

PENERAPAN TEKNOLOGI IOT DALAM MONITORING DAN
PENGEDALIAN DAYA LISTRIK



LAPORAN TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat Untuk Menyelesaikan
Pendidikan Diploma Tiga (D-3) Program Studi Teknik Telekomunikasi
Jurusan Teknik Elektro
Politeknik Negeri Ujung Pandang

HILDA RIANYSKA

322 19 032

MUHAMMAD ICHSAN ARIEF

MARZUKI

322 19 038

PROGRAM STUDI D-3 TEKNIK TELEKOMUNIKASI

JURUSAN TEKNIK ELEKTRO

POLITEKNIK NEGERI UJUNG PANDANG

MAKASSAR

2022

HALAMAN PENGESAHAN

Laporan Tugas Akhir ini dengan judul "Penerapan Teknologi IoT dalam Monitoring dan Pengendalian Daya Listrik" oleh Hilda Rianyska NIM 322 19 032 dan Muhammad Ichsan Arief Marzuki NIM 322 19 038 dinyatakan layak untuk diujikan.

Makassar, September 2022

Pembimbing 1

Menyetujui,

Pembimbing 2

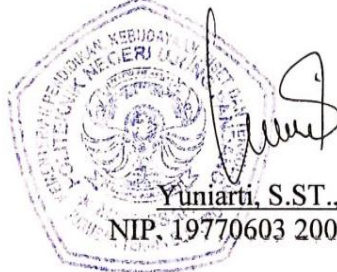


Nuraeni Umar, S.T., M.T.
NIP. 19629012 198803 2 004



Nurul Khaerani Hamzidah, S.T., M.T.
NIP. 19890814 201903 2 020

Mengetahui,
Ketua Program Studi
D-3 Teknik Telekomunikasi



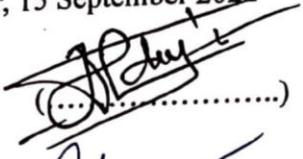





Yuniarti, S.ST., M.T.
NIP. 19770603 200212 2 002

HALAMAN PENERIMAAN

Pada hari ini, Selasa tanggal 20 September 2022, tim penguji ujian tugas akhir telah menerima hasil seminar tugas akhir oleh mahasiswa Hilda Rianyska NIM 322 19 032 dan Muhammad Ichsan Arief Marzuki NIM 322 19 038 dengan judul “Penerapan Teknologi IoT dalam Monitoring dan Pengendalian Daya Listrik”.

Makassar, 15 September 2022

Tim Penguji Ujian Sidang Tugas Akhir:

1	Lidemar Halide, S.T.,M.T	Ketua	
2	Misnawati, S.T.,M.T	Sekretaris	
3	M.Ahyar, S.T.,M.T	Anggota	
4	Nuraeni Umar, S.T.,M.T	Anggota	
5	Zaini, S.ST.,M.T	Anggota	
6	Nurul Khaerani Hamzidah, S.T.,M.T	Anggota	

KATA PENGANTAR

Puji syukur kami panjatkan atas kehadiran Allah SWT karena atas Rahmat dan HidayahNya-lah, penulisan Laporan Tugas Akhir ini yang berjudul **“Penerapan Teknologi IoT dalam Monitoring dan Pengendalian Daya Listrik”** ini dapat diselesaikan dengan baik.

Tugas akhir ini dibuat dengan tujuan untuk memenuhi persyaratan kelulusan Program Studi Diploma III Teknik Telekomunikasi Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.

Dalam menyelesaikan tugas akhir ini, penulis berpegang pada teori yang penulis dapatkan dan pihak-pihak lain yang sangat membantu hingga terselesaikannya tugas akhir ini.

Penulis sangat menyadari bahwa laporan tugas akhir ini masih sangat banyak terdapat kekeliruan dan masih memerlukan perbaikan secara menyeluruh. Hal ini tidak lain karena keterbatasan ilmu dan kemampuan yang dimiliki penulis, karena itu berbagai masukan dan saran yang sifatnya membangun sangatlah diharapkan demi kesempurnaan laporan tugas akhir ini.

Penulis menyadari bahwa dari awal hingga selesainya laporan tugas akhir ini, banyak sekali pihak yang telah terlibat dan berperan serta dalam mewujudkan terselesaikannya tugas akhir ini, oleh karena itu penulis ingin menyampaikan rasa hormat dan ucapan terima kasih yang setinggi-tingginya kepada mereka yang secara moril maupun materil telah banyak membantu penulis untuk merampungkan laporan tugas akhir ini hingga selesai. Maka pada kesempatan kali ini pula penulis ingin menyampaikan ucapan terima kasih yang sedalam-dalamnya

kepada:

1. Kedua orang tua penulis, yang penuh kesabaran mendukung lewat doa dan semangat secara moril maupun materil kepada penulis sehingga dapat menyelesaikan studi pada jenjang yang lebih tinggi.
2. Bapak Prof. Ir. Muhammad Anshar, M.Si., Ph.D. selaku direktur Politeknik Negeri Ujung Pandang.
3. Bapak Ahmad Rizal Sultan, S.T., M.T., Ph.D. selaku Ketua Jurusan Teknik Elektro Politeknik Negeri Ujung Pandang.
4. Ibu Yuniarti, S.ST., M.T. selaku Koordinator Program Studi Diploma III Teknik Telekomunikasi Politeknik Negeri Ujung Pandang.
5. Ibu Nuraeni Umar, S.T., M.T. selaku Dosen Pembimbing I yang telah meluangkan waktu untuk membimbing, mengarahkan, dan memberi semangat dan dorongan kepada penulis dalam pengerjaan tugas akhir ini hingga selesai.
6. Ibu Nurul Khaerani Hamzidah, S.T., M.T. selaku Dosen Pembimbing II yang telah sabar dalam membimbing dan mendampingi penulis serta banyak membantu selama proses pengerjaan Tugas Akhir ini.
7. Seluruh staff pengajar Politeknik Negeri Ujung Pandang yang telah membimbing dan memberikan materi perkuliahan kepada penulis.
8. Teman-teman kelas B Teknik Telekomunikasi angkatan 2019, yang telah memberikan dukungan serta masukan kepada penulis.
9. Seluruh pihak-pihak terkait yang tidak bisa disebutkan namanya satu-persatu, yang telah banyak membantu penulis hingga terselesaikannya tugas akhir ini.

Semoga laporan Tugas Akhir ini dapat berguna bagi penulis dan bagi pembaca

pada umumnya. Penulis sadar sekali bahwa laporan ini masih memiliki kekurangan. Oleh karena itu, penulis mengharapkan saran dan kritikan yang sifatnya membangun.

Makassar, 17 September 2022



Penulis

DAFTAR ISI

HALAMAN SAMPUL	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PENERIMAAN	iii
KATA PENGANTAR.....	iv
DAFTAR ISI.....	vii
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xi
SURAT PERNYATAAN.....	1
RINGKASAN	2
BAB I PENDAHULUAN	3
1.1 Latar Belakang.....	3
1.2 Rumusan Masalah	4
1.3 Ruang Lingkup Kegiatan.....	4
1.4 Tujuan Kegiatan	5
1.5 Manfaat Kegiatan	5
BAB II TINJAUAN PUSTAKA.....	6
2.1 Pengertian Arus Listrik AC (<i>Alternating Current</i>)	8
2.2 Monitoring	11
2.3 Menghitung Tagihan Listrik Pascabayar dan Prabayar.....	12

2.3	<i>Internet of Things</i>	14
2.4	Program Processing Arduino IDE	16
2.5	ESP8266	19
2.6	Sensor PZEM004T	21
2.7	<i>Relay</i>	23
2.8	RTC DS3231	25
2.9	LCD (<i>Liquid Crystal Display</i>)	28
2.10	ESP32	30
2.11	Mengukur persentase kesalahan	32
BAB III METODE KEGIATAN		34
3.1	Tempat dan Waktu Perancangan	34
3.2	Alat dan Bahan Perancangan	34
3.3	Prosedur Perancangan	36
3.4	Pengambilan Data	40
BAB IV HASIL DAN PEMBAHASAN		42
4.1	Data Hasil Penelitian	42
4.2	Pengujian Sensor PZEM-004T untuk pembacaan tegangan dan arus	51
4.3	Pengujian Pembacaan <i>Timer</i> /RTC	54
4.4	Pengujian <i>Relay</i> /Stopkontak	55
4.5	Tampilan <i>Interface</i> pada aplikasi Blynk	57

