keamanan, atau teknologi komunikasi tertentu seperti LoRaWAN. Masih terdapat celah dalam pengembangan sistem terpadu yang menggabungkan fungsi monitoring dan kontrol secara menyeluruh dalam satu platform yang efisien, mudah diakses, dan hemat biaya untuk aplikasi rumahan.

Penelitian ini menghadirkan kebaruan dengan merancang dan mengimplementasikan prototipe sistem fisik menggunakan ESP32 dan platform

Thingier.io. Fokusnya adalah pada penciptaan solusi yang terintegrasi dan aplikatif, dengan evaluasi kinerja berdasarkan akurasi data, latensi komunikasi, serta respons sistem terhadap perintah kontrol secara bersamaan.

2.22. Kerangka Konseptual

Berdasarkan tinjauan pustaka, kerangka konseptual penelitian ini adalah:

1. Desain Sistem:

- Pemilihan sensor, mikrokontroler, dan protokol komunikasi.
- Perancangan antarmuka pengguna berbasis web.

2. Implementasi:

- Integrasi hardware dan software.
- Kalibrasi sensor dan uji coba di lingkungan laboratorium.

3. Evaluasi:

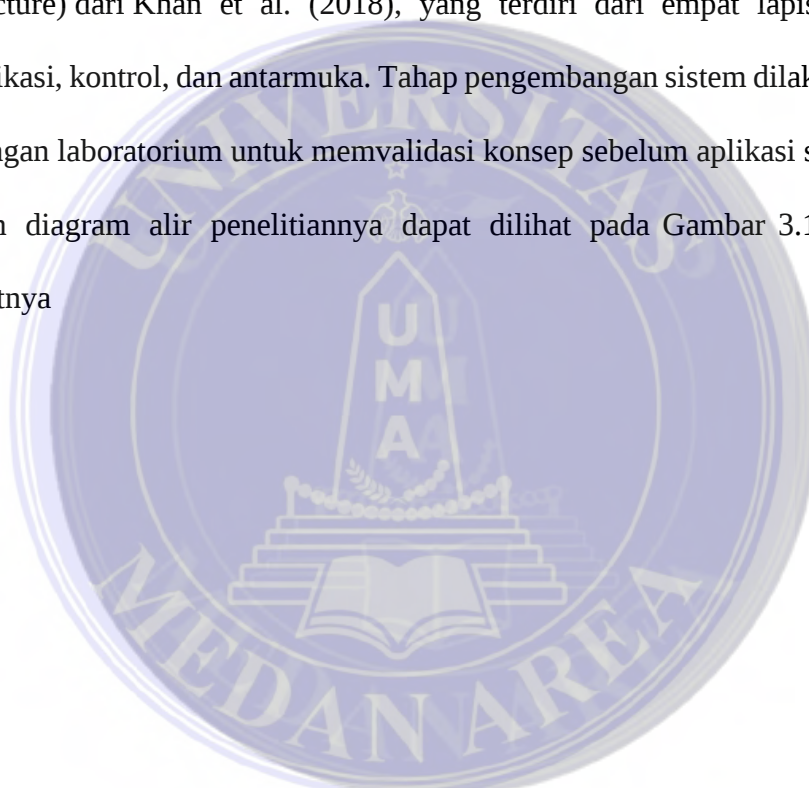
- Parameter: Akurasi data, waktu respons, dan konsumsi daya sistem.

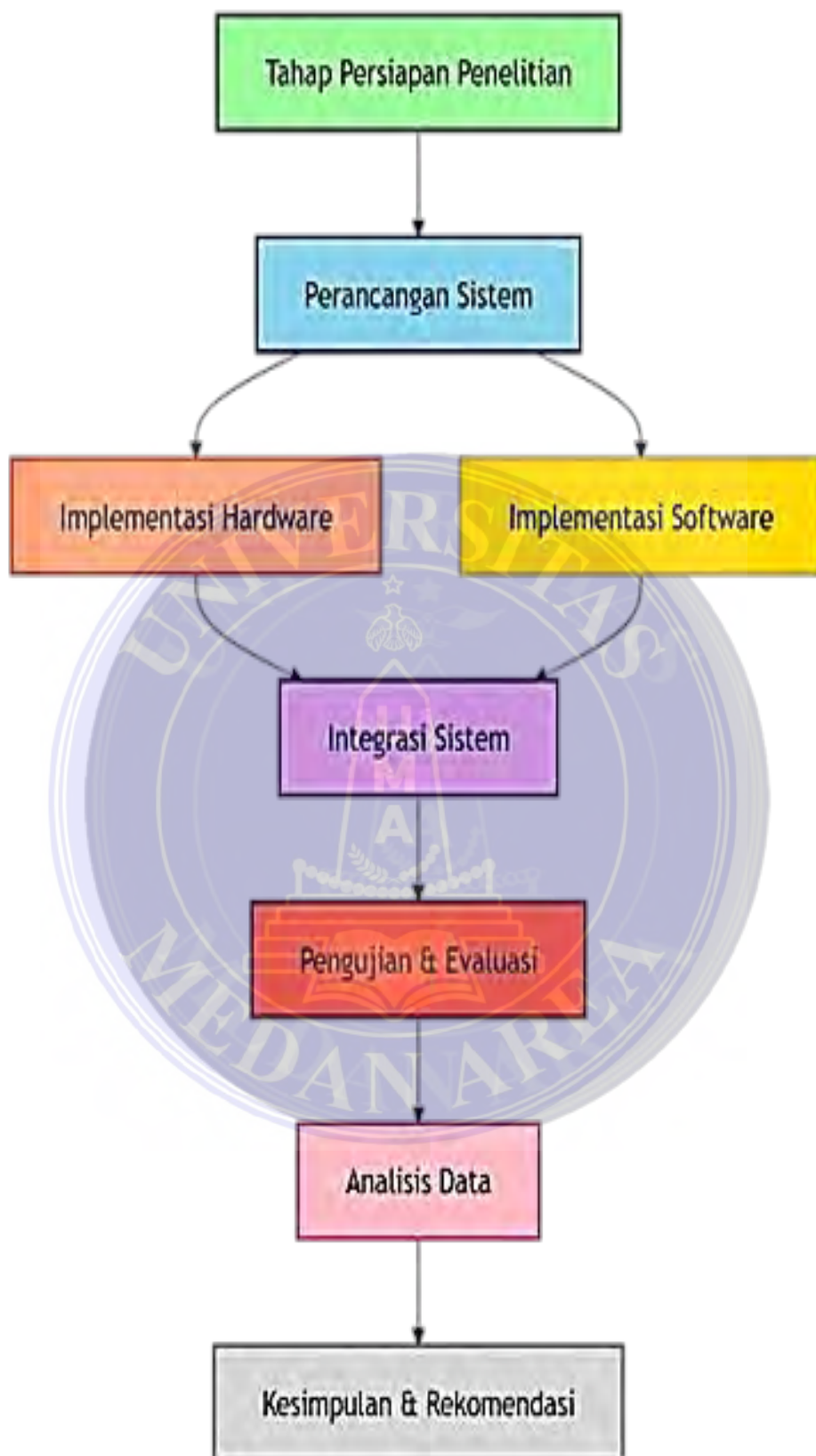
BAB III

METODE PENELITIAN

3.1. Desain Penelitian

Penelitian ini menggunakan pendekatan Research and Development (R&D) dengan tahapan perancangan, implementasi, dan evaluasi sistem. Model desain yang diadopsi mengacu pada kerangka kerja IoT berbasis lapisan (layered architecture) dari Khan et al. (2018), yang terdiri dari empat lapisan: sensor, komunikasi, kontrol, dan antarmuka. Tahap pengembangan sistem dilakukan dalam lingkungan laboratorium untuk memvalidasi konsep sebelum aplikasi skala penuh. Adapun diagram alir penelitiannya dapat dilihat pada Gambar 3.1 di halaman selanjutnya





Gambar 3. 1. : Diagram Blok Alir Penelitian

(Sumber: Diolah oleh Penulis menggunakan <https://mermaid.js.org>)

3.2. Alat dan Bahan

Adapun penelitian ini memanfaatkan berbagai macam komponen elektronik dan juga perangkat lunak untuk membangun sistem monitoring dan kontrol terintegrasi untuk manajemen smart grid berbasis IOT. Adapun bahan yang akan digunakan dalam penelitian ini dapat dilihat pada tabel 3.1 sebagai berikut :

Tabel 3. 1. Alat dan Bahan yang akan digunakan

No.	Alat	Bahan	Spesifikasi Bahan
1.	Personal Computer (PC) atau Laptop	Panel Surya	30 Wp Monocrystalline merek Technical Data
2.	Solder listrik	Solar Charge control (SCC)	Type PWM 50 A
3.	Timah	Box Plastik	18 cm x 11,5 cm x 6 cm
4.	Bor Listrik	Modul ESP32	ESP-32S
5.	Tang Potong	Modul Relay	2 Channel
6.	Gerinda	DC-DC Konverter	LM2596 Step Down
7.	Obeng	Modul kontroller Charge baterai	XH-M604
8.	Mesin Las Listrik	Modul Sensor Suhu	DS18B20
9.	Kawat Las	Modul Sensor Tegangan dan Arus	INA219
10.		Baterai	12 Volt, 10 Ah
11.		Besi Holow	3 mm
12.		Kabel	NYAF 2,5 mm ² dan 0,75 mm ²
13.		Watt Meter DC	60 V, 150 A
14.		Switch	12 V DC

3.3. Perancangan dan Pembuatan Sistem

Penelitian ini mencakup perancangan dan konstruksi sistem pemantauan serta kontrol yang terintegrasi untuk mengevaluasi kinerja Sistem Monitoring dan Kontrol Berbasis IoT. Sistem ini dikembangkan dengan tujuan utama untuk mengumpulkan data kelistrikan secara real-time, melakukan pengolahan data melalui mekanisme manajemen smart grid, serta menyajikan informasi hasil pengolahan dalam format yang akurat, informatif, dan mudah dipahami oleh pengguna.

