

**PERANCANGAN *SMART GRID* DENGAN DISTRIBUSI
BEBAN LISTRIK BERBASIS IOT**

SKRIPSI

Disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik (S.T.)



Disusun oleh:

VIERY RACHMANSYAH PUTRA

3332170073

**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNIK
UNIVERSITAS SULTAN AGENG TIRTAYASA
2022**

LEMBAR PERNYATAAN KEASLIAN SKRIPSI

Dengan ini saya sebagai penulis Skripsi berikut:

Judul : Perancangan *Smart Grid* Dengan Distribusi Beban Listrik Berbasis IoT.

Nama Mahasiswa : Viery Rachmansyah Putra

NPM : 3332170073

Fakultas/Jurusan : Teknik / Teknik Elektro

Menyatakan dengan sesungguhnya bahwa skripsi tersebut di atas adalah benar-benar hasil karya asli saya dan tidak memuat hasil karya orang lain, kecuali dinyatakan melalui rujukan yang benar dan dapat dipertanggung jawabkan. Apabila dikemudian hari ditemukan hal-hal yang menunjukkan bahwa sebagian atau seluruh karya ini bukan karya saya, maka saya bersedia dituntut melalui hukum yang berlaku. Saya juga bersedia menanggung segala akibat hukum yang timbul dari pernyataan yang secara sadar dan sengaja saya nyatakan melalui lembar ini.

Cilegon, Juni 2022



Viery Rachmansyah Putra
3332170073

LEMBAR PENGESAHAN

Dengan ini ditetapkan bahwa Skripsi berikut.

Judul : Perancangan *Smart Grid* Dengan Distribusi Beban Listrik Berbasis IoT.

Nama Mahasiswa : Viery Rachmansyah Putra

NPM : 3332170073

Fakultas/Jurusan : Teknik / Teknik Elektro

Telah diuji dan dipertahankan pada tanggal ____ Juni 2022 melalui Sidang Skripsi di Fakultas Teknik Universitas Sultan Ageng Tirtayasa Cilegon dan dinyatakan **LULUS**

Dewan Penguji

Pembimbing I : Dr. Eng. Rocky Alfanz, M.Sc.

Penguji I : Dr. Irma Saraswati, S.Si., M.T.

Penguji II : Masjudin, S.T., M.Eng.

Tanda Tangan



Mengetahui,

Ketua Jurusan



Dr. Romi Wiryadinata, S.T., M. Eng.
NIP. 198307032009121006

PRAKATA

Puji syukur saya panjatkan kepada Tuhan Yang Maha Esa, karena atas berkat dan rahmat-Nya, penulis dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik, Jurusan Teknik Elektro pada Fakultas Teknik Universitas Sultan Ageng Tirtayasa. Penulis menyadari bahwa, tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi penulis untuk menyelesaikan skripsi ini. Oleh karena itu, penulis mengucapkan terima kasih kepada:

1. Ayahanda dan ibunda, yang tidak henti-hentinya memberikan semangat, doa dan selalu memotivasi penulis dalam penulisan skripsi ini.
2. Dr. Romi Wiryadinata, S.T., M.Eng., selaku Ketua Jurusan Teknik Elektro Fakultas Teknik Universitas Sultan Ageng Tirtayasa.
3. Dr. Eng. Rocky Alfanz, S.T., M.Sc., selaku Dosen Pembimbing I Skripsi yang telah membantu serta memberikan arahan, saran dan bimbingan kepada penulis selama proses menyelesaikan Skripsi.
4. Ceri Ahendriyanti, S.T., M.Eng., selaku Dosen Pembimbing Akademik penulis yang telah memberikan arahan dan motivasi kepada penulis.
5. Ucapan terima kasih juga penulis sampaikan kepada seluruh dosen, staf akademik dan teman-teman Jurusan Teknik Elektro Fakultas Teknik Universitas Sultan Ageng Tirtayasa yang telah membantu penulis berupa semangat, dukungan, dan saran dalam menyelesaikan skripsi ini.

Akhir kata, penulis berharap Tuhan Yang Maha Esa berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini membawa manfaat bagi pengembangan ilmu.

Cilegon, Juni 2022

Penulis

ABSTRAK

Viery Rachmansyah Putra
Teknik Elektro

PERANCANGAN *SMART GRID* DENGAN DISTRIBUSI BEBAN LISTRIK BERBASIS IoT

Kebutuhan energi listrik untuk pemakaian beban listrik rumah tangga mengalami peningkatan setiap harinya, maka dari itu sistem Pembangkit Listrik Tenaga Surya (PLTS) yang sudah dibangun tidak dapat mencukupi kebutuhan beban listrik diluar dari rancangan awal, sehingga PLTS tidak sanggup menyuplai daya listrik untuk digunakan oleh beban listrik. Tujuan penelitian ini dirancang sebuah alat untuk mengoptimalkan pemakaian daya listrik keluran PLTS dengan alat *smart grid*, sehingga dapat menghemat penggunaan daya listrik dari Perusahaan Listrik Negara (PLN). Metode dari alat *smart grid* meliputi otomatis distribusi yang dapat mengalihkan daya dari sumber utama ke sumber tambahan, dan alat *smart grid* dapat mengendalikan dan memantau penggunaan daya listrik dari sumber PLTS dan PLN secara *real time* melalui *Internet of Thing* (IoT). Hasil dari alat *smart grid* ini menggunakan beban sebesar 770 Wh dan menunjukkan rata-rata persentasi dari penggunaan daya listrik dari PLN sebesar 70,65% dan penghematan daya listrik sebesar 29,35%.

Kata Kunci : *Smart Grid*, PLTS, *Internet of Things*

ABSTRACT

Vierey Rachmansyah Putra
Electrical Engineering

SMART GRID DESIGN WITH IoT-BASED ELECTRIC LOAD DISTRIBUTION

The need for electrical energy for the use of household electrical loads is increasing every day, therefore the Solar Power Plant (PLTS) system that has been built cannot cover the needs of electrical loads outside of the initial design, so that PLTS is not able to supply electrical power to be used by the load. electricity. In this study, a tool is designed to optimize the use of PLTS output power with a smart grid device, so as to save the use of electricity from the State Electricity Company (PLN). The methods of smart grid tools include automatic distribution that can divert power from the main source to additional sources, and smart grid tools can control and monitor the use of electric power from PLTS and PLN sources in real time through the Internet of Things (IoT). The results of this smart grid tool use a load of 770 Wh and show an average percentage of electricity usage from PLN of 70.65% and electricity saving of 29.35%.

Key word : *Smart Grid, PLTS, Internet of Things*

DAFTAR ISI

	Halaman
LEMBAR HALAMAN.....	i
LEMBAR PERNYATAAN KEASLIAN SKRIPSI	ii
LEMBAR PENGESAHAN	iii
PRAKATA.....	iv
ABSTRAK	v
<i>ABSTRACT</i>	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xii
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 II TINJAUAN PUSTAKA.....	6
2.1 <i>Smart Grid</i>	6
2.2 Peralatan Listrik Rumah Tangga	7
2.3 Pembangkit Listrik Tenaga Surya (PLTS).....	8
2.4 PLTS <i>Off Grid</i>	8
2.4.1 Panel Surya	9
2.4.2 <i>Inverter</i>	11
2.4.3 Baterai.....	12
2.5 ESP32.....	13
2.6 Sensor PZEM-004t.....	13
2.7 Sensor ACS712	14
2.8 Sensor Tegangan.....	15
2.9 Modul RTC (<i>Real Time Clock</i>).....	15

2.10	Relai	16
2.11	<i>Liquid Crystal Display (LCD)</i>	17
2.12	<i>Platform Thinger.io</i>	17
2.13	Kajian Pustaka	18
BAB III METODOLOGI PENELITIAN		21
3.1	Perancangan Penelitian	21
3.1.1	Diagram Blok.....	21
3.1.2	<i>Flowchart</i> Sistem.....	22
3.2	Perancangan Sistem <i>Smart Grid</i>	23
3.2.1	Rangkaian Mikrokontroler Sistem <i>Smart Grid</i>	24
3.2.2	Rangkaian <i>Input</i> Sistem <i>Smart Grid</i>	25
3.2.3	Rangkaian <i>Output</i> Sistem <i>Smart Grid</i>	26
3.3	Perancangan Sistem <i>Monitoring Smart Grid</i>	27
3.4	Perancangan Pembangkit Listrik Tenaga Surya	28
3.4.1	Kebutuhan Beban.....	28
3.4.2	Komponen Sistem PLTS	29
3.4.3	Perancangan Sistem PLTS.....	30
BAB IV HASIL DAN PEMBAHASAN		31
4.1	Hasil Perancangan Alat.....	31
4.2	Pengujian Sensor PZEM-004t.....	33
4.3	Pengujian Sensor ACS712.....	36
4.4	Pengujian Sensor Tegangan.....	38
4.5	Pengujian Modul RTC	39
4.6	Pengujian Sistem <i>Monitoring Smart Grid</i>	39
4.7	Pengujian Sistem Kendali Jarak Jauh <i>Smart Grid</i>	41
4.8	Pengujian Sistem Kendali Otomatis <i>Smart Grid</i>	42
4.8.1	Pengujian <i>Setpoint</i> Otomatis Beban Listrik.....	42
4.8.2	Pengujian <i>Setpoint</i> Otomatis Lampu	43
4.9	Hasil Pengujian Sistem <i>Smart Grid</i>	44
BAB V PENUTUP.....		50
5.1	Kesimpulan	50
5.2	Saran	50

DAFTAR PUSTAKA	51
LAMPIRAN A PERALATAN LISTRIK.....	A-1
LAMPIRAN B HASIL PENGUJIAN	B-1
LAMPIRAN C <i>LISTING PROGRAM</i>	C-1

DAFTAR GAMBAR

Gambar 2.1 Konsep <i>Smart Grid</i>	6
Gambar 2.2 Skema Dasar PLTS <i>Off Grid</i>	8
Gambar 2.3 Proses Efek <i>Photovoltaic</i>	9
Gambar 2.4 Bagian-bagian Panel Surya	10
Gambar 2.5 <i>Inverter</i>	11
Gambar 2.6 Baterai	12
Gambar 2.7 <i>Pin Out</i> Mikrokontroler ESP32.....	13
Gambar 2.8 Sensor PZEM-004t.....	14
Gambar 2.9 Sensor ACS712	14
Gambar 2.10 Sensor Tegangan	15
Gambar 2.11 Modul RTC (<i>Real Time Clock</i>)	16
Gambar 2.12 Relai	16
Gambar 2.13 LCD 20x4 I2C.....	17
Gambar 2.14 Model Komunikasi Thinger.io	18
Gambar 3.1 Diagram Bloak Alat	21
Gambar 3.2 <i>Flowchart Sistem</i>	22
Gambar 3.3 Diagram Blok Sistem <i>Smart Grid</i>	24
Gambar 3.4 Rangkaian Sistem <i>Smart Grid</i>	25
Gambar 3.5 Rangkaian <i>Input</i> Mikrokontroler.....	25
Gambar 3.6 Rangkaian <i>Output</i> Mikrokontroler.....	26
Gambar 3.7 <i>Dashboard Monitoring</i> Web Thinger.io	27
Gambar 3.8 <i>Data Bucket</i> Thinger.io	28
Gambar 3.9 Rancangan Penempatan PLTS	30
Gambar 4.1 <i>Box Panel Smart Grid</i>	31
Gambar 4.2 PLTS	32
Gambar 4.3 <i>Storage System</i>	32
Gambar 4.4 Pengujian Nilai Tegangan Sensor 1 PZEM-004t.....	33
Gambar 4.5 Pengujian Nilai Tegangan Sensor 2 PZEM-004t.....	34
Gambar 4.6 Pengujian Nilai Arus Sensor 1 PZEM-004t.....	34

Gambar 4.7 Pengujian Nilai Arus Sensor 2 PZEM-004t.....	35
Gambar 4.8 Pengujian Nilai Frekuensi Sensor 1 PZEM-004t.....	35
Gambar 4.9 Pengujian Nilai Frekuensi Sensor 2 PZEM-004t.....	36
Gambar 4.10 Pengujian Nilai Arus Sensor 1 ACS712	37
Gambar 4.11 Pengujian Nilai Arus Sensor 2 ACS712	37
Gambar 4.12 Pengujian Nilai Sensor Tegangan	38
Gambar 4.13 Pengujian Modul RTC	39
Gambar 4.14 Pengujian <i>Dashboard</i> Thinger.io	40
Gambar 4.15 Pengujian <i>Data Bucket</i> Thinger.io	40
Gambar 4.16 <i>Dashboard</i> Kendali Jarak Jauh Thinger.io.....	41
Gambar 4.17 Pengujian Alat <i>Smart Grid</i> Hari ke-1.....	44
Gambar 4.18 Pengujian Nilai Tegangan Baterai Hari ke-1	45
Gambar 4.19 Pengujian Alat <i>Smart Grid</i> Hari ke-2.....	46
Gambar 4.20 Pengujian Nilai Tegangan Baterai Hari ke-2	46
Gambar 4.21 Pengujian Alat <i>Smart Grid</i> Hari ke-3.....	47
Gambar 4.22 Pengujian Nilai Tegangan Baterai Hari ke-3	48
Gambar 4.23 Konsumsi Daya Listrik	49

DAFTAR TABEL

Tabel 3.1 Perencanaan Kebutuhan Beban.....	29
Tabel 3.2 Komponen Sistem PLTS.....	30
Tabel 4.1 Pengujian Kendali Jarak Jauh Thinger.io	41
Tabel 4.2 Pengujian <i>Setpoint</i> Otomatis Beban Listrik.....	42
Tabel 4.3 Pengujian <i>Setpoint</i> Otomatis Lampu	43

BAB I

PENDAHULUAN

1.1 Latar Belakang

Penggunaan energi di Indonesia dalam skala nasional masih didominasi oleh energi tak terbarukan yang berasal dari fosil, terutama minyak bumi dan batu bara [1]. Penggunaan energi fosil dapat menyebabkan emisi karbon meningkat dari hari ke hari, menyebabkan fenomena yang disebut pemanasan global yang dapat merusak lingkungan [2]. Pemerintah mengambil tindakan untuk mencegah terjadinya kerusakan lingkungan dengan membentuk peraturan Menteri Energi dan Sumber Daya Mineral (ESDM). Peraturan tersebut mengatur penggunaan sumber Energi Baru Terbarukan (EBT) sebagai sumber tenaga listrik, salah satunya adalah penggunaan energi surya. Energi surya atau matahari diubah menjadi energi listrik menggunakan panel surya [3].

Panel surya merupakan komponen utama dari Pembangkit Listrik Tenaga Surya (PLTS), yang dimana PLTS termasuk sebagai pembangkit listrik EBT karena bersumber dari matahari, yang mana energi matahari selalu ada tiap tahunnya [4]. PLTS memperoleh energi listrik melalui sel surya yang bekerja dengan efek *photovoltaic*. Efek *photovoltaic* mengubah atau mengkonversikan radiasi sinar foton matahari menjadi energi listrik. Sistem PLTS salah satunya adalah sistem PLTS *off grid* yang keluarannya berupa listrik bolak-balik atau *Alternating Current* (AC) dan dapat digunakan pada peralatan listrik rumah tangga [5].

Kebutuhan energi listrik sangat penting untuk kehidupan, sehingga perlu adanya energi listrik yang berkelanjutan. Gangguan atau ketidakmampuan pasokan tenaga listrik biasanya disebabkan oleh pemadaman listrik yang dilakukan oleh Perusahaan Listrik Negara (PLN) karena keterbatasan pembangkit atau gangguan jaringan. Cara untuk mengantisipasi gangguan perlu adanya sumber energi alternatif untuk pengganti suplai dari PLN yang mengalami *black out* atau terputus, seperti PLTS [6]. Perpindahan energi listrik dari sumber listrik PLN terhadap sumber listrik PLTS tidak boleh adanya jeda waktu perpindahan karena menyebabkan perangkat elektronika tidak berjalan, sehingga ditambahkan sistem pendukung

berupa otomatis distribusi [7]. Otomatis distribusi ini memungkinkan untuk mengalihkan daya dari sumber utama ke sumber cadangan, tetapi teknologi ini masih dianggap konvensional, karena sistem otomatis hanya dapat memindahkan daya dan tidak dapat dipantau atau dikendalikan dari jarak jauh [8].

Penggunaan teknologi yang tidak tepat dalam mengalihkan sumber energi listrik utama ke sumber EBT dapat menyebabkan masalah stabilitas, keandalan, dan kualitas daya pada jaringan listrik utama atau cadangan. Penggunaan teknologi *smart grid* adalah kunci untuk memecahkan masalah ini [9]. Teknologi *smart grid* dapat membentuk sistem jaringan cerdas yang dapat beralih dari sumber satu ke sumber lainnya, dan daya listrik dari sumber yang digunakan dapat didistribusikan ke jala-jala beban listrik. Sumber energi listrik yang berasal dari EBT harus mengikuti standar kualitas daya listrik dan jaringan distribusi dari PLN [10]. Teknologi *smart grid* ini juga dapat memantau titik-titik tertentu yang perlu dipantau dan dapat dikendalikan oleh pengguna agar dapat menyediakan sumber daya yang efisien, berkelanjutan, ekonomis, dan aman. [11].

Perkembangan teknologi saat ini tumbuh semakin cepat, memungkinkan *monitoring* dan pengendalian semula hanya dapat dilakukan jarak dekat, sekarang dapat dilakukan jarak jauh, yang mana konsep tersebut dinamakan *Internet of Things* (IoT). IoT semakin banyak digunakan sebagai teknologi dan *platform* yang menghubungkan manusia, sistem, perangkat, dan berbagai hal lainnya [12]. IoT memastikan dunia terhubung dan memiliki tujuan untuk memperluas manfaat untuk mempermudah pekerjaan manusia [13]. IoT pada sistem perangkat keras menggunakan sarana komunikasi internet, sehingga IoT dapat dipantau secara *real time* [14]. IoT memungkinkan pengguna untuk secara otomatis mengendalikan sistem melalui internet tanpa mengenal jarak untuk mengumpulkan, memproses, dan bertukar data yang bermanfaat [15].

Berdasarkan penjelasan di atas, penelitian ini merancang sistem *smart grid* dengan distribusi beban listrik antara sumber PLN dengan PLTS begitu juga dengan sebaliknya. Perancangan alat ini juga dapat memantau konsumsi daya listrik yang telah digunakan oleh konsumen yang bersumber dari PLTS maupun PLN yang dapat memanfaatkan IoT sehingga pemantauan dapat dilakukan secara jarak jauh.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang sudah dipaparkan di atas maka rumusan masalah yang diangkat pada penelitian ini sebagai berikut:

1. Bagaimana merancang sistem *smart grid* dengan distribusi beban listrik antara sumber listrik PLN dengan PLTS?
2. Bagaimana merancang sistem *smart grid* yang dapat memantau penggunaan daya dari sumber listrik PLTS dan jaringan listrik PLN menggunakan metode IoT?
3. Seberapa besar penghematan konsumsi daya jaringan listrik PLN bila menggunakan sistem *smart grid* yang dipadukan dengan sumber listrik PLTS?

1.3 Tujuan Penelitian

Tujuan penelitian yang ingin dicapai dalam penelitian ini sebagai berikut:

1. Merancang sistem *smart grid* dengan distribusi beban listrik antara sumber listrik PLTS dan PLN.
2. Merancang sistem *monitoring* penggunaan listrik yang berumber dari PLTS dan jaringan listrik PLN memanfaatkan IoT.
3. Menghemat penggunaan daya listrik dari jaringan PLN dengan sistem kendali *smart grid* yang dikombinasikan dengan daya listrik dari PLTS.

1.4 Manfaat Penelitian

Manfaat yang dapat diberikan dari penelitian ini sebagai berikut:

1. Hasil penelitian diharapkan dapat menjadi referensi untuk akademisi yang melakukan kajian tentang sistem *smart grid* untuk PLTS.
2. Hasil penelitian diharapkan dapat memberikan wawasan dan pengetahuan dalam bidang PLTS.
3. Hasil penelitian diharapkan menjadi perkembangan dan pembaruan informasi bidang PLTS.

1.5 Batasan Masalah

Batasan masalah pada penelitian ini sebagai berikut:

1. Sumber listrik berasal dari sumber listrik PLN dengan kapasitas 1300 VA dan sumber listrik PLTS dengan kapasitas 2790 Wp.
2. Sistem PLTS yang dirancang adalah *off grid*.
3. Mikrokontroler yang digunakan untuk melakukan pemrosesan data adalah mikrokontroler ESP32.
4. Parameter dari alat *smart grid* berdasarkan dari pembacaan tegangan, arus, frekuensi, tegangan baterai, dan *timer*.
5. Pengaturan *timer* diperuntukan untuk menyalakan beban lampu pada waktu 18:00 s.d. 05:00.
6. Pembahasan hanya terfokus pada hasil monitoring yaitu, pembacaan arus, tegangan dan frekuensi dari PLTS dan PLN, dan juga menganalisis daya yang digunakan dari PLTS dan PLN.
7. Pengujian alat ini dilakukan dengan pengambilan data secara *real time*.

1.6 Sistematika Penulisan

Sistematika penulisan merupakan urutan garis besar penulisan skripsi. Berikut ini dituliskan pokok bahasan untuk memberikan gambaran yang jelas mengenai susunan materi yang dibahas dari tiap-tiap bab.

BAB I PENDAHULUAN

Bab ini berisi tentang penjelasan mengenai latar belakang masalah, tujuan penelitian, rumusan masalah, batasan masalah, manfaat penelitian, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Bab ini berisi penjelasan tentang konsep dasar perangkat yang dipakai dalam PLTS dan sistem *smart grid* untuk mendukung sistem distribusi dan *monitoring* menggunakan IoT.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan tentang metode penelitian, alur penelitian, perancangan perangkat keras (*hardware*) maupun perangkat lunak (*software*).

BAB IV HASIL DAN PEMBAHASAN

Bab ini membahas tentang hasil penelitian dan pembahasan yang disampaikan berupa penjelasan dari hasil pengolahan data penelitian yang dikerjakan.

BAB V PENUTUP

Bab ini berisi kesimpulan yang menguraikan rangkuman yang disimpulkan dari hasil penelitian dan pembahasan yang dilakukan beserta saran untuk pengembangan penelitian lebih lanjut.

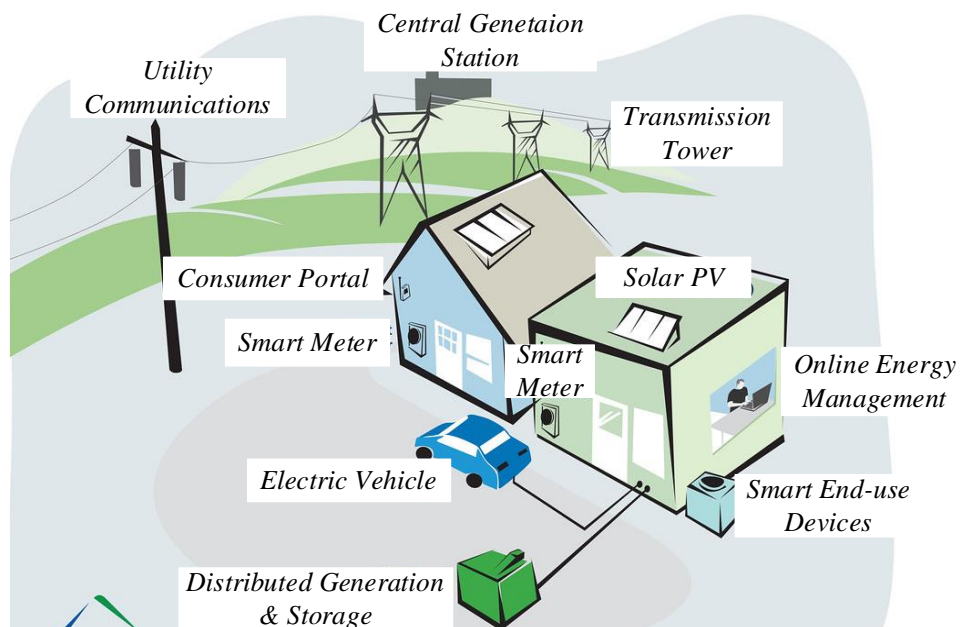
BAB II

TINJAUAN PUSTAKA

2.1 *Smart Grid*

Smart grid merupakan konsep jaringan listrik modern yang perencanaan, pengembangan, dan penelitiannya telah dimulai sejak 10 tahun yang lalu. Penelitian *smart grid* mempunyai visi bahwa jaringan *power system* harus fleksibel dalam memenuhi kebutuhan konsumen, mudah diakses, reliabel, dan ekonomis.