4.6	Pembahasan	58
BAB V PENUTUP.....		60
5.1	Kesimpulan.....	60
5.2	Saran	60
DAFTAR PUSTAKA		61
LAMPIRAN		63



DAFTAR GAMBAR

Gambar 2.1 <i>Internet of Things</i>	15
Gambar 2.2 Program <i>Processing</i> Arduino IDE	17
Gambar 2.3 Bagian-bagian Arduino IDE.....	18
Gambar 2.4 ESP8266	20
Gambar 2.5 Pin-pin NodeMCU	20
Gambar 2.6 Sensor PZEM004T	21
Gambar 2.7 <i>Relay</i>	24
Gambar 2.8 RTC DS3231	26
Gambar 2.9 Konfigurasi pin pada RTCDS3231	27
Gambar 2.10 LCD (<i>Liquid Crystal Display</i>).....	28
Gambar 2.11 Modul 12C.....	29
Gambar 2.12 ESP32	31
Gambar 3.1 Diagram Alir Perancangan.....	36
Gambar 3.2 Diagram Blok Sistem.....	37
Gambar 3.3 Rangkaian Keseluruhan.....	37
Gambar 3.4 <i>Flowchart Software</i>	39
Gambar 4.1 Tampilan pada aplikasi Blynk.....	41
Gambar 4.2 Tampilan LCD pada sistem rancangan.....	42
Gambar 4.3 Tampilan waktu, hari, tanggal, bulan dan tahun pada LCD.....	54
Gambar 4.4 Tampilan pengendalian stop kontak pada aplikasi Blynk.....	55
Gambar 4.5 Nilai kWh pada beban.....	58
Gambar 4.6 Biaya pemakaian beban.....	59

DAFTAR TABEL

Tabel 2.1 Penggunaan listrik perhari dengan Daya listrik: 900 VA	12
Tabel 2.2 Fungsi Pin Sensor PZEM-004T	23
Tabel 2.3 <i>Four-Channel Relay Module Pinout</i>	25
Tabel 2.4 Penjelasan Pin RTC DS3231	27
Tabel 2.5 Spesifikasi LCD	28
Tabel 2.6 Spesifikasi <i>Module I2C</i>	30
Tabel 3.1 Bahan yang digunakan dalam perangkat lunak.....	34
Tabel 3.2 Bahan yang digunakan dalam perangkat keras.....	35
Tabel 4.1 Pengujian pada Beban A (<i>Charger Laptop Lenovo IdeaPad S340</i>).....	44
Tabel 4.2 Pengujian pada Beban B (<i>Charger Smartphone Asus Max Pro M2</i>) ...	45
Tabel 4.3 Pengujian pada Beban C (Kipas Angin Mitochiba).....	46
Tabel 4.4 Pengujian pada Beban D (Setrika).....	47
Tabel 4.5 Pengujian beban A, B, C, dan D dihubungkan secara bersama.....	48
Tabel 4.6 Perbandingan hasil pengamatan pada skema 1 dan skema 2.....	49
Tabel 4.7 Perbandingan besar biaya antara hasil perhitungan dan perangkat	50
Tabel 4.8 Hasil pengujian tegangan dan arus pada perangkat.....	52
Tabel 4.9 Perbandingan spesifikasi beban dengan hasil ukur perangkat.....	53
Tabel 4.10 Pengujian Data Pengontrolan Secara Jarak Jauh.....	55
Tabel 4.11 Tampilan <i>interface</i> pada aplikasi Blynk.....	57

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Hilda Rianyska / Muhammad Ichsan Arief Marzuki

NIM : 32219032 / 32219038

Menyatakan dengan sebenar-benarnya bahwa segala pernyataan dalam Laporan Tugas Akhir ini yang berjudul “Penerapan Teknologi IoT dalam Monitoring dan Pengendalian Daya Listrik” merupakan gagasan dan hasil karya saya sendiri dengan arahan komisi pembimbing dan belum pernah diajukan dalam bentuk apapun pada perguruan tinggi dan instansi manapun.

Semua data dan informasi yang digunakan telah dinyatakan secara jelas dan dapat diperiksa kebenarannya. Sumber informasi yang berasal atau dikutip dari karya yang telah diterbitkan dari penulis telah di sebutkan dalam naskah dan dicantumkan dalam Laporan Tugas Akhir ini.

Jika pernyataan saya tersebut diatas tidak benar, saya siap menanggung resiko yang ditetapkan oleh Politeknik Negeri Ujung Pandang.

Makassar, 15 September 2022

Mahasiswa I



Hilda Rianyska

32219032

Mahasiswa II

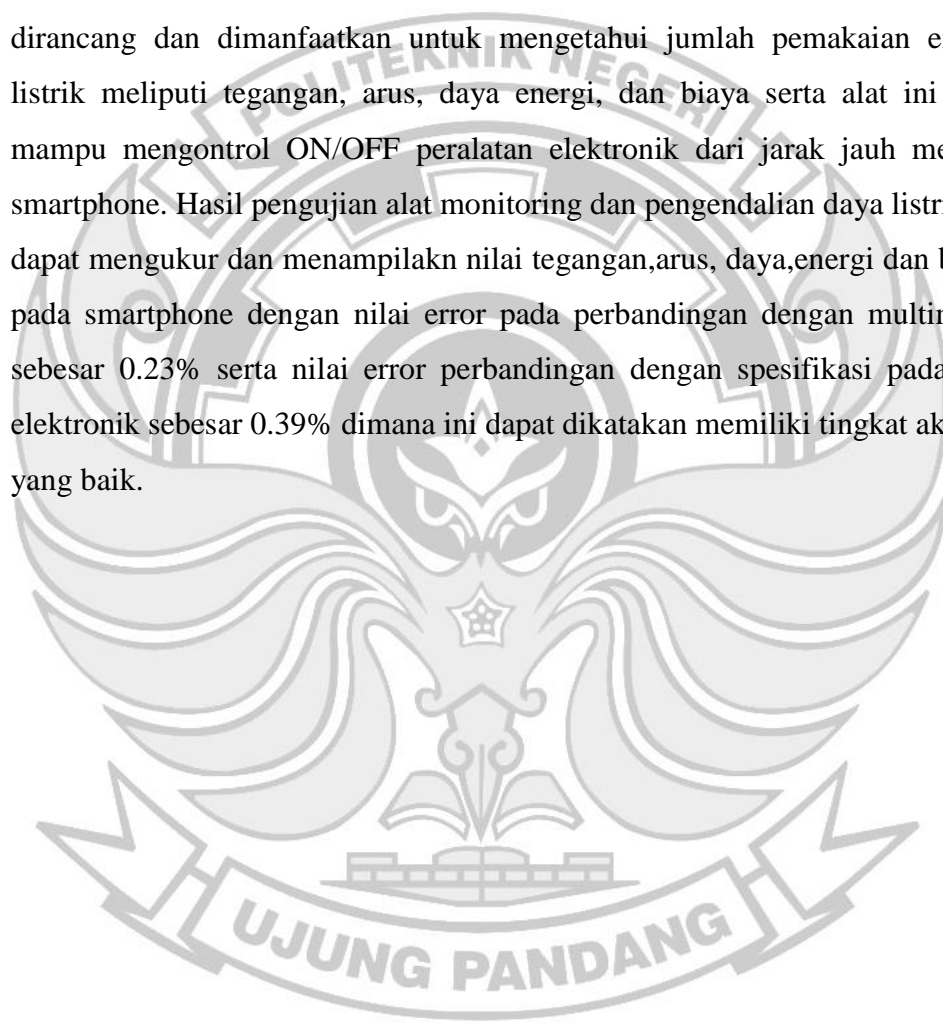


Muhammad Ichsan Arief Marzuki

32219038

RINGKASAN

Kelalaian manusia dalam pemakaian energi listrik akan menyebabkan keborosan yang juga berdampak pada kenaikan biaya pemakaian energi listrik. Oleh sebab itu, dibutuhkan alat yang mampu memonitoring pemakaian energi listrik dan mengendalikan perangkat elektronik melalui *smartphone*, walaupun pengguna energi listrik tidak berada dirumah. Alat monitoring dan pengendalian daya listrik adalah sebuah sistem yang dirancang dan dimanfaatkan untuk mengetahui jumlah pemakaian energi listrik meliputi tegangan, arus, daya energi, dan biaya serta alat ini juga mampu mengontrol ON/OFF peralatan elektronik dari jarak jauh melalui *smartphone*. Hasil pengujian alat monitoring dan pengendalian daya listrik ini dapat mengukur dan menampilkan nilai tegangan, arus, daya, energi dan biaya pada *smartphone* dengan nilai error pada perbandingan dengan multimeter sebesar 0.23% serta nilai error perbandingan dengan spesifikasi pada alat elektronik sebesar 0.39% dimana ini dapat dikatakan memiliki tingkat akurasi yang baik.