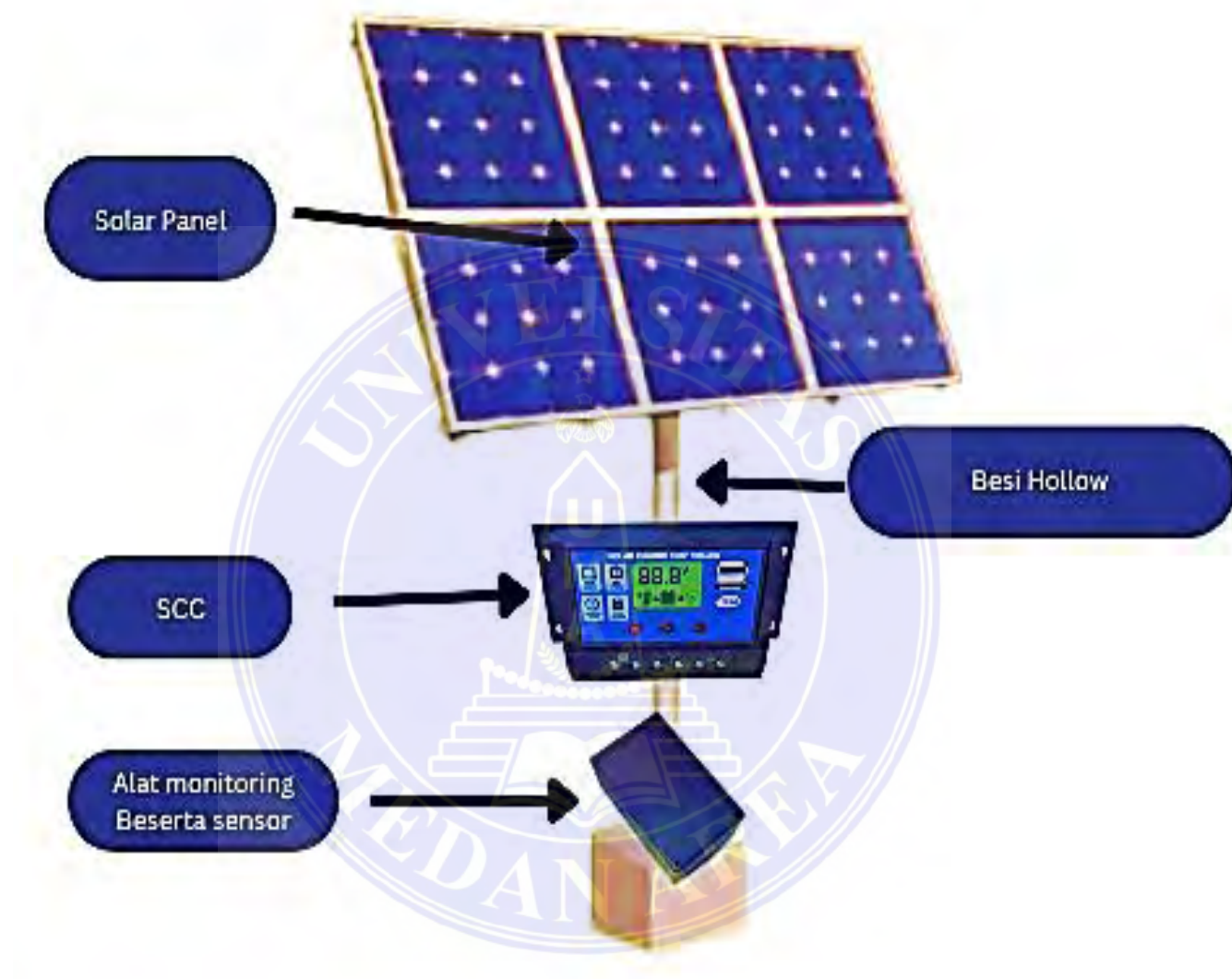
Melalui integrasi antara sensor, mikrokontroler, dan platform IoT berbasis cloud, sistem ini tidak hanya mampu memonitor parameter penting seperti tegangan, arus, daya, dan suhu, tetapi juga memberikan kemampuan kontrol beban secara langsung dari jarak jauh. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi nyata terhadap upaya pengembangan smart grid skala rumah tangga maupun skala laboratorium, yang menekankan pada efisiensi energi, keandalan sistem, dan kemudahan akses informasi.

Selain itu, hasil rancangan ini dirancang agar bersifat modular dan skalabel, sehingga dapat dengan mudah diperluas untuk mencakup jumlah beban atau sensor yang lebih banyak sesuai kebutuhan implementasi di lapangan. Evaluasi performa dilakukan dengan menguji fungsi monitoring, keakuratan sensor, waktu respons sistem, serta stabilitas komunikasi IoT, sehingga diperoleh gambaran menyeluruh mengenai efektivitas sistem yang telah dibangun. Berikut adalah beberapa penjelasan tentang rancangan yang dapat memperlihatkan gambaran alat :

3.3.1. Perancangan dan Pembuatan Sistem

Dalam pembuatan Sistem Monitoring dan Kontrol Terintegrasi Berbasis IoT, desain sistem memegang peranan yang sangat penting karena berfungsi sebagai visualisasi awal dari sistem yang akan direalisasikan. Melalui desain, seluruh aspek mulai dari alur kerja data, arsitektur perangkat keras, integrasi sensor, hingga koneksi ke platform IoT berbasis cloud dapat dipetakan secara jelas. Berikut adalah gambar desain sistem yang dapat dilihat pada gambar 3.2 pada halaman selanjutnya.





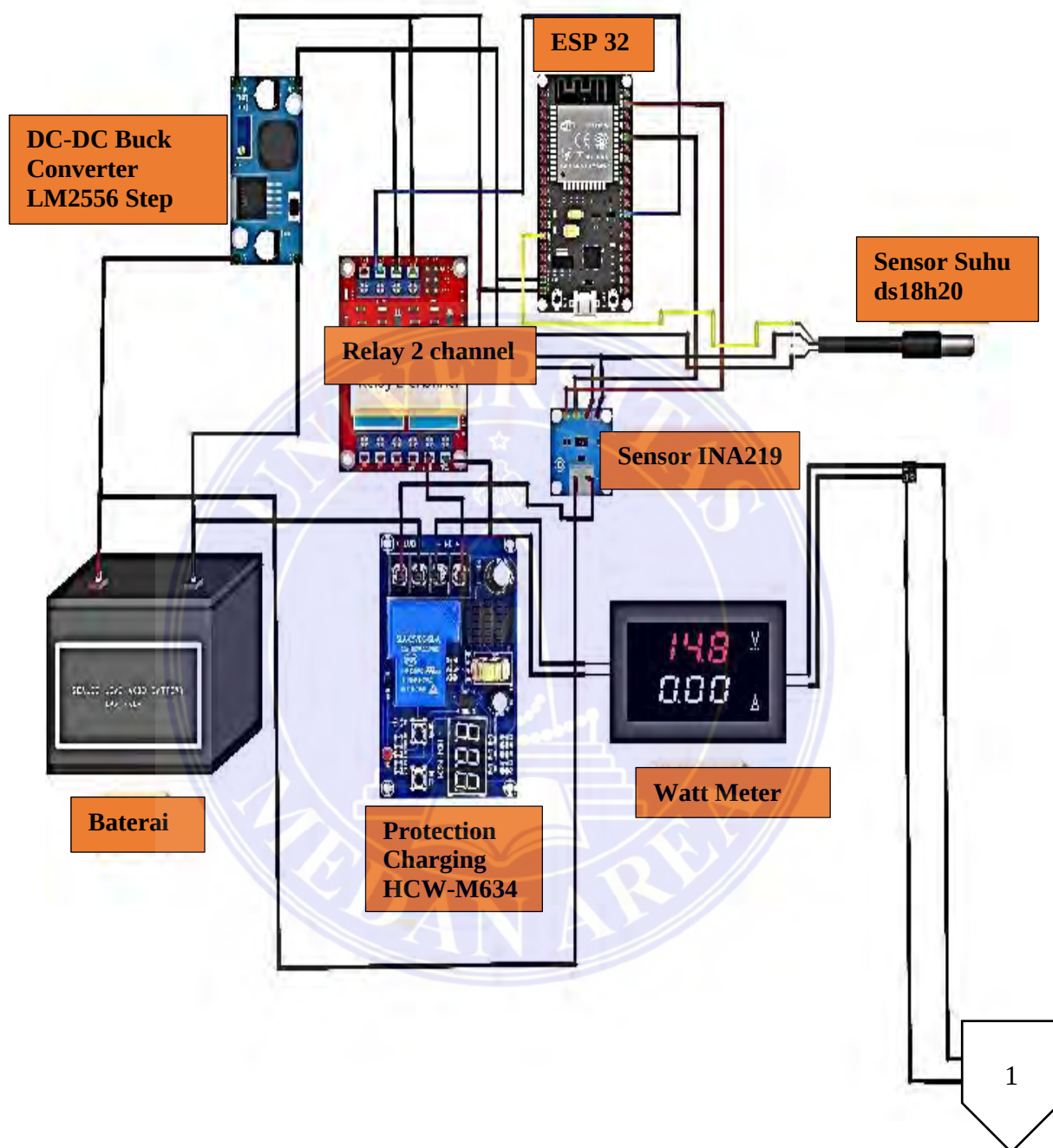
Gambar 3. 2. :Desain Sistem

Sumber: Diolah oleh peneliti, 2025

3.3.2. Perancangan dan Pembuatan Sistem

Gambar rangkaian sistem berperan penting dalam proses pembangunan Sistem Monitoring dan Kontrol Terintegrasi Berbasis IoT, karena memberikan panduan visual yang jelas mengenai hubungan antar komponen. Dengan adanya rancangan rangkaian, peneliti dapat melihat bagaimana sensor, mikrokontroler, modul komunikasi, relay, serta sumber daya saling terhubung dalam satu sistem yang utuh. Hal ini sangat membantu dalam tahap implementasi nyata, karena rangkaian berfungsi sebagai peta teknis yang meminimalkan kesalahan pemasangan, mempercepat proses perakitan, serta memastikan setiap jalur koneksi sudah sesuai dengan kebutuhan fungsional sistem.