Kelayakan dan keandalan suplai listrik merupakan salah satu parameter yang vital dalam distribusi energi khususnya untuk lintas aplikasi atau operator. Informasi yang tersedia di masing-masing area pembangkit, transmisi, dan distribusi biasanya hanya untuk masing-masing jaringan listrik lokal dan data sistem tersebut belum berbasiskan data yang *real time*. Permasalahan tersebut yang menjadi tantangan ke depan agar jaringan listrik mampu lebih berkualitas dengan tingkat keandalan yang tinggi serta aman dari gangguan. Gambar 2.1 adalah konsep dari *smart grid*.



Gambar 2.1 Konsep *Smart Grid* [16]

Gambar 2.1 menunjukkan konsep serta model *smart grid*. Jaringan *smart grid* terdapat beberapa peralatan digital seperti data kolektor dan *recording*, otomatisasi,

sensor, *smart meter*, *real time data display*, data manajemen, dan aplikasi komunikasi dua arah ditambahkan dalam jaringan listrik [16].

2.2 Peralatan Listrik Rumah Tangga

Peralatan listrik rumah tangga ialah peralatan yang besumber dari sambungan tegangan listrik AC, baik yang bersumber dari PLN maupun sumber lain. Tegangan listrik AC terpasang pada rumah tangga harus berstandar terutama di Indonesia, implementasinya adalah tegangan standar antara hantaran fasa ke hantaran netral adalah 220 Volt dengan frekuensi 50 Hz [17].

Ketika beban peralatan listrik rumah tangga aktif, beban tersebut mengonsumsi tegangan dan arus terhadap waktu yang disebut dengan daya. Daya aktif (*aktif power*) adalah daya yang terpakai untuk melakukan energi sebenarnya, dan satuan daya aktif adalah Watt. Daya aktif dapat dihitung dalam Persamaan (2.1) [13]:

$$P = V.I.\cos\varphi \quad (2.1)$$

Keterangan:

P = Daya

V = Tegangan

I = Arus

$\cos\varphi$ = Faktor daya

Persamaan (2.1) adalah persamaan daya aktif yang dimana, hasil daya aktif dihasilkan dari perkalian tegangan, arus, dan faktor daya. Perhitungan konsumsi daya sehari dalam Persamaan (2.2) [18]:

$$\text{Konsumsi Daya}(Wh) = t.P.n \quad (2.2)$$

t = Lama beban dihidupkan (jam)

P = Daya beban (watt)

n = Jumlah beban

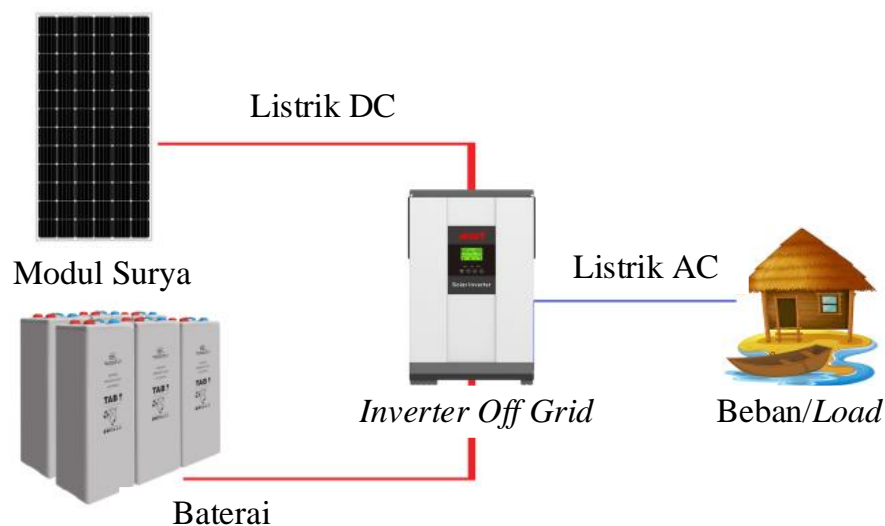
Persamaan (2.2) adalah persamaan konsumsi daya, dalam perhitungan konsumsi daya dihasilkan melalui besar daya listrik yang digunakan beban dan lama beban digunakan dalam satu hari [18].

2.3 Pembangkit Listrik Tenaga Surya (PLTS)

Pembangkit Listrik Tenaga Surya (PLTS) adalah suatu sistem pembangkit listrik dimana energi matahari diubah menjadi energi listrik dengan memanfaatkan teknologi *PhotoVoltaic* (PV) atau sel surya. *Photovoltaic* atau sel surya adalah piranti semikonduktor yang dapat merubah radiasi matahari secara langsung menjadi arus listrik searah atau *Direct Current* (DC) dengan menggunakan kristal Silicon (Si) yang tipis. Proses di dalam sel surya, tegangan yang dihasilkan tergantung dari intensitas energi yang terkandung dalam sinar matahari yang sampai ke permukaan bumi [18].

2.4 PLTS *Off Grid*

PLTS *off grid* sering disebut juga PLTS *stand alone* artinya sistem hanya disuplai oleh panel surya saja tanpa ada pembangkit jenis lain misalnya PLTD. Sistem tipe ini hanya tergantung pada matahari seutuhnya, karena panel tidak mungkin mendapatkan sinar matahari terus menerus terutama malam hari, maka sistem ini membutuhkan media penyimpan yaitu baterai. Gambar 2.2 adalah skema dari PLTS tipe *off grid*.



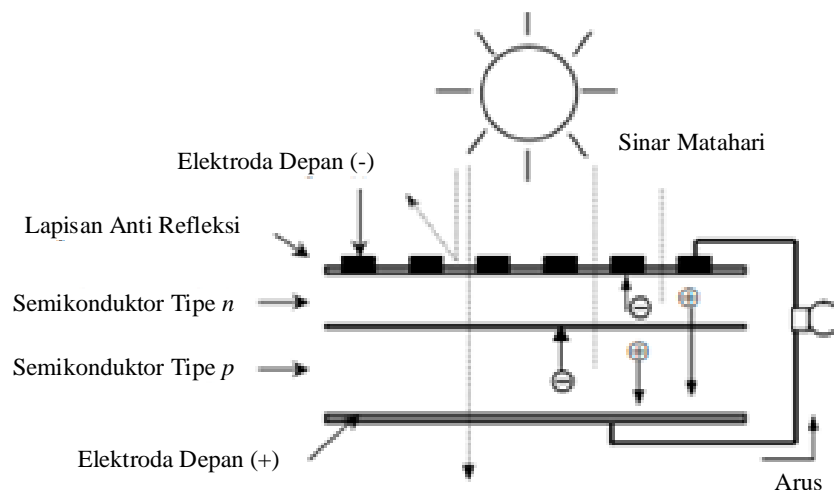
Gambar 2.2 Skema Dasar PLTS *Off Grid* [19]

Gambar 2.2 adalah skema PLTS *off grid*, suatu PLTS dengan konfigurasi *off grid* memiliki komponen yang diantaranya panel surya, *Solar Charge Controller*

(SCC), baterai dan *inverter*. Menentukan kapasitas panel dan baterai untuk digunakan harus dihitung secara akurat, karena jika perhitungannya tidak tepat mengakibatkan ketidaksanggupan PLTS menyuplai beban listrik karena baterai kehabisan daya listrik. Sistem *off grid* umumnya kapasitas baterai ditambah untuk mengantisipasi dihari tersebut tidak ada sinar matahari atau kondisi cuaca berawan yang disebut *Days of Autonomy (DoA)*, dan berdasarkan pertimbangan biaya, kapasitas baterai ditambahkan dari 1 s.d. 2 kali periodenya [19].

2.4.1 Panel Surya

Secara sederhana sel surya terdiri dari persambungan bahan semikonduktor bertipe *p* dan *n* (*p-n junction semiconductor*) yang jika terkena matahari maka akan terjadi aliran elektron, aliran elektron inilah yang disebut sebagai aliran arus listrik. Proses efek *photovoltaic* dapat dilihat Gambar 2.3 di bawah ini.



Gambar 2.3 Proses Efek *Photovoltaic* [18]

Gambar 2.3 adalah proses dari efek *photovoltaic*. Prinsip kerja sel surya Ketika sinar matahari terdiri dari foton-foton jatuh pada permukaan bahan sel surya (*absorber*), hanya foton dengan tingkat energi tertentu yang membebaskan elektron dari ikatan atomnya, sehingga mengalir arus listrik. Tingkat energi ini disebut energi *band-gap* yang didefinisikan sebagai sejumlah energi elektron menjadi pembawa *n* dan meninggalkan *hole*, pembawa *p*. Pembawa *p* akan bergerak menuju persambungan demikian juga pembawa *n* akan bergerak ke persambungan, perpindahan tersebut menghasilkan beda potensial. Arus dan daya yang dihasilkan

PV ini dapat dialirkan ke rangkaian luar. Energi foton terlalu besar dari energi *band-gap*, maka ekstra energi tersebut akan dirubah dalam bentuk panas pada sel surya [18].

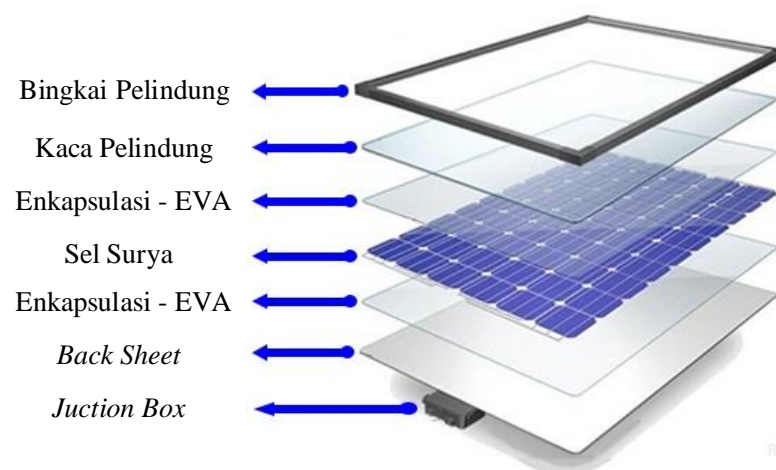
Sel surya adalah bentuk sel PV yang kapasitas daya panel surya diukur dalam satuan Wp (Watt *peak*). Perhitungan untuk kebutuhan banyak panel surya akan digunakan dalam sistem PLTS dapat dilihat di Persamaan (2.3) [18]:

$$\text{Kapasitas panel surya} = \frac{\text{Total konsumsi harian}}{\text{waktu penyinaran matahari}} \quad (2.3)$$

Persamaan (2.3) adalah persamaan untuk mencari kebutuhan panel surya yang dihasilkan oleh pembagian dari total konsumsi harian dengan waktu penyinaran matahari. Panel surya memiliki jenis-jenis yang berbeda yang digolongkan berdasarkan teknologi pembuatannya, dan salah satu jenis sel surya yang di pakai dalam penelitian ini adalah bertipe *monocrystalline*.

1. *Monocrystalline*

Monocrystalline terbuat dari batangan kristal silikon murni yang diiris tipis-tipis. Jenis panel surya *monocrystalline* terdiri dari batang kristal Silikon murni yang diiris tipis dan memerlukan teknologi khusus untuk memotong kristal Silikon menjadi potongan tipis. Teknologi sel surya *Monocrystalline* menjadi paling efisien dibandingkan jenis sel surya lainnya yang memiliki nilai efisiensi sekitar 15% s.d. 20% [18]. Bagian-bagian dari panel surya *monocrystalline* dapat dilihat di Gambar 2.4.



Gambar 2.4 Bagian-bagian Panel Surya [18]

Gambar 2.4 adalah bagian-bagian panel surya yang memiliki fungsi berbeda-beda, seperti bingkai aluminium berfungsi untuk melindungi tepi bagian laminasi yang menampung sel-sel dan untuk pemasangan modul surya pada sistem PLTS. *Etilen Vinil Asetat* (EVA) merupakan lapisan polimer yang dirancang khusus untuk menahan sel surya pada posisinya selama pembuatan. *Backsheet* adalah bagian belakang modul surya yang berfungsi sebagai isolasi listrik dan penghalang kelembaban. *Junction box* modul surya merupakan hubungan output daya modul surya yang berada pada bagian belakang dan berfungsi sebagai interkoneksi dengan modul surya lainnya [20].

2.4.2 Inverter

Inverter adalah bagian penting dalam sistem PLTS, *inverter* berfungsi mengubah arus searah atau *direct current* (DC) yang dihasilkan oleh panel surya menjadi arus bolak-balik atau *Alternating Current* (AC). Salah satu contoh *inverter* dapat dilihat pada Gambar 2.5.



Gambar 2.5 *Inverter* [19]

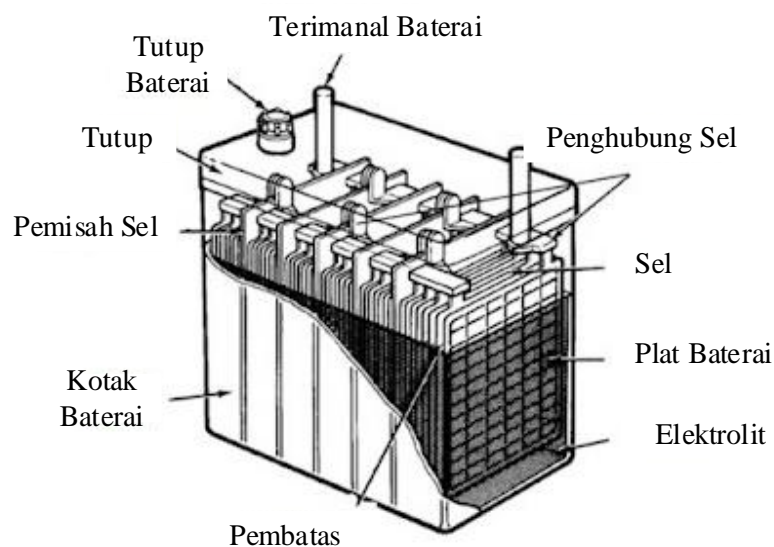
Gambar 2.5 adalah salah satu inverter *off grid* yang mana bekerja karena adanya tegangan DC dari panel surya. Tegangan DC cenderung tidak konstan karena radiasi matahari yang berubah-ubah, dengan adanya *inverter* menjadi tegangan AC yang konstan yang siap disambungkan pada alat elektronik. Parameter tegangan dan arus pada keluaran *inverter* harus sesuai dengan standar baku nasional atau internasional [19].

2.4.3 Baterai

Baterai berfungsi untuk menyimpan sementara listrik yang dihasilkan modul surya, agar dapat digunakan pada saat energi matahari tidak ada. Kapasitas baterai dinyatakan dalam Ah atau Ampere *hours*. Perhitungan untuk kebutuhan banyak baterai akan digunakan dalam sistem PLTS dapat dilihat di Persamaan (2.4) [18]:

$$\text{Kapasitas baterai} = \frac{\text{Total konsumsi harian}}{\text{Tegangan baterai}} \quad (2.4)$$

Persamaan (2.4) adalah persamaan untuk menghitung kebutuhan kapasitas baterai yang dihasilkan dari pembagian total konsumsi harian dengan tegangan baterai. Beberapa teknologi baterai yang umum dikenal adalah *lead acid*, *alkalin*, NiFe, Ni-Cad, dan Li-ion. Masing-masing jenis baterai memiliki kelemahan dan kelebihan baik dari segi teknis maupun ekonomi (harga). Baterai yang digunakan dari sistem PLTS dapat dilihat pada Gambar 2.6.

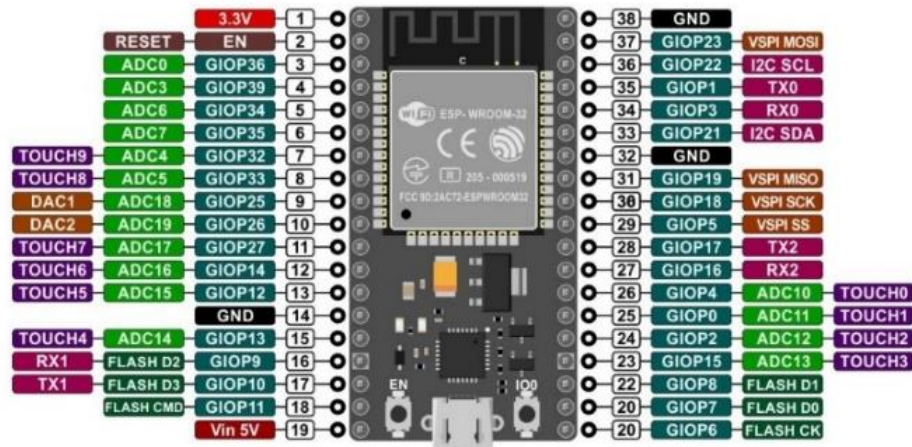


Gambar 2.6 Baterai [18]

Gambar 2.6 adalah baterai *lead acid* yang dinilai lebih unggul dari jenis lain jika mempertimbangkan segi teknis dan ekonomi. Kapasitas baterai yang diperlukan tergantung pada pola operasi PLTS dan juga harus mempertimbangkan seberapa banyak isi baterai akan dikeluarkan dalam sekali pengeluaran [18].

2.5 ESP32

ESP32 adalah mikrokontroler yang dikenalkan oleh *Espressif System* merupakan penerus dari mikrokontroler ESP8266. Berikut adalah susunan kaki-kaki *bord* ESP32 diperlihatkan di Gambar 2.7.

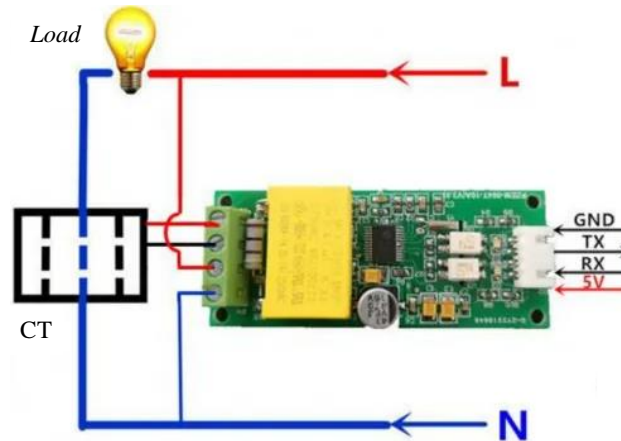


Gambar 2.7 *Pin Out* Mikrokontroler ESP32 [20]

Gambar 2.7 merupakan penjelasan *pin out* dari mikrokontroler ESP32. ESP32 memiliki modul WiFi dalam *chip* sehingga sangat mendukung untuk membuat sistem aplikasi IoT. ESP32 ini juga dapat di program melalui aplikasi Arduino IDE. Pemilihan ESP32 memudahkan di pemograman dan memiliki *pin input* dan *input output* yang memadai dan dapat mengakses jaringan internet untuk mengirim data [20].

2.6 Sensor PZEM-004t

Sensor PZEM-004t adalah alat sensor yang berfungsi untuk mengukur parameter dari tegangan, arus, daya aktif, dan konsumsi daya (Wh). Sensor PZEM-004t dapat digunakan untuk menampilkan konsumsi energi listrik secara waktu nyata. Sensor PZEM-004t dapat dilihat pada Gambar 2.8.

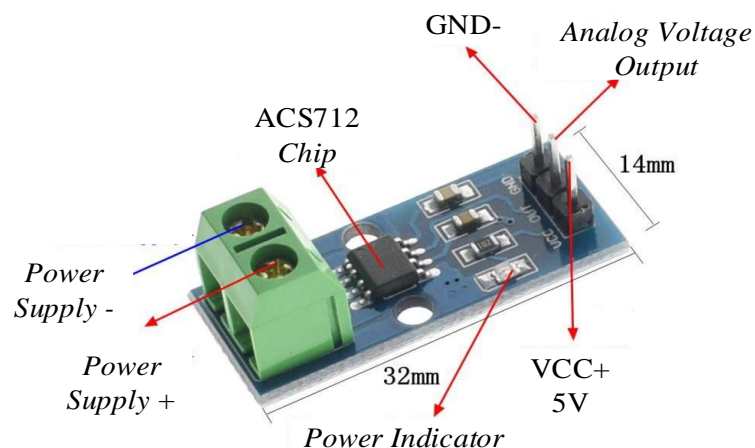


Gambar 2.8 Sensor PZEM-004t [21]

Gambar 2.8 adalah sensor PZEM-004t yang memiliki sistem kabel yang digunakan pada modul ini memiliki 2 bagian, yaitu dari kabel terminal masukan tegangan dan arus, serta kabel komunikasi serial. Berdasarkan pada kebutuhan, modul ini memiliki papan *pin* TTL untuk mendukung komunikasi data serial antar perangkat keras. Jalur komunikasi PZEM-004t menggunakan *port* USB atau RS-232 [21].

2.7 Sensor ACS712

ACS712 merupakan suatu IC terpaket yang mana berguna sebagai sensor arus menggantikan transformator arus yang relatif besar dalam hal ukuran. Sensor ACS712 dapat dilihat pada Gambar 2.9.



Gambar 2.9 Sensor ACS712 [22]

Gambar 2.9 merupakan sensor ACS712, pada prinsipnya ACS712 sama dengan sensor *efek hall* lainnya yaitu memanfaatkan medan magnetik disekitar arus kemudian dikonversi menjadi tegangan yang linier dengan perubahan arus. Nilai variabel dari sensor ini merupakan *input* untuk mikrokontroler yang kemudian diolah. Keluaran dari sensor ini berupa sinyal tegangan DC dan dibaca oleh *pin* dari mikrokontroler [22].

2.8 Sensor Tegangan

Sensor tegangan digunakan untuk mengukur tegangan yang didasari prinsip rangkaian pembagi tegangan. Tegangan sumber yang akan dibaca kemudian akan dibagi oleh rangkaian komponen resistor yang akan menurunkan tegangan sumber menjadi tegangan 0 s.d. 3,3 Volt DC, sehingga dapat dibaca oleh mikrokontroler. Sensor tegangan dapat dilihat pada Gambar 2.10.

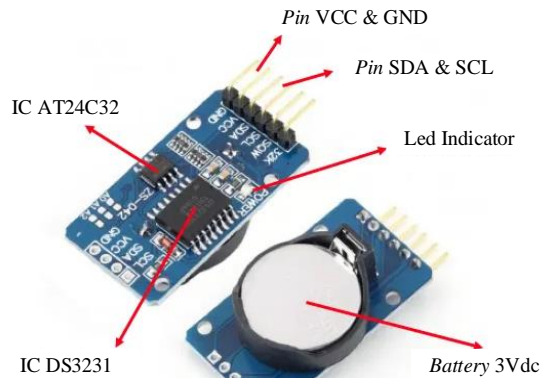


Gambar 2.10 Sensor Tegangan [22]

Gambar 2.10 adalah sensor tegangan yang merupakan sebuah rangkaian sederhana yang bisa mengubah arus listrik tegangan besar menjadi tegangan yang lebih kecil. Fungsi utama dari pembagi tegangan di dalam rangkaian elektronika adalah untuk membagi tegangan *input* menjadi satu atau beberapa tegangan *output* yang dibutuhkan oleh komponen lain dalam rangkaian [22].