BAB I

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi saat ini berkembang dengan pesat dan mempermudah masyarakat dalam melakukan berbagai kegiatan. Banyak kegiatan masyarakat yang mulai dilakukan menggunakan *smartphone*, seperti berbelanja, transaksi bank, belajar, dan masih banyak lagi. Mayoritas teknologi menggunakan listrik sebagai tenaga utama. Hal ini yang menyebabkan tingkat konsumsi listrik yang semakin banyak dan menyebabkan tagihan listrik dalam jumlah yang tinggi. Berdasarkan data kementerian Energi dan Sumber Daya Mineral (ESDM), konsumsi listrik di Indonesia pada tahun 2021 naik menjadi 1.123 kWh/kapita dari tahun sebelumnya yaitu di tahun 2020 sebesar 1.089 kWh/kapita. Adapun pada tahun 2022, Kementerian ESDM memproyeksikan konsumsi daya listrik per-kapitanya akan naik signifikan menjadi 1.268 kWh/kapita. Padahal dalam kenyataannya, tidak seluruh teknologi digunakan secara terus menerus dan penggunaan listrik dapat dihemat dengan mematikan peralatan listrik yang tidak digunakan. Namun seringkali seseorang lupa, sehingga menyebabkan konsumsi listrik menjadi tinggi.

Untuk mengatasi hal tersebut, dilakukan penelitian dengan membuat sebuah sistem yang dapat memonitoring dan mengendalikan perangkat listrik melalui aplikasi pada *smartphone*. Sistem monitoring dan pengendali ini menggunakan prinsip IoT (*Internet Of Things*) dimana semua perangkat listrik dapat dihidupkan dan dimatikan menggunakan aplikasi yang terhubung dengan internet serta dapat memonitoring penggunaan daya, tegangan, arus dan energi yang digunakan oleh perangkat listrik. Salah satu kelebihan IoT dibandingkan pengontrolan

menggunakan prinsip *Bluetooth* ataupun *Wifi* adalah penggunaan yang mudah dan cepat serta pengaturan perangkat yang dapat diatur dengan jarak yang jauh.

Berdasarkan penjelasan dari latar belakang diatas, kemudian muncul ide untuk membuat sebuah rancangan Penerapan Teknologi IoT dalam Monitoring dan Pengendalian Daya Listrik yang dapat diakses dimana saja selagi koneksi internet masih terhubung serta dapat mengendalikan beban daya listrik agar tidak melebihi kapasitas daya yang semestinya.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya, maka permasalahan yang ada dapat dirumuskan sebagai berikut:

1. Bagaimana merancang sistem monitoring dan pengendalian daya listrik berbasis IoT?
2. Bagaimana memonitoring konsumsi daya listrik melalui *smartphone*?

1.3 Ruang Lingkup Kegiatan

Agar permasalahan yang dibahas lebih berfokus dan tidak melebar, penulis membatasi permasalahan yang akan dibahas. Adapun beberapa hal tersebut antara lain:

1. Menggunakan *relay* sebagai saklar digital
2. Menggunakan koneksi internet *wifi* sebagai sistem IoT (*Internet of Things*).
3. Koneksi internet yang lancar dan stabil.

1.4 Tujuan Kegiatan

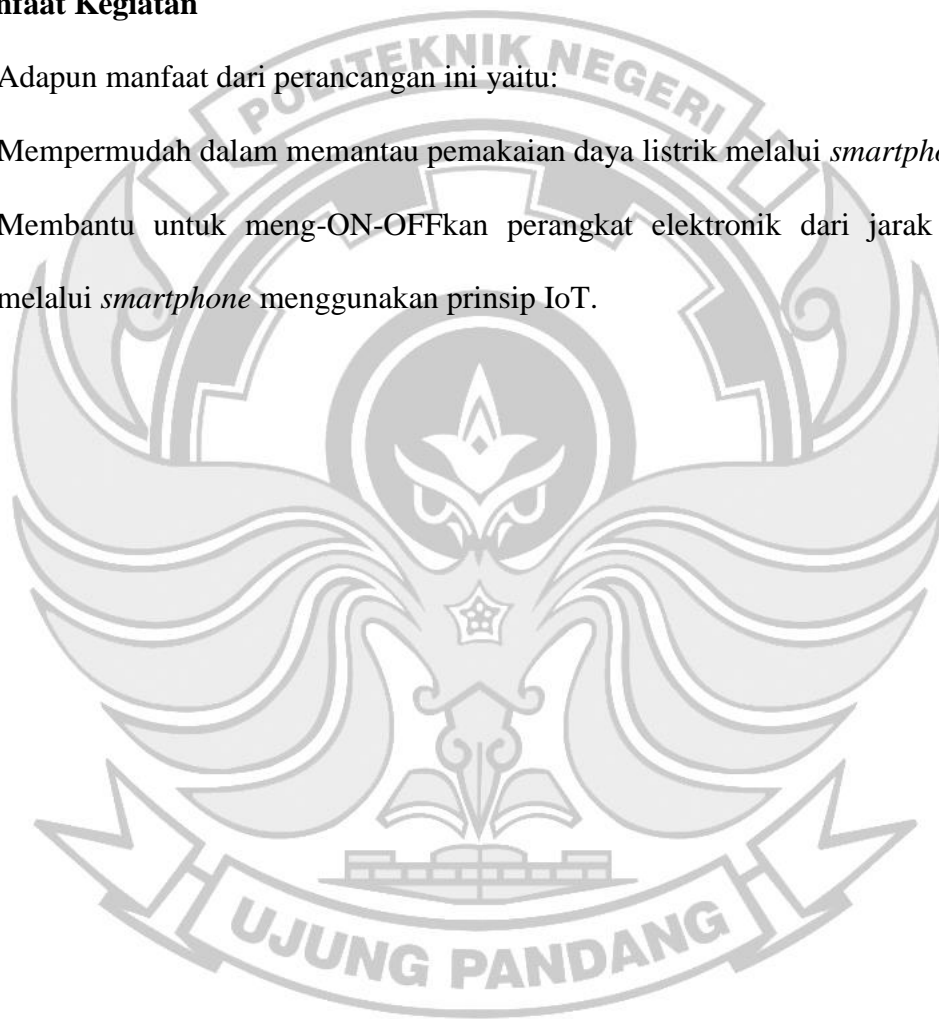
Tujuan dari perancangan ini yaitu:

1. Merancang sistem monitoring konsumsi dan pengendalian daya listrik berbasis IoT; dan
2. Memonitoring konsumsi daya listrik melalui *smartphone*.

1.5 Manfaat Kegiatan

Adapun manfaat dari perancangan ini yaitu:

1. Mempermudah dalam memantau pemakaian daya listrik melalui *smartphone*.
2. Membantu untuk meng-ON-OFFkan perangkat elektronik dari jarak jauh melalui *smartphone* menggunakan prinsip IoT.



BAB II

TINJAUAN PUSTAKA

Dalam jurnal (Alpin Nugraha dkk) menjelaskan tentang bagaimana merancang dan membuat prototipe sistem yang dapat memantau dan mengontrol pemakaian alat elektronik berbasis IoT atau melalui internet dimana data pemakaian listrik untuk disimpan di *Firestore realtime* database menggunakan sensor arus ACS712, sensor tegangan ZMPT101B serta mikrokontroler Arduino Uno dan NodeMCU V3. Alat ini akan mengirimkan data arus, tegangan serta watt kedalam database dan kemudian akan dikonversi menjadi kwh dan juga biaya yang ditimbulkan dari pemakaian listrik berdasarkan tarif dasar listrik PLN dan hasilnya dapat dilihat secara langsung melalui ponsel *Smartphone*. Sistem ini dapat memonitor pemakaian listrik untuk 3 unit alat elektronik yaitu pompa air aquarium (unit 1), TV CRT 14" (unit 2) dan Dispenser (unit 3). Sistem ini juga berhasil menurunkan pemakaian listrik yaitu pada unit 2 penurunan terjadi sebanyak 27.49% dan unit 3 sebanyak 57.28%.

Penelitian yang dilakukan oleh (Adam) mengenai Sistem Monitoring Arus dan Tegangan Menggunakan SMS Gateway. Penelitian ini didasari akibat digitalisasi yang mempengaruhi perkembangan sistem monitoring pemakaian listrik untuk peralatan rumah. Namun masih banyak pengguna energi listrik yang belum bisa memonitoring daya listrik secara detail. Sehingga penelitian ini membahas tentang sistem monitoring besaran listrik seperti arus dan tegangan. Sistem ini memanfaatkan sensor arus AC SCT 013-000 dan sensor tegangan ZMPT101B untuk membaca besaran listrik. Arduino mega memproses data yang dihasilkan oleh sensor-sensor dan mengirimkan data menggunakan 800L ke

handphone melalui SMS sehingga pengguna mampu memantau dan mengontrol pemakaian energi listrik secara cepat dan tepat. Data tegangan dan arus listrik juga tersimpan di data *logger* yang digunakan sebagai data cadangan jika sistem tidak bekerja secara normal/sistem *error*. Penggunaan SMS dirasakan handal karena hanya membutuhkan minimal jaringan GPRS/*Edge* untuk pengiriman data yang dikirim dengan karakter yang cukup panjang sehingga semua informasi data tegangan dan arus dapat dikirim seluruhnya ke pengguna.