Selain itu, gambar rangkaian juga mempermudah proses analisis dan troubleshooting. Jika terjadi masalah pada sistem, peneliti dapat merujuk langsung ke skema rangkaian untuk mengidentifikasi titik kesalahan, misalnya pada jalur tegangan, pin koneksi sensor, atau pengendalian relay. Dengan demikian, keberadaan gambar rangkaian tidak hanya memudahkan pembuatan sistem secara real, tetapi juga meningkatkan akurasi, efisiensi, dan keandalan dalam pengembangan serta pengujian prototipe. Adapun gambar rangkaian keseluruhan yang dapat dilihat pada gambar 3.3. dihalaman selanjutnya sebagai berikut :





Gambar 3. 3. : Rangkaian Keseluruhan

Sumber: Diolah Oleh peneliti, 2025

Secara garis besar, sistem ini bekerja dengan menggabungkan alur energi (power flow) dari panel surya ke baterai dan beban, serta alur data (data flow) dari sensor ke mikrokontroler lalu ke platform IoT. Integrasi keduanya memungkinkan sistem tidak hanya menghasilkan dan menyalurkan energi, tetapi juga dapat dimonitoring, dianalisis, dan dikontrol secara real-time. Adapun Sistem tenaga surya mini ini menyediakan listrik melalui tahapan berikut:

- Panel Surya → SCC (Solar Charge Controller)

Cahaya matahari diubah menjadi arus DC oleh panel surya, kemudian dialirkan ke SCC. SCC bertugas mengatur pola pengisian baterai agar sesuai dengan kapasitasnya, mencegah overcharging, serta memastikan arus yang masuk stabil.

- SCC → Baterai → Beban

Energi yang dihasilkan disimpan dalam baterai 12V 10Ah. Baterai berfungsi sebagai cadangan energi sehingga sistem tetap menyala meskipun tidak ada sinar matahari. Dari baterai, energi dapat langsung digunakan untuk beban DC atau dialirkan melalui rangkaian IoT untuk keperluan monitoring.

- Proteksi & Konversi Tegangan

Karena komponen cerdas (ESP32, sensor, relay) bekerja pada tegangan rendah (3.3–5V), maka digunakan DC-DC Buck Converter (LM2596) untuk menurunkan tegangan baterai dari

12V menjadi tegangan operasi yang sesuai. Modul proteksi tambahan (misalnya XH-M604) mencegah baterai over-discharge, menjaga keawetan baterai.

- Distribusi ke Beban & Monitoring Lokal

Daya dialirkan ke relay sebagai saklar elektronik yang dapat dikontrol, serta melalui Watt Meter DC untuk validasi manual. Dengan demikian, sistem tidak hanya dapat menyalurkan energi tetapi juga memberikan diagnosis langsung jika ada masalah kelistrikan.

- ESP32 sebagai Otak Sistem

ESP32 (SoC dengan Wi-Fi & Bluetooth) mengendalikan seluruh proses monitoring dan kontrol. Semua data dari sensor masuk ke ESP32, kemudian diproses dan dikirimkan ke server cloud Thingier.io.

- Monitoring Sensor Real-Time

- o INA219 mengukur tegangan, arus, dan daya pada jalur baterai → memberikan data konsumsi dan kapasitas energi yang tersisa.

- o DS18B20 mengukur suhu (lingkungan atau baterai) → berguna untuk mencegah overheating.

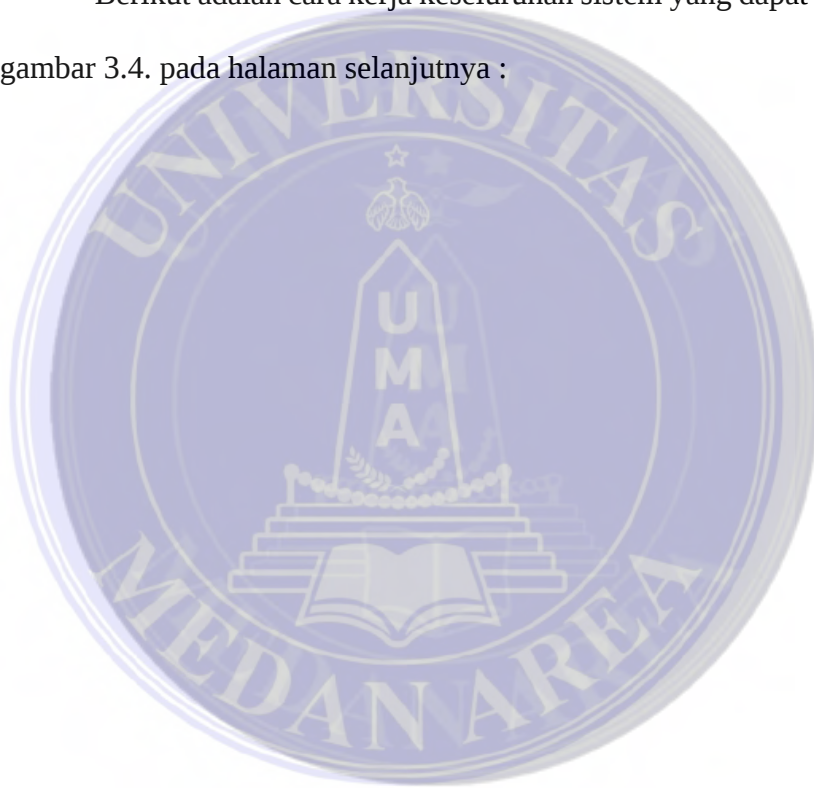
- Komunikasi Data ke Cloud Data yang diperoleh sensor dikirimkan oleh ESP32 ke internet melalui jaringan Wi-Fi. Dengan ini, pengguna dapat memantau tegangan, arus, daya, dan suhu secara real-time menggunakan smartphone atau laptop,

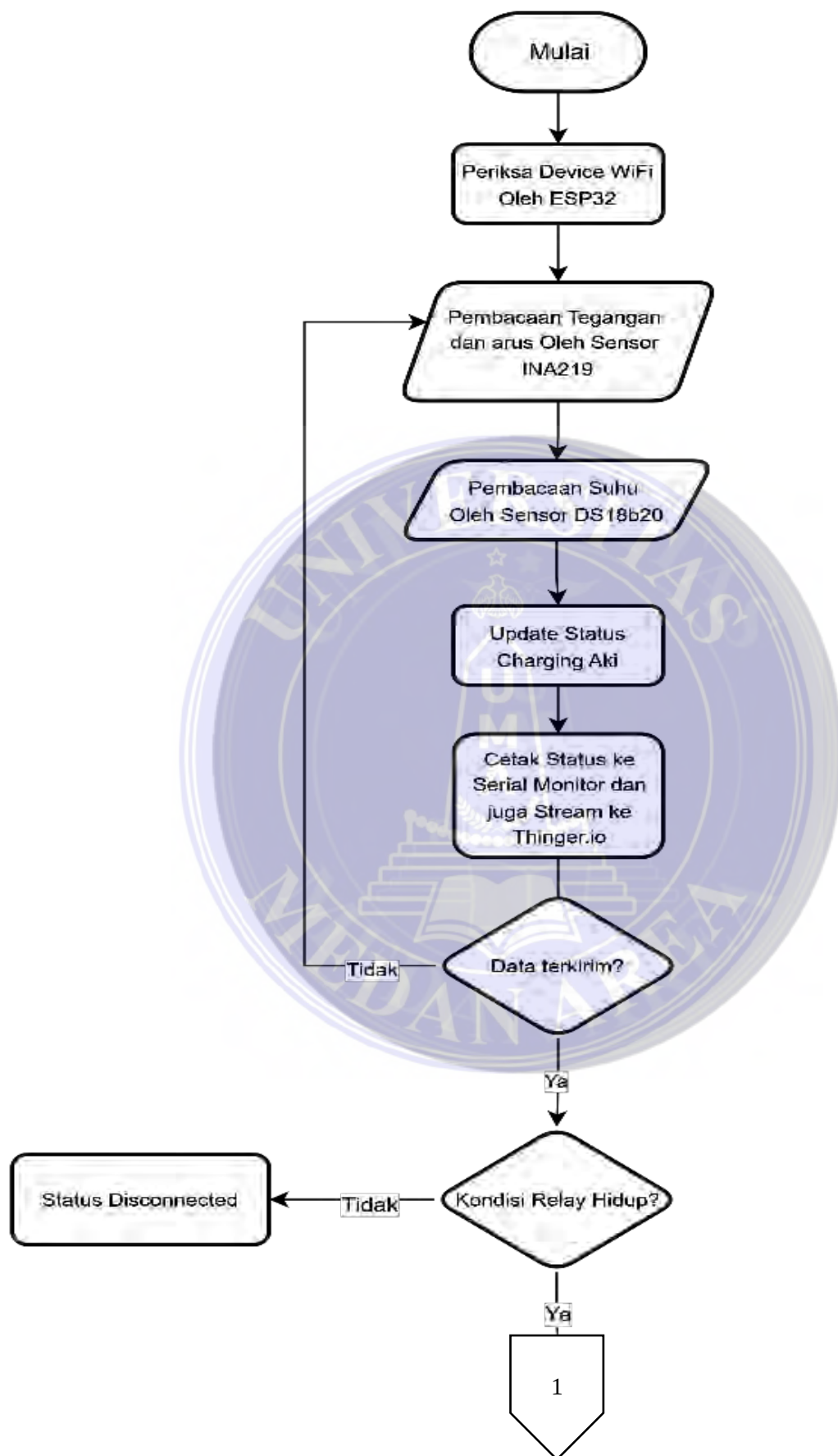
tanpa harus berada di lokasi sistem.