2.9 Modul RTC (*Real Time Clock*)

Real time clock (RTC) adalah perangkat untuk menghasilkan waktu yang tepat karena dilengkapi pembangkit waktu dan dilengkapi baterai. Modul RTC dapat dilihat pada Gambar 2.11

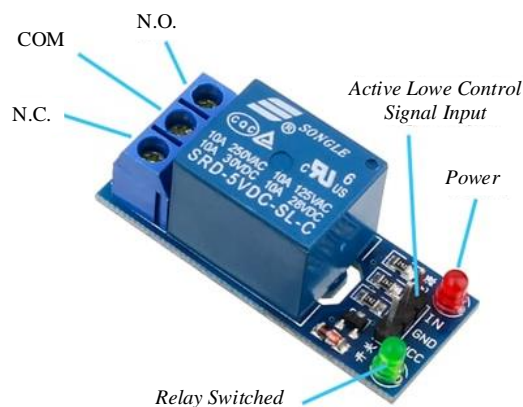


Gambar 2.11 Modul RTC (*Real Time Clock*) [23]

Gambar 2.11 adalah modul RTC DS3231. RTC DS3231 mampu mengakses data waktu secara *real time* mulai dari detik, menit, jam, hari, bulan, dan tahun. RTC sebelum digunakan dikalibrasi ulang jika waktu di RTC sudah tidak menunjukkan waktu yang tepat [23].

2.10 Relai

Relai adalah komponen elektronika yang berupa saklar atau *switch* elektrik yang dioperasikan arus secara programmable. Relai digunakan untuk mengatur kondisi-kondisi untuk perpindahan catu daya. Modul relai dapat dilihat pada Gambar 2.12.



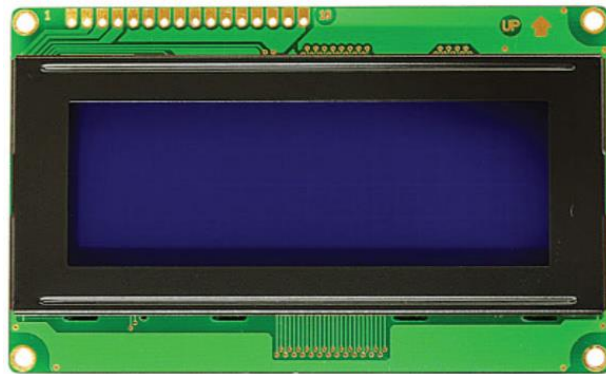
Gambar 2.12 Relai [23]

Gambar 2.12 merupakan relai atau biasa disebut sebagai komponen *electromechanical* atau elektromekanikal yang terdiri dari dua bagian utama yaitu *coil* atau electromagnet dan kontak saklar atau mekanikal. Komponen relai

menggunakan prinsip elektromagnetik sebagai penggerak kontak saklar, sehingga dengan menggunakan arus listrik yang kecil atau *low power* dapat menghantarkan arus listrik yang memiliki tegangan lebih tinggi [23].

2.11 *Liquid Crystal Display (LCD)*

Liquid Crystal Display (LCD) adalah perangkat yang berfungsi sebagai media penampil dengan memanfaatkan kristal cair sebagai objek penampil utama. LCD 20x4 dapat dilihat pada Gambar 2.13.

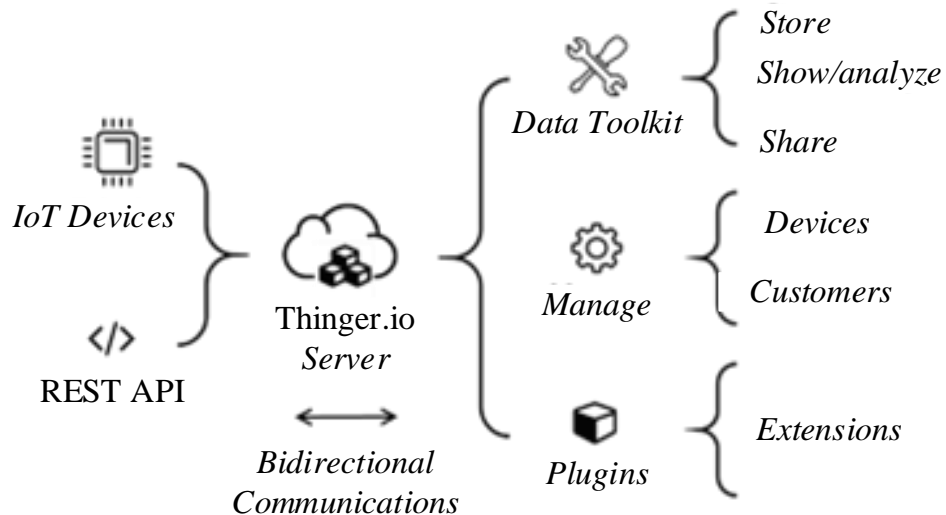


Gambar 2.13 LCD 20x4 I2C [21]

Gambar 2.13 adalah LCD yang digunakan adalah LCD berukuran 20x4 karakter dengan tambahan *chip* modul I2C untuk mempermudah programmer nantinya dalam mengakses LCD. Penggunaan modul I2C akan lebih menghemat penggunaan *pin* pada mikrokontroler, dengan hanya diperlukan 4 buah *pin*, yaitu *pin* SCL, *pin* SDA, *pin* VCC, dan *pin* GND [21].

2.12 *Platform Thinger.io*

Platform Thinger.io merupakan *platform Internet of Things (IoT)* berbasis *cloud open-source*. Model komunikasi Thinger.io dapat dilihat pada Gambar 2.14.



Gambar 2.14 Model Komunikasi Thingier.io [14]

Gambar 2.14 merupakan model komunikasi dari Thingier.io ini berfungsi untuk media pembacaan dan penerimaan data sensor yang digunakan untuk mempermudah suatu proses *monitoring*, sehingga dapat menghubungkan perangkat keras yang akan mengirimkan data sensor menuju *cloud* Thingier.io. Thingier.io dapat dihubungkan ke berbagai jenis *bord* arduino yang menggunakan *ethernet*, Wi-Fi, GSM atau *bord* lain yang didukung seperti TI CC3200, NodeMCU, dan ESP32. *Client library* memungkinkan menghubungkan perangkat IoT pengguna ke *platform cloud* Thingier.io. Pemakaian *library* Thingier.io sudah dirancang untuk Arduino IDE, yang mana dapat mudah memprogram perangkat untuk dihubungkan ke Thingier.io dalam waktu singkat [14].

2.13 Kajian Pustaka

Kajian pustaka digunakan sebagai perbandingan dari penelitian sejenis serta sumber informasi penelitian seperti penelitian mengenai perancangan pengatur suplai daya listrik pada sistem gabungan pembangkit listrik tenaga surya dengan jala-jala listrik PLN. Rancangan pengatur suplai daya memanfaatkan PLTS *off grid* dan sumber listrik PLN dengan parameter pergantiannya berdasarkan radiasi matahari 600 s.d. 700 Watt/m². Rancangan pengatur suplai daya listrik ketika radiasi matahari mencapai 1000 Watt/m² maka seluruh beban menggunakan sumber PLTS, sedangkan ketika radiasi matahari di bawah 600 Watt/m² maka beban

menggunakan sumber PLN [24]. Hubungan penelitian yang diperoleh dari penelitian ini adalah pendistribusian sumber PLN ataupun PLTS dapat dibagikan ke lebih dari satu jala-jala distribusi beban listrik.

Penelitian selanjutnya membahas tentang *smart grid hybrid* sistem (*fotovoltaik-PT. PLN*) berbasis *Internet of Things* (IoT). Penelitian *smart grid hybrid* sistem membuat rancangan *prototype hybrid* memanfaatkan PLTS *on grid* dengan kapasitas 20 Wp dan sumber listrik PLN secara bersamaan. Penelitian *smart grid hybrid* sistem menghasilkan penghematan energi sebesar 18,9% [25]. Hubungan penelitian yang diperoleh dari penelitian ini adalah dapat memantau penggunaan energi listrik dengan bantuan IoT.

Penelitian selanjutnya membahas tentang rancang bangun dan implementasi *Automatic Transfer Switch* (ATS) menggunakan Arduino Uno dan relai. Penelitian ini membuat perangkat ATS untuk pensaklaran dari sumber utama ke sumber cadangan catu daya. Penelitian rancang bangun ATS menghasilkan bahwa, alat dapat berjalan dengan baik untuk memindahkan sumber listrik utama ke sumber listrik cadangan [7]. Hubungan penelitian yang diperoleh dari penelitian ini adalah implementasi dari kendali distribusi daya listrik menggunakan relai.

Penelitian selanjutnya membahas tentang rancang bangun ATS sebagai pengalih catu daya otomatis berbasis *Programmable Logic Control* (PLC). Penelitian rancang bangun ATS berbasis PLC dapat memindahkan catu daya utama menuju catu daya cadangan secara otomatis dan cepat, saat catu daya utama mengalami gangguan ataupun terjadi pemutusan suplai ke beban. ATS ini dikendalikan dengan PLC sebagai kendali otomatis sehingga memudahkan dalam pembuatan dan meminimalisir biaya dalam penggunaan komponen [8]. Hubungan penelitian yang diperoleh dari penelitian ini adalah perpindahan sumber listrik PLN ke PLTS berdasarkan tegangan dan frekuensi yang sesuai standar.

Penelitian selanjutnya membahas tentang rancang bangun ATS dengan parameter arus, frekuensi, dan suhu. Penelitian rancang bangun ATS dapat bekerja untuk memindahkan suplai listrik dari satu sumber ke sumber lain jika sudah melampaui batas parameter yang sudah ditentukan berupa arus, frekuensi, dan suhu [26]. Hubungan penelitian yang diperoleh dari penelitian ini adalah sistem ATS dapat berjalan menggunakan mikrokontroler sebagai pusat kendali.

Berdasarkan referensi yang digunakan maka dari itu, penelitian merancang alat *smart grid* dengan distribusi beban listrik antara sumber PLN dengan PLTS begitu juga dengan sebaliknya. Perancangan alat ini juga dapat memantau konsumsi daya listrik yang telah digunakan oleh konsumen yang bersumber dari PLTS maupun PLN yang dapat memanfaatkan *Internet of Thing* (IoT) sehingga pemantauan dapat dilakukan secara jarak jauh.

BAB III

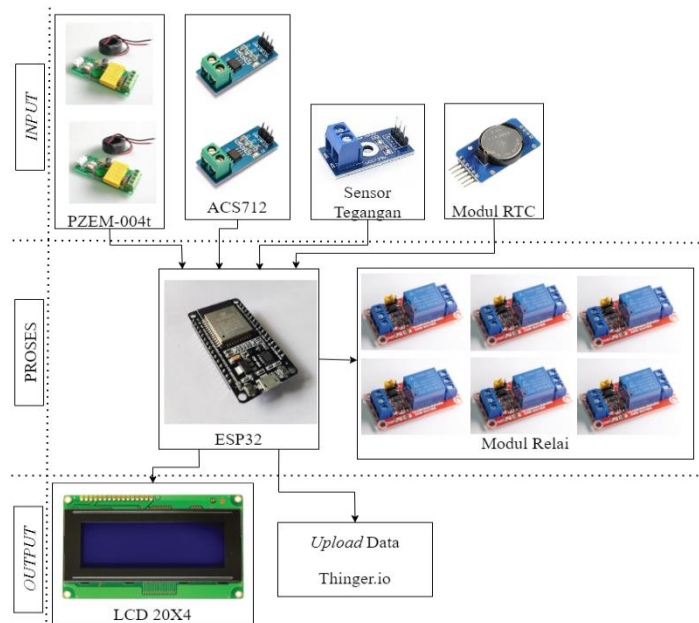
METODOLOGI PENELITIAN

3.1 Perancangan Penelitian

Perancangan penelitian sistem *smart grid* dengan distribusi beban listrik berbasis IoT membahas tentang keseluruhan tahapan dari pembuatan alat agar berjalan secara teratur dan sistematis baik yang mencakup perangkat keras maupun perangkat lunak.

3.1.1 Diagram Blok

Diagram blok perancangan penelitian dibuat agar mempermudah dalam merealisasikan terbentuknya sistem *smart grid* dengan sistem otomatis distribusi berbasis IoT. Gambaran mengenai blok diagram perancangan alat dapat dilihat pada Gambar 3.1.



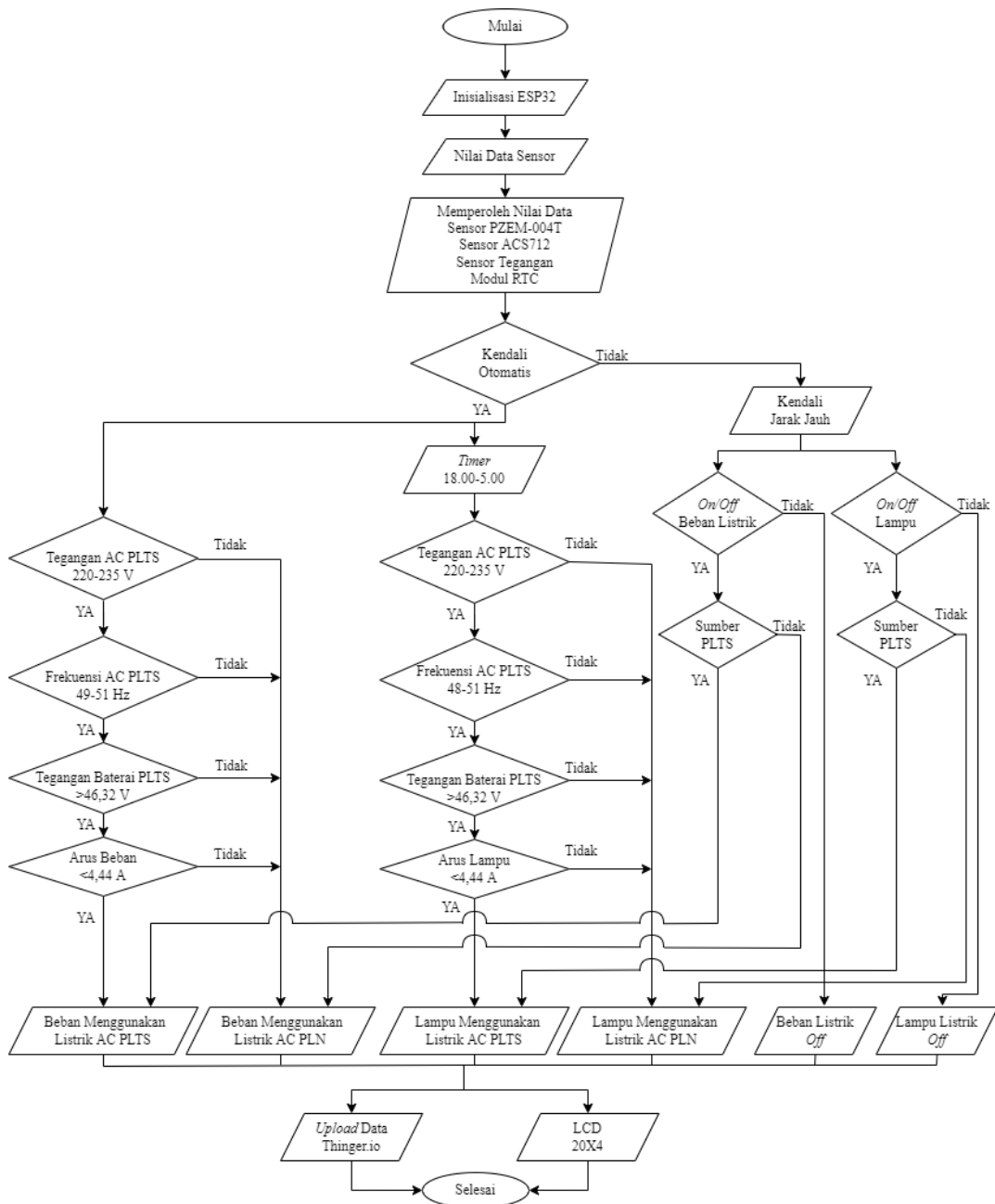
Gambar 3.1 Diagram Blok Alat

Gambar 3.1 adalah rangkaian rancang bangun sistem *smart grid* dengan sistem otomatis distribusi berbasis IoT yang terdiri dari beberapa blok rangkaian *input*, *proses*, dan *output*. Diagram perancangan merupakan perancangan sistem yang dibuat dari beberapa komponen-komponen yang tersusun menjadi satu

kesatuan sistem. Diagram tersebut yang menggambarkan proses yang dilakukan dalam pembuatan alat. Komponen tersebut meliputi sensor tegangan, sensor PZEM-004t, sensor ACS712, modul RTC, mikrokontroler ESP32, relai, dan LCD 20x4 I2C.

3.1.2 Flowchart Sistem

Flowchart sistem *smart grid* merupakan tahapan atau proses kerja secara menyeluruh. Proses kerja dari sistem *smart grid* dapat dilihat pada Gambar 3.2.



Gambar 3.2 Flowchart Sistem

Gambar 3.2 adalah *flowchart* sistem dari sistem *smart grid*, tahapan pertama adalah menginisiasikan mikrokontroler ESP32, langkah selanjutnya merupakan pengambilan data sensor diperoleh dari pembacaan sensor, seperti sensor tegangan untuk pembacaan tegangan baterai, sensor PZEM-004t untuk pembacaan listrik bolak-balik dari PLTS dan PLN, sensor ACS712 untuk pembacaan nilai arus dari PV dan beban yang terpakai, dan modul RTC menampilkan waktu, kemudian seluruh data tersebut dikirimkan ke mikrokontroler untuk diproses datanya.

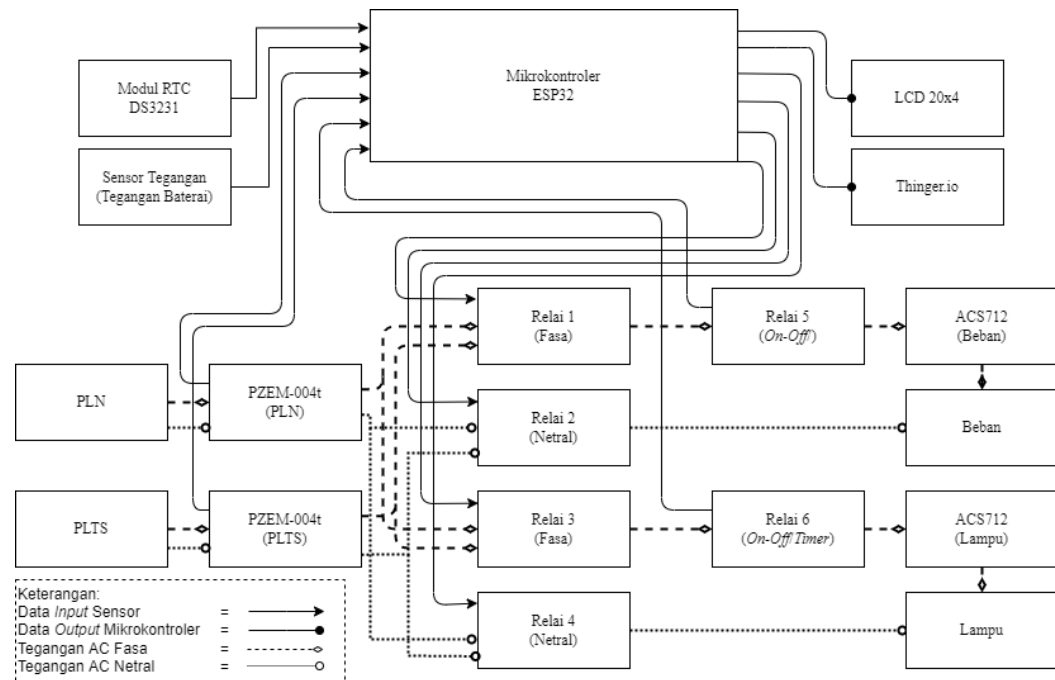
Sistem *smart grid* ini memiliki dua kendali, yaitu kendali otomatis dan kendali jarak jauh, jika sistem berjalan dengan kendali jarak jauh, *switch* pemakain sumber dan *switch on-off* dapat dikendalikan langsung tanpa harus memenuhi parameter yang ada. Pengendalian jarak jauh ini dapat dikendalikan melalui Thinger.io, jika sistem berjalan dengan kendali otomatis, *switch* pemakain sumber untuk menghidupkan beban listrik dan lampu harus memenuhi parameter yang sudah ditentukan. Parameter tersebut salah satunya adalah listrik bolak-balik atau *Alternating Current (AC)* yang harus memenuhi standar Indonesia yaitu tegangan sebesar 220 s.d. 235 V dan frekuensi 49 s.d. 51 Hz. Parameter selanjutnya adalah batas minimal tegangan baterai berfungsi untuk mengetahui kapasitasnya, batas tegangan baterai jika kapasitasnya 20% adalah 46,32 V dan kapasitasnya 100% adalah 54,48 V. Parameter selanjutnya adalah batas maksimal dari konsumsi daya, yang dimana arus pada beban harus kurang dari batas maksimal *ouput inverter* 4,44 A, jika seluruh parameter dari sistem *smart grid* terpenuhi, relai akan memindahkan sumber PLTS untuk menyalakan beban listrik, jika parameter tersebut tidak terpenuhi relai akan memindahkan sumber PLN untuk menyalakan beban listrik. Hasil dari pembacaan nilai dari sensor dan kendali yang digunakan dapat dilihat dari LCD dan *platform* Thinger.io.

3.2 Perancangan Sistem *Smart Grid*

Perancangan *smart grid* menggunakan beberapa komponen yang dirakit menjadi suatu sistem dengan berbagai instrument meliputi perancangan mikrokontroler, rangkaian *input*, dan *output*. Perangkat keras ini dirancang untuk dapat memantau dan mendistribusikan sistem tenaga listrik.

3.2.1 Rangkaian Mikrokontroler Sistem *Smart Grid*

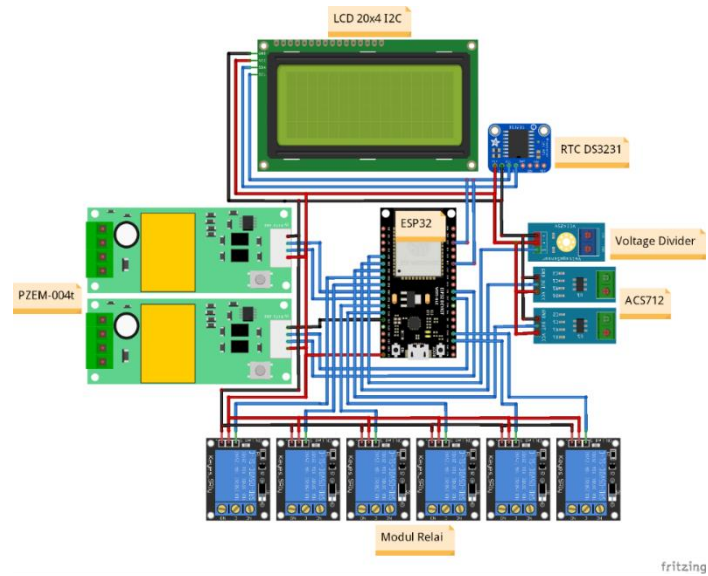
Perancangan sistem *smart grid* meliputi beberapa komponen sensor digital, *smart meter*, dan *online monitoring* yang akan digunakan sebagai komponen utama yang berfungsi untuk mendistribusikan sistem tenaga listrik terhadap distribusi beban listrik berbasis IoT. Perancangan sistem *smart grid* dapat dilihat pada Gambar 3.3.