Dalam jurnal (Hesti dan Marniati) menjelaskan tentang Rancang bangun terminal stop kontak Via SMS (*Short Message Service*) Berbasis mikrokontroler. Kelebihan dari alat ini adalah dapat mengendalikan alat-alat yang menggunakan energi listrik untuk menjalankan sistem yang dibawa didalam alat tersebut dari jarak jauh Sistem monitoring listrik berkembang seiring berjalannya waktu seperti penelitian yang dilakukan oleh (Mustamin dan Ardi, 2019) yaitu rancang bangun Sistem Kontrol KWH Meter Pelanggan Satu Phasa Berbasis *Internet of Things* yang menggunakan Arduino mega pro mini sebagai pengontrol utamanya. Kelebihan dari sistem ini yakni dapat memonitoring hingga jarak 20 meter melalui internet berupa tampilan data pada aplikasi *Smartphone* yang dirancang menggunakan aplikasi Inventor.

Penelitian yang dilakukan oleh (Rosmauli Situngkir) mengenai Rancang bangun pengendalian jarak jauh perangkat listrik serta monitoring penggunaan daya listrik berbasis IoT, Penelitian ini bertujuan untuk merancang sistem monitoring daya listrik berbasis mikrokontroler yang dapat memantau seberapa besar daya yang kita gunakan dan akan ditampilkan pada lcd. Penelitian dilakukan dengan merancang sebuah perangkat kendali jarak jauh melalui jaringan

komputer. Pada *system* kontrol dan monitoring lampu ini, bekerja berdasarkan pusat pengontrolan yang dalam system ini yaitu *website*. Jika lampu pada alat hidup maka didalam aplikasi bylink akan menampilkan ON. Dan sebaliknya jika lampu kita mati maka menampilkan OFF. Dan pada alat ini juga ditunjukkan besar daya yang digunakan.

2.1 Pengertian Arus Listrik AC (*Alternating Current*)

Arus Listrik AC adalah merupakan jenis arus yang tidak mengalir secara searah, melainkan secara bolak-balik. Arus AC memiliki nilai dan arah yang selalu berubah-ubah dan akan membentuk suatu gelombang yang bernama gelombang sinusoida. Pada arus listrik AC, dikenal yang namanya frekuensi. Yang mana besarnya frekuensi ini berbeda-beda di setiap negara.

Di Indonesia, arus listrik AC yang ditetapkan oleh PLN memiliki frekuensi sebesar 50 Hertz. Sedangkan tegangan standar untuk arus bolak-bali 1 fasa di Indonesia adalah 220 Volt. Contoh penggunaan dari arus listrik AC pun sangat banyak. Hampir semua alat-alat yang ada di rumah anda menggunakan arus listrik AC.

Kelebihan Arus Listrik AC (*Alternating Current*)

Arus Listrik AC biasanya dipergunakan untuk menyalurkan listrik menuju tempat yang jauh dikarenakan arus AC memiliki kerugian yang lebih kecil dibandingkan arus DC. Listrik disalurkan menggunakan voltage yang tinggi yang sudah di *step up* dari trafo sehingga menjadi pilihan yang tepat untuk menyalurkan listrik menuju ketempat yang jauh sehingga berbeda dengan arus DC. Arus AC sangat mudah untuk didapatkan hanya dengan menggunakan generator sedangkan untuk arus DC sulit.

Kekurangan Arus Listrik AC (*Alternating Current*)

Arus AC tidak dapat disimpan dalam waktu yang lama dan juga tidak dapat dipindahkan untuk keperluan yang tiba-tiba. Berbeda dengan arus DC yang bisa kita dapatkan atau kita pindahkan dalam bentuk aki dan baterai.

Daya Listrik

Daya listrik didefinisikan sebagai laju hantaran energi listrik dalam rangkaian listrik. Daya listrik dibagi menjadi tiga, yaitu: daya aktif, daya reaktif, dan daya semu. Daya dengan satuan joule/detik atau watt disebut sebagai daya aktif. Daya aktif adalah daya sebenarnya yang dipakai oleh beban. Daya nyata dapat dihitung dengan persamaan 2.1 berikut:

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

Keterangan:

P = Daya

V = Tegangan

I = Arus

Cos φ = Faktor Daya

Daya reaktif adalah daya yang dibutuhkan untuk pembentukan medan magnet atau daya yang ditimbulkan oleh beban yang bersifat induktif. Daya reaktif satuannya adalah VAR (*Voltampere Reactive*). Persamaan daya reaktif dapat dilihat pada persamaan 2.2 berikut ini:

$$P = V \times I \times \sin\varphi \quad (2.2)$$

Keterangan:

Q = Daya Reaktif

V = Tegangan

$I = \text{Arus}$

$\sin\phi = \text{Faktor Daya}$

Daya semu adalah hasil dari penjumlahan trigonometri daya aktif dan reaktif yang memiliki lambang S . Dengan satuannya adalah VA (Voltampere). Persamaan daya semu dapat dilihat pada persamaan 2.3 berikut:

$$S = V \times I \quad (2.3)$$

Keterangan:

$S = \text{Daya Semu}$

$V = \text{Tegangan}$

$I = \text{Arus}$

Faktor Daya

Faktor daya adalah perbandingan antara daya aktif dengan daya semu. Faktor daya atau faktor kerja menggambarkan sudut fasa antara daya aktif dan daya semu. Daya aktif digunakan untuk mengoperasikan beban-beban pada pelanggan listrik. Daya semu dihasilkan oleh generator pembangkit yang ditransmisikan ke pelanggan listrik. Daya reaktif yang bertambah akan menyebabkan turunnya faktor daya listrik. Cara yang mudah untuk mengantisipasi turunnya faktor daya listrik dapat dilakukan dengan memilih beban-beban yang mempunyai faktor daya besar juga dapat dilakukan dengan memasang kapasitor.

Suatu beban dengan faktor daya 1.0 merupakan beban yang hanya mengandung nilai resistansi murni dan merupakan pembebanan yang paling efisiensi. Beban dengan faktor daya yang rendah (0.5) merupakan beban yang mengandung nilai induktansi yang menyebabkan kerugian yang lebih tinggi dalam sistem suplai tenaga listrik. Faktor daya yang rendah berhubungan dengan

beda fasa antara arus dan tegangan pada terminal beban. Sudut fasa arus beban yang rendah biasanya diakibatkan oleh penggunaan beban induktif seperti transformator, motor induksi, lampu TL dan beban elektronik lainnya.

2.2 Monitoring

Monitoring menurut *Webster's New Collegiate Dictionary* (1981) adalah: “*a device for observing or giving admonition or warning*”. Sementara itu menurut *Webster's New World Dictionary*, maka pengertian “monitoring adalah *something that reminds or warns' or any of various devices for checking or regular the performance*”.

Menurut pengertian yang diberikan oleh kedua kamus internasional tersebut, dapat disimpulkan yang dimaksud dengan “monitoring“ yaitu kegiatan yang dilakukan untuk mengecek penampilan dari aktivitas yang sedang dikerjakan. Monitoring adalah bagian dari kegiatan pengawasan, dalam pengawasan ada aktivitas memantau (monitoring). Pemantauan umumnya dilakukan untuk tujuan tertentu, untuk memeriksa apakah program yang telah berjalan itu sesuai dengan sasaran atau sesuai dengan tujuan dari program. Secara umum Monitoring bertujuan mendapatkan umpan balik bagi kebutuhan program proses pembelajaran yang sedang berjalan, dengan mengetahui kebutuhan ini pelaksanaan program akan segera mempersiapkan kebutuhan dalam pembelajaran tersebut. Kebutuhan bisa berupa biaya, waktu, personel, dan alat. Pelaksanaan program akan mengetahui berapa biaya yang dibutuhkan, berapa lama waktu yang tersedia untuk kegiatan tersebut.