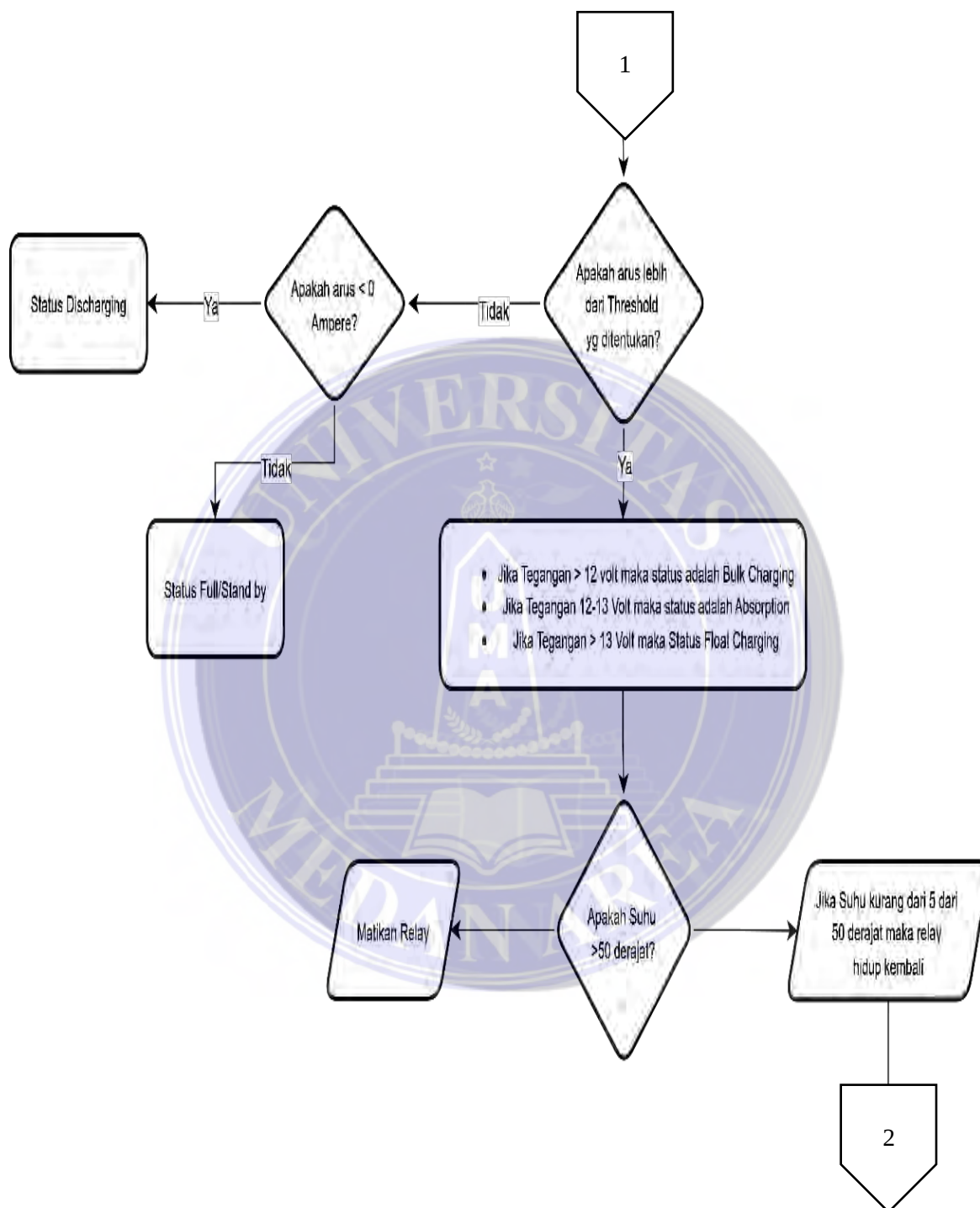
- Kontrol Otomatis & Manual Berdasarkan data sensor atau perintah pengguna melalui aplikasi IoT, ESP32 dapat mengontrol Relay 2 Channel untuk menyalakan/mematikan Charger pada baterai.

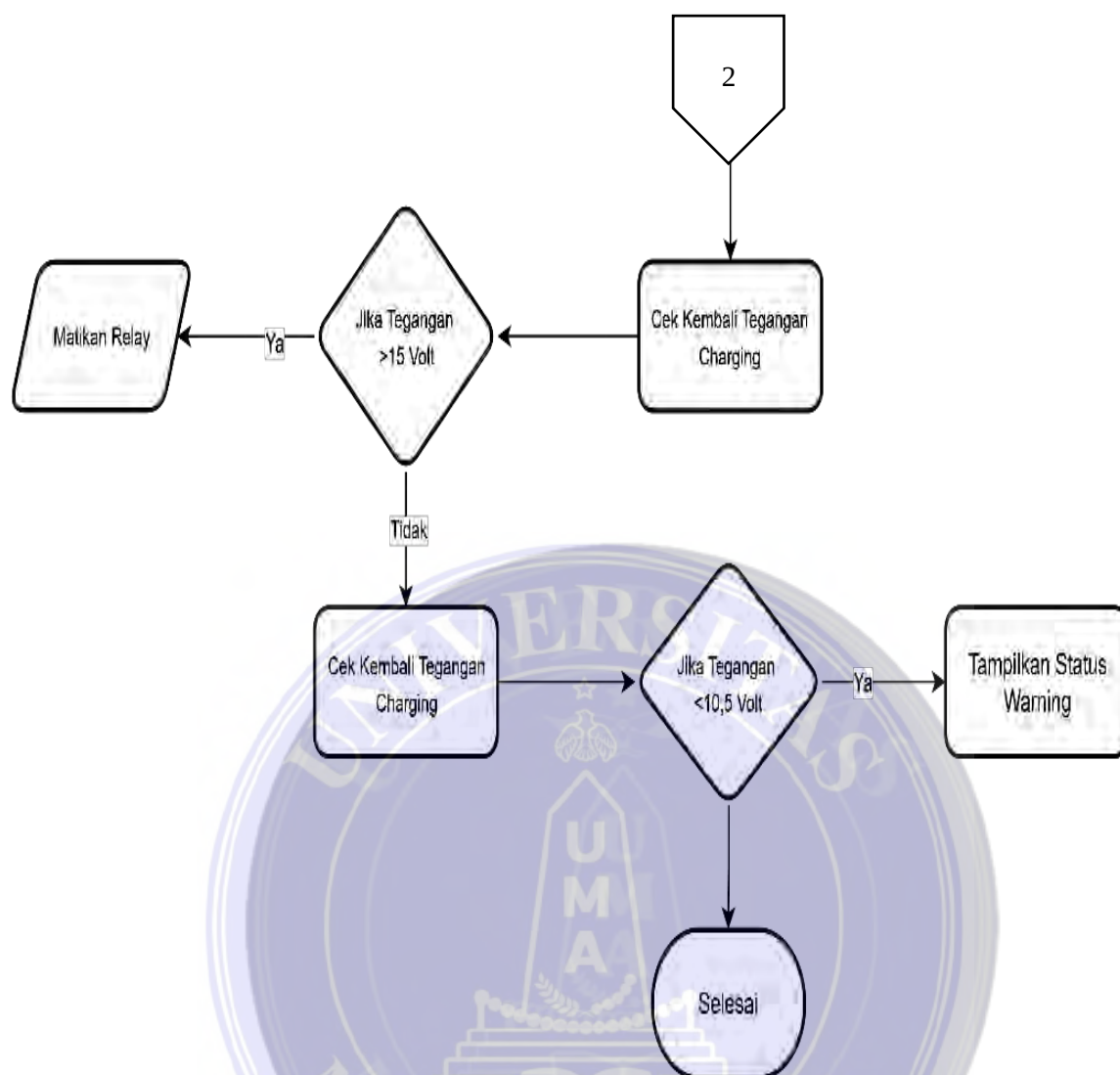
3.3.3. Flowchart Cara Kerja Keseluruhan Sistem

Berikut adalah cara kerja keseluruhan sistem yang dapat dilihat pada gambar 3.4. pada halaman selanjutnya :









Gambar 3. 4. : Diagram Blok Sistem Monitoring dan Kontrol Terintegrasi

Sumber: Diolah oleh peneliti, 2025

Sistem monitoring dan kontrol berbasis IoT ini bekerja secara siklus berulang (looping process), di mana setiap siklus mencakup proses pengumpulan data, pengolahan, pengiriman, analisis status, dan proteksi. Berikut tahapan lengkapnya:

Tahap 1 : Inisialisasi dan Pengumpulan Data

- Mulai → Sistem memulai eksekusi dari titik awal.
- Koneksi WiFi → ESP32 menginisialisasi perangkat

keras, kemudian memeriksa koneksi WiFi. Koneksi ini penting karena seluruh data hasil pembacaan sensor akan dikirimkan ke server/cloud. Jika koneksi gagal, sistem akan mencoba berulang hingga terhubung.

- Pembacaan Sensor INA219 → ESP32 membaca tegangan (Volt) dan arus (Ampere) dari baterai. Data ini penting untuk menentukan status energi.
- Pembacaan Sensor DS18B20 → ESP32 juga membaca data suhu baterai/lingkungan sebagai indikator kondisi termal sistem.

Tahap 2 : Pengolahan dan Pengiriman Data

- Update Status Charging Aki → Data tegangan dan arus yang masuk diproses untuk menghasilkan status baterai (misalnya: mengisi, penuh, siaga, atau digunakan).
- Cetak & Kirim Status →
 1. Ditampilkan pada Serial Monitor untuk debugging.
 2. Dikirim ke platform Thingspeak agar bisa dipantau secara real-time dari jarak jauh.
- Pengecekan Pengiriman Data → Sistem memverifikasi apakah data berhasil terkirim.
 - o Jika Tidak, sistem mencoba mengirim ulang.
 - o Jika Ya, proses berlanjut ke tahap analisis logika.

Tahap 3 : Analisis Logika Status Baterai

- Cek Arus vs Threshold → Sistem membandingkan arus pengisian dengan nilai ambang batas.
 - o Jika Arus > Threshold → Baterai sedang mengisi.
 - Tegangan > 12V → Status: Bulk Charging (pengisian utama).
 - Tegangan > 13V → Status: Float Charging (pengisian pemeliharaan, baterai hampir penuh).
 - o Jika Arus $\leq$ Threshold → Baterai tidak sedang diisi aktif.
 - Jika Arus < 0A → Status: Discharging (baterai sedang dipakai).
 - Jika Arus $\approx$ 0A → Status: Full/Standby (baterai penuh atau idle).