Gambar 3.3 Diagram Blok Sistem *Smart Grid*

Gambar 3.3 merupakan alur sistem *smart grid*, yang dimana mikrokontroler ESP32 berfungsi untuk pemrosesan data dari sensor dan dapat difungsikan sebagai pengendali. Mikrokontroler akan mendapatkan data atau nilai yang telah ditentukan dari pembacaan *input sensor*, yang nantinya data tersebut akan diproses dan menghasilkan perintah untuk mengaktifkan relai. Komponen-komponen yang digunakan adalah sensor tegangan yang berfungsi sebagai pembacaan tegangan baterai, sensor PZEM-004t berfungsi untuk pembacaan tegangan listrik AC dari PLTS dan PLN, sensor ACS712 berfungsi untuk pembacaan arus berdasarkan konsumsi daya beban listrik, dan modul RTC berfungsi untuk memberikan informasi terkait waktu. Sistem *smart grid* relai berfungsi untuk pengatur suplai listrik yang akan digunakan. *Output* dari sistem *smart grid* yang berupa hasil

pembacaan nilai dari *sensor*, dapat di tampilkan di LCD ataupun Thingier. Rangkaian sistem *smart grid* dapat dilihat pada Gambar 3.4.

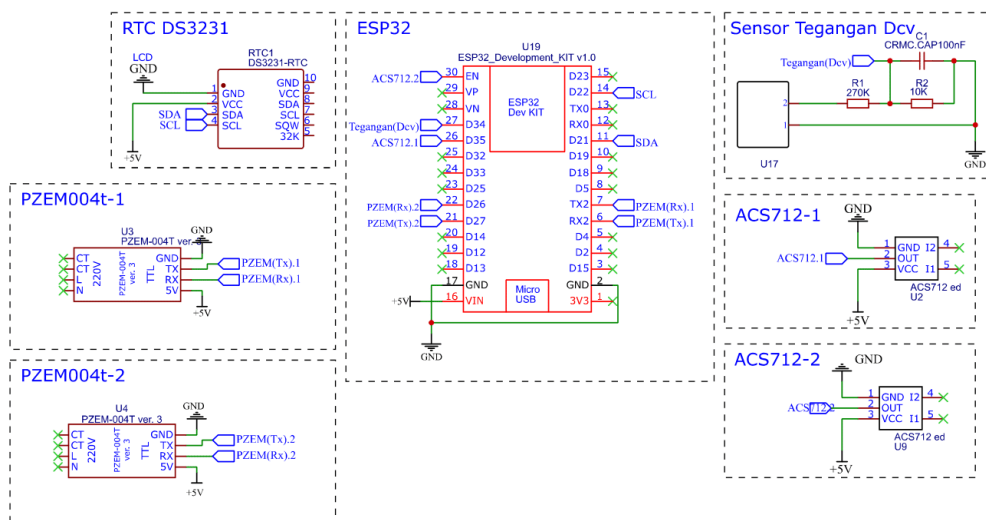


Gambar 3.4 Rangkaian Sistem *Smart Grid*

Gambar 3.4 adalah rangkaian mikrokontroler yang berguna untuk mengetahui gambaran keseluruhan hubungan komponen yang digunakan seperti sensor digital, *smart meter* dan *output* dengan *pin* pada mikrokontroler ESP32.

3.2.2 Rangkaian *Input* Sistem *Smart Grid*

Rangkaian *input* dari sensor sistem *smart grid* pada penelitian ini dapat dilihat pada Gambar 3.5.

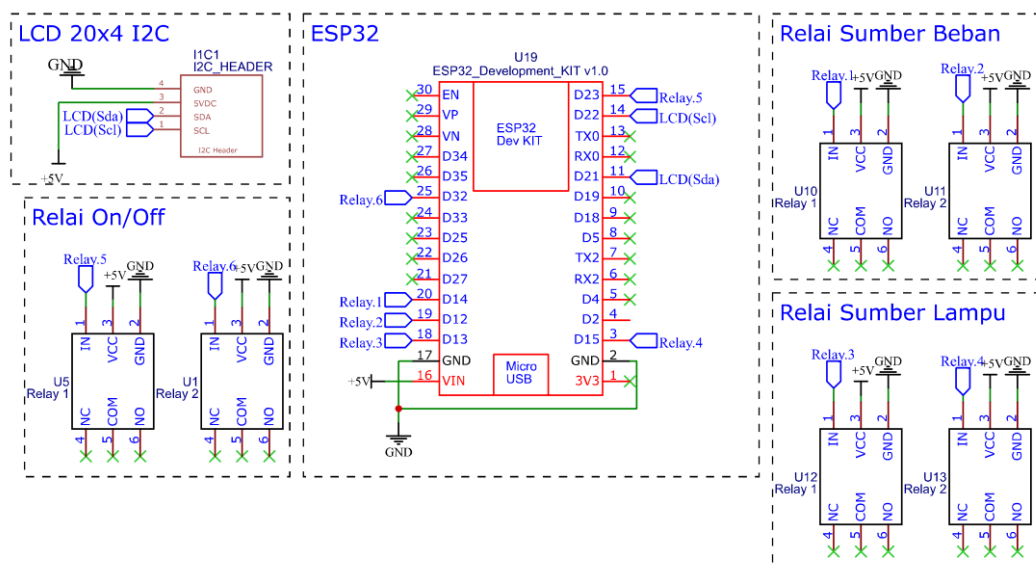


Gambar 3.5 Rangkaian *Input* Mikrokontroler

Rangkaian *input* dari sistem *smart grid* dapat dilihat Gambar 3.5, dalam perancangan ini untuk mengetahui hubungan atau koneksi dari beberapa sensor sebagai komponen untuk pembacaan data dengan mikrokontroler ESP32. Komponen sensor yang digunakan pada rangkaian *input* terdiri dari sensor dari PZEM-004t, ACS712, sensor tegangan, dan modul RTC. Komponen sensor ini akan terkoneksi terhadap *pin* dari ESP32, seperti sensor PZEM-004t yang berjumlah 2 masing-masing terhubung pada *pin* RX2, *pin* TX2, *pin* 26 dan *pin* 27. Sensor ACS712 berjumlah 2 masing-masing terhubung pada *pin* 35 dan *pin* 36. Sensor tegangan terhubung pada *pin* 34. Modul RTC terhubung pada *pin* SCL dan *pin* SDA. Seluruh komponen sensor memerlukan tegangan 5 V sebagai tegangan kerja, yang dimana seluruh *pin* VCC dan *pin* GND pada komponen sensor dihubungkan pada *pin* Vin dan *pin* GND.

3.2.3 Rangkaian *Output* Sistem *Smart Grid*

Rangkaian *Output* dari sistem *smart grid* pada penelitian ini dapat dilihat pada Gambar 3.6.



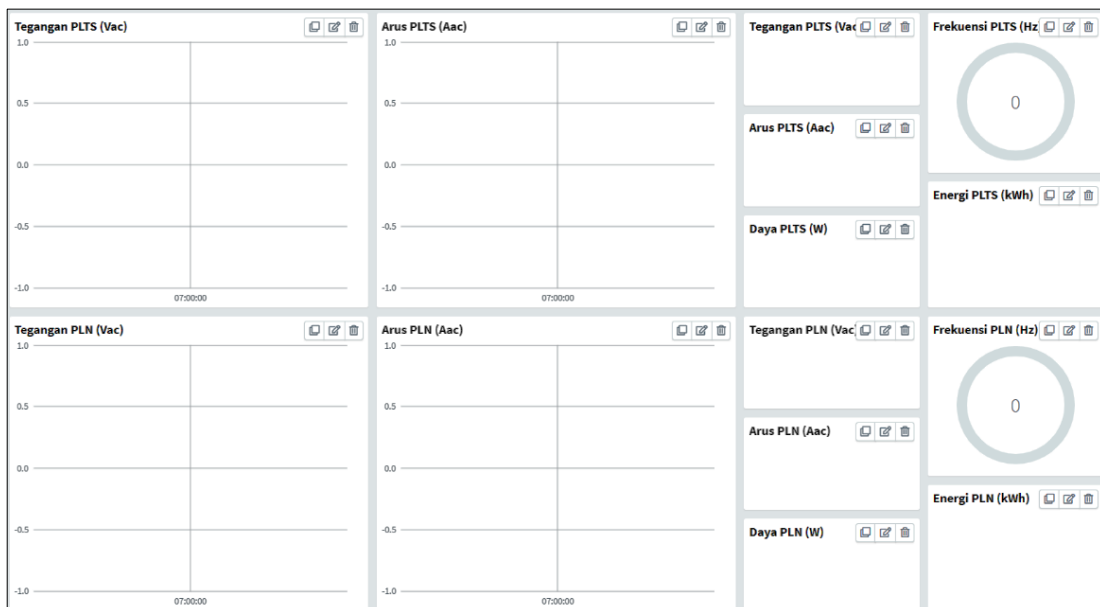
Gambar 3.6 Rangkaian *Output* Mikrokontroler

Rangkaian *output* dari sistem *smart grid* dapat dilihat Gambar 3.6, dalam perancangan ini untuk mengetahui hubungan atau koneksi dari beberapa komponen untuk dikendalikan dan untuk menampilkan data dengan mikrokontroler ESP32.

Komponen yang digunakan pada rangkaian *output* terdiri dari 6 relai dan LCD I2C. Komponen ini akan terkoneksi terhadap *pin* dari ESP32, seperti 6 relai masing-masing terhubung pada *pin* 12, *pin* 13, *pin* 14, *pin* 15, *pin* 21, dan *pin* 32. LCD I2C digunakan untuk menampilkan data dari hasil sensor secara langsung dan LCD I2C terhubung dengan mikrokontroler melalui koneksi I2C yang terkoneksi pada *pin* SCL dan *pin* SDA dari mikrokontroler.

3.3 Perancangan Sistem *Monitoring Smart Grid*

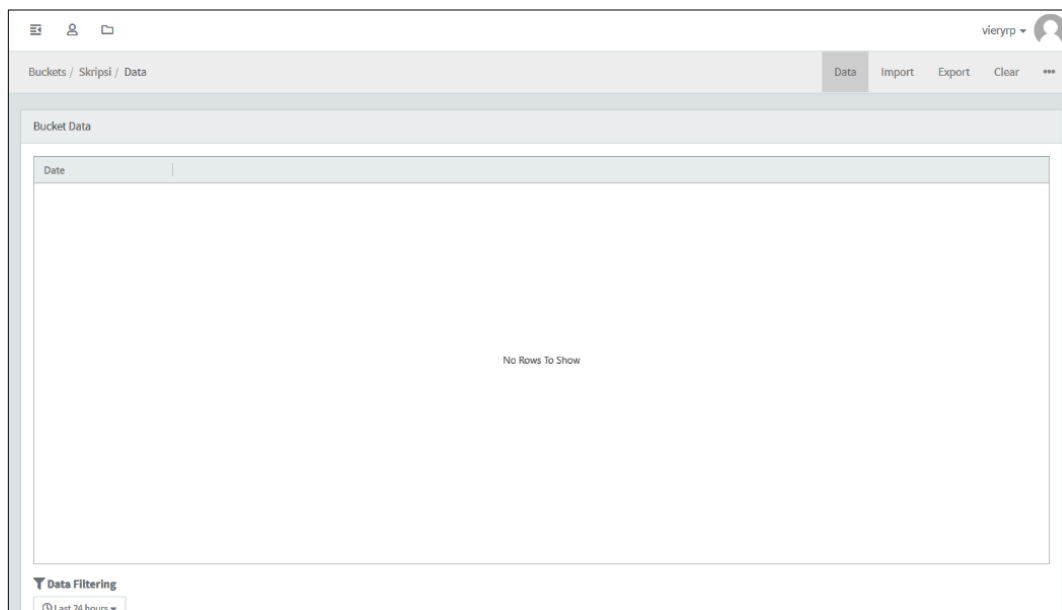
Sistem *monitoring smart grid* pada penelitian ini menggunakan perangkat lunak yaitu *platform* Thingier.io, dapat dilihat pada Gambar 3.7 merupakan hasil rancangan tampilan *monitoring* pada *dashboard* Thingier.io.



Gambar 3.7 *Dashboard Monitoring Web* Thingier.io

Gambar 3.7 adalah hasil rancangan sistem *monitoring smart grid* menggunakan perangkat lunak yaitu *platform* Thingier.io. *Platform* Thingier.io dapat digunakan untuk menampilkan hasil pembacaan sensor, mengendalikan sistem, dan pencatatan data dari hasil pembacaan sensor menggunakan konektivitas internet sehingga dapat dipantau secara *real time*. Rancangan *dashboard monitoring* dari *platform* Thingier.io memiliki fasilitas untuk menampilkan grafik

atau nilai pembacaan sensor dari keseluruhan sistem. *Dashboard* dari *data bucket* dari *platform* Thinger.io dapat dilihat pada Gambar 3.8.



Gambar 3.8 *Data Bucket* Thinger.io

Gambar 3.8 adalah *dashboard* dari *data bucket* dari *platform* Thinger.io yang berfungsi sebagai penyimpanan data yang berasal dari pembacaan sensor yang dikumpulkan dalam satuan waktu yang sudah disesuaikan.

3.4 Perancangan Pembangkit Listrik Tenaga Surya

Perancangan pembangkit listrik tenaga surya harus melewati tahapan yang sesuai dengan ketentuan, seperti menghitung kebutuhan beban listrik yang akan digunakan, menghitung kebutuhan komponen-komponen PLTS agar mencukupi daya listrik sesuai kebutuhan beban listrik, dan perencanaan penempatan sistem PLTS tersebut dipasang.

3.4.1 Kebutuhan Beban

Perencanaan beban dilakukan untuk mengetahui kapasitas alat yang akan digunakan yaitu untuk menentukan kapasitas dari komponen PLTS. Data-data yang digunakan adalah daya dari beban listrik yang terpasang pada rumah dengan pelanggan golongan RTM R-1/TR 1300 VA dengan penjabaran komponen-

komponen beban terpasang sebagai berikut. Perencanaan kebutuhan beban dapat dilihat pada Tabel 3.1.

Tabel 3.1 Perencanaan Kebutuhan Beban

No.	Beban	Jumlah	Daya (W)	Lama Penggunaan (Jam)
1	Lampu	6	90	12
2	Kulkas	1	120	24
3	Penanak Nasi	1	50	4
4	Dispenser	1	60	24

Tabel 3.1 adalah penggunaan beban listrik yang digunakan untuk penelitian seperti lampu, kulkas, penanak nasi, dan dispenser. Penggunaan beban listrik dihitung berdasarkan daya yang digunakan dan lama penggunaannya yang menghasilkan total kebutuhan daya listrik selama satu hari menggunakan Persamaan (2.2).

3.4.2 Komponen Sistem PLTS

Dalam penyusunan PLTS kebutuhan dari panel surya harus sesuai ataupun mendekati dengan besar konsumsi daya harian maka dari itu banyaknya kebutuhan dari PLTS dapat dilihat di bawah ini.

1. Kebutuhan panel yang dibutuhkan

Kebutuhan banyaknya panel surya dapat dihitung berdasarkan perbandingan konsumsi energi dengan lamanya penyinaran matahari yang dapat dihitung menggunakan Persamaan (2.3).

2. Kebutuhan *inverter*

Kebutuhan besaran *inverter* berdasarkan kebutuhan keluaran listrik DC menjadi listrik AC yang sesuai dengan standar PLN yaitu listrik AC dan mempertimbangkan beban listrik yang digunakan.

3. Kebutuhan baterai yang dibutuhkan

Kebutuhan banyaknya baterai dapat dihitung berdasarkan total dari beban listrik yang digunakan yang dapat dihitung menggunakan Persamaan (2.4).

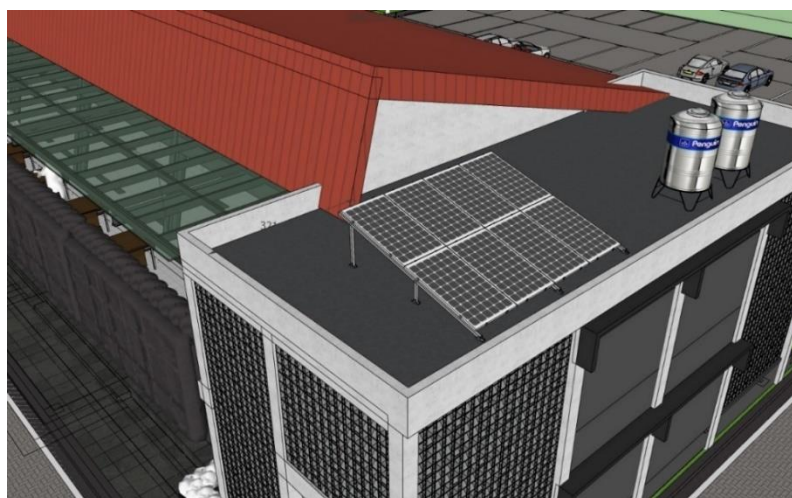
3.4.3 Perancangan Sistem PLTS

Perancangan sistem PLTS dibutuhkan komponen-komponen yang dibutuhkan berdasarkan dengan hasil hitungan, berikut adalah komponen yang dibutuhkan oleh sistem PLTS pada tabel 3.2.

Tabel 3.2 Komponen Sistem PLTS

Nama Komponen	Tipe	Spesifikasi
Panel Surya	<i>Monocrystalline</i>	310 Wp
<i>Inverter</i>	<i>Inverter Off Grid</i>	<i>Output Power</i> 1000 W
Baterai	Baterai VRLA	12 V, 150 Ah

Tabel 3.2 adalah komponen sistem PLTS yang dibutuhkan seperti panel surya dengan tipe *monocrystalline*, *inverter* dengan tipe *off grid*, dan baterai dengan tipe VRLA. Rancangan penempatan panel surya dapat dilihat pada Gambar 3.9.



Gambar 3.9 Rancangan Penempatan PLTS

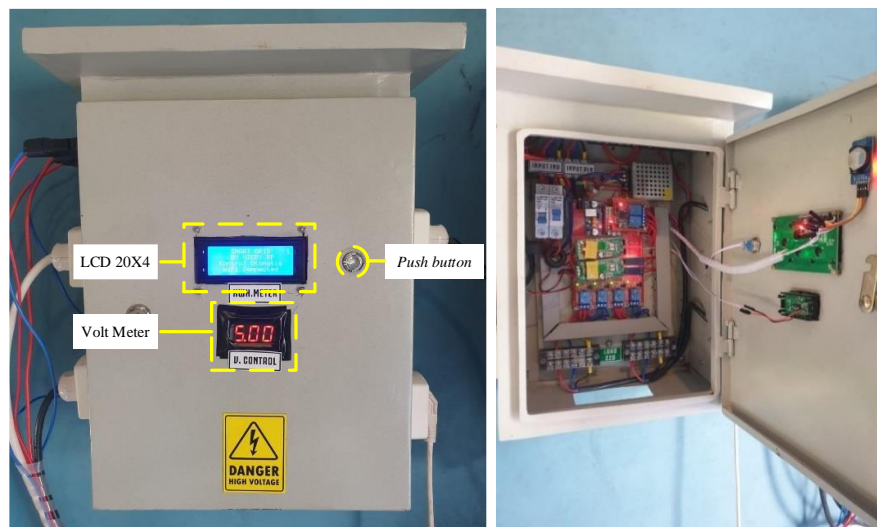
Gambar 3.9 adalah rancangan penempatan PLTS yang diletakan di atas Kantin Hijrah FT UNTIRTA. Sistem *array* dari panel surya harus ditempatkan pada tempat yang terbuka agar tidak terkena *shading* sehingga hasil daya listrik dapat maksimal.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Perancangan Alat

Penelitian ini membahas tentang sistem *smart grid* dengan distribusi beban listrik berbasis IoT. Hasil dari perancangan sistem *smart grid* pada sistem ini dapat dilihat pada Gambar 4.1.



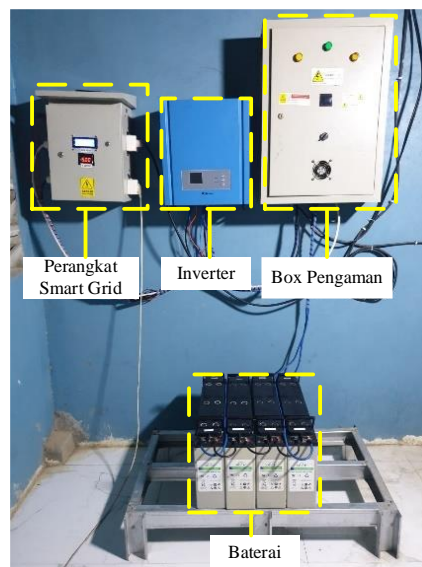
Gambar 4.1 *Box Panel Smart Grid* (a) Tampak Luar (b) Tampak Dalam

Gambar 4.1 merupakan hasil dari perancangan dari alat *smart grid*. Seluruh rangkaian komponen yang sudah disatukan berada di dalam panel *box* agar aman dari kontak langsung sekitar, karena sistem ini memiliki tegangan listrik. *Box panel* pada alat *smart grid* terdapat modul LCD 20x4 dan I2C dan *display* tegangan kendali yang diletakkan pada posisi tengah agar mudah untuk dilakukan *monitoring* secara langsung oleh pengguna, dan terdapat *push button* dimana digunakan untuk pergantian tampilan pada LCD 20x4 agar seluruh parameter dapat dilihat. Alat *smart grid* juga terhubung dengan sumber dari PLN, sumber dari PLTS, dan sumber baterai. Alat ini memiliki 2 *output* yang dimana berfungsi untuk memberikan suplai listrik untuk beban listrik yaitu lampu, kulkas, penanak nasi, dan dispenser seperti pada (lampiran A-1). Beban listrik yang digunakan memiliki konsumsi energi listrik sebesar 11.000 Wh yang diperoleh dari Persamaan (2.2) dengan rincian hasil seperti pada (lampiran B-8). Sistem *array* dari panel surya dapat dilihat pada Gambar 4.2.



Gambar 4.2 PLTS

Gambar 4.2 merupakan hasil dari perancangan PLTS yang terintegrasi dengan alat *smart grid*. PLTS tersebut menggunakan modul surya *monocrystalline* sebesar 310 Wp, yang terintegrasi dengan modul lain yang menghasilkan 2790 Wp diperoleh dari Persamaan (2.3) dengan rincian hasil seperti pada (lampiran B-9). Penempatan sistem *array* dari panel surya yang tidak tepat bisa mengakibatkan penurunan hasil daya listrik dari panel surya, maka dari itu peletakan sistem *array* diletakkan di atas kantin Hijrah FT UNTIRTA. *Storage system* dapat dilihat pada Gambar 4.3.

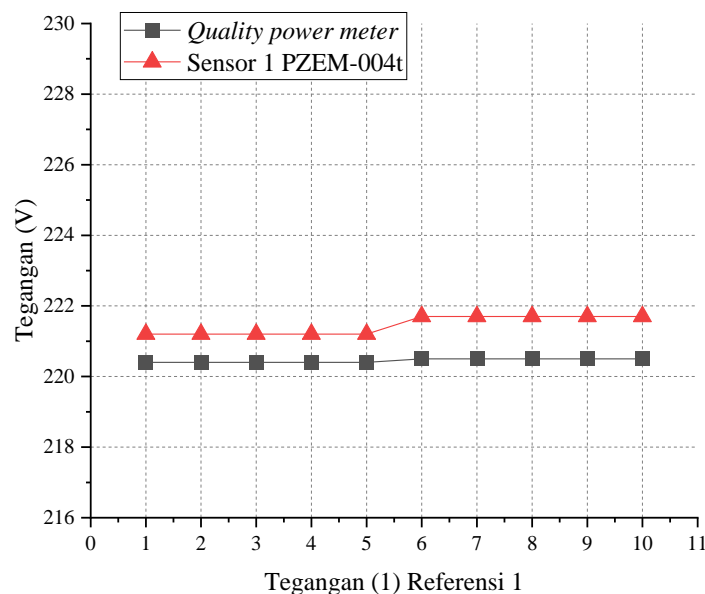


Gambar 4.3 *Storage System*

Gambar 4.3 merupakan hasil dari *storage system* yang terintegrasi dengan alat *smart grid*. *Storage system* terdapat *inverter* dan baterai yang saling terhubung dengan panel surya maupun alat dari *smart grid*, *inverter* yang digunakan memiliki batas pemakaian sebesar 1000 Watt dan baterai yang digunakan memiliki kapasitas sebesar 150 Ah dan tegangan 48 Volt diperoleh dari Persamaan (2.4) dengan rincian hasil seperti pada (lampiran B-9).