2.3 Menghitung Tagihan Listrik Pascabayar dan Prabayar

Pasca Bayar

Berikut adalah contoh cara perhitungan listrik pasca bayar besar daya dan barang-barang yang digunakan:

Tabel 2.1 Penggunaan listrik perhari dengan Daya listrik: 900 VA

Nama Beban	Jumlah	Waktu Pemakaian	kWh
Setrika 350 watt	1	2 jam/hari	0,70 kWh/hari
Pompa air 150 watt	1	3 jam/hari	0,45 kWh/hari
Kulkas 100 watt	1	6 jam/hari	0,60 kWh/hari
TV 110 watt	1	6 jam/hari	0,66 kWh/hari
<i>Rice cooker</i>	1	2 jam/hari	0,60 kWh/hari
Lampu 20 watt	6	6 jam/hari	0.72 kWh/hari
Lampu 10 watt	4	6 jam/hari	0,24 kWh/hari

Jumlah kebutuhan listrik perhari 3,91 kWh

Jumlah Kebutuhan listrik per bulan $3,91 \text{ kWh} \times 30 = 117,30 \text{ kWh}$ Golongan

Tarif R1 900 VA(Rumah Tangga) dengan pemakaian 117,30kWh:

Rumus Perhitungannya = Pemakaian x Tarif Dasar Listrik

1 Blok 1 (20 kWh pertama) = $20 \text{ kWh} \times \text{Rp } 275 = \text{Rp } 5500$

2 Blok 2 (40 kWh berikutnya) = $40 \text{ kWh} \times \text{Rp } 445 = \text{Rp } 17800$

3 Blok 3 (diatas 60 kWh) = $57,3 \text{ kWh} \times \text{Rp } 495 = \text{Rp } 28363,5$

4 Jumlah = $117,30 \text{ kWh} = \text{Rp } 51663.5$

$117,30 \text{ kWh} = \text{Rp } 51663.5$ (Ini belum biaya Abodemen dan PJU)

Berikut adalah cara perhitungan abodemen:

Rumus Perhitungan Abodemen PLN = (Daya / 1000) x (Rp/kVA)

Jadi : (900/1000) X Rp. 20000 0.9 X Rp. 20000 = Rp.18000

Total : Rp. 51663.5 + Rp.18000 = Rp. 69663.5

PAJAK PJU (3% s.d 10 %)

Rumus Perhitungan Pajak PJU = 3% x Total Tagihan Listrik Plus Abodemen :

3% x Rp 69663.5,- = Rp 2089.905 (dibulatkan Rp 2100,-) ADMIN BANK (Rp. 1600 s.d Rp. 5000)

Jadi Seluruhnya Rp. 51663.5 + Rp.18000 + Rp.2100 = RP. 71763.5

Tambah Admin Bank Rp. 1600 (ambil yang termurah)

TOTAL Rp. 71763.5 + Rp. 1600 = Rp. 73363.5

Tagihan Listrik Pasca Bayar tiap bulan kurang lebih Rp. 73363.5

Pra Bayar

Berikut adalah perhitungan prabayar dengan kondisi seperti contoh pasca bayar diatas:

{(NominalBELI – Adm Bank) – (NominalBELI-Adm Bank)x PPJ} : Biaya per Kwh

Misal voucher yang dibeli Rp. 75.000

{(75.000 - 1.600) – (75.000 – 1.600) x 3% } : 605

(73.400 - 2202) : 605 = 117.68264

Jika membeli Token Rp. 75.000 maka jumlah KWH yang didapat kurang lebih 117.68264. Ini cukup untuk 1 bulan dengan situasi sama seperti diatas, dan kalau diperhatikan antara Pasca bayar dan Pra Bayar hampir sama tidak jauh beda.

Biaya perbeban akan dihitung menggunakan persamaan 2.4 berikut ini:

$$Biaya = \text{nilai kWh} \times Rp. 1.352 \quad (2.4)$$

Keterangan:

Berikut daftar tarif listrik per kWh 2022 untuk golongan tarif listrik non-subsidi:

- Golongan R-1/TR daya 900 VA, Rp 1.352 per kWh.
- Golongan R-1/ TR daya 1.300 VA, Rp 1.444,70 per kWh.
- Golongan R-1/ TR daya 2.200 VA, Rp 1.444,70 per kWh.
- Golongan R-2/ TR daya 3.500-5.500 VA, Rp 1.699,53 per kWh.
- Golongan R-3/ TR daya 6.600 VA ke atas, Rp 1.699,53 per kWh.
- Golongan B-2/ TR daya 6.600 VA-200 kVA, Rp 1.444,70 per kWh.
- Golongan I-3/ TM daya di atas 200 kVA, Rp 1.114,74 per kWh.
- Golongan I-4/ Tegangan Tinggi (TT) daya 30.000 kVA ke atas, Rp 996,74 per kWh.

2.3 *Internet of Things*

Internet of Things adalah semua hal dan perangkat yang menggunakan jaringan *wireless* maupun jaringan internet pada sistemnya. Perangkat tersebut akan memiliki kemampuan untuk mengirimkan data dan transmisi melalui sebuah jaringan tanpa adanya campur tangan manusia. Berkat adanya *chip* komputer canggih dan juga banyaknya jaringan nirkabel saat ini, hampir semua perangkat bisa menjadi bagian dari *Internet of Things*. Ini berarti *Internet of Things* tidak terbatas pada industri teknologi informasi atau IT saja. Bahkan kita bisa menemukan banyak contoh IoT dalam sehari-hari. Intinya, IoT adalah bentuk komunikasi antar mesin dan jaringan.

Manfaat utama dari IoT adalah kemudahan proses konektivitas yang ditawarkannya. Melalui IoT, koneksi bisa dilakukan dengan lebih cepat dan juga praktis tanpa menggunakan kabel. Selain itu, IoT juga menggunakan teknologi cerdas yang memastikan koneksi jaringan dan perangkat berjalan dengan lancar. Tentunya ini membuat pekerjaan apapun menjadi mudah. Manfaat IoT yang lainnya adalah efisiensi kerja yang ditawarkannya. Berkat adanya konektivitas melalui teknologi pintar, IoT mampu menurunkan waktu yang dibutuhkan sebuah pekerjaan. Selain itu, IoT juga bisa digunakan untuk melakukan monitoring efektivitas pekerjaan.



Gambar 2.1 *Internet of Things*

Internet of Things sendiri terbentuk dari beberapa komponen dasar. Berikut adalah beberapa unsur dari IoT yaitu:

- *Artificial Intelligence* (AI), salah satu komponen dasar IoT adalah kecerdasan buatan yang berfungsi sebagai “otak” sebuah perangkat.
- Konektivitas, tanpa sebuah hubungan jaringan, IoT tidak akan bisa berfungsi dengan baik. Konektivitas adalah sebuah komponen utama IoT yang membantu perangkat untuk terhubung ke jaringan yang ditentukan.
- Sensor, sensor pada perangkat IoT bertujuan untuk mendefinisikan

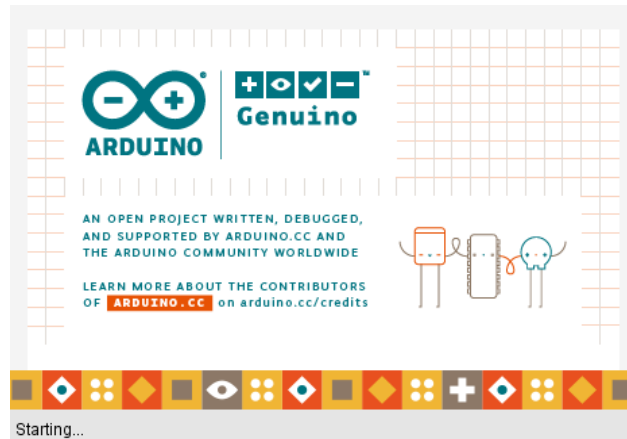
instrumen dan mengubah perangkat tersebut menjadi sebuah sistem aktif yang sanggup melakukan fungsinya dengan sesuai.

- *Active engagement*, banyak perangkat konvensional yang masih menggunakan metode engagement secara pasif. IoT menerapkan keterlibatan secara aktif yang aktif dalam berbagai produk, dan layanan yang tersedia.
- Perangkat yang kecil dan ringkas, di masa yang semakin modern ini, banyak perangkat berukuran kecil yang mampu melakukan banyak hal, smartphone contohnya.

2.4 Program Processing Arduino IDE

Arduino IDE (*Integrated Development Environment*) adalah *software* yang digunakan untuk memprogram di arduino, dengan kata lain Arduino IDE sebagai media untuk memprogram *board* Arduino. Arduino IDE bisa di *download* secara gratis di *website* resmi Arduino IDE.

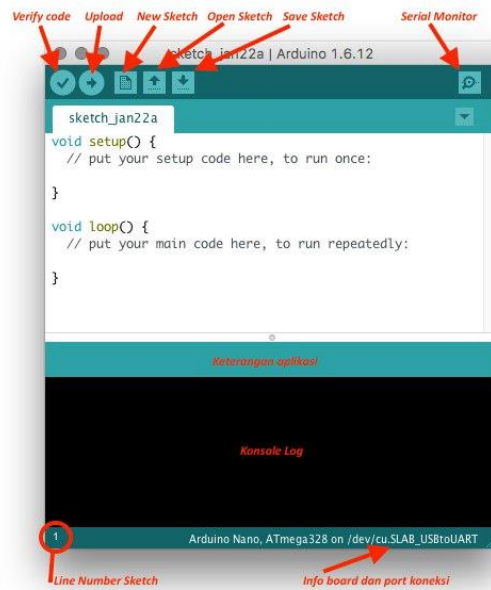
Arduino IDE ini berguna sebagai *text editor* untuk membuat, mengedit, dan juga mevalidasi kode program. bisa juga digunakan untuk mengupload ke *board* Arduino. Kode program yang digunakan pada Arduino disebut dengan istilah Arduino “*sketch*” atau disebut juga source code arduino, dengan ekstensi file source code .ino



Gambar 2.2 Program *Processing* Arduino IDE

Bagian-bagian Arduino IDE

Editor Programming pada umumnya memiliki fitur untuk *cut/paste* dan untuk *find/replace* teks, demikian juga pada Arduino IDE. Pada bagian keterangan aplikasi memberikan pesan balik saat menyimpan dan mengekspor serta sebagai tempat menampilkan kesalahan. Konsol log menampilkan teks log dari aktifitas Arduino IDE, termasuk pesan kesalahan yang lengkap dan informasi lainnya. Pojok kanan bawah menampilkan port serial yang di gunakan. Tombol *toolbar* terdapat ikon tombol pintas untuk memverifikasi dan meng-upload program, membuat, membuka, dan menyimpan *sketch*, dan membuka monitor serial.