Tahap 4 : Logika Kontrol & Proteksi Relay

- Cek Suhu (DS18B20) → Sistem memeriksa data suhu baterai/lingkungan.
 - o Normal (5–50°C) → Relay tetap aktif, sistem bekerja normal.
 - o Overheat (>50°C) → Relay dimatikan otomatis untuk proteksi.
 - Setelah relay mati, sistem menunggu hingga tegangan turun < 13V sebagai indikator penurunan intensitas pengisian, lalu kembali

melanjutkan operasi.

Tahap 5 : Proteksi Tegangan Rendah & Penyelesaian Siklus

- Cek Tegangan $< 10.5V \rightarrow$ Jika baterai mencapai tegangan kritis, sistem memberikan peringatan (Warning) agar pengguna segera melakukan tindakan (misalnya memutus beban atau mengganti baterai).
- Selesai $\rightarrow$ Restart Siklus $\rightarrow$ Proses satu loop selesai, kemudian sistem kembali ke tahap awal untuk memulai siklus monitoring berikutnya.

Dengan alur kerja ini, sistem tidak hanya memantau kondisi energi dan suhu secara real-time, tetapi juga melakukan kontrol otomatis (melalui relay) serta memberikan proteksi cerdas terhadap kondisi kritis baterai (overheat dan low voltage). Sistem ini dirancang agar dapat berjalan secara kontinu dan berulang, sehingga data selalu ter-update dan keamanan sistem tetap terjaga.

3.3.4. Pembuatan Dashboard Thinger.io

Dasbor pada sistem ini berfungsi sebagai antarmuka visual interaktif yang menghubungkan pengguna dengan perangkat IoT. Melalui dasbor, pengguna dapat:

- Memantau kondisi sistem secara real-time berdasarkan data yang dikirimkan sensor.
- Mengelola perangkat IoT, seperti menghidupkan/mematikan beban melalui relay.
- Melakukan analisis data historis untuk evaluasi performa

sistem.

Salah satu platform IoT yang dipilih untuk penelitian ini adalah Thingier.io, karena mendukung visualisasi data, kontrol perangkat, serta integrasi dengan ESP32 secara fleksibel.

A. Pendaftaran pada Website Thingier.io

Langkah awal sebelum menggunakan dasbor adalah membuat akun pada platform Thingier.io. Tahapan yang dilakukan sebagai berikut:

1. Mengakses Website Resmi Pengguna membuka browser lalu mengunjungi alamat <https://thingier.io>.
2. Memulai Proses Pendaftaran Pada halaman utama, pilih opsi *Sign Up* atau *Register* untuk memulai pembuatan akun baru.
3. Mengisi Formulir Pendaftaran Isi data diri berupa:
 - Nama lengkap
 - Alamat email aktif
 - Kata sandi yang diinginkan

Catatan: Email harus valid karena proses aktivasi akun memerlukan verifikasi melalui email.

1. Menyetujui Persyaratan Centang kotak persetujuan *Terms and Conditions* sebelum melanjutkan.
2. Konfirmasi Pendaftaran Setelah menekan tombol *Register*, sistem akan mengirimkan email konfirmasi. Pengguna perlu membuka email tersebut lalu mengklik tautan aktivasi untuk mengaktifkan akun Thingier.io.

3. Akses Akun Setelah proses aktivasi selesai, pengguna dapat masuk (login) ke akun Thinger.io dengan alamat email dan kata sandi yang sudah didaftarkan.

Dengan akun aktif, pengguna bisa melanjutkan ke tahap berikutnya, yaitu menghubungkan perangkat IoT ke Thinger.io dan mengatur dashboard untuk monitoring dan kontrol.



Gambar 3. 5. : Antarmuka pendaftaran pada *web* thinger.io

Sumber: thinger.io

B. Login ke Akun dan Membuat Devices

1. Login ke Akun Thinger.io Setelah proses registrasi selesai dan akun berhasil diaktifkan, pengguna dapat masuk ke platform Thinger.io:

- Kunjungi halaman utama <https://thinger.io>.
- Pilih menu Login.
- Masukkan alamat email dan kata sandi yang sudah terdaftar.
- Klik tombol Masuk (*Sign In*).

Jika informasi yang dimasukkan sesuai, sistem akan mengarahkan pengguna ke halaman utama Dashboard Thinger.io. Dari sini, seluruh pengaturan perangkat IoT dan visualisasi data dapat dilakukan.

2. Membuat Devices Baru Setelah berhasil masuk ke dashboard, langkah berikutnya adalah menambahkan perangkat IoT (ESP32) ke dalam akun agar dapat dikelola dan dipantau. Langkah-langkahnya adalah:

- Pada menu utama, pilih opsi Devices atau Perangkat.
- Klik tombol Add Device (*Tambahkan Perangkat Baru*).
- Isi informasi perangkat, antara lain:
 - o Device ID / Nama Perangkat → Berikan nama unik yang mudah dikenali, misalnya *ESP32_SmartGrid*.
 - o Description (Opsional) → Tambahkan keterangan singkat tentang fungsi perangkat.
 - o Device Credentials → Sistem akan memberikan User

ID, Device ID, dan Device Credential (Password) yang nantinya digunakan untuk autentikasi perangkat.

- Pilih Tipe Protokol Komunikasi yang digunakan. Dalam penelitian ini digunakan protokol IoTMP (Thinger Protocol) yang sudah terintegrasi dengan ESP32 dan memungkinkan komunikasi data dua arah.

3. Simpan Konfigurasi Perangkat Setelah semua informasi diisi dengan benar, klik tombol Create Device. Perangkat baru akan muncul di daftar device yang tersedia, siap untuk dihubungkan dengan ESP32 melalui kode program.

Dengan perangkat yang sudah terdaftar di Thinger.io, langkah selanjutnya adalah melakukan pengaturan koneksi pada sisi ESP32 dengan memasukkan kredensial (User ID, Device ID, dan Device Credential) ke dalam kode program, sehingga ESP32 dapat mengirim dan menerima data melalui platform cloud.

Gambar 3. 6. : Antarmuka penambahan *device* untuk *monitor*

Sumber: thinger.io

C. Penambahan Widget pada Dashboard

Setelah perangkat berhasil ditambahkan pada akun Thingiverse.io, tahap berikutnya adalah membuat dashboard visual untuk menampilkan data secara real-time. Dashboard berfungsi sebagai antarmuka pengguna (user interface) yang intuitif sehingga data dari

sensor dapat dipantau dengan mudah tanpa harus mengakses kode program secara langsung.

1. Mengakses Menu Dashboard

- Dari halaman utama, pilih menu Dashboards.
- Klik tombol Create Dashboard untuk membuat tampilan baru.
- Beri nama yang sesuai, misalnya Smart Grid Monitoring.

2. Menambahkan Widget

Widget adalah elemen visual yang menampilkan data dari perangkat IoT. Dalam sistem monitoring berbasis IoT ini, widget dipilih sesuai dengan parameter yang dipantau (tegangan, arus, daya, dan suhu). Beberapa widget penting yang digunakan adalah:

- Gauge Widget
 - o Digunakan untuk menampilkan nilai tegangan, arus, dan suhu secara instan dalam bentuk jarum analog.
 - o Memberikan gambaran cepat mengenai status sistem, misalnya apakah tegangan baterai masih dalam batas normal atau sudah mendekati batas kritis.
- Chart (Time Series) Widget
 - o Menampilkan data dalam bentuk grafik perubahan terhadap waktu (time-based).
 - o Cocok untuk memantau tren jangka panjang, seperti fluktuasi arus dari panel surya pada siang hingga malam,

atau perubahan daya yang dikonsumsi oleh beban.

- Pie Chart Widget (opsional)
 - o Dapat digunakan untuk menampilkan distribusi energi dari berbagai sumber (misalnya panel surya).
 - o Mempermudah analisis proporsi kontribusi tiap sumber energi.

3. Konfigurasi Data pada Widget

Setiap widget perlu dikaitkan dengan data sensor yang dikirim oleh perangkat ESP32. Prosesnya adalah:

- Pilih widget yang ingin ditambahkan.
- Hubungkan ke perangkat (device) yang sudah dibuat.
- Tentukan resource yang sesuai, misalnya:
 - o Voltage → untuk data tegangan dari sensor INA219.
 - o Current → untuk data arus dari sensor INA219.
 - o Power → hasil perhitungan daya dari tegangan $\times$ arus.
 - o Temperature → dari sensor DS18B20.
- Simpan konfigurasi.

4. Hasil Akhir Dashboard

Setelah semua widget ditambahkan dan dikonfigurasi, dashboard akan menampilkan kondisi sistem secara real-time.

Pengguna dapat dengan mudah:

- Melihat tegangan, arus, dan daya dari panel surya.
- Memantau suhu baterai untuk alasan keamanan.
- Melakukan analisis pola konsumsi dan produksi energi

dalam periode tertentu.

Dengan demikian, dashboard pada Thinger.io bukan hanya berfungsi sebagai tampilan monitoring, tetapi juga sebagai alat analisis visual yang membantu pengguna dalam mengevaluasi performa sistem smart grid rumahan.