4.2 Pengujian Sensor PZEM-004t

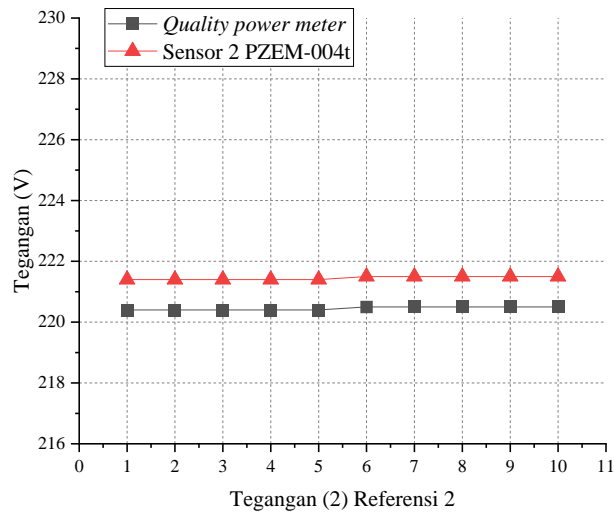
Pengujian sensor PZEM-004t dilakukan untuk mengetahui sensitivitas sensor dan untuk memastikan tingkat ketelitian pembacaan nilai pembacaan yang dihasilkan oleh sensor PZEM-004t dengan alat ukur. Sensor PZEM-004t yang digunakan berjumlah 2 buah dapat membaca nilai tegangan, arus, dan frekuensi dari listrik bolak-balik atau *Alternating Current (AC)*. Hasil pengujian pembacaan tegangan dari sensor PZEM-004t yang pertama dapat dilihat berdasarkan grafik pada Gambar 4.3.



Gambar 4.4 Pengujian Nilai Tegangan Sensor 1 PZEM-004t

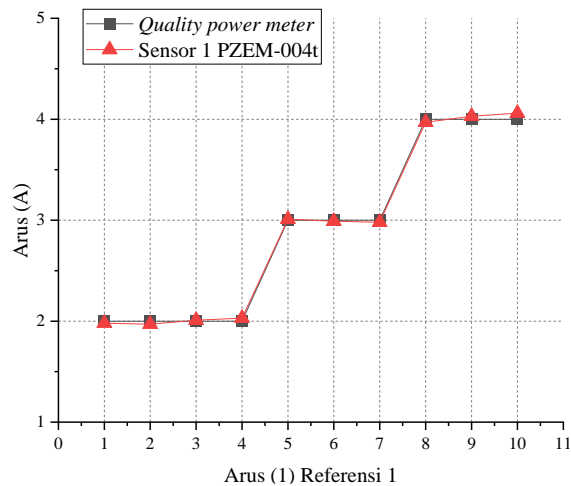
Gambar 4.4 merupakan pengujian pada sensor PZEM-004t yang pertama dengan pembandingan alat ukur *quality power meter*, didapat bahwa nilai rata-rata persentase *error* pada sensor PZEM-004t pertama yaitu 0,45%. Hasil pengujian

pembacaan tegangan dari sensor PZEM-004t yang kedua dapat dilihat berdasarkan grafik pada Gambar 4.4.



Gambar 4.5 Pengujian Nilai Tegangan Sensor 2 PZEM-004t

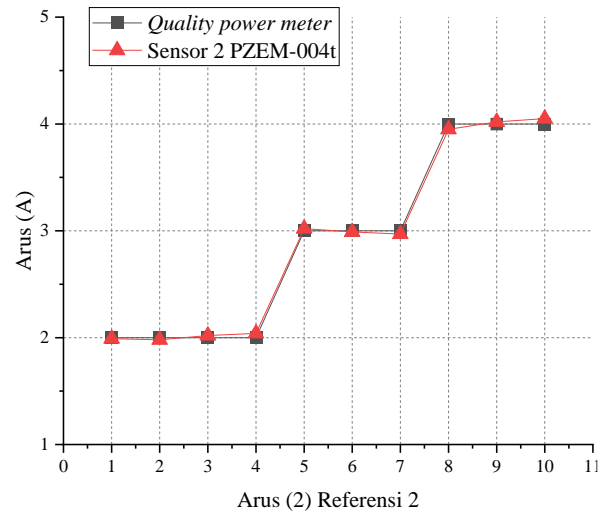
Gambar 4.5 merupakan pengujian pada sensor PZEM-004t yang kedua dengan pembandingan alat ukur *quality power meter*, didapat bahwa nilai rata-rata persentase *error* pada sensor PZEM-004t kedua yaitu 0,45%. Hasil pengujian pembacaan arus dari sensor PZEM-004t yang pertama dapat dilihat berdasarkan grafik pada Gambar 4.6.



Gambar 4.6 Pengujian Nilai Arus Sensor 1 PZEM-004t

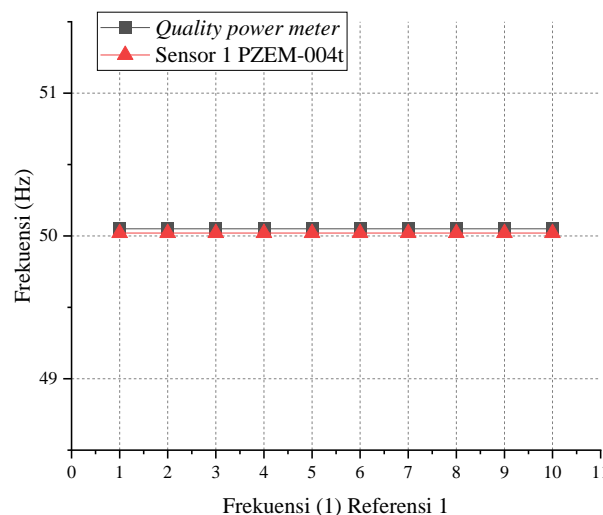
Gambar 4.6 adalah pengujian pada sensor PZEM-004t yang pertama dengan pembandingan alat ukur *quality power meter* didapat bahwa nilai rata-rata persentase

error pada sensor PZEM-004t pertama yaitu 0,88%. Hasil pengujian pembacaan arus dari sensor PZEM-004t yang kedua dapat dilihat berdasarkan grafik pada Gambar 4.7.



Gambar 4.7 Pengujian Nilai Arus Sensor 2 PZEM-004t

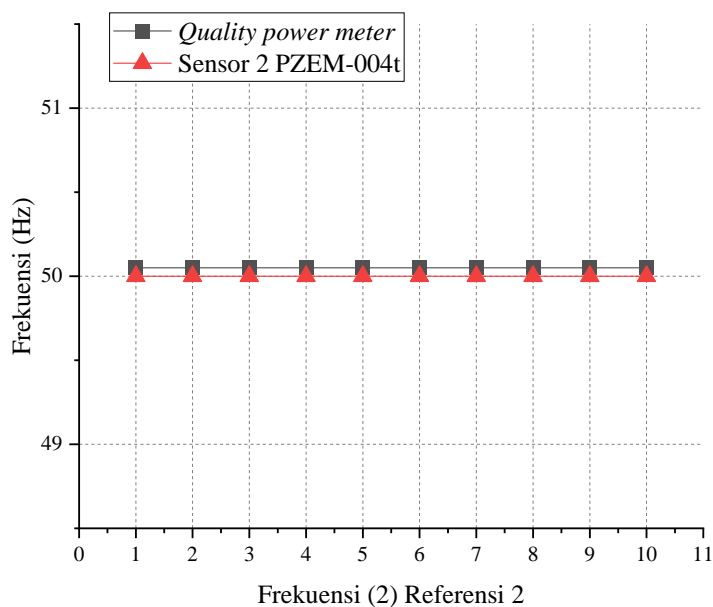
Gambar 4.7 adalah pengujian pada sensor PZEM-004t kedua dengan pembandingan alat ukur *quality power meter* didapat bahwa nilai rata-rata persentase *error* pada sensor PZEM-004t kedua yaitu 0,95%. Hasil pengujian pembacaan frekuensi dari sensor PZEM-004t yang pertama dapat dilihat berdasarkan grafik pada Gambar 4.8.



Gambar 4.8 Pengujian Nilai Frekuensi Sensor 1 PZEM-004t

Gambar 4.8 merupakan pengujian pada sensor PZEM-004t yang pertama dengan pembandingan alat ukur *quality power meter* didapat bahwa nilai rata-rata

persentase *error* pada sensor PZEM-004t pertama yaitu 0,06%. Hasil pengujian pembacaan frekuensi dari sensor PZEM-004t yang kedua dapat dilihat berdasarkan grafik pada Gambar 4.9.

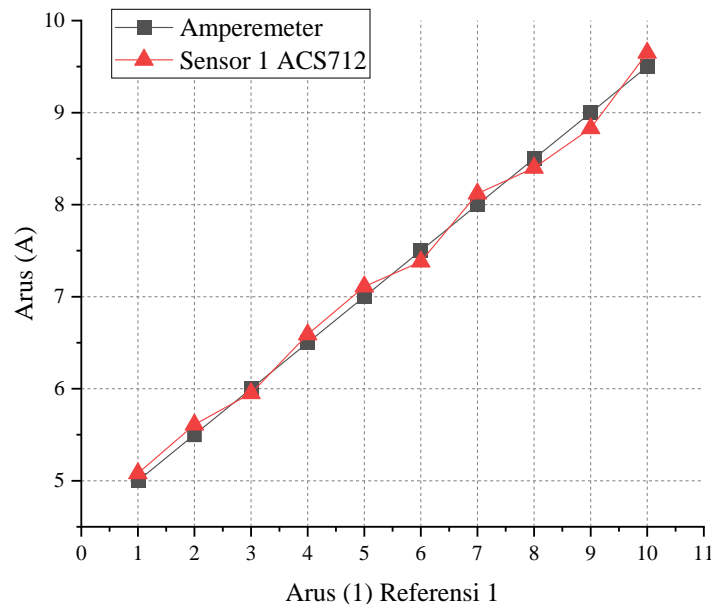


Gambar 4.9 Pengujian Nilai Frekuensi Sensor 2 PZEM-004t

Gambar 4.9 merupakan pengujian pada sensor PZEM-004t yang kedua dengan pembandingan alat ukur *quality power meter* didapat bahwa nilai rata-rata persentase *error* pada sensor PZEM-004t kedua yaitu 0,1%. Pengujian kelurusan dari hasil pengujian nilai tegangan, arus dan frekuensi pada sensor PZEM-004t memiliki nilai *error* yang dihasilkan dapat dikatakan baik dan dapat digunakan sebagai alat untuk pengambilan data.

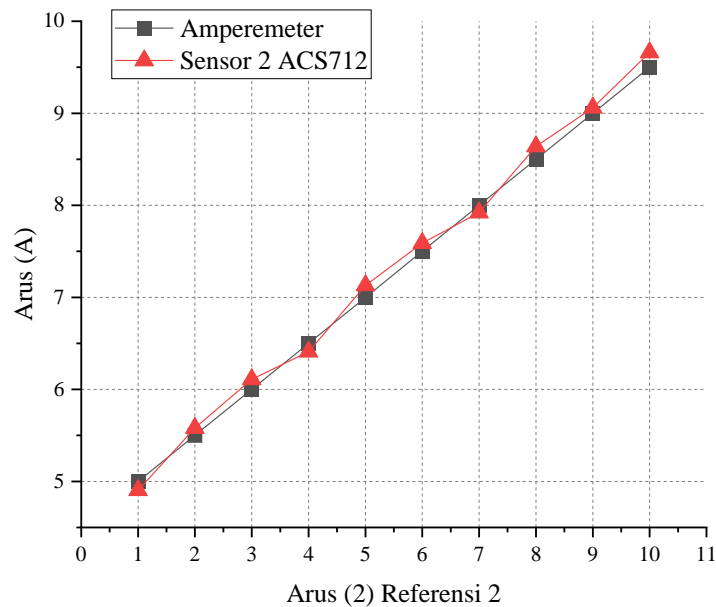
4.3 Pengujian Sensor ACS712

Pengujian sensor ACS712 dilakukan untuk mengetahui kesesuaian nilai yang diperoleh dari pembacaan sensor menggunakan alat ukur dan untuk menemukan kesalahan pada pembacaan yang sebenarnya. Sensor ACS712 yang digunakan berjumlah 2 buah dan pembacaan maksimum adalah 20 A. Hasil pengujian sensor ACS712 yang pertama dapat dilihat berdasarkan grafik pada Gambar 4.10.



Gambar 4.10 Pengujian Nilai Arus Sensor 1 ACS712

Gambar 4.10 merupakan pengujian pada sensor ACS712 yang pertama dengan pembandingan alat ukur *ampere meter* didapat bahwa nilai rata-rata persentase *error* pada sensor ACS712 pertama yaitu 1,51%. Hasil pengujian sensor ACS712 yang kedua dapat dilihat berdasarkan grafik pada Gambar 4.11.

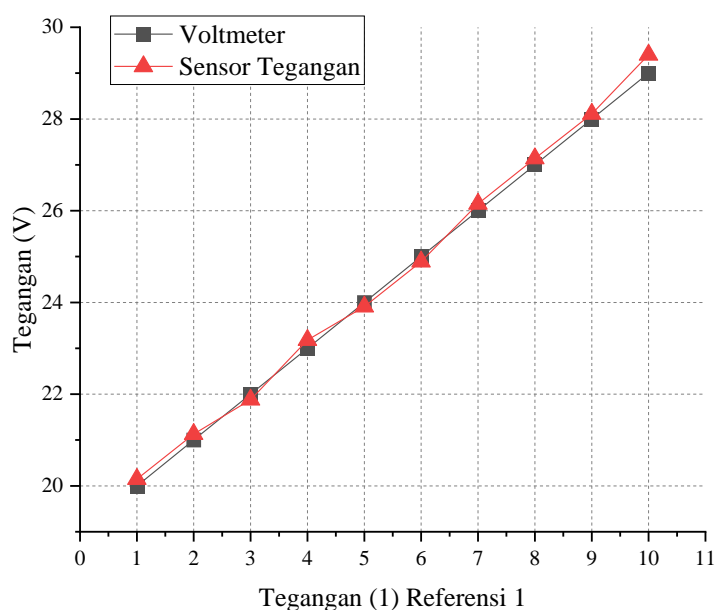


Gambar 4.11 Pengujian Nilai Arus Sensor 2 ACS712

Gambar 4.11 merupakan pengujian pada sensor ACS712 yang kedua dengan pembandingan alat ukur *ampere meter* didapat bahwa nilai rata-rata persentase *error* pada sensor ACS712 kedua yaitu 1,46%. Pengujian keseluruhan dari hasil pengujian nilai arus sensor ACS712 dapat dilihat nilai persentase *error* yang dihasilkan memiliki akurasi yang baik dan dapat digunakan sebagai alat untuk pengambilan nilai arus.

4.4 Pengujian Sensor Tegangan

Pengujian sensor tegangan pada penelitian ini merupakan konfigurasi pembagi tegangan. Pengujian kali ini dilakukan untuk mengetahui kesesuaian nilai yang diperoleh dari pembacaan sensor menggunakan alat ukur dan untuk menemukan kesalahan pada pembacaan yang sebenarnya. Sensor tegangan ini memiliki pembacaan maksimum adalah 158,4 Volt. Hasil pengujian sensor tegangan dapat dilihat berdasarkan grafik pada Gambar 4.12.



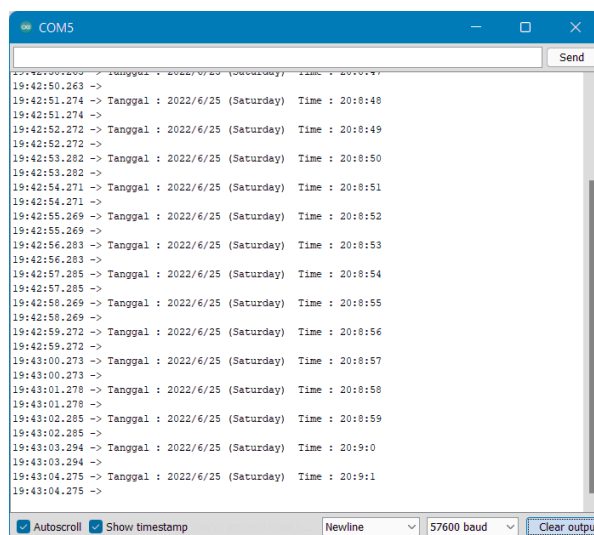
Gambar 4.12 Pengujian Nilai Sensor Tegangan

Hasil pengujian sensor tegangan yang dapat di lihat pada Gambar 4.12 menunjukkan bahwa pembacaan sensor tegangan dengan alat ukur tidak teralalu besar perbedaannya. Hasil pembacaan sensor tegangan dengan volt meter didapat bahwa nilai rata-rata persentase *error* sensor tegangan yaitu 0,64%. Nilai *error* tertinggi didapat saat pengukuran tegangan 29 Volt sebesar 1,38%. Pengujian

keseluruhan dari hasil pengujian nilai sensor tegangan dapat dilihat persentase *error* yang dihasilkan memiliki akurasi yang baik dan dapat digunakan sebagai alat pengambilan nilai tegangan.

4.5 Pengujian Modul RTC

Pengujian modul *Real Time Clock* (RTC) pada penelitian ini yaitu untuk mengetahui apakah waktu yang dihasilkan dapat sesuai dengan waktu yang sama dilokasi ini. Pengejuaian RTC memerlukan hubungan koneksi ke PC atau laptop agar dapat menyesuaikan waktu yang sama dengan PC atau laptop tersebut. Pengujian RTC dapat dilihat pada Gambar 4.13.

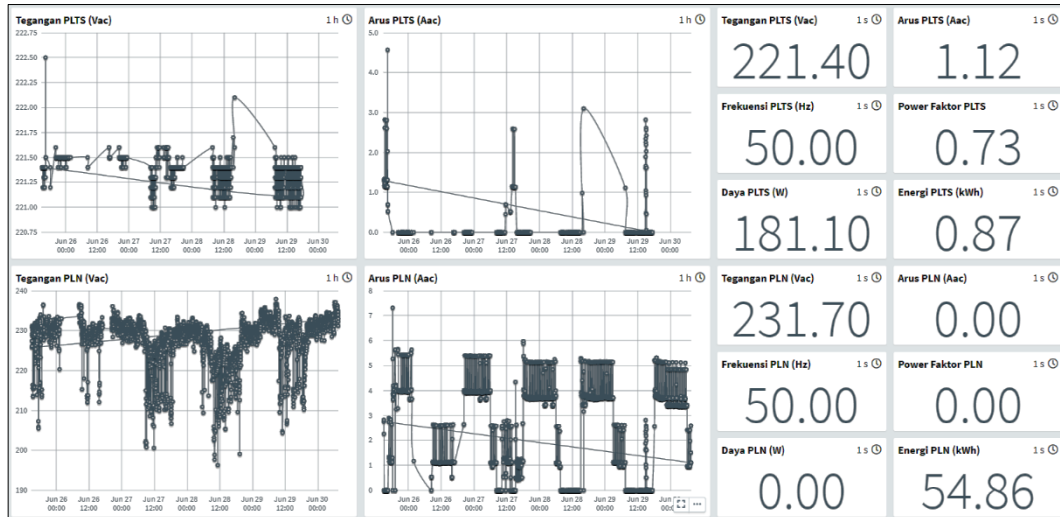


Gambar 4.13 Pengujian Modul RTC

Hasil dari pengujian RTC pada Gambar 4.13 menunjukkan bahwa modul RTC bekerja sesuai dengan waktu lokasi saat ini. Pengujian keseluruhan dari hasil pengujian modul RTC memiliki akurasi yang baik dan dapat digunakan sebagai alat untuk penambilan parameter waktu.

4.6 Pengujian Sistem *Monitoring Smart Grid*

Pengujian sistem *monitoring smart grid* pada Thinger.io bertujuan untuk mengetahui sistem *monitoring* dan penyimpanan data pada *platform* Thinger.io dapat berjalan dengan baik dan bekerja secara *real time*. Hasil pengujian *dashboard* Thinger.io dapat dilihat pada Gambar 4.14.



Gambar 4.14 Pengujian *Dashboard* Thingier.io

Gambar 4.14 merupakan hasil dari pengujian yang didapat bahwa sistem *monitoring* selalu berubah dalam satuan waktu, yang memandakan perubahan nilai yang didapat dilakukan secara *real time*. Hasil pengujian *data bucket* Thingier.io dapat dilihat pada Gambar 4.15.

Date	ArusBeban	ArusLampu	ArusPLN	ArusPLTS	DayaBeban	DayaLampu
6/23/2022, 10:41:47 PM	0.029082393273711205	0.06821141391992569	0	0	6.735482692718506	15.79776
6/23/2022, 10:40:48 PM	0.04987601563334465	0.058063603937625885	0	0	11.541309356689453	13.43591
6/23/2022, 10:39:48 PM	0.03275185450911522	0.07582227885723114	0	0	7.578779220581055	17.54527
6/23/2022, 10:38:44 PM	0.035198163241147995	0.052989695221185684	0	0	8.148374557495117	12.26711
6/23/2022, 10:37:46 PM	0.035198163241147995	0.07201685011386871	0	0	8.141335487365723	16.65749
6/23/2022, 10:36:47 PM	0.022966623306274414	0.09611791372299194	0	0	5.309883117917675781	22.22246
6/23/2022, 10:35:49 PM	0.026636086404323578	0.05172121897339821	0	0	6.152935981750488	11.94760
6/23/2022, 10:34:49 PM	0.03886762633919716	0.03903644531965256	0	0	9.001742362976074	9.040841
6/23/2022, 10:33:47 PM	0.052322328090667725	0.03142558038234711	0	0	12.112618446350098	7.275022
6/23/2022, 10:32:49 PM	0.04376024380326271	0.03523101285099983	0	0	10.095488548278809	8.127795
6/23/2022, 10:31:50 PM	0.04376024380326271	0.07201685011386871	0	0	10.091113090515137	16.60708
6/23/2022, 10:30:52 PM	0.020520316436886787	0.10753420740365982	0	0	4.76071357270508	24.94793
6/23/2022, 10:29:51 PM	0.033975012600421906	0.08850704878568649	0	0	7.878805160522461	20.52478
6/23/2022, 10:28:52 PM	0.027859242632985115	0.04411035403609276	0	0	6.466130256652832	10.23801
6/23/2022, 10:27:39 PM	0.020520316436886787	0.05425817519426346	null	null	0	0
6/23/2022, 10:25:32 PM	0.015627702698111534	0.04030492156744003	null	null	0	0

Gambar 4.15 Pengujian *Data Bucket* Thingier.io

Gambar 4.15 adalah pengujian *data bucket* pada Thingier.io yang dapat dilihat berjalan dengan baik dimana *data bucket* dapat mencatat data dari pembacaan sensor. Jika dilihat dari yang dihasilkan, sistem *monitoring smart grid* pada Thingier.io dapat berjalan baik dan dapat digunakan sebagai alat untuk *monitoring*.

4.7 Pengujian Sistem Kendali Jarak Jauh *Smart Grid*

Pengujian kendali jarak jauh dilakukan dengan pengendalian jarak jauh menggunakan Thingier.io dengan konektivitas *internet*. Thingier.io dilengkapi dengan fungsi *switch* yang dapat terintegrasi dengan alat dari *smart grid*. Fungsi *switch* dapat mengaktifkan atau merubah kondisi relai yang dimana relai akan hidup jika kondisinya *HIGH* dan mati jika kondisinya *LOW*. Tampilan *dashboard* kendali jarak jauh dari Thingier.io dapat dilihat pada Gambar 4.16.