Gambar 2.3 Bagian-bagian Arduino IDE

Berikut ini penjelasan mengenai bagian-bagian pada Arduino IDE,

- *Verify* pada versi sebelumnya dikenal dengan istilah *Compile*. Sebelum aplikasi di-*upload* ke *board* Arduino, biasakan untuk memverifikasi terlebih dahulu *sketch* yang dibuat. Jika ada kesalahan pada *sketch*, nanti akan muncul *error*. Proses *Verify/Compile* mengubah *sketch* ke *binary code* untuk diupload ke mikrokontroler.
- *Upload* tombol ini berfungsi untuk mengupload *sketch* ke *board* Arduino. Walaupun kita tidak mengklik tombol *verify*, maka *sketch* akan di-*compile*, kemudian langsung diupload ke *board*. Berbeda dengan tombol *verify* yang hanya berfungsi untuk memverifikasi *source code* saja.
- *New Sketch* Membuka *window* dan membuat *sketch* baru.
- *Open Sketch* Membuka *sketch* yang sudah pernah dibuat. *Sketch* yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi file *.ino*
- *Save Sketch* menyimpan *sketch*, tapi tidak disertai dengan mengkompile.

- *Serial Monitor* Membuka *interface* untuk komunikasi serial, nanti akan kita diskusikan lebih lanjut pada bagian selanjutnya.
- Keterangan Aplikasi pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal *Compiling* dan *Done Uploading* ketika kita mengcompile dan mengupload sketch ke *board* Arduino
- Konsol log Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang sketch akan muncul pada bagian ini. Misal, ketika aplikasi mengcompile atau ketika ada kesalahan pada *sketch* yang kita buat, maka informasi error dan baris akan diinformasikan di bagian ini.
- Baris *Sketch* bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada *sketch*.
- Informasi *Board* dan *Port* Bagian ini menginformasikan *port* yang dipakai oleh *board* Arduino.

2.5 ESP8266

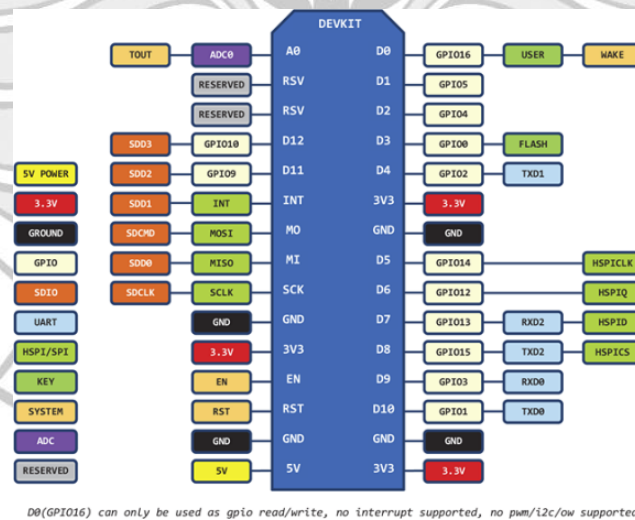
ESP8266 merupakan modul wifi yang berfungsi sebagai perangkat tambahan mikrokontroler seperti Arduino agar dapat terhubung langsung dengan *wifi* dan membuat koneksi TCP/IP. Modul ini membutuhkan daya sekitar 3.3v dengan memiliki tiga mode wifi yaitu *Station*, *Access Point*, dan *Both* (Keduanya). Modul ini juga dilengkapi dengan prosesor, memori dan GPIO dimana jumlah pin bergantung dengan jenis ESP8266 yang kita gunakan. Sehingga modul ini bisa berdiri sendiri tanpa menggunakan mikrokontroler apapun karena sudah memiliki perlengkapan layaknya mikrokontroler

Spesifikasi dari NodeMCU sebagai berikut :

1. 10 port pin GPIO
2. Fungsionalitas PWM
3. Antarmuka I2C dan SPI
4. Antarmuka 1 Wire
5. ADC



Gambar 2.4 ESP8266



Gambar 2.5 Pin-pin NodeMCU

Untuk tegangan kerja ESP8266 menggunakan standar tegangan JEDEC (tegangan 3.3V) untuk bisa berfungsi. Tidak seperti mikrokontroler AVR dan sebagian besar *board* Arduino yang memiliki tegangan TTL 5 volt. Meskipun

begitu, NodeMCU masih bisa terhubung dengan 5V namun melalui *port micro* USB atau pin Vin yang disediakan oleh board-nya. Namun karena semua pin pada ESP8266 tidak toleran terhadap masukan 5V.

2.6 Sensor PZEM004T

PZEM-004T adalah sebuah modul elektronik yang berfungsi untuk mengukur tegangan, arus, daya, *frekuensi*, *energy* dan *power factor*. Dengan kelengkapan fungsi ini, maka modul PZEM-004T sangat ideal untuk digunakan sebagai proyek maupun eksperimen alat pengukur daya pada sebuah jaringan listrik seperti rumah atau gedung. Modul PZEM-004T diproduksi oleh sebuah perusahaan bernama Peacefair, ada yang model 10 ampere dan 100 ampere.

Untuk spesifikasi PZEM-004T itu sendiri:

1. *Working voltage* : 80~260VAC.
2. *Rated power* : 100A/22000W.
3. *Working Frequency*: 45-65Hz.
4. *Measurement accuracy*: 1.0



Gambar 2.6 Sensor PZEM004T

Sensor PZEM-004T adalah sensor yang dapat mengukur arus, tegangan, power dan energy dari arus AC. Sensor ini mengeluarkan output dengan komunikasi serial. Jika dihubungkan dengan arduino maka komunikasi yang digunakan adalah komunikasi serial. Sensor PZEM-004T cukup mudah digunakan karena output nya langsung bisa dibaca, baik berupa arus, tegangan, power maupun energi. Untuk kekurangan sensor ini sendiri tidak mampu membaca arus AC dengan ketelitian mili Ampere. Modul PZEM-004T adalah sebuah modul sensor multifungsi yang berfungsi untuk mengukur daya, tegangan, arus, dan energi yang terdapat pada sebuah aliran listrik. Modul ini sudah dilengkapi dengan sensor tegangan dan sensor arus (CT) yang sudah terintegrasi.

Dalam penggunaannya, alat ini khusus penggunaan dalam ruangan (indoor) dan beban yang terpasang tidak diperbolehkan melebihi daya yang sudah ditetapkan. PZEM-004T adalah hardware berfungsi untuk mengukur parameter dari tegangan, arus, daya aktif, dan konsumsi daya (kWh). Pengkabelan dari modul ini memiliki 2 bagian, yaitu dari pengkabelan terminal masukan tegangan dan arus, serta pengkabelan komunikasi serial. Berdasarkan pada kebutuhan, modul ini memiliki papan pin TTL untuk mendukung komunikasi data serial antar hardware. Jika pengguna ingin mengkomunikasikan PZEM-004T ini dengan perangkat yang memiliki port USB atau RS-232 (seperti komputer), diperlukan lagi kabel converter (TTL ke USB, TTL ke RS232).

Tabel 2.2 Fungsi Pin Sensor PZEM-004T

No.	Pin	Fungsi
1	VCC	+5V
2	GND	Ground
3	TX	RX(<i>Software Serial/Hardware Serial</i>)
4	RX	TX(<i>Software Serial/Hardware Serial</i>)

Fungsi Sensor PZEM-004T

Modul PZEM-004T terutama digunakan untuk mengukur tegangan AC, arus, daya aktif, frekuensi, faktor daya, dan *energy* aktif. Modul tanpa fungsi tampilan, dan dibaca melalui interface TTL. Interface TTL dari modul ini adalah *interface* pasif membutuhkan catu daya eksternal 5V, yang berarti ketika berkomunikasi keempat *port* harus terhubung (5V, RX, TX, GND) jika tidak ia tidak dapat berkomunikasi. PZEM-004T 10A dengan rentang pengukuran 10A (*Built-in Shunt*). PZEM-004T 100A dengan rentang pengukuran 100A (*External Transformer*).

2.7 Relay

Relay merupakan komponen elektronika berupa saklar atau *switch* yang dioperasikan menggunakan listrik. *Relay* juga biasa disebut sebagai komponen elektromekanikal yang terdiri dari dua bagian utama yaitu *coil* atau elektromagnet dan saklar atau mekanikal. Komponen *relay* 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.