Gambar 3. 7. : Antar muka penambahan *widget* ke *dashboard*

Sumber: [thinger.io](https://things.io)

3.4. Prosedur Pengumpulan Data

Prosedur pengumpulan data merupakan tahapan krusial yang dilakukan setelah sistem selesai dirakit dan diimplementasikan. Tujuan dari tahap ini adalah untuk memperoleh data kuantitatif yang digunakan dalam mengevaluasi dan menganalisis kinerja prototipe Sistem Monitoring dan Kontrol Terintegrasi untuk Manajemen Smart Grid Berbasis IoT sesuai dengan tujuan penelitian. Secara umum, pengumpulan data dibagi menjadi tiga tahapan utama:

1. Persiapan sistem dan lingkungan,
2. Pengujian fungsi monitoring, dan
3. Pengujian fungsi kontrol.

3.4.1. Tahap Persiapan Sistem dan Lingkungan

Sebelum proses pengambilan data dimulai, beberapa langkah persiapan dilakukan agar sistem siap diuji dalam kondisi optimal, yaitu:

1. Perakitan dan Instalasi Sistem

Seluruh komponen perangkat keras, yaitu Panel Surya 30 Wp, Baterai 12V 10Ah, ESP32, Sensor INA219, Sensor Suhu DS18B20, dan Modul Relay, dirangkai menjadi satu kesatuan sesuai dengan skema pada Gambar 3.3 Rangkaian Keseluruhan.

2. Konfigurasi Perangkat Lunak

Kode program (sketch) yang dikembangkan dengan Arduino IDE diunggah ke mikrokontroler ESP32. Kode ini mencakup konfigurasi koneksi Wi-Fi dan integrasi dengan platform Thinger.io.

3. Penyiapan Dashboard

Dashboard Thinger.io disusun dengan menambahkan widget yang relevan untuk memantau tegangan, arus, daya, dan suhu secara real-time.

4. Kalibrasi dan Validasi Awal

Sistem dijalankan pertama kali, kemudian hasil pembacaan sensor pada dashboard dibandingkan dengan alat ukur pembanding (Watt Meter DC). Hal ini memastikan akurasi sistem agar tidak terdapat perbedaan nilai signifikan.

3.4.2. Tahap Pengumpulan Data Monitoring

Tahap ini dilakukan untuk memperoleh data operasional sistem secara berkelanjutan.

- Aktivasi Sistem

Prototipe ditempatkan di lokasi penelitian dan dijalankan dalam kondisi normal, di mana panel surya mendapat paparan sinar matahari untuk mengisi baterai.

- Pencatatan Data Otomatis

Selama periode pengujian (misalnya 3 hari), sistem bekerja secara kontinu untuk menangkap variasi data pada kondisi cuaca berbeda (cerah, mendung). ESP32 secara otomatis:

1. Membaca tegangan dan arus dari Sensor INA219,
2. Membaca suhu dari Sensor DS18B20,
3. Mengirimkan data tersebut ke Thinger.io dengan interval tertentu (misalnya 1 menit).

3.4.3. Tahap Pengumpulan Data Uji Fungsi Kontrol

Tahap ini bertujuan menguji kemampuan sistem dalam merespons perintah kontrol, baik manual maupun otomatis.

1. Pengujian Kontrol Manual Jarak Jauh

- Peneliti mengakses dashboard Thinger.io menggunakan laptop atau smartphone.
- Perintah *ON/OFF* pada charger baterai dikirim melalui tombol kontrol di dashboard.
- Waktu respons (*latency*) diukur, yaitu selisih waktu antara perintah dikirim hingga relay berubah status. Pengujian diulang beberapa kali untuk mendapatkan nilai rata-rata.

2. Pengujian Kontrol Otomatis (Skenario Proteksi)

- Proteksi Suhu Berlebih: Suhu $> 50^{\circ}\text{C}$ disimulasikan dengan pemberian panas terkontrol di sekitar sensor DS18B20. Respon sistem, yaitu relay yang otomatis mati, diamati dan dicatat.
- Proteksi Tegangan Rendah: Sistem dibiarkan *discharging* hingga tegangan baterai $< 10.5\text{V}$. Dashboard harus menampilkan status *Warning* sebagai bukti fungsi proteksi bekerja sesuai rancangan.

3.5. Lokasi Penelitian

Penelitian dilakukan di CV. AB Electronics untuk memastikan kontrol lingkungan dan keamanan perangkat, dan dilaksanakan di Workshop AB. Electronics, jalan Gurila, Medan Area, dengan pertimbangan:

1. Ketersediaan perangkat pendukung seperti mikrokontroler, sensor, dan alat ukur listrik.
2. Akses ke jaringan listrik terkontrol untuk simulasi smart grid.
3. Dukungan infrastruktur internet untuk uji komunikasi IoT.

3.6. Jadwal Penelitian

Jadwal dibuat secara fleksibel dengan mempertimbangkan ketersediaan alat dan uji coba iterative dan pada di bulan ke-6 difokuskan pada penyusunan laporan akhir dan publikasi di jurnal atau internal kampus. Berikut adalah Tabel jadwal penelitian yang sudah direncanakan pada halaman selanjutnya

Tabel 3. 2. jadwal penelitian yang direncanakan selama 1 - 6 bulan:

N O	Kegiatan Penelitian	Bulan																							
		MARET				APRIL				MEI				JUNI				JULI				AGUSTUS			
		1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1	Studi literatur, penyusunan skripsi, dan persiapan alat/bahan.	✓	✓	✓																					
2	Perancangan sistem (desain hardware, algoritma kontrol, dan antarmuka IoT).		✓	✓	✓	✓	✓	✓																	
3	Implementasi hardware (pemasangan sensor, kalibrasi, koneksi modul IoT).	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓													
4	Implementasi software (pengembangan kode, konfigurasi platform IoT).					✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓									
5	Pengujian sistem (akurasi sensor, latensi komunikasi, respons kontrol).							✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓				
6	Analisis data, penyusunan laporan.										✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan seluruh tahapan perancangan, implementasi, dan pengujian yang telah dilaksanakan, penelitian ini berhasil merealisasikan sebuah prototipe sistem monitoring dan kontrol terintegrasi untuk manajemen *Smart Grid* skala rumah tangga berbasis *Internet of Things* (IoT). Sistem dirancang dengan memanfaatkan panel surya 30 WP sebagai sumber energi terbarukan, baterai 12 V 10 Ah sebagai media penyimpanan energi, serta mikrokontroler ESP32 sebagai unit pemroses utama yang mengintegrasikan sensor INA219 dan DS18B20 dengan platform *Thinger.io*. Secara keseluruhan, sistem mampu beroperasi secara *real-time* dalam memantau parameter kelistrikan utama sekaligus menjalankan fungsi kontrol beban secara jarak jauh.

Dari sisi kinerja monitoring, sistem menunjukkan tingkat akurasi yang memadai untuk penerapan skala kecil. Hasil kalibrasi sensor menunjukkan bahwa sensor tegangan dan sensor suhu memiliki rata-rata galat masing-masing sebesar 1,65% dan 4,09%, yang mencerminkan tingkat presisi yang tinggi. Meskipun sensor arus mencatatkan rata-rata galat yang lebih besar, yaitu sebesar 12,93% pada kondisi arus rendah, secara fungsional sistem tetap mampu menampilkan tren produksi dan konsumsi energi secara konsisten melalui dashboard *IoT*. Selain itu, platform *Thinger.io* terbukti responsif sebagai antarmuka pengguna dengan latensi pengiriman data yang relatif rendah.

Lebih lanjut, sistem ini menunjukkan keandalan dalam menjalankan fungsi proteksi otomatis guna menjaga keamanan komponen. Logika kontrol yang

diterapkan mampu memutus proses pengisian secara otomatis ketika suhu baterai terdeteksi melampaui ambang batas 50°C, sehingga potensi terjadinya *overheat* dapat dihindari. Selain itu, mekanisme proteksi tegangan berfungsi secara efektif dengan memutus beban saat tegangan turun di bawah 10 Volt serta menghentikan proses pengisian ketika tegangan melebihi 12,5 Volt. Keberhasilan penerapan fitur ini memastikan baterai terlindungi dari risiko kerusakan permanen akibat *deep discharge* maupun *overcharging*, sehingga sistem dapat dikategorikan sebagai solusi manajemen energi yang aman dan efisien.

5.1. Saran

Untuk penyempurnaan sistem di masa mendatang, beberapa aspek masih dapat dikembangkan lebih lanjut. Dari sisi perangkat keras, peningkatan akurasi pembacaan arus dapat dilakukan dengan menggunakan sensor yang memiliki resolusi lebih tinggi atau dengan menerapkan metode kalibrasi multi-titik pada mikrokontroler, khususnya untuk mendeteksi arus dalam orde miliampere. Selain itu, pengembangan sistem agar kompatibel dengan beban arus bolak-balik (AC) sangat disarankan melalui integrasi inverter dan sensor energi AC, sehingga sistem dapat diaplikasikan secara lebih luas pada peralatan elektronik rumah tangga konvensional.

Dari sisi kecerdasan sistem, penelitian selanjutnya dapat mengembangkan pendekatan kontrol dari yang semula berbasis ambang batas sederhana menuju penerapan algoritma kecerdasan buatan (*Artificial Intelligence*) atau *Machine Learning*. Pendekatan ini memungkinkan sistem untuk melakukan prediksi ketersediaan energi matahari berdasarkan pola cuaca serta karakteristik konsumsi beban, sehingga proses manajemen energi dapat berlangsung secara lebih

prediktif dan efisien.

Sebagai upaya peningkatan keandalan operasional, sistem disarankan untuk dilengkapi dengan modul penyimpanan data lokal (*data logger*) serta opsi konektivitas alternatif seperti GSM/4G atau *LoRaWAN*. Fitur ini penting untuk mengantisipasi potensi kehilangan data ketika koneksi *Wi-Fi* mengalami gangguan, sekaligus memungkinkan penerapan sistem pada lokasi terpencil. Selain itu, pengembangan aplikasi mobile khusus juga direkomendasikan guna meningkatkan pengalaman pengguna melalui penyediaan notifikasi langsung ke perangkat pengguna apabila sistem mendeteksi kondisi darurat.