Gambar 4.16 *Dashboard* Kendali Jarak Jauh Thingier.io

Gambar 4.16 merupakan *dashboard* kendali di dalam Thingier.io terdapat fungsi *switch* yang dapat di kendalikan jarak jauh, yang dimana berfungsi untuk mengendalikan relai-relai pada alat *smart grid*. Hasil pengujian pengendalian jarak jauh dapat dilihat di Tabel 4.1.

Tabel 4.1 Pengujian Kendali Jarak Jauh Thingier.io

Kondisi Awal Relai	Kondisi Awal Thingier.io	<i>Switch</i> Thingier.io	Kondisi	Relai
<i>Pin</i> Virtual	<i>Switch</i> Kendali <i>Off</i>	Kendali Otomatis	<i>HIGH</i>	<i>HIGH</i>
		Kendali Jarak Jauh	<i>LOW</i>	<i>LOW</i>
Relai 1 Mati	<i>Switch</i> Sumber Beban <i>Off</i>	<i>Switch</i> PLTS	<i>HIGH</i>	<i>HIGH</i>
		<i>Switch</i> PLN	<i>LOW</i>	<i>LOW</i>
Relai 2 Mati	<i>Switch</i> <i>On/Off</i> Beban	<i>Switch</i> ON	<i>HIGH</i>	<i>HIGH</i>
		<i>Switch</i> OFF	<i>LOW</i>	<i>LOW</i>
Relai 3 Mati	<i>Switch</i> Sumber Lampu <i>Off</i>	<i>Switch</i> PLTS	<i>HIGH</i>	<i>HIGH</i>
		<i>Switch</i> PLN	<i>LOW</i>	<i>LOW</i>
Relai 4 Mati	<i>Switch</i> <i>On/Off</i> Lampu	<i>Switch</i> ON	<i>HIGH</i>	<i>HIGH</i>
		<i>Switch</i> OFF	<i>LOW</i>	<i>LOW</i>

Hasil pengujian relai dapat dilihat pada Tabel 4.1, pada kondisi *switch* Thinger.io aktif, Thinger.io akan memberikan perintah ke mikrokontroler melalui jaringan internet untuk mengaktifkan relai agar aktif, dan sebaliknya jika *switch* Thinger.io tidak aktif, Thinger.io tidak akan memberikan perintah ke mikrokontroler. Hasil pengujian kendali jarak jauh menggunakan Thinger.io bergantung pada jaringan internet, jika konektivitas internet bagus maka mikrokontroler akan cepat merespon dan jika perintah sedang berjalan tetapi alat *smart grid* tidak dapat koneksi internet, maka alat akan terhenti pada perintah terakhir dan dapat melanjutkan kembali jika sudah terkoneksi dengan internet.

4.8 Pengujian Sistem Kendali Otomatis *Smart Grid*

Pengujian kendali otomatis dilakukan untuk menegengetahuin kinerja dari relai terhadap nilai *setpoint* dari *flowchat* sistem. Kendali otomatis ini terbagi dua, yaitu kendali otomatis terhadap beban listrik dan yang kedua kendali otomatis beban lampu. Hasil dari pengujian dapat dilihat di bawah ini.

4.8.1 Pengujian *Setpoint* Otomatis Beban Listrik

Pengujian kendali otomatis pada beban listrik dilakukan untuk mengetahui kinerja dari relai terhadap masing-masing parameter atau *setpoint* dari pembacaan sensor PZEM-004t, ACS712 dan sensor tegangan. Hasil pengujian dari *setpoint* kendali otomatis beban listrik dapat dilihat pada Tabel 4.2.

Tabel 4.2 Pengujian *Setpoint* Otomatis Beban Listrik

No	Tegangan AC PLTS	Frekuensi AC PLTS	Tegangan Baterai	Arus Beban	Sumber Listrik
1	205,5	50	54,48	4	PLN
2	220,0	46	54,48	4	PLN
3	220,0	50	46,15	4	PLN
4	220,0	50	54,48	5	PLN
5	220,0	50	54,48	4	PLTS

Tabel 4.2 merupakan hasil pengujian *setpoint* kendali otomatis pada beban listrik. *Setpoint* yang ditentukan adalah hasil dari pembacaan yang sudah sesuai dari parameter, ketika semua parameter sudah memenuhi maka relai akan memindahkan sumber PLN ke PLTS dan sebaliknya jika salah satu parameter tidak terpenuhi maka sumber yang digunakan adalah sumber dari PLN. Dapat disimpulkan bahwa kinerja dari relai terhadap masing-masing nilai *setpoint* berjalan dengan baik dan dapat digunakan.

4.8.2 Pengujian *Setpoint* Otomatis Lampu

Pengujian kendali otomatis pada lampu dilakukan untuk mengetahui kinerja dari relai terhadap masing-masing parameter atau *setpoint* dari pembacaan sensor PZEM-004t, ACS712, dan sensor tegangan. Hasil pengujian dari *setpoint* kendali otomatis beban lampu dapat dilihat pada Tabel 4.3.

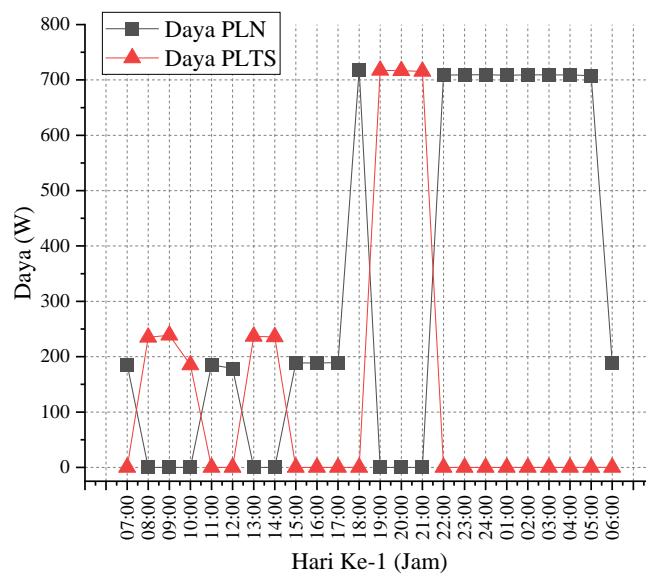
Tabel 4.3 Pengujian *Setpoint* Otomatis Lampu

No	Waktu	Tegangan AC PLTS	Frekuensi AC PLTS	Tegangan Baterai	Arus Beban	Sumber Listrik
1	13:00:00	220,0	50	54,48	0	-
2	18:00:00	205,5	50	54,48	4	PLN
3	20:00:00	220,0	46	54,48	4	PLN
4	23:00:00	220,0	50	46,15	4	PLN
5	0:00:00	220,0	50	54,48	5	PLN
6	3:00:00	220,0	50	54,48	4	PLTS

Tabel 4.3 merupakan hasil pengujian *setpoint* kendali otomatis pada lampu. *Setpoint* pada otomatis lampu berbeda dengan otomatis beban listrik karna adanya timer untuk menyalakan lampu. *Setpoint* yang ditentukan adalah hasil dari pembacaan yang sudah sesuai dari parameter, ketika semua kondisi sudah memenuhi maka relai akan memindahkan sumber PLN ke PLTS. Dapat disimpulkan bahwa kinerja dari relai terhadap masing-masing nilai *setpoint* berjalan dengan baik dan dapat digunakan.

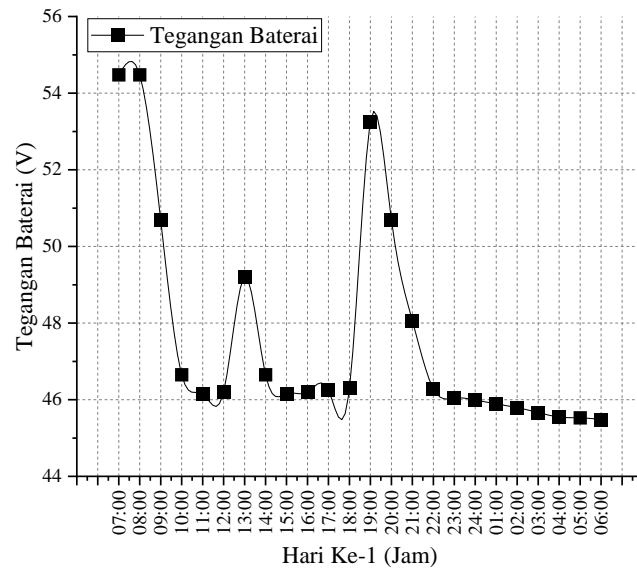
4.9 Hasil Pengujian Sistem *Smart Grid*

Pengujian sistem *smart grid* merupakan pengujian dari keseluruhan sistem untuk melihat pengaruh penggunaan alat dalam mengendalikan pemakaian sumber dari PLTS dan PLN secara otomatis. Pengujian ini menggunakan beban listrik seperti kulkas, dispenser, penanak nasi, dan lampu. Pengamatan dilakukan dalam kurun waktu 3 hari. Gambar 4.17 merupakan hasil dari penggunaan alat *smart grid* pada hari ke-1.



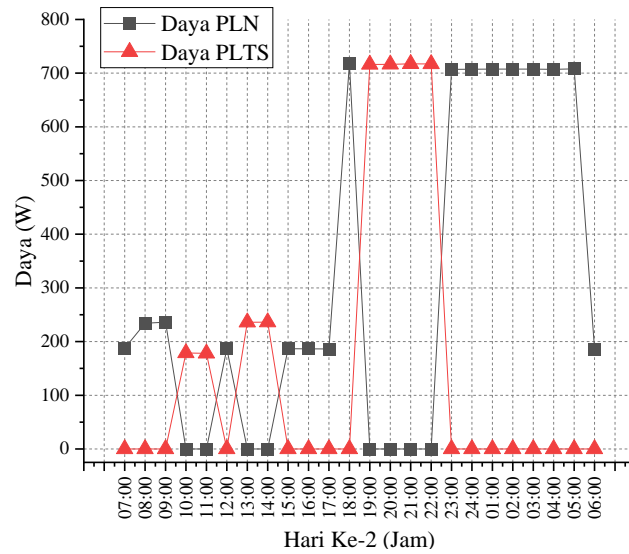
Gambar 4.17 Pengujian Alat *Smart Grid* Hari ke-1

Gambar 4.17 adalah hasil pengujian dari alat *smart grid* yang melakukan kendali otomatis terhadap sumber yang digunakan pada hari ke-1. Pengujian dilakukan dalam kurun waktu 1 hari dan data yang disimpan pada *platform* Thinger.io adalah setiap 1 jam sekali. Data yang ditampilkan merupakan grafik penggunaan daya yang terpakai dari PLTS maupun PLN. Grafik pada Gambar 4.17 dapat dilihat di dalam grafik pada jam 18:00 terdapat peningkatan daya yang signifikan dari 189,3 Watt menjadi 718,9 Watt yang menandakan kendali otomatis dari alat *smart grid* menyalakan beban lampu. Gambar 4.18 merupakan hasil dari nilai tegangan baterai pada hari ke-1.



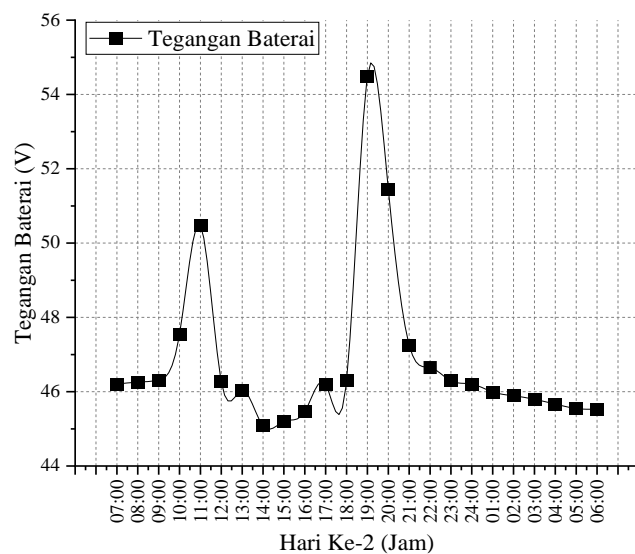
Gambar 4.18 Pengujian Nilai Tegangan Baterai Hari ke-1

Kendali otomatis dari alat *smart grid* dapat mengendalikan sumber listrik yang digunakan dari PLTS ataupun PLN, dapat dilihat pada Gambar 4.17 dan Gambar 4.18 dari data grafik tersebut dapat dilihat bahwa penggunaan sumber listrik dari PLTS dimulai dari jam 08.00 sampai dengan jam 10.00, karena kondisi tegangan baterai 54,48 Volt menunjukkan keadaan *full*. Penggunaan sumber PLTS untuk pemakaian sumber beban listrik dapat berbarengan dengan pengisian baterai. Dapat dilihat dalam grafik ketika jam 13.00 sampai dengan jam 14.00 dengan nilai tegangan baterai 49,21 Volt, sumber dari PLTS digunakan untuk pemakaian sumber beban listrik. Pemakaian sumber PLTS dapat digunakan ketika tidak adanya sumber energi matahari, dapat dilihat dalam grafik ketika jam 18.00 sampai dengan 21.00 dengan nilai tegangan baterai 53,25 Volt, sumber yang digunakan untuk menyalakan beban listrik menggunakan PLTS, karena kondisi baterai sudah mengalami pengisian di siang hari. Gambar 4.19 merupakan hasil dari penggunaan alat *smart grid* pada hari ke-2.



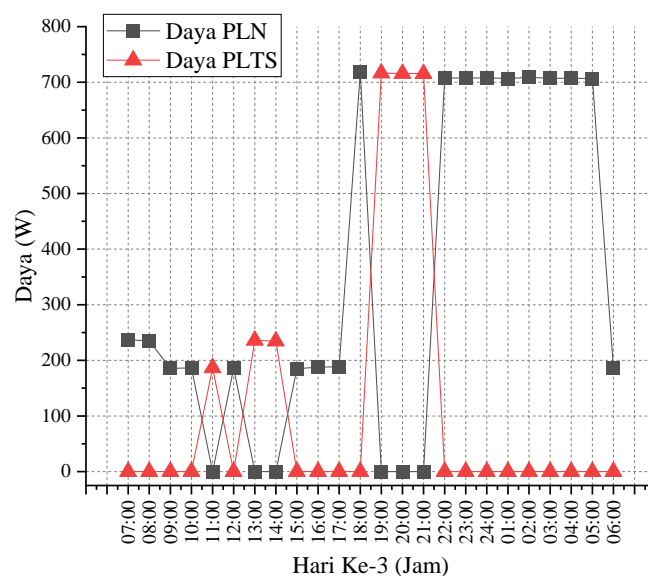
Gambar 4.19 Pengujian Alat *Smart Grid* Hari ke-2

Gambar 4.19 adalah hasil pengujian dari alat *smart grid* yang melakukan kendali otomatis terhadap sumber yang digunakan pada hari ke-2. Pengujian dilakukan dalam kurun waktu 1 hari dan data yang disimpan pada *platform* Thinger.io adalah setiap 1 jam sekali. Data yang ditampilkan merupakan grafik penggunaan daya yang terpakai dari PLTS maupun PLN. Grafik pada Gambar 4.19 dapat dilihat di dalam grafik pada jam 18:00 terdapat peningkatan daya yang signifikan dari 186,24 Watt menjadi 718,2 Watt yang menandakan kendali otomatis dari alat *smart grid* menyalakan beban lampu. Gambar 4.20 merupakan hasil dari nilai tegangan baterai pada hari ke-2.



Gambar 4.20 Pengujian Nilai Tegangan Baterai Hari ke-2

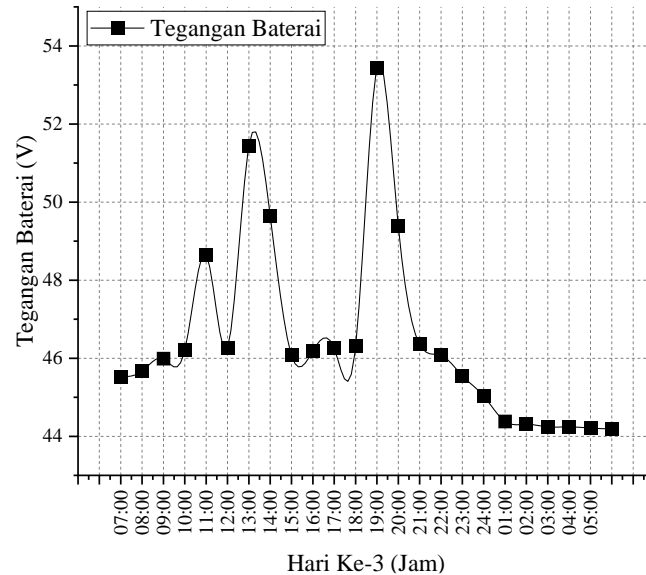
Kendali otomatis dari alat *smart grid* dapat mengendalikan sumber listrik yang digunakan dari PLTS ataupun PLN, dapat dilihat pada Gambar 4.19 dan Gambar 4.20 dari data grafik tersebut dapat dilihat bahwa penggunaan sumber PLTS dimulai dari jam 10.00 sampai dengan jam 11.00 dengan nilai tegangan baterai 47,54 Volt, dan dari jam 13.00 sampai dengan jam 14.00 dengan nilai tegangan baterai 51,31 Volt karena pemakaian sumber beban listrik dapat berbarengan dengan pengisian baterai. Pemakaian sumber PLTS dapat digunakan ketika tidak adanya sumber energi matahari, dapat dilihat dalam grafik ketika jam 18.00 sampai dengan 22.00 dengan nilai tegangan baterai 54,48 Volt sumber yang digunakan untuk menyalakan beban listrik menggunakan PLTS, karena kondisi baterai sudah mengalami pengisian di siang hari. Gambar 4.21 merupakan hasil dari penggunaan alat *smart grid* pada hari ke-3.



Gambar 4.21 Pengujian Alat *Smart Grid* Hari ke-3

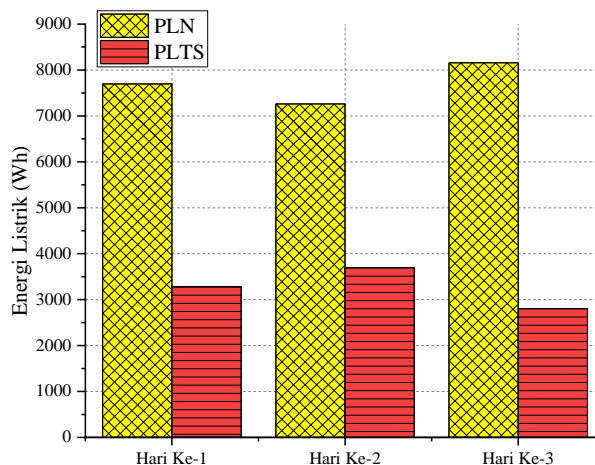
Gambar 4.21 adalah hasil pengujian dari alat *smart grid* yang melakukan kendali otomatis terhadap sumber yang digunakan pada hari ke-3. Pengujian dilakukan dalam kurun waktu 1 hari dan data yang disimpan pada *platform* Thingier.io adalah setiap 1 jam sekali. Data yang ditampilkan merupakan grafik penggunaan daya yang terpakai dari PLTS maupun PLN. Grafik pada Gambar 4.21 dapat dilihat di dalam grafik pada jam 18:00 terdapat peningkatan daya yang

signifikan dari 188,7 Watt menjadi 718,6 Watt yang menandakan kendali otomatis dari alat *smart grid* menyalakan beban lampu. Gambar 4.22 merupakan hasil dari nilai tegangan baterai pada hari ke-3.



Gambar 4.22 Pengujian Nilai Tegangan Baterai Hari ke-3

Kendali otomatis dari alat *smart grid* dapat mengendalikan sumber listrik yang digunakan dari PLTS ataupun PLN, dapat dilihat pada Gambar 4.21 dan Gambar 4.22 dari data grafik tersebut dapat dilihat bahwa penggunaan sumber listrik dari PLTS dimulai dari jam 11.00 sampai dengan jam 12.00 dengan nilai tegangan baterai 48,65 Volt, karena kondisi baterai mengalami kekurangan pengisian dari PLTS disebabkan cuaca hujan. Penggunaan sumber listrik dari PLTS digunakan kembali ketika kondisi cuaca cerah dari jam 13.00 sampai dengan jam 14.00 dengan nilai tegangan baterai 51,44 Volt. Pemakaian sumber PLTS dapat digunakan ketika tidak adanya sumber energi matahari, dapat dilihat dalam grafik ketika jam 18.00 sampai dengan 21.00 dengan nilai tegangan baterai 53,44 Volt sumber yang digunakan untuk menyalakan beban listrik menggunakan PLTS, karena kondisi baterai sudah mengalami pengisian di siang hari. Gambar 4.23 merupakan hasil konsumsi daya listrik dari alat *smart grid*.



Gambar 4.23 Konsumsi Daya Listrik

Berdasarkan pengujian yang telah dilakukan pada Gambar 4.23, terjadi perbedaan penggunaan konsumsi sumber daya listrik PLN dan PLTS, dan menghasilkan nilai persentase penggunaan konsumsi sumber daya listrik dengan rincian hasil yang terdapat seperti pada (lampiran B-10). Percobaan hari ke-1 didapat penggunaan konsumsi sumber daya listrik PLN sebesar 7.696,6 Wh dan penggunaan konsumsi sumber daya listrik PLTS sebesar 3.281 Wh, yang dimana alat *smart grid* pada hari ke-1 dapat menghemat konsumsi sumber daya listrik dari PLN sebesar 29,89%. Percobaan hari ke-2 didapat penggunaan konsumsi sumber daya listrik PLN sebesar 7.260,9 Wh dan penggunaan konsumsi sumber daya listrik PLTS sebesar 3.696,4 Wh, yang dimana alat *smart grid* pada hari ke-2 dapat menghemat konsumsi sumber daya listrik dari PLN sebesar 33,73%. Percobaan hari ke-3 didapat penggunaan konsumsi sumber daya listrik PLN sebesar 8.677,3 Wh dan penggunaan konsumsi sumber daya listrik PLTS sebesar 2.804,9 Wh, yang dimana alat *smart grid* pada hari ke-3 dapat menghemat konsumsi sumber daya listrik dari PLN sebesar 24,43%.

Percobaan ini dapat dilihat bahwa pada hari ke-3 terjadi penurunan hasil sumber daya dari PLTS yang disebabkan faktor cuaca yang menyebabkan PLTS tidak dapat optimal untuk menghasilkan energi listrik dan pada hari ke-2 didapat peningkatan hasil sumber daya listrik dari PLTS karena kondisi cuaca yang cerah. Percobaan keseluruhan dapat dilihat, sistem *smart grid* dapat mengoptimalkan penggunaan PLTS atau menghemat konsumsi dari daya PLN dengan rata-rata persentase pemakaian dibandingkan dengan total beban adalah 29,35%.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil dari pengolahan data dan analisis pada penelitian dengan judul “Perancangan *Smart Grid* Dengan Distribusi Beban Listrik Berbasis IoT” didapatkan beberapa kesimpulan sebagai berikut.

1. Sistem *smart grid* dalam *switching* antara sumber PLN dengan sumber PLTS berjalan baik dengan sistem otomatis dengan parameter yang sudah ditentukan.
2. Sistem *monitoring* berhasil bekerja dengan baik dalam pembacaan tegangan, arus, dan frekuensi dari sumber listrik PLN, PLTS, dan mampu *memonitoring* konsumsi daya listrik yang terpakai secara *real time* dan dimana saja menggunakan IoT.
3. Sistem *smart grid* berhasil bekerja dengan baik yang dapat mengoptimalkan penggunaan PLTS dengan menggunakan beban sebesar 770 Wh dan menunjukkan rata-rata persentasi dari penggunaan daya listrik dari PLN sebesar 70,65% dan penghematan daya listrik sebesar 29,35%.

5.2 Saran

Penelitian selanjutnya diharapkan dapat menyempurnakan dan mengembangkan penelitian ini, untuk itu saran penelitian selanjutnya sebagai berikut.