Gambar 2.7 Relay

Relay tersusun dari empat komponen dasar, yaitu elektromagnet (*coil*), armature, *switch contact point* (saklar), dan *spring*. Kontak poin terbagi lagi menjadi dua jenis, antara lain adalah:

- *Normally Close* (NC): Kondisi awal sebelum diaktifkan, akan selalu berada di posisi *close* (tertutup).
- *Normally Open* (NO): Kondisi awal sebelum diaktifkan akan selalu berada di posisi *open* (terbuka)

Untuk cara kerja *relay*, besi atau *iron core* dililit oleh kumparan *coil*. Ketika kumparan *coil* dialiri arus listrik, maka akan muncul gaya elektromagnetik yang bisa menarik tuas armature, sehingga posisi kontak *switch* berubah, dari NC (*normally closed*) menjadi NO (*normally open*).

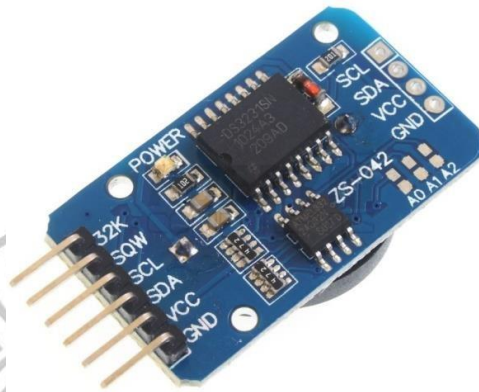
Tabel 2.3 Four-Channel Relay Module Pinout

Nomor Pin	Nama Pin	Deskripsi
1	GND	<i>Ground reference for the module</i>
2	IN1	<i>Input untuk mengaktifkan relay 1</i>
3	IN2	<i>Input untuk mengaktifkan relay 2</i>
4	IN3	<i>Input untuk mengaktifkan relay 3</i>
5	IN4	<i>Input untuk mengaktifkan relay 4</i>
6	VCC	<i>Power supply untuk modul relay</i>
7	VCC	<i>Power supply selection jumper</i>
8	JD-VCC	<i>Pin daya alternatif untuk modul relay</i>

2.8 RTC DS3231

RTC merupakan alat yang digunakan untuk mengakses data waktu dan kalender. RTC yang digunakan adalah DS3231 yang merupakan pengganti dari serial RTC tipe DS1307 dan DS1302. RTC mampu mengakses informasi data waktu mulai dari detik, menit, jam, hari, tanggal, bulan dan tahun. Akhir tanggal pada setiap bulan akan disesuaikan secara otomatis dengan kurang dari 31 hari dan juga mampu mengoreksi tahun kabisat. Pada DS3231 Operasi jam bisa diformat dalam 24 jam atau 12 jam (AM/ PM). Untuk tatap muka dengan suatu mikroprosesor dapat disederhanakan dengan menggunakan sinkronisasi komunikasi serial I2C dengan kecepatan *clock* 400Khz. Hanya membutuhkan 2 saluran untuk komunikasi dengan *clock*/RAM: SCL (*serial clock*), SDA (*Serial I/O data*), dan juga dilengkapi dengan keluaran SQW/*Out* yang dapat *deprogram*

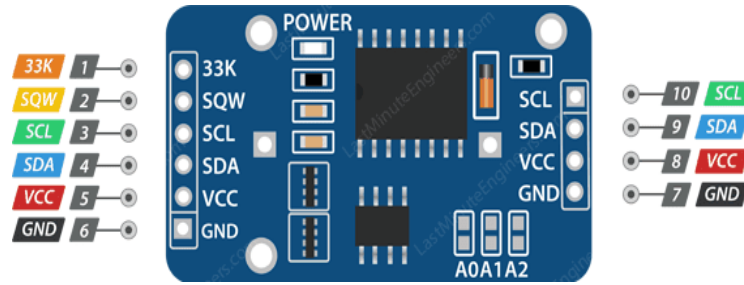
15 untuk mengetahui perubahan data waktu pada RTC dan pin RST. DS3231 didesain untuk mengoperasikan pada power yang sangat rendah dan mempertahankan data dan informasi waktu ± 1 microwatt.



Gambar 2.8 RTC DS3231

Adapun karakteristik dari RTC tipe DS3231 yaitu:

- RTC menghitung detik, menit, jam, tanggal, bulan, hari setiap minggu dan tahun dengan benar sampai tahun 2100
- Serial I2C untuk pin minimum proses komunikasi RTC
- 2.0 – 5.5 Volt *full operation*
- Mempunyai kemasan 16 pin SOICs
- 3 *simple wire interface* (I2C dan SQW/Out)
- *Square wave output* yang dapat diprogram
- Mempunyai sensor temperatur dengan akurasi $\pm 3^\circ$ Celcius



Gambar 2.9 Konfigurasi pin pada RTCDS3231

Tabel 2.4 Penjelasan Pin RTC DS3231

Pin	Fungsi
VCC1, VBAT	sebagai <i>power supply</i> . Jika $VCC2 > VCC1 (+0.2V)$ maka VCC2 menjadi power DS3231, begitu juga sebaliknya
SCL	untuk sinkronisasi data pada <i>serial interface (clock)</i>
SDA	pin data <i>bidireksional (input/output)</i>
INT/SQW _{OUT}	<i>Output</i> interupsi dari RTC yang dapat <i>deprogram</i> sebagai pemberi informasi perubahan waktu
32Khz	<i>Output</i> gelombang kotak yang dapat diprogram
RST	Pin <i>Reset</i> yang RTC
Clock/ kalender	memuat data dalam bentuk BCD dan memiliki 7 <i>register write/ read</i> .
AM-PM/12-24	7 bit <i>register</i> ditetapkan sebagai mode 12 atau 24 jam.
Write protect bit	pada 7 bit pertama (bit 0.....6) berlogika 0 sampai pada proses <i>read</i> , bit 7 harus berlogika 0 sebelum ada operasi penulisan untuk clock atau RAM
Clock/ calendar Burst Mode	bagian dari operasi burst mode, secara teratur dapat dibaca atau ditulis mulai dengan bit 0 pada alamat 0

2.9 LCD (*Liquid Crystal Display*)

LCD (*Liquid Crystal Display*) adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. Di pasaran tampilan LCD sudah tersedia dalam bentuk modul yaitu tampilan LCD beserta rangkaian pendukungnya. LCD mempunyai pin data, kontrol catu daya, dan pengatur kontras tampilan. LCD juga merupakan perangkat *display* yang paling umum dipasangkan di Mikrokontroler, Mengingat ukurannya yang kecil dan kemampuannya menampilkan karakter atau grafik yang lebih dibandingkan *display seven segment*.



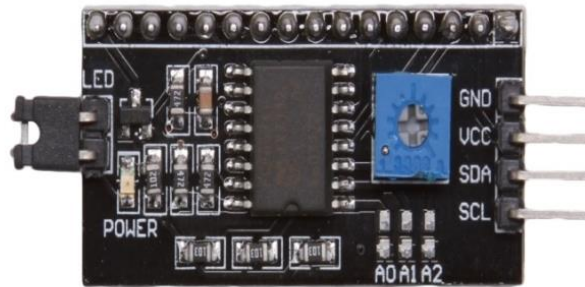
Gambar 2.10 LCD (*Liquid Crystal Display*)

Tabel 2.5 Spesifikasi LCD

No.	Nama	Spesifikasi
1	<i>Blue backlight</i>	12C
2	<i>Display Format</i>	16 Characters x 4 lines
3	<i>Supply voltage</i>	5V
4	<i>Back lit</i>	<i>Blue with White char color</i>
5	<i>Supply voltage</i>	5V
6	<i>Pcb Size</i>	60mm 99mm
7	<i>Contrast Adjust</i>	<i>Potentiometer</i>
8	<i>Backlight Adjust</i>	Jumper

I2C Module

I2C / TWI modul LCD2004 adalah sebuah sistem peraga menggunakan LCD dot matrix 16X2 karakter berbasis IC Hitachi HD44780 dengan I2C serial bus kecepatan tinggi yang diproduksi oleh DFRobot. Sistem peraga LCD dot matrix 16x2 karakter berbasis IC HD44780 dapat dihubungkan ke *board* Arduino Uno hanya menggunakan 2 (dua) buah kaki Analog A4 dan A5 selain sumber tegangan DC +5 Volt. Kaki Analog A4 dan A5 dari Arduino Uno dihubungkan ke kaki SDA dan kaki SCL dari serial board. Diperlukan sebuah file *library LiquidCrystal_I2C.h* agar sebuah *board* Arduino Uno dapat digunakan untuk menggerakkan LCD dot matrix 16x2 karakter berbasis IC Hitachi HD44780 dengan I2C serial bus.