DAFTAR PUSTAKA

- Abir, S. M. A. A., Anwar, A., Choi, J., & Kayes, A. S. M. (2021). IoT-Enabled Smart Energy Grid: Applications and Challenges. *IEEE Access*, 9, 50961–50981.
- Al-Ali, A. R., Al Nabulsi, A., Kalbounieh, S., McKeown, D., & Al-Ali, A. (2023). A Survey on IoT-Enabled Smart Grids: Technologies, Architectures, Applications, and Challenges. *Sustainability*, 15(1), 717.
- Alexander, C. K., & Sadiku, M. N. O. (2021). *Fundamentals of Electric Circuits* (7th ed.). New York, NY: McGraw-Hill Education.
- Andika, R. (2022). Implementasi IoT Menggunakan Platform Thinger.io pada Sistem Monitoring Daya. *Jurnal Teknologi Elektro*.
- Andrianto, H., Fudholi, A., & Sarno, R. (2022). Blockchain and IoT Integration for Secure Smart Grids: A Comprehensive Review. *Journal of Network and Computer Applications*, 200, 103311.
- Astudillo-Salinas, F., Vazquez-Rodas, A., & Minchala, L. I. (2022). Thinger.io: An Open Source Platform for Deploying Data Fusion Applications in IoT Environments. *Sensors*, 22(5).
- Gungor, V. C., Sahin, D., Kocak, T., Ergut, S., Buccella, C., Cecati, C., & Hancke, G. P. (2011). Smart Grid Technologies: Communication Technologies and Standards. *IEEE Transactions on Industrial Informatics*, 7(4), 529–539.
- Gunoto, P., & Sofyan, S. (2020). Perancangan Pembangkit Listrik Tenaga Surya 100 Wp Untuk Penerangan Lampu Di Ruang Selasar Fakultas Teknik Universitas Riau Kepulauan. *Sigma Teknika*, 3(2), 96-106.
- Hassan, M. K., Rehmani, M. H., & Chen, J. (2023). Privacy preservation in blockchain based IoT systems: Integration issues, prospects, challenges, and future research directions. *Future Generation Computer Systems*, 112, 496-512.
- Hossain, M. S., Madloul, N. A., Rahim, N. A., Selvaraj, J., Pandey, A. K., & Khan, A. F. (2021). Role of smart grid in renewable energy: An overview. *Renewable and Sustainable Energy Reviews*, 60, 1168-1184.
- Javaid, S., Javaid, N., Tahir, M., & Khan, Z. A. (2023). A smart energy management system for residential buildings using IoT. *IEEE Access*, 11, 12345-12356.

- Kabir, M. H., Dousty, M., & Pour, S. (2024). Efficient IoT Communication Protocols for Smart Grid Applications: A Review. *Journal of Network and Computer Applications*, 210, 103540.
- Khan, A. A., Khan, H. A., Azam, M. A., & Tariq, M. (2018). A Layered Architecture for IoT-Enabled Smart Grid Systems. *IEEE Access*, 6, 44266–44273.
- Li, Z., Wang, H., & Liu, Y. (2020). Machine Learning for Automated Control in Smart Grids: A Reinforcement Learning Approach. *Applied Energy*, 277, 115584.
- Rahardjo, P. (2022). Sistem penyiraman otomatis menggunakan sensor kelembaban tanah berbasis mikrokontroler Arduino Mega 2560 pada tanaman mangga harum manis Buleleng Bali. *Majalah Ilmiah Teknologi Elektro*, 21(1), 31.
- Rahman, M. S., Al-Ahmari, A., & Al-Sulaiman, F. (2021). LoRaWAN-Based Smart Grid Monitoring: A Case Study on Fault Detection and Localization. *IEEE Sensors Journal*, 21(16), 18312–18321.
- Sandi, G. H., & Fatma, Y. (2023). Pemanfaatan Teknologi Internet of Things (Iot) Pada Bidang Pertanian. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(1), 1-5.
- Shaukat, N., Ali, B., Ko, S. M., & Su, S. F. (2021). A survey on consumers empowerment, communication technologies, and renewable generation penetration within Smart Grid. *Renewable and Sustainable Energy Reviews*, 81, 1453-1475.
- Sinaga, D. H., Sasue, R. R. O., & Hutahaeen, H. D. (2021). Pemanfaatan Energi Terbarukan Dengan Menerapkan Smart Grid Sebagai Jaringan Listrik Masa Depan: Pemanfaatan Energi Terbarukan Dengan Menerapkan Smart Grid Sebagai Jaringan Listrik Masa Depan. *Journal Zetroem*, 3(1), 11-17.
- Yaqoob, I., Ahmed, E., Hashem, I. A. T., Ahmed, A. I. A., Gani, A., Imran, M., & Guizani, M. (2021). Internet of things architecture: Recent advances, taxonomy, requirements, and open challenges. *IEEE Wireless Communications*, 24(3), 10-16.
- Y. Efendi, “Internet Of Things (Iot) Sistem Pengendalian Lampu Menggunakan Raspberry Pi Berbasis Mobile,” *J. Ilm. Ilmu Komput.*, vol. 4, no. 2, pp. 21–27, 2018, doi: 10.35329/jiik.v4i2.41.
- Zhang, Y., Chen, Y., & Wang, L. (2020). IoT-Based Real-Time Monitoring System for Smart Grids. *Energies*, 13(12), 3159.

LAMPIRAN

Lampiran 1. Gambar Alat



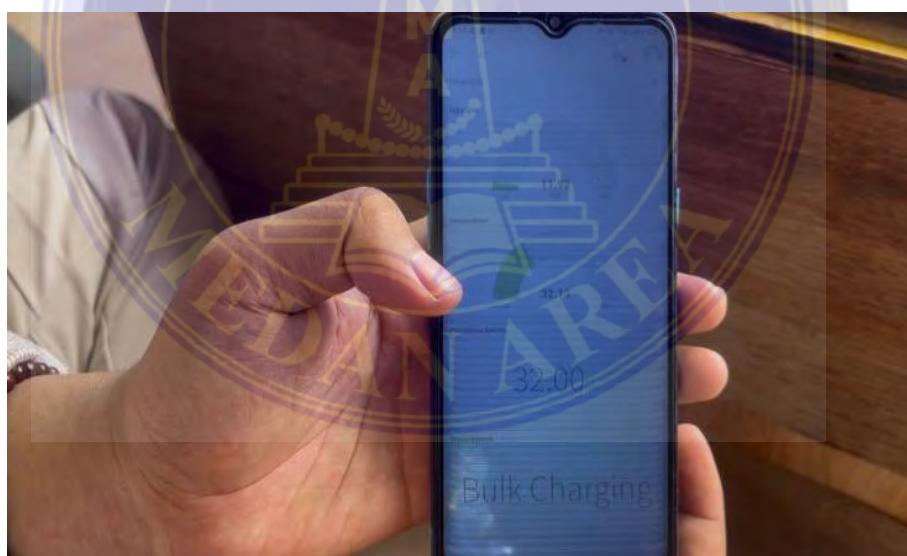
Gambar 1: Peneliti sedang melakukan penelitian dengan alat yang diteliti.



Gambar 2: Tampilan monitoring data yang masuk pada SCC.



Gambar 3: Tampilan thinger.io pada desktop.



Gambar 4: Tampilan thinger.io Mobile.

Lampiran 2. Kodingan

```
#include <WiFi.h>
#include <ThingyESP32.h>
#include <Wire.h>
#include <Adafruit_INA219.h>
#include <OneWire.h>
#include <DallasTemperature.h>
#include <math.h>

// Konfigurasi WiFi dan Thingy.io
#define USERNAME "aelonelo"
#define DEVICE_ID "ESP32"
#define DEVICE_CREDENTIAL "tqt42FgKzIcl!U5X"
#define SSID "Wifi Wannabe"
#define SSID_PASSWORD "lintas12345"

// Pin definitions
#define RELAY_PIN 5
#define DS18B20_PIN 27 // Pin untuk sensor DS18B20 (bisa GPIO 4, 5, 16, 17, dll)
#define LED_STATUS_PIN 2

// Konstanta aki (Lead-Acid Battery)
#define BATTERY_MIN_VOLTAGE 10.5 // Voltage minimum aki 12V (0% - deep discharge)
#define BATTERY_MAX_VOLTAGE 12.6 // Voltage maximum aki 12V (100% - full charge)
#define FLOAT_VOLTAGE 13.8 // Voltage float charging untuk aki 12V
#define BULK_VOLTAGE 15.1 // Voltage bulk charging untuk aki 12V
#define MAX_CHARGING_TEMP 50.0 // Suhu maksimum pengecasan aki (°C)
#define CHARGING_CURRENT_THRESHOLD 0.1 // Threshold arus untuk deteksi charging (A)

// Setup DS18B20
OneWire oneWire(DS18B20_PIN);
DallasTemperature ds18b20(&oneWire);

// Inisialisasi objek
ThingyESP32 thing(USERNAME, DEVICE_ID, DEVICE_CREDENTIAL);
Adafruit_INA219 ina219;

// Variabel global
float voltage = 0.0;
float current = 0.0;
float temperature = 0.0;
float batteryPercent = 0.0;
```

```

bool relayState = true; // true = ON (charging), false = OFF (disconnected)
bool manualControl = false;
bool ds18b20Connected = false;
String chargingStatus = "Disconnected";
unsigned long lastReadTime = 0;
const unsigned long READ_INTERVAL = 1000; // Baca sensor setiap 1 detik

void setup() {
  Serial.begin(115200);

  // Inisialisasi pin
  pinMode(RELAY_PIN, OUTPUT);
  pinMode(LED_STATUS_PIN, OUTPUT);
  digitalWrite(RELAY_PIN, HIGH); // Relay aktif (charging ON)
  digitalWrite(LED_STATUS_PIN, LOW);

  // Inisialisasi I2C untuk INA219
  Wire.begin();

  // Inisialisasi INA219
  if (!ina219.begin()) {
    Serial.println("Gagal menginisialisasi INA219!");
    while (1) { delay(10); }
  }

  // Konfigurasi INA219 untuk aki 12V dengan arus yang lebih besar
  ina219.setCalibration_32V_2A(); // Atau gunakan 32V_2A untuk arus lebih
  besar

  // Inisialisasi DS18B20
  Serial.println("Inisialisasi DS18B20...");
  ds18b20.begin();

  // Cek jumlah sensor DS18B20 yang terdeteksi
  int deviceCount = ds18b20.getDeviceCount();
  Serial.printf("Ditemukan %d sensor DS18B20\n", deviceCount);

  if (deviceCount > 0) {
    ds18b20Connected = true;
    ds18b20.setResolution(12); // Set resolusi 12-bit (0.0625°C)
    Serial.println("DS18B20 berhasil diinisialisasi!");

    // Print alamat sensor (untuk debugging multiple sensors)
    DeviceAddress tempDeviceAddress;
    for (int i = 0; i < deviceCount; i++) {
      if (ds18b20.getAddress(tempDeviceAddress, i)) {
        Serial.printf("Sensor %d address: ", i);
        for (uint8_t j = 0; j < 8; j++) {