1. Menambahkan *monitoring* sistem *array* panel surya untuk mengetahui perfoma dari sistem panel surya.
2. Menambahkan sumber Energi Baru Terbarukan (EBT) lain untuk dapat meningkatkan kapasitas daya listrik yang dibutuhkan.

DAFTAR PUSTAKA

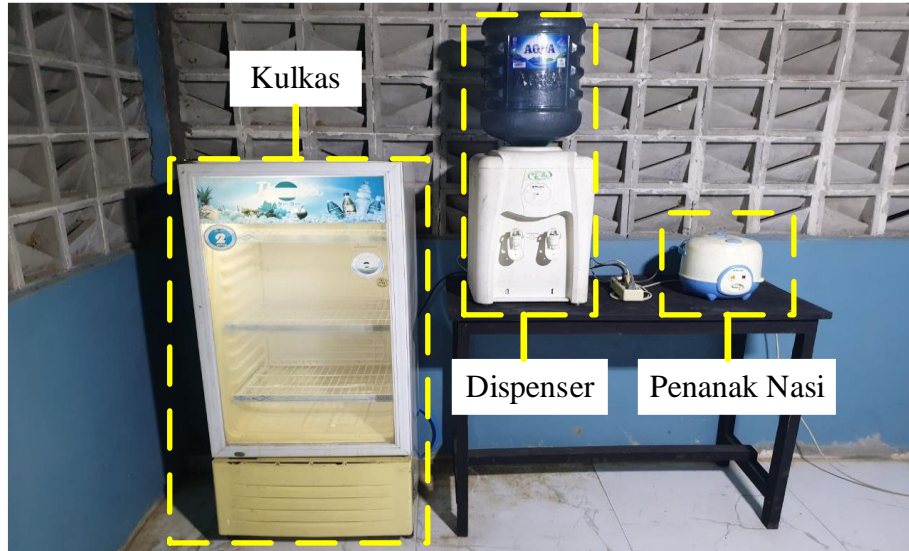
- [1] Primadona N., “Variabilitas Angin dan Gelombang Laut Sebagai Energi Terbarukan di Pantai Selatan Jawa Barat,” *Jurnal Akuatika*, vol. 1, pp. 8–15, Mar. 2014.
- [2] Iqbal M. S., Y. A. K. Niazi, U. Amir Khan, dan B. W. Lee, “Real-time fault detection system for large scale grid integrated solar photovoltaic power plants,” *International Journal of Electrical Power and Energy Systems*, vol. 130, Sep. 2021, doi: 10.1016/j.ijepes.2021.106902.
- [3] Anonim, *Peraturan Menteri Energi Sumber Daya Mineral Nomor 53 Tahun 2018*.
- [4] Yu D., A. G. Ebadi, K. Jermisittiparsert, N. H. Jabarullah, M. V. Vasiljeva, dan S. Nojavan, “Risk-Constrained Stochastic Optimization of a Concentrating Solar Power Plant,” *IEEE Trans Sustain Energy*, vol. 11, no. 3, pp. 1464–1472, Jul. 2020, doi: 10.1109/TSTE.2019.2927735.
- [5] Naim M., “Rancangan Sistem Kelistrikan PLTS Off Grid 1000 Watt Di Desa Mahalona Kecamatan Towuti,” *DINAMIKAJurnalIlmiahTeknikMesin*, vol. 9, no. 1, pp. 27–32, Nov. 2017.
- [6] Asriyadi, A. Wawan Indrawan, S. Pranoto, A. Rizal Sultan, dan R. Ramadhan, “Rancang Bangun Automatic Transfer Switch (ATS) Pada PLTS dan PLN serta Genset,” *ELEKTRIKA*, no. 2, pp. 225–235, Nov. 2016.
- [7] Pakpahan R., D. N. Ramadan, dan S. Hadiyoso, “Rancang Bangun Dan Implementasi Automatic Transfer Switch (ATS) Menggunakan Arduino Uno Dan Relai,” *Jurnal Elektro Telekomunikasi Terapan*, pp. 332–341, Dec. 2017.
- [8] Rahman F., A. Natsir, dan W. W. Giri, “Rancang Bangun ATS/AMF Sebagai Pengalih Catu Daya Otomatis Berbasis Programmable Logic Control,” *Dielektrika*, vol. 2, no. 2, pp. 164–172, Aug. 2017.
- [9] Rauf S., A. Wahab, M. Rizwan, S. Rasool, dan N. Khan, “Application of dc-grid for Efficient use of solar PV System in Smart Grid,” in *Procedia Computer Science*, 2016, vol. 83, pp. 902–906. doi: 10.1016/j.procs.2016.04.182.

- [10] Bouzguenda M., A. Gastli, A. H. Al Badi, dan T. Salmi, "Solar Photovoltaic Inverter Requirements for Smart Grid Applications," *IEEE PES Conference on Innovative Smart Grid Technologies*, Jun. 2012, doi: 10.1109/ISGT-MidEast.2011.6220799.
- [11] Suyanto H. dan A. Hariyanto, "Pengembangan Teknologi Sistem Smart Microgrid Di Sekolah Tinggi Teknik PLN," *Jurnal Sutet*, vol. 7, no. 1, pp. 1–67, May 2017.
- [12] Alreshidi A. dan A. Ahmad, "Architecting Software for the Internet of Thing Based Systems," *Future Internet*, vol. 11, no. 7, pp. 1–30, Jul. 2019, doi: 10.3390/fi11070153.
- [13] Alipudin A. M., D. Notosudjono, dan D. B. Fiddiansyah, "Rancang Bangun Alat Monitoring Biaya Listrik Terpakai Berbasis Internet of Things (IoT)," *Program Studi Teknik Elektro, Fakultas Teknik-Universitas Pakuan*, pp. 1–11, 2019.
- [14] Indra D., E. I. Alwi, dan M. A. Mubarak, "Prototipe Sistem Kontrol Pemadam Kebakaran Pada Rumah Berbasis Arduino Uno dan ESP8266," *Komputika : Jurnal Sistem Komputer*, vol. 11, no. 1, pp. 1–8, Sep. 2021, doi: 10.34010/komputika.v11i1.4801.
- [15] Eryawan B., A. E. Jayati, dan S. Heranurweni, "Rancang Bangun Prototype Smart Home Dengan Konsep Internet Of Things (IoT) Menggunakan Raspberry Pi Berbasis Web," *eLEKTRIKAL*, vol. 11, no. 2, pp. 1–5, 2019.
- [16] Hidayatullah N. A. dan D. E. J. Sudirman, "Desain Dan Aplikasi Internet Of Thing (IoT) Untuk Smart Grid Power System," *VOLTJurnal Ilmiah Pendidikan Teknik Elektro*, vol. 2, no. 1, pp. 35–44, Apr. 2017.
- [17] Tjandi Y. dan S. Kasim, "Kendali Peralatan Listrik Berbasis Smartphone," *Jurnal SCIENTIFIC PINI*, vol. 1, no. 1, pp. 73–81, Oct. 2015.
- [18] Safitr N. dan T. Rihayat, *Buku Teknologi Photovoltaic*. YayasanPuga Aceh Riset, 2019. Available: <https://www.researchgate.net/publication/341909134>
- [19] Sianipar R., "Dasar Perencanaan Pembangkit Listrik Tenaga Surya," *JETri*, vol. 11, pp. 61–78, 2014.

- [20] Muliadi, A. Imran, dan M. Rasul, “Pengembangan Tempat Sampah Pintar Menggunakan Esp32,” *Jurnal MEDIA ELEKTRIK*, vol. 17, no. 2, pp. 2721–9100, Apr. 2020.
- [21] Nirwan S. dan M. Hafidz, “Rancang Bangun Aplikasi Untuk Prototipe Sistem Monitoring Konsumsi Energi Listrik Pada Peralatan Elektronik Berbasis Pzem-004t,” *Jurnal Teknik Informatika*, vol. 12, no. 2, pp. 22–28, 2020.
- [22] Fitriandi A., E. Komalasari, dan H. Gusmedi, “Rancang Bangun Alat Monitoring Arus dan Tegangan Berbasis Mikrokontroler dengan SMS Gateway,” *Jurnal Rekayasa dan Teknologi Elektro*, vol. 10, no. 2, pp. 87–98, May 2016.
- [23] Kresnha P. E., S. N. Ambo, dan Y. Sosrowiguno, “Smart Outdoor Hidroponik Dengan Pengaturan Penyinaran Matahari dan Hujan Berbasis Mikrokontroller,” *Jurnal Sains, Teknologi dan Industri*, vol. 16, no. 1693–2390, pp. 77–82, Dec. 2018.
- [24] Cahya G. I. dan B. Winardi, “Perancangan Pengatur Suplai Daya Listrik Pada Sistem Hibrid Pembangkit Listrik Tenaga Surya Dengan Jala-Jala Listrik Pln,” *TRANSIENT*, vol. 7, no. 2, Jun. 2018.
- [25] Saputra A. A., D. Notosudjono, dan B. B. Rijadi, “Smart Grid Hybrid System (Fotovoltaik-PT. PLN) Berbasis IoT (Internet Of Things),” *Program Studi Teknik Elektro, Fakultas Teknik-Universitas Pakuan*, vol. 2, no. 1, 2019.
- [26] Fikra Y., D. Suryadi, dan R. R. Yacoub, “Rancang Bangun Automatic Transfer Switch (ATS) Dengan Parameter Arus, Frekuensi Dan Suhu,” *Program Studi Teknik Elektro Fakultas Teknik UNTAN Pontianak*, vol. 4, no. 3, 2018.

LAMPIRAN A PERALATAN LISTRIK

- Beban Listrik Peralatan Rumah tangga



- Beban Listrik Lampu Jalan



LAMPIRAN B HASIL PENGUJIAN

Tabel 1 Hasil Pengukuran Sensor 1 PZEM-004t

Tegangan(V)		
PZEM-004T	Quality power meter	Error(%)
221,2	220,4	0,36%
221,2	220,4	0,36%
221,2	220,4	0,36%
221,2	220,4	0,36%
221,2	220,4	0,36%
221,7	220,5	0,54%
221,7	220,5	0,54%
221,7	220,5	0,54%
221,7	220,5	0,54%
221,7	220,5	0,54%
Rata-rata		0,45%

Arus(A)		
PZEM-004T	Quality power meter	Error(%)
1,98	2	1,00%
1,97	2	1,50%
2,01	2	0,50%
2,03	2	1,50%
3,01	3	0,33%
2,99	3	0,33%
2,98	3	0,67%
3,97	4	0,75%
4,03	4	0,75%
4,06	4	1,50%
Rata-rata		0,88%

Frekuensi(Hz)		
PZEM-004T	Quality power meter	Error(%)
50,02	50,05	0,06%
50,02	50,05	0,06%
50,02	50,05	0,06%
50,02	50,05	0,06%
50,02	50,05	0,06%
50,02	50,05	0,06%
50,02	50,05	0,06%
50,02	50,05	0,06%
50,02	50,05	0,06%
50,02	50,05	0,06%
Rata-rata		0,06%

Tabel 2 Hasil Pengukuran Sensor 2 PZEM-004t

Tegangan(V)		
PZEM-004T	Quality power meter	Error(%)
221,4	220,4	0,45%
221,4	220,4	0,45%
221,4	220,4	0,45%
221,4	220,4	0,45%
221,4	220,4	0,45%
221,5	220,5	0,45%
221,5	220,5	0,45%
221,5	220,5	0,45%
221,5	220,5	0,45%
221,5	220,5	0,45%
Rata-rata		0,45%

Arus(A)		
PZEM-004T	Quality power meter	Error(%)
1,99	2	0,50%
1,98	2	1,00%
2,02	2	1,00%
2,04	2	2,00%
3,02	3	0,67%
2,99	3	0,33%
2,97	3	1,00%
3,95	4	1,25%
4,02	4	0,50%
4,05	4	1,25%
Rata-rata		0,95%

Frekuensi(Hz)		
PZEM-004T	Quality power meter	Error(%)
50	50,05	0,10%
50	50,05	0,10%
50	50,05	0,10%
50	50,05	0,10%
50	50,05	0,10%
50	50,05	0,10%
50	50,05	0,10%
50	50,05	0,10%
50	50,05	0,10%
50	50,05	0,10%
Rata-rata		0,10%

Tabel 3 Hasil Pengukuran Sensor 1 ACS712

Arus(A)		
ACS712 (A)	Ampere Meter (A)	Error
5,08	5	1,60%
5,61	5,5	2,00%
5,95	6	0,83%
6,59	6,5	1,38%
7,11	7	1,57%
7,38	7,5	1,60%
8,12	8	1,50%
8,4	8,5	1,18%
8,83	9	1,89%
9,65	9,5	1,58%
Rata-rata		1,51%

Tabel 4 Hasil Pengukuran Sensor 2 ACS712

Arus(A)		
ACS712 (A)	Ampere Meter (A)	Error(%)
4,905	5	1,90%
5,58	5,5	1,45%
6,11	6	1,83%
6,41	6,5	1,38%
7,13	7	1,86%
7,59	7,5	1,20%
7,92	8	1,00%
8,64	8,5	1,65%
9,06	9	0,67%
9,66	9,5	1,68%
Rata-rata		1,46%

Tabel 5 Hasil Pengukuran Sensor Tegangan

Tegangan(V)		
Sensor Tegangan (V)	Volt Meter (V)	Error(%)
20,15	20	0,75%
21,13	21	0,62%
21,88	22	0,55%
23,18	23	0,78%
23,91	24	0,37%
24,89	25	0,44%
26,15	26	0,58%
27,14	27	0,52%
28,11	28	0,39%
29,4	29	1,38%
Rata-rata		0,64%

Tabel 6 Hasil Pengujian Alat *Smart Grid* Hari ke-1

Hari Ke-1	Tegangan PLN (V)	Arus PLN (A)	Frekuensi PLN (Hz)	Faktor Daya PLN	Daya PLN (W)	Tegangan PLTS (V)	Arus PLTS (A)	Frekuensi PLTS (Hz)	Faktor Daya PLTS	Daya PLTS (W)
7:00:00 AM	225,40	0,85	50,00	0,97	185,80	0,00	0,00	0,00	0,00	0,00
8:00:00 AM	224,50	0,00	50,90	0,00	0,00	221,20	1,31	50,00	0,81	235,10
9:00:00 AM	223,00	0,00	50,00	0,00	0,00	221,20	1,33	50,00	0,81	238,80
10:00:00 AM	223,40	0,00	50,00	0,00	0,00	221,30	1,03	50,00	0,81	185,20
11:00:00 AM	223,20	1,01	50,00	0,82	185,30	221,10	0,00	50,00	0,00	0,00
12:00:00 PM	222,70	0,98	50,00	0,82	178,90	221,10	0,00	50,00	0,00	0,00
1:00:00 PM	222,70	0,00	50,00	0,00	0,00	221,40	1,30	50,00	0,82	236,70
2:00:00 PM	223,40	0,00	50,00	0,00	0,00	221,40	1,42	50,00	0,75	235,80
3:00:00 PM	223,10	1,03	49,90	0,82	188,90	221,20	0,00	50,00	0,00	0,00
4:00:00 PM	223,00	1,04	50,00	0,81	188,70	221,30	0,00	50,00	0,00	0,00
5:00:00 PM	222,60	1,05	49,90	0,81	189,30	221,30	0,00	50,00	0,00	0,00
6:00:00 PM	222,30	4,37	50,00	0,74	718,90	221,40	0,00	50,00	0,00	0,00
7:00:00 PM	223,30	0,00	50,00	0,75	0,00	221,30	3,95	50,00	0,82	717,40
8:00:00 PM	223,30	0,00	49,90	0,00	0,00	221,30	3,95	50,00	0,82	716,60
9:00:00 PM	221,00	0,00	50,00	0,00	0,00	221,30	4,20	50,00	0,77	715,40
10:00:00 PM	218,60	4,38	50,00	0,74	708,70	221,10	0,00	50,00	0,00	0,00
11:00:00 PM	223,40	4,29	49,90	0,74	709,20	221,30	0,00	50,00	0,00	0,00
12:00:00 AM	223,70	4,29	50,00	0,74	709,40	221,20	0,00	50,00	0,00	0,00
1:00:00 AM	226,30	3,87	50,00	0,81	708,60	221,20	0,00	50,00	0,00	0,00
2:00:00 AM	225,80	4,24	50,00	0,74	709,30	221,20	0,00	50,00	0,00	0,00
3:00:00 AM	225,70	4,25	50,00	0,74	709,20	221,30	0,00	50,00	0,00	0,00
4:00:00 AM	225,00	3,25	50,00	0,97	709,00	221,20	0,00	50,00	0,00	0,00
5:00:00 AM	223,70	4,28	50,00	0,74	707,80	221,30	0,00	50,00	0,00	0,00
6:00:00 AM	223,70	1,15	50,00	0,74	189,60	221,20	0,00	50,00	0,00	0,00

Tabel 7 Hasil Pengujian Alat *Smart Grid* Hari ke-2

Hari Ke-2	Tegangan PLN (V)	Arus PLN (A)	Frekuensi PLN (Hz)	Faktor Daya PLN	Daya PLN (W)	Tegangan PLTS (V)	Arus PLTS (A)	Frekuensi PLTS (Hz)	Faktor Daya PLTS	Daya PLTS (W)
7:00:00 AM	223,70	1,04	50,00	0,81	187,60	221,20	0,00	50,00	0,00	0,00
8:00:00 AM	223,70	1,29	49,90	0,81	234,10	221,30	0,00	50,00	0,00	0,00
9:00:00 AM	227,40	1,37	50,00	0,76	236,20	221,20	0,00	50,00	0,00	0,00
10:00:00 AM	225,10	0,00	50,09	0,00	0,00	221,30	1,06	50,00	0,76	178,90
11:00:00 AM	225,10	0,00	49,90	0,00	0,00	221,30	0,98	50,00	0,82	178,20
12:00:00 PM	225,70	0,97	50,00	0,85	186,80	221,30	0,00	50,00	0,00	0,00
1:00:00 PM	223,80	0,00	50,00	0,00	0,00	221,10	1,30	50,00	0,82	235,90
2:00:00 PM	223,60	0,00	49,90	0,00	0,00	221,30	1,30	50,00	0,82	236,20
3:00:00 PM	226,90	0,97	49,90	0,85	186,80	221,20	0,00	50,00	0,00	0,00
4:00:00 PM	226,10	0,99	50,00	0,83	186,60	221,30	0,00	50,00	0,00	0,00
5:00:00 PM	226,70	1,03	50,00	0,80	186,20	221,30	0,00	50,00	0,00	0,00
6:00:00 PM	225,00	3,85	50,00	0,83	718,20	221,20	0,00	50,00	0,00	0,00
7:00:00 PM	226,10	0,00	50,00	0,00	0,00	221,30	4,32	50,00	0,75	716,30
8:00:00 PM	226,10	0,00	50,00	0,00	0,00	221,20	3,95	50,00	0,82	716,50
9:00:00 PM	225,00	0,00	49,90	0,00	0,00	221,30	4,32	50,00	0,75	717,30
10:00:00 PM	225,00	0,00	50,00	0,00	0,00	221,30	4,32	50,00	0,75	717,10
11:00:00 PM	223,90	3,90	50,00	0,81	707,20	221,30	0,00	50,00	0,00	0,00
12:00:00 AM	225,80	3,26	50,00	0,96	707,40	221,30	0,00	50,00	0,00	0,00
1:00:00 AM	225,70	3,73	49,90	0,84	707,50	221,50	0,00	50,00	0,00	0,00
2:00:00 AM	225,70	3,73	50,00	0,84	707,50	221,50	0,00	50,00	0,00	0,00
3:00:00 AM	221,80	4,14	49,90	0,77	707,50	221,50	0,00	50,00	0,00	0,00
4:00:00 AM	225,70	4,29	50,00	0,73	707,20	221,50	0,00	50,00	0,00	0,00
5:00:00 AM	225,70	3,87	50,00	0,81	708,00	221,50	0,00	50,00	0,00	0,00
6:00:00 AM	225,70	1,02	50,00	0,81	186,10	221,19	0,00	50,00	0,00	0,00

Tabel 8 Hasil Pengujian Alat *Smart Grid* Hari ke-3

Hari Ke-3	Tegangan PLN (V)	Arus PLN (A)	Frekuensi PLN (Hz)	Faktor Daya PLN	Daya PLN (W)	Tegangan PLTS (V)	Arus PLTS (A)	Frekuensi PLTS (Hz)	Faktor Daya PLTS	Daya PLTS (W)
7:00:00 AM	221,80	1,46	50,00	0,73	236,60	221,39	0,00	50,00	0,00	0,00
8:00:00 AM	221,80	1,43	50,00	0,74	235,40	221,39	0,00	50,00	0,00	0,00
9:00:00 AM	221,80	0,83	50,00	0,96	185,60	221,30	0,00	50,00	0,00	0,00
10:00:00 AM	225,30	0,86	50,00	0,96	186,40	221,60	0,00	50,00	0,00	0,00
11:00:00 AM	223,50	0,00	50,00	0,00	0,00	221,60	1,04	50,00	0,81	186,40
12:00:00 PM	223,50	1,14	50,00	0,73	185,80	221,60	0,00	50,00	0,00	0,00
1:00:00 PM	223,50	0,00	50,00	0,00	0,00	221,50	1,32	50,00	0,81	236,10
2:00:00 PM	223,50	0,00	50,00	0,00	0,00	221,60	1,31	50,00	0,81	234,70
3:00:00 PM	226,70	1,12	50,00	0,73	185,20	221,60	0,00	50,00	0,00	0,00
4:00:00 PM	226,10	1,14	50,00	0,73	187,60	221,60	0,00	50,00	0,00	0,00
5:00:00 PM	225,70	1,03	50,00	0,81	188,70	221,60	0,00	50,00	0,00	0,00
6:00:00 PM	223,50	3,87	50,00	0,83	718,60	221,60	0,00	50,00	0,00	0,00
7:00:00 PM	223,50	0,00	50,00	0,00	0,00	221,50	3,30	50,00	0,98	716,40
8:00:00 PM	225,70	0,00	50,00	0,00	0,00	221,50	3,94	50,00	0,82	715,80
9:00:00 PM	225,70	0,00	50,00	0,00	0,00	221,60	3,36	50,00	0,96	715,50
10:00:00 PM	221,80	4,37	50,00	0,73	707,60	221,50	0,00	50,00	0,00	0,00
11:00:00 PM	221,80	4,37	50,00	0,73	707,90	221,50	0,00	50,00	0,00	0,00
12:00:00 AM	221,80	4,09	50,00	0,78	707,90	221,50	0,00	50,00	0,00	0,00
1:00:00 AM	227,40	4,04	50,00	0,77	706,90	221,40	0,00	50,00	0,00	0,00
2:00:00 AM	227,00	4,06	50,00	0,77	708,80	221,50	0,00	50,00	0,00	0,00
3:00:00 AM	226,80	4,05	50,00	0,77	707,30	221,50	0,00	50,00	0,00	0,00
4:00:00 AM	221,80	4,14	50,00	0,77	707,20	221,50	0,00	50,00	0,00	0,00
5:00:00 AM	224,20	3,58	50,00	0,88	706,90	221,50	0,00	50,00	0,00	0,00
6:00:00 AM	221,80	4,14	50,00	0,77	706,90	221,50	0,00	50,00	0,00	0,00

Tabel 9 Hasil Pengukuran Tegangan Baterai, Arus Beban, dan Arus Lampu

Waktu	Hari ke-1			Hari ke-2			Hari ke-3		
	Tegangan Baterai (V)	Arus Beban (A)	Arus Lampu (A)	Tegangan Baterai (V)	Arus Beban (A)	Arus Lampu (A)	Tegangan Baterai (V)	Arus Beban (A)	Arus Lampu (A)
7:00:00 AM	54,48	0,81	0,00	46,20	0,91	0,00	45,53	0,83	0,00
8:00:00 AM	54,48	1,23	0,00	46,26	1,21	0,00	45,67	1,28	0,00
9:00:00 AM	50,68	1,16	0,00	46,31	1,34	0,00	45,99	1,26	0,00
10:00:00 AM	46,65	0,81	0,00	47,54	0,86	0,00	46,20	0,87	0,00
11:00:00 AM	46,16	0,88	0,00	50,48	0,82	0,00	46,26	0,84	0,00
12:00:00 PM	46,20	0,88	0,00	46,27	0,88	0,00	46,31	0,98	0,00
1:00:00 PM	49,21	1,18	0,00	46,04	1,26	0,00	51,44	1,35	0,00
2:00:00 PM	46,65	1,21	0,00	45,08	1,34	0,00	48,65	1,34	0,00
3:00:00 PM	46,16	0,90	0,00	45,20	0,84	0,00	46,07	0,82	0,00
4:00:00 PM	46,20	0,82	0,00	45,46	0,85	0,00	46,19	0,88	0,00
5:00:00 PM	46,25	0,80	0,00	46,19	0,82	0,00	46,25	0,82	0,00
6:00:00 PM	46,30	0,88	3,42	46,31	0,82	3,38	46,31	0,88	3,37
7:00:00 PM	53,25	0,97	3,29	54,48	0,86	3,28	51,44	0,85	3,44
8:00:00 PM	50,70	0,85	3,36	51,44	0,22	3,42	49,38	0,96	3,42
9:00:00 PM	48,07	0,83	3,43	47,26	0,93	3,35	46,35	0,86	3,29
10:00:00 PM	46,28	0,92	3,32	46,65	0,97	3,30	46,07	0,83	3,28
11:00:00 PM	46,04	0,99	3,46	46,29	0,92	3,32	45,55	0,86	3,29
12:00:00 AM	45,99	0,97	3,27	46,20	0,91	3,32	45,03	0,86	3,30
1:00:00 AM	45,89	0,88	3,47	45,99	0,84	3,28	44,38	0,85	3,28
2:00:00 AM	45,79	0,94	3,37	45,89	0,99	3,31	44,31	0,89	3,48
3:00:00 AM	45,67	0,94	3,30	45,79	0,30	3,36	44,24	0,88	3,39
4:00:00 AM	45,55	0,85	3,36	45,67	0,96	3,48	44,24	0,83	3,45
5:00:00 AM	45,53	0,88	3,32	45,55	0,85	3,39	44,21	0,97	3,28
6:00:00 AM	45,48	0,86	0,00	45,53	0,91	0,00	44,19	0,96	0,00

Tabel 10 Hasil Pengukuran Energi Selama 3 Hari

Tanggal	Energi PLN (Wh)	Energi PLTS (Wh)	Total Energi (Wh)	Pemakaian PLN	Pemakaian PLTS
Hari Ke-1	7696,60	3281,00	10977,60	70,11%	29,89%
Hari Ke-2	7260,90	3696,40	10957,30	66,27%	33,73%
Hari Ke-3	8677,30	2804,90	11482,20	75,57%	24,43%

A. Perhitungan Pembangkit Listrik Tenaga Surya

1. Kebutuhan Beban

Nilai kebutuhan konsumsi energi didapat dari perhitungan Persamaan (2.2), sebagai contoh pada penggunaan beban lampu yaitu sebagai berikut:

a) Beban Lampu

Lama lampu menyala = 12 Hours

Daya lampu = 90 Watt

Jumlah lampu = 6 Buah

Maka:

$$\text{Konsumsi Daya Lampu(Wh)} = t.P.n$$

$$\text{Konsumsi Daya Lampu(Wh)} = 12 \text{ Hours} \times 90 \text{ Watt} \times 6 \text{ Buah}$$

$$\text{Konsumsi Daya Lampu(Wh)} = 6480$$

No.	Beban	Konsumsi Energi (Wh)
1	Lampu	6480
2	Kulkas	2880
3	Penanak Nasi	200
4	Dispenser	1440
Total Konsumsi Harian (Wh)		11000

2. Kebutuhan Panel Surya

Kebutuhan panel surya didapat dari perhitungan Persamaan (2.3), perhitungan tersebut dapat dilihat sebagai berikut:

$$Kapasitas\ panel\ surya = \frac{Total\ konsumsi\ harian}{waktu\ penyinaran\ matahari}$$

$$Kapasitas\ panel\ surya = \frac{11000\ Wh}{4}$$

$$Kapasitas\ panel\ surya = 2790\ Wp$$

Berdasarkan perhitungan di atas didapat banyaknya panel surya yang dibutuhkan 9 buah, dengan kapasitas masing-masing 310 Wp dipasang 3 seri dan 3 paralel menghasilkan 2790 Wp .

3. Kebutuhan Inverter

Kebutuhan besaran *inverter* berdasarkan kebutuhan keluaran listrik AC yang dapat dilihat dari beban daya maksimal dalam satuan waktunya. Beban maksimal yang digunakan dalam satuan waktu adalah 770 Watt, maka dari itu menggunakan *inverter* dengan spesifikasi 1000 Watt dengan tegangan kerja adalah 48 Volt, yang memiliki keluaran listrik AC dengan tegangan kurang lebih dari 220 Volt dan frekuensinya kurang lebih dari 50 Hz.

4. Kebutuhan Baterai

Kebutuhan baterai didapat dari perhitungan Persamaan (2.4), perhitungan tersebut dapat dilihat sebagai berikut:

$$Kapasitas\ baterai = \frac{Total\ konsumsi\ harian}{Tegangan\ baterai}$$

$$Kapasitas\ baterai = \frac{11000Wh}{48V} = 229,1\ Ah$$

Penelitian ini memiliki kapasitas baterai yang digunakan lebih kecil dari kebutuhan energi, agar sitem *smart grid* dapat berjalan ketika PLTS tidak dapat menyuplai keseluruhan total dari kebutuhan baterai yang harusnya dipenuhi sebesar 229,1 Ah, sedangkan baterai digunakan sebesar 150 Ah.

B. Perhitungan Persentasi Pemakaian Listrik

Perhitungan persentasi pemakaian listrik berdasarkan dari pemakaian yang digunakan oleh salah satu sumber dibagi dengan pemakaian keseluruhan dan dikali dengan 100%. Sebagai contoh perhitungan pada penggunaan sumber listrik PLTS pada hari ke-1 yaitu sebagai berikut:

$$\text{Energi PLTS} = 3281,00 \text{ Wh}$$

$$\text{Energi Total} = 10977,60 \text{ Wh}$$

Maka:

$$\text{Persentasi Pemakaian PLTS} = \frac{\text{Energi PLTS}}{\text{Energi Total}} \times 100\%$$

$$\text{Persentasi Pemakaian PLTS} = \frac{3281,00 \text{ Wh}}{10977,60 \text{ Wh}} \times 100\%$$

$$\text{Persentasi Pemakaian PLTS} = 29,89\%$$

LAMPIRAN C LISTING PROGRAM

```
// Deklarasi Thinger.Io //
#define THINGER_SERIAL_DEBUG
#include <ThingerESP32.h>

#define USERNAME "vieryrp"
#define DEVICE_ID "skripsi"
#define DEVICE_CREDENTIAL "VJILW%6czQVL1WZD"

#define SSID "SMARTGRID"
#define SSID_PASSWORD "smartgrid"

ThingerESP32 thing(USERNAME, DEVICE_ID, DEVICE_CREDENTIAL);
// ===== //

// Pembacaan Libarary //
#include <WiFi.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h> //Library I2C LCD 20X4
#include <PZEM004Tv30.h> //Library PZEM-004T
#include "RTCLib.h" //library DS3231
// ===== //

// Pembacaan LCD 20X4 //
LiquidCrystal_I2C lcd(0x27,20,4); // set the LCD address to 0x27
for a 16 chars and 2 line display
// ===== //

// Pembacaan Pin Pushbutton dan Variabel //
const int buttonPin = 2;
int buttonState = 0;
int count_value = 0;
```

```

int prestate;
// ===== //

// Pembacaan Pin Switch Thinger.io //
//#define Switch 12 //Pin switch dari Thinger.io
const int Switch = 12;
int buttonSwitch = 0;
// ===== //

// Waktu RTC DS3231 //
RTC_DS3231 rtc;
int jam, menit, detik;
// ===== //

// Pembacaan Relay dan Variabel //
#define relay0 14 //relay0 Sumber Beban
#define relay1 13 //relay1 Sumber Lampu
#define relay2 32 //relay2 On-Off Beban
#define relay3 23 //relay3 Timer / On-Off Lampu
// ===== //

// Pembacaan Voltage Divider //
int VPin = 34; // voltage divider Baterai 48 V
double volt0;
float R1 = 47000.0; //47k Ohm
float R2 = 1000.0; //1k Ohm
// ===== //

// Pembacaan Sensor Arus Lampu ACS712(20A)- 1 //
const int sensorIn1 = 33;
int mVperAmp1 = 100; // use 100 for 20A Module and 66 for 30A Module
double Voltage1 = 0;
double VRMS1 = 0;

```

```

double AmpsRMS1 = 0;
double Arus1 = 0;
// ===== //

// Pembacaan Sensor Arus Beban ACS712(20A)- 2 //
const int sensorIn2 = 36;
int mVperAmp2 = 100; // use 100 for 20A Module and 66 for 30A Module
double Voltage2 = 0;
double VRMS2 = 0;
double AmpsRMS2 = 0;
double Arus2 = 0;
// ===== //

// Sensor PLN PZEM-004T - 1 //
#if !defined(RXD1) && !defined(TXD1) && !defined(SERIAL1)
#define RXD1 16
#define TXD1 17
#define SERIAL1 Serial1
#endif
PZEM004Tv30 pzem1(SERIAL1, RXD1, TXD1); // PZEM-004T - 1
float TeganganPLN, ArusPLN, FrekuensiPLN, DayaPLN, EnergiPLN,
PowerFactorPLN;
// ===== //

// Sensor PLTS PZEM-004T - 2 //
#if !defined(RXD2) && !defined(TXD2) && !defined(SERIAL2)
#define RXD2 27
#define TXD2 26
#define SERIAL2 Serial2
#endif
PZEM004Tv30 pzem2(SERIAL2, RXD2, TXD2); // PZEM-004T - 2
float TeganganPLTSVac, ArusPLTSAac, FrekuensiPLTS, DayaPLTS,
EnergiPLTS, PowerFactorPLTS;
// ===== //

```

```

// Tambahan //
String s1, s2, Wifi, Kontrol, KontrolTh; // Inisiasi huruf
float d2, d3; // Inisiasi angka
// ===== //

int NUMBR_OF_SAMPLES = 50;

void setup()
{
  Serial.begin(921600);

  // Memulai Thingier.Io //
  WiFi.begin(SSID, SSID_PASSWORD);
  thing.add_wifi(SSID, SSID_PASSWORD);
  // ===== //

  // Memulai LCD 20X4 //
  lcd.init();
  lcd.backlight();
  // ===== //

  // Memulai Push Button //
  pinMode(buttonPin, INPUT);
  // ===== //

  // Memulai Switch Thingier.io //
  pinMode(Switch, OUTPUT);
  thing["Switch"] << digitalPin(Switch);
  // ===== //

  //Memulai Relay //
  pinMode(relay0, OUTPUT); //relay0 Sumber Beban

```



```

pinMode(relay1, OUTPUT); //relay1 Sumber Lampu
pinMode(relay2, OUTPUT); //relay2 On-Off Beban
pinMode(relay3, OUTPUT); //relay3 Timer / On-Off Lampu

digitalWrite(relay0, LOW); // R0 Sumber Beban (LOW = Mati)
digitalWrite(relay1, LOW); // R1 Sumber Lampu (LOW = Mati)
digitalWrite(relay2, HIGH); // R2 On-Off Beban (LOW = Mati)
digitalWrite(relay3, HIGH); // R3 Timer / On-Off Lampu (LOW =
Mati)

// ===== //

// Memulai RTC //
rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
// ===== //

// Pengiriman data ke platform THINGER.IO //
thing["DATA"] >> [](pson & out) {
  out["TeganganBATT"] = volt0;
  out["ArusLampu"] = Arus1;
  out["ArusBeban"] = Arus2;
  out["TeganganPLN"] = TeganganPLN;
  out["ArusPLN"] = ArusPLN;
  out["FrekuensiPLN"] =FrekuensiPLN;
  out["DayaPLN"] = DayaPLN;
  out["EnergiPLN"] = EnergiPLN;
  out["PowerFactorPLN"] = PowerFactorPLN;
  out["TeganganPLTS"] = TeganganPLTSVac;
  out["ArusPLTS"] = ArusPLTSAac;
  out["FrekuensiPLTS"] = FrekuensiPLTS;
  out["DayaPLTS"] = DayaPLTS;
  out["EnergiPLTS"] = EnergiPLTS;
  out["PowerFactorPLTS"] = PowerFactorPLTS;
  out["DayaLampu"] = d2;
  out["DayaBeban"] = d3;

```

```

    out["SumberLampu"] = s1;
    out["SumberBeban"] = s2;
    out["KontrolTh"] = KontrolTh;
};
// ===== //

}

void loop()
{
    // Thinger.Io Data //
    if (WiFi.status() != WL_CONNECTED) {
        Wifi = ("WiFi Disconnected");
        WiFi.begin(SSID, SSID_PASSWORD);
        thing.add_wifi(SSID, SSID_PASSWORD);
        Serial.println ("WiFi Disconnected");
    } else {
        Wifi = (" WiFi Connected");
        Serial.println ("WiFi Connected");
        thing.handle();
        // Thinger.Io Bucket //
        thing.write_bucket("Skripsi","DATA");
        // ===== //
    }
    // ===== //

    // Waktu RTC DS3231 //
    DateTime now = rtc.now();
    jam      = now.hour();
    menit    = now.minute();
    detik    = now.second();
    // ===== //

    // voltage divider Baterai 48 V //

```

```

double avg0 = 0;

    volt0 = 0;

for (int i = 0; i < NUMBR_OF_SAMPLES; i++){
    avg0 += (analogRead(VPin) * 3.6405241878) / 4095.0;
}

    volt0 = (avg0/NUMBR_OF_SAMPLES)/ (R2/(R1+R2)); // voltage
divider Baterai 48 V

// ===== //

// Sensor Arus Lampu ACS712-1 //
Voltage1 = getVPP1();
if(Voltage1 >= 0.04){
    VRMS1 = (Voltage1/2.0) *0.707; //root 2 is 0.707
    AmpsRMS1 = (VRMS1 * 1000)/mVperAmp1;
    Arus1 = ((0.44528 * AmpsRMS1)- 0.04722)* 1.28;
}
// ===== //

// Sensor Arus Beban ACS712-2 //
Voltage2 = getVPP2();
if(Voltage2 >= 0.04){
    VRMS2 = (Voltage2/2.0) *0.707; //root 2 is 0.707
    AmpsRMS2 = (VRMS2 * 1000)/mVperAmp2;
    Arus2 = ((0.42937 * AmpsRMS2)- 0.04553) * 1.28;
}
// ===== //

// Penampilan Data Sensor PLN PZEM-004T - 1 //
TeganganPLN = pzem1.voltage();
ArusPLN = pzem1.current();
FrekuensiPLN = pzem1.frequency();
DayaPLN = pzem1.power();

```

```

EnergiPLN = pzem1.energy();
PowerFactorPLN = pzem1.pf();
// ===== //

// Penampilan Data Sensor PLTS PZEM-004T - 2 //
TeganganPLTSVac = pzem2.voltage();
ArusPLTSAac = pzem2.current();
FrekuensiPLTS = pzem2.frequency();
DayaPLTS = pzem2.power();
EnergiPLTS = pzem2.energy();
PowerFactorPLTS = pzem2.pf();
// ===== //

buttonSwitch = digitalRead(Switch);
if (buttonSwitch == LOW){
    // Kotrol Otomatis //
    Kontrol = ("Kontrol Otomatis");
    KontrolTh = ("Otomatis");
    // Otomatis Relay Lampu //
    if
        (TeganganPLTSVac >= 220.0 && TeganganPLTSVac <= 235.0
        && FrekuensiPLTS >= 49.0 && FrekuensiPLTS <= 51.0
        && volt0 > 46.32
        && Arus1 < 4.44)
    {
        digitalWrite(relay1, HIGH);
        digitalWrite(relay0, HIGH);
        d2 = TeganganPLTSVac * Arus1;
        s1 = ("PLTS");
    }
    else if
        (TeganganPLN >= 220.0 && TeganganPLN <= 235.0
        && FrekuensiPLN >= 49.0 && FrekuensiPLN <= 51.0)

```

```

{
    digitalWrite(relay1, LOW);
    digitalWrite(relay0, LOW);
    d2 = TeganganPLN * Arus1;
    s1 = ("PLN");
}
// ===== //

// Otomatis Relay Beban //
if
(TeganganPLTSVac >= 220.0 && TeganganPLTSVac <= 235.0
&& FrekuensiPLTS >= 49.0 && FrekuensiPLTS <= 51.0
&& volt0 > 46.32
&& Arus2 < 4.44)
{
    digitalWrite(relay0, HIGH);
    digitalWrite(relay1, HIGH);
    d3 = TeganganPLTSVac * Arus2;
    s2 = ("PLTS");
}
else if
(TeganganPLN >= 220.0 && TeganganPLN <= 235.0
&& FrekuensiPLN >= 49.0 && FrekuensiPLN <= 51.0)
{
    digitalWrite(relay0, LOW);
    digitalWrite(relay1, LOW);
    d3 = TeganganPLN * Arus2;
    s2 = ("PLN");
}
// ===== //

// Otomatis Waktu Lampu //
if

```

```

(jam >= 18.0 && jam < 24.0 || jam >= 0.0 && jam < 6.0)
{
    digitalWrite(relay3, HIGH);
} else {
    digitalWrite(relay3, LOW);
}
// ===== //
}

else if (buttonSwitch == HIGH) {
    // Kotrol Manual //
    Kontrol = (" Kontrol Manual");
    KontrolTh = ("Manual");
    // Penerimaan data dari platform THINGER.IO //
    thing["relay0"] << digitalPin(relay0);
    thing["relay1"] << digitalPin(relay1);
    thing["relay2"] << digitalPin(relay2);
    thing["relay3"] << digitalPin(relay3);
    // ===== //
}
// ===== //

// Perubahan Tampilan LCD 20x4 //
buttonState = digitalRead(buttonPin);
if (buttonState == LOW && prestate == 0) {
    count_value++;
    prestate = 1;
    lcd.clear();
} else if (buttonState == HIGH) {
    prestate = 0;
}
if (count_value > 5) {
    count_value = 1;
}

```

```

}
// ===== //

// Tampilan LCD 20x4 //
if (count_value == 1){
    lcd.clear();
    lcd.setCursor(19,0); // Column, line
    lcd.print("1");
    lcd.setCursor(5,0);
    lcd.print("SMART GRID");
    lcd.setCursor(5,1);
    lcd.print("BY VIERY RP");
    lcd.setCursor(2,2);
    lcd.print(Kontrol);
    lcd.setCursor(2,3);
    lcd.print(Wifi);
}
else if (count_value == 2){
    lcd.clear();
    lcd.setCursor(19,0); // Column, line
    lcd.print("2");
    lcd.setCursor(2,0); // Column, line
    lcd.print("LISTRIK AC PLN");
    lcd.setCursor(0,1);
    lcd.print("V: "); lcd.print(TeganganPLN,1); lcd.print("V");
    lcd.setCursor(10,1);
    lcd.print("I: "); lcd.print(ArusPLN,2); lcd.print("A");
    lcd.setCursor(0,2);
    lcd.print("F: "); lcd.print(FrekuensiPLN); lcd.print("Hz");
    lcd.setCursor(10,2);
    lcd.print("PF: "); lcd.print(PowerFactorPLN);
    lcd.setCursor(0,3);
    lcd.print("P: "); lcd.print(DayaPLN); lcd.print("W");
}

```

```

        lcd.setCursor(10,3);
        lcd.print("E:"); lcd.print(EnergiPLN,1); lcd.print("Kwh");
    }
else if (count_value == 3){
    lcd.clear();
    lcd.setCursor(19,0); // Column, line
    lcd.print("3");
    lcd.setCursor(2,0); // Column, line
    lcd.print("LISTRIK AC PLTS");
    lcd.setCursor(0,1);
    lcd.print("V:          ");          lcd.print(TeganganPLTSVac,1);
lcd.print("V");

    lcd.setCursor(10,1);
    lcd.print("I: "); lcd.print(ArusPLTSAac,2); lcd.print("A");
    lcd.setCursor(0,2);
    lcd.print("F: "); lcd.print(FrekuensiPLTS); lcd.print("Hz");
    lcd.setCursor(10,2);
    lcd.print("PF: "); lcd.print(PowerFactorPLTS);
    lcd.setCursor(0,3);
    lcd.print("P: "); lcd.print(DayaPLTS); lcd.print("W");
    lcd.setCursor(10,3);
    lcd.print("E:"); lcd.print(EnergiPLTS,1); lcd.print("Kwh");
}
else if (count_value == 4){
    lcd.clear();
    lcd.setCursor(19,0); // Column, line
    lcd.print("4");
    lcd.setCursor(2,0); // Column, line
    lcd.print("PEMAKAIAN LAMPU");
    lcd.setCursor(7,1);
    lcd.print(String() +jam+":" +menit+":"+detik+" ");
    lcd.setCursor(0,2);
    lcd.print("BATT: "); lcd.print(volt0,2); lcd.print("V");
    lcd.setCursor(13,2);

```



```

    lcd.print("S: "); lcd.print(s1);
    lcd.setCursor(0,3);
    lcd.print("A : "); lcd.print(Arus1,2); lcd.print("A");
    lcd.setCursor(10,3);
    lcd.print("P : "); lcd.print(d2,1); lcd.print("w");
}
else if (count_value == 5){
    lcd.clear();
    lcd.setCursor(19,0); // Column, line
    lcd.print("5");
    lcd.setCursor(2,0); // Column, line
    lcd.print("PEMAKAIAN BEBAN");
    lcd.setCursor(7,1);
    lcd.print(String() +jam+":" +menit+":"+detik+" ");
    lcd.setCursor(0,2);
    lcd.print("BATT: "); lcd.print(volt0,2); lcd.print("V");
    lcd.setCursor(13,2);
    lcd.print("S: "); lcd.print(s2);
    lcd.setCursor(0,3);
    lcd.print("A : "); lcd.print(Arus2,2); lcd.print("A");
    lcd.setCursor(10,3);
    lcd.print("P : "); lcd.print(d3,1); lcd.print("w");
}
// ===== //

delay(100);

}

// Sensor Arus Lampu ACS712-2 //
float getVPP1()
{
    float result;

```

```

int readValue; //value read from the sensor
int maxValue = 0; // store max value here
int minValue = 4095; // store min value here

uint32_t start_time = millis();
while((millis()-start_time) < 1000) //sample for 3 Sec
{
    readValue = analogRead(sensorIn1);
    // see if you have a new maxValue
    if (readValue > maxValue)
    {
        /*record the maximum sensor value*/
        maxValue = readValue;
    }
    if (readValue < minValue)
    {
        /*record the minimum sensor value*/
        minValue = readValue;
    }
}

// Subtract min from max
result = ((maxValue - minValue) * 3.3)/4095.0;
return result;
}
// ===== //

// Sensor Arus Beban ACS712-3 //
float getVPP2()
{
    float result;
    int readValue; //value read from the sensor
    int maxValue = 0; // store max value here

```

```

int minValue = 4095; // store min value here

uint32_t start_time = millis();
while((millis()-start_time) < 1000) //sample for 3 Sec
{
    readValue = analogRead(sensorIn2);
    // see if you have a new maxValue
    if (readValue > maxValue)
    {
        /*record the maximum sensor value*/
        maxValue = readValue;
    }
    if (readValue < minValue)
    {
        /*record the minimum sensor value*/
        minValue = readValue;
    }
}

// Subtract min from max
result = ((maxValue - minValue) * 3.3)/4095.0;
return result;
}
// ===== //

```