```

```

        Serial.printf("%02X", tempDeviceAddress[j]);
    }
    Serial.println();
}
}
} else {
    Serial.println("PERINGATAN: Tidak ada sensor DS18B20 yang terdeteksi!");
    Serial.println("Sistem akan tetap berjalan dengan suhu default 25°C");
    ds18b20Connected = false;
}

Serial.println("Smart Grid Aki (Lead-Acid) Charging System dengan
DS18B20");
Serial.println("Memulai koneksi WiFi...");

// Konfigurasi Thingier.io
thing.add_wifi(SSID, SSID_PASSWORD);

// Resource untuk membaca data sensor (untuk dashboard)
thing["sensors"] >> [](pson& out) {
    out["voltage"] = voltage;
    out["current"] = current;
    out["temperature"] = temperature;
    out["battery_percent"] = batteryPercent;
    out["charging_status"] = chargingStatus;
    out["relay_state"] = relayState;
    out["ds18b20_status"] = ds18b20Connected;
};

// Resource individual untuk widget dashboard
thing["voltage"] >> [](pson& out) {
    out = voltage;
};

thing["current"] >> [](pson& out) {
    out = current;
};

thing["temperature"] >> [](pson& out) {
    out = temperature;
};

thing["battery_percent"] >> [](pson& out) {
    out = batteryPercent;
};

thing["charging_status"] >> [](pson& out) {
    out = chargingStatus;
};

```

```

};

thing["relay_state"] >> [](pson& out) {
    out = relayState;
};

// Resource untuk kontrol manual relay
thing["manual_control"] << [](pson& in) {
    if (in.is_empty()) {
        in = manualControl;
    } else {
        manualControl = in;
        if (manualControl) {
            digitalWrite(RELAY_PIN, LOW); // Matikan relay (putus charging)
            relayState = false;
            Serial.println("Manual control: Charging DISABLED");
        } else {
            if (temperature < MAX_CHARGING_TEMP) {
                digitalWrite(RELAY_PIN, HIGH); // Nyalakan relay (charging ON)
                relayState = true;
                Serial.println("Manual control: Charging ENABLED");
            }
        }
    }
};

// Resource untuk tombol emergency stop
thing["emergency_stop"] << [](pson& in) {
    digitalWrite(RELAY_PIN, LOW); // Matikan relay
    relayState = false;
    manualControl = true;
    Serial.println("EMERGENCY STOP ACTIVATED!");
};

Serial.println("Setup selesai!");
digitalWrite(LED_STATUS_PIN, HIGH); // Indikator sistem siap
}

void loop() {
    thing.handle();

    // Baca sensor setiap interval tertentu
    if (millis() - lastReadTime >= READ_INTERVAL) {
        readSensors();
        updateChargingStatus();
        safetyCheck();
        printStatus();
    }
}

```

```

    // Stream data ke Thinger.io untuk real-time monitoring
    thing.stream("sensors");

    lastReadTime = millis();
}

delay(100);
}

void readSensors() {
    // Baca INA219
    voltage = ina219.getBusVoltage_V();
    current = ina219.getCurrent_mA() / 1000.0; // Konversi ke Ampere

    // Baca DS18B20
    temperature = readDS18B20Temperature();

    // Hitung persentase baterai
    batteryPercent = calculateBatteryPercent(voltage);

    // Batasi nilai dalam range yang wajar
    if (batteryPercent > 100.0) batteryPercent = 100.0;
    if (batteryPercent < 0.0) batteryPercent = 0.0;
}

float readDS18B20Temperature() {
    if (!ds18b20Connected) {
        Serial.println("DS18B20 tidak terhubung, menggunakan suhu default");
        return 25.0; // Return default temperature
    }

    // Request pembacaan suhu dari semua sensor
    ds18b20.requestTemperatures();

    // Tunggu konversi selesai (750ms untuk 12-bit resolution)
    delay(750);

    // Baca suhu dari sensor pertama (index 0)
    float tempC = ds18b20.getTempCByIndex(0);

    // Cek apakah pembacaan valid
    if (tempC == DEVICE_DISCONNECTED_C) {
        Serial.println("ERROR: DS18B20 terputus atau gagal membaca!");
        ds18b20Connected = false;
        return 25.0; // Return default temperature
    }

    // Cek range suhu yang masuk akal

```

```

    if (tempC < -55.0 || tempC > 125.0) {
        Serial.printf("WARNING: Suhu DS18B20 di luar range normal: %.2f°C\n",
tempC);
        return 25.0;
    }

    Serial.printf("DS18B20 Temperature: %.2f°C\n", tempC);
    return tempC;
}

float calculateBatteryPercent(float voltage) {
    // Perhitungan persentase aki berdasarkan voltage curve lead-acid
    // Aki 12V memiliki karakteristik discharge yang berbeda dengan Li-ion

    if (voltage >= 12.6) {
        return 100.0;
    } else if (voltage >= 12.4) {
        return 90.0 + ((voltage - 12.4) / 0.2) * 10.0; // 90-100%
    } else if (voltage >= 12.2) {
        return 70.0 + ((voltage - 12.2) / 0.2) * 20.0; // 70-90%
    } else if (voltage >= 12.0) {
        return 40.0 + ((voltage - 12.0) / 0.2) * 30.0; // 40-70%
    } else if (voltage >= 11.8) {
        return 20.0 + ((voltage - 11.8) / 0.2) * 20.0; // 20-40%
    } else if (voltage >= 11.6) {
        return 10.0 + ((voltage - 11.6) / 0.2) * 10.0; // 10-20%
    } else if (voltage >= BATTERY_MIN_VOLTAGE) {
        return ((voltage - BATTERY_MIN_VOLTAGE) / (11.6 -
BATTERY_MIN_VOLTAGE)) * 10.0; // 0-10%
    } else {
        return 0.0; // Deep discharge
    }
}

void updateChargingStatus() {
    if (!relayState) {
        chargingStatus = "Disconnected";
    } else if (current > CHARGING_CURRENT_THRESHOLD) {
        // Tentukan mode charging berdasarkan voltage
        if (voltage < 12.0) {
            chargingStatus = "Bulk Charging";
        } else if (voltage >= 12.0 && voltage < 13.0) {
            chargingStatus = "Absorption";
        } else if (voltage >= 13.0) {
            chargingStatus = "Float Charging";
        } else {
            chargingStatus = "Charging";
        }
    }
}

```

```

    } else if (current < -CHARGING_CURRENT_THRESHOLD) {
        chargingStatus = "Discharging";
    } else {
        if (batteryPercent >= 95.0 && voltage >= 12.4) {
            chargingStatus = "Full";
        } else {
            chargingStatus = "Standby";
        }
    }
}

void safetyCheck() {
    static bool overTempWarning = false;

    // Cek suhu berlebih
    if (temperature > MAX_CHARGING_TEMP && !manualControl) {
        if (!overTempWarning) {
            Serial.println("WARNING: Suhu berlebih! Memutus charging...");
            overTempWarning = true;
        }
        digitalWrite(RELAY_PIN, LOW); // Matikan relay
        relayState = false;

        // Blink LED sebagai indikator warning
        digitalWrite(LED_STATUS_PIN, (millis() / 500) % 2);

    } else if (temperature < (MAX_CHARGING_TEMP - 5.0) && !manualControl)
    {
        // Hysteresis 5°C untuk menghindari osilasi
        if (overTempWarning) {
            Serial.println("Suhu normal kembali. Melanjutkan charging...");
            overTempWarning = false;
        }
        digitalWrite(RELAY_PIN, HIGH); // Nyalakan relay
        relayState = true;
        digitalWrite(LED_STATUS_PIN, HIGH);
    }

    // Cek voltage berlebih (overcharge protection untuk aki)
    if (voltage > BULK_VOLTAGE && !manualControl) {
        Serial.println("WARNING: Voltage charging berlebih! Memutus charging...");
        digitalWrite(RELAY_PIN, LOW);
        relayState = false;
    }

    // Cek voltage terlalu rendah (deep discharge protection)
    if (voltage < BATTERY_MIN_VOLTAGE) {
        Serial.println("WARNING: Aki dalam kondisi deep discharge!");
    }
}

```

```

    // Tetap izinkan charging untuk recovery aki
}
}

void printStatus() {
    Serial.println("=== Smart Grid Aki Charging Status ===");
    Serial.printf("Voltage: %.2f V\n", voltage);
    Serial.printf("Current: %.3f A\n", current);
    Serial.printf("Temperature: %.1f °C\n", temperature);
    Serial.printf("DS18B20 Status: %s\n", ds18b20Connected ? "Connected" :
"Disconnected");
    Serial.printf("Aki Level: %.1f%%\n", batteryPercent);
    Serial.printf("Status: %s\n", chargingStatus.c_str());
    Serial.printf("Relay: %s\n", relayState ? "ON" : "OFF");
    Serial.printf("Manual Control: %s\n", manualControl ? "ACTIVE" : "AUTO");

    // Tambahan info untuk aki
    if (voltage < 10.8) {
        Serial.println("ALERT: Aki perlu segera di-charge!");
    } else if (voltage > 14.9) {
        Serial.println("ALERT: Voltage charging terlalu tinggi!");
    }

    // Status sensor DS18B20
    if (temperature > MAX_CHARGING_TEMP - 5.0) {
        Serial.printf("WARNING: Suhu mendekati batas maksimum (%.1f°C)\n",
MAX_CHARGING_TEMP);
    }

    Serial.println("=====");
    Serial.println();
}

```