Gambar 2.11 Modul I2C

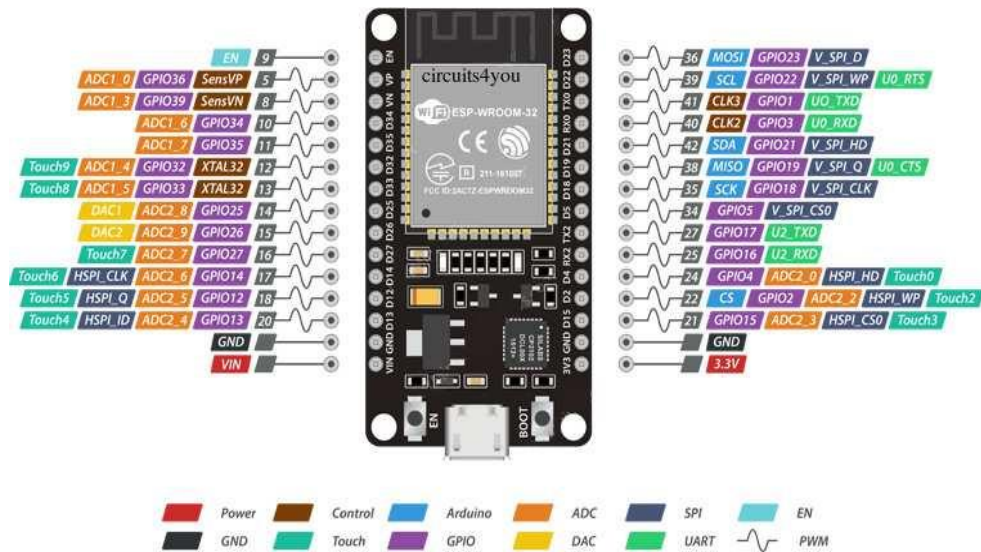
Tabel 2.6 Spesifikasi Module I2C

No.	Nama	Spesifikasi
1	Tegangan kerja	VCC, GND, DO, AO
2		Mendukung protokol I2C, <i>coding</i> lebih singkat
3		Dilengkapi Trimpot pengatur lampu dan kontras layar
4		Hanya 4 pin utk pengendalian (SDA, SCL, VCC dan GND)
5	<i>Device Address</i>	0x27 atau 0x3F
6		Dapat digunakan untuk LCD 16x2 ataupun 20x4 7
7	Ukuran	41.5x19x15.3mm

I2C Merupakan komunikasi serial dua arah menggunakan dua saluran yang didisain khusus untuk mengirim maupun menerima data. Sistem I2C terdiri dari saluran SCL (Serial Clock) dan SDA (Serial Data) yang membawa informasi data antara I2C dengan pengontrolnya.

2.10 ESP32

Gambar 2.12 menunjukkan mikrokontroler ESP32 sebagai penerus dari mikrokontroler ESP8266. Pada mikrokontroler ini sudah tersedia modul *wifi* dan *bluetooth* sehingga sangat mendukung untuk membuat sistem aplikasi *Internet of Things*. Memiliki 18 ADC (*Analog Digital Converter*), 2 DAC, 16 PWM, 10 Sensor sentuh, 2 jalur antarmuka UART, pin antarmuka I2C, I2S, dan SPI



ESP32 Dev. Board Pinout

Gambar 2.12 ESP32

Spesifikasi ESP32 Devkit V1

- *Wireless connectivity*
 - WiFi: 150.0 Mbps *data rate* dengan HT40
 - *Bluetooth*: BLE (*Bluetooth Low Energy*) dan *legacy Bluetooth*.
- *Processor*: Tensilica Xtensa *Dual-Core* 32-bit LX6 *microprocessor*, pada 160 / 240MHz
- ROM: 448KB
- SRAM: 520KB
- *Low Power*: jaminan masih bisa menjalankan *ADC conversion*, contohnya, saat *deep sleep*.
- *Peripheral Input/Output*:
 - *Peripheral interface with DMA that includes capacitive touch*
 - *ADC (Analog-to-Digital Converter)*
 - *DAC (Digital-to-Analog Converter)*

- I²C (*Inter-Integrated Circuit*)
- UART (*Universal Asynchronous Receiver/Transmitter*)
- SPI (*Serial Peripheral Interface*)
- I²S (*Integrated Interchip Sound*)
- RMII (*Reduced Media-Independent Interface*)
- PWM (*Pulse-Width Modulation*).
- *Security: hardware accelerators for AES and SSL/TLS*

2.11 Mengukur persentase kesalahan

Persen kesalahan tujuan dari perhitungan persentase kesalahan adalah untuk mengukur seberapa dekat nilai yang diukur dengan nilai sebenarnya. Persen kesalahan (persentase kesalahan) adalah selisih antara nilai eksperimental dan nilai teoritis, dibagi dengan nilai teoritis, dikalikan dengan 100 menghasilkan persen. Menghitung tingkat kesalahan dapat dilakukan perhitungan dengan persamaan 2.5 dan 2.6 sebagai berikut:

$$\% \text{ Kesalahan} = \left| \frac{V_{\text{out Perhitungan}} - V_{\text{out Pengukuran}}}{V_{\text{out Perhitungan}}} \right| \times 100\% \quad (2.5)$$

$$\% \text{ Kesalahan Rata - rata} = \left| \frac{\sum \% \text{ Kesalahan}}{n} \right| \quad (2.6)$$

Dimana n adalah banyaknya pengujian yang dilakukan.

Suatu alat ukur dikatakan tepat jika mempunyai akurasi yang baik yaitu hasil ukur menunjukkan ketidakpastian yang kecil. Dapat juga dipahami sebagai seberapa dekat hasil ukur dengan nilai benarnya. Dalam hal ini sebelum sebuah alat ukur digunakan, harus dipastikan bahwa kondisi alat benar-benar dalam keadaan baik dan layak untuk digunakan, yaitu alat dalam keadaan terkalibrasi dengan baik. Kalibrasi yang buruk akan menyebabkan ketidakpastian hasil ukur

menjadi besar. Klasifikasi Kelas Meter Untuk mendapatkan hasil pengukuran yang mendekati dengan harga sebenarnya. Perlu memperhatikan batas kesalahan yang tertera pada alat ukur tersebut. Klasifikasi alat ukur listrik menurut Standar IEC no. 13B-23 menspesifikasikan bahwa ketelitian alat ukur dibagi menjadi 8 kelas, yaitu : 0,05; 0,1 ; 0,2 ; 0,5 ; 1,0 ; 1,5 ; 2,5 ; dan 5. Kelas-kelas tersebut artinya bahwa besarnya kesalahan dari alat ukur pada batas-batas ukur masing-masing kali $\pm 0,05 \%$, $\pm 0,1 \%$, $\pm 0,2 \%$, $\pm 0,5 \%$, $\pm 1,0 \%$, $\pm 1,5 \%$, $\pm 2,5 \%$, $\pm 5 \%$ dari relatif harga maksimum.

00:43

Dari 8 kelas alat ukur tersebut digolongkan menjadi 4 golongan sesuai dengan daerah pemakaiannya, yaitu :

1. Golongan dari kelas 0,05, 0,1, 0,2 termasuk alat ukur presisi yang tertinggi. Biasa digunakan di laboratorium yang standar.
2. Golongan alat ukur dari kelas 0,5 mempunyai ketelitian dan presisi tingkat berikutnya dari kelas 0,2 alat ukur ini biasa digunakan untuk pengukuran-pengukuran presisi. Alat ukur ini biasanya portabel.
3. Golongan dari kelas 1,0 mempunyai ketelitian dan presisi pada tingkat lebih rendah dari alat ukur kelas 0,5. Alat ini biasa digunakan pada alat ukur portabel yang kecil atau alat-alat ukur pada panel.
4. Golongan dari kelas 1,5, 2,5, dan 5 alat ukur ini dipergunakan pada panel-panel yang tidak begitu memperhatikan presisi dan ketelitian.

BAB III

METODE KEGIATAN

3.1 Tempat dan Waktu Perancangan

Pelaksanaan kegiatan dari proses perancangan simulasi, hingga fabrikasi dilaksanakan di kampus 1 Politeknik Negeri Ujung Pandang. Dengan uraian perancangan dan fabrikasi dilaksanakan di Bengkel Perancangan dan proses simulasi dilaksanakan di Laboratorium Mikrokontroler. Keseluruhan kegiatan dari Bulan Januari hingga Bulan Agustus 2022.

3.2 Alat dan Bahan Perancangan

Dalam pelaksanaan perancangan, perlengkapan yang digunakan terdiri dari perangkat lunak dan perangkat keras.

a. Perangkat lunak (*software*):

Perangkat lunak yang digunakan dalam perancangan sistem monitoring dan pengendalian daya listrik ini terdiri dari:

Tabel 3.1 Bahan yang digunakan dalam perangkat lunak

NO.	Nama Software	Jumlah
1	Blynk	1
2	Blynk cloud	1
3	Arduino ide	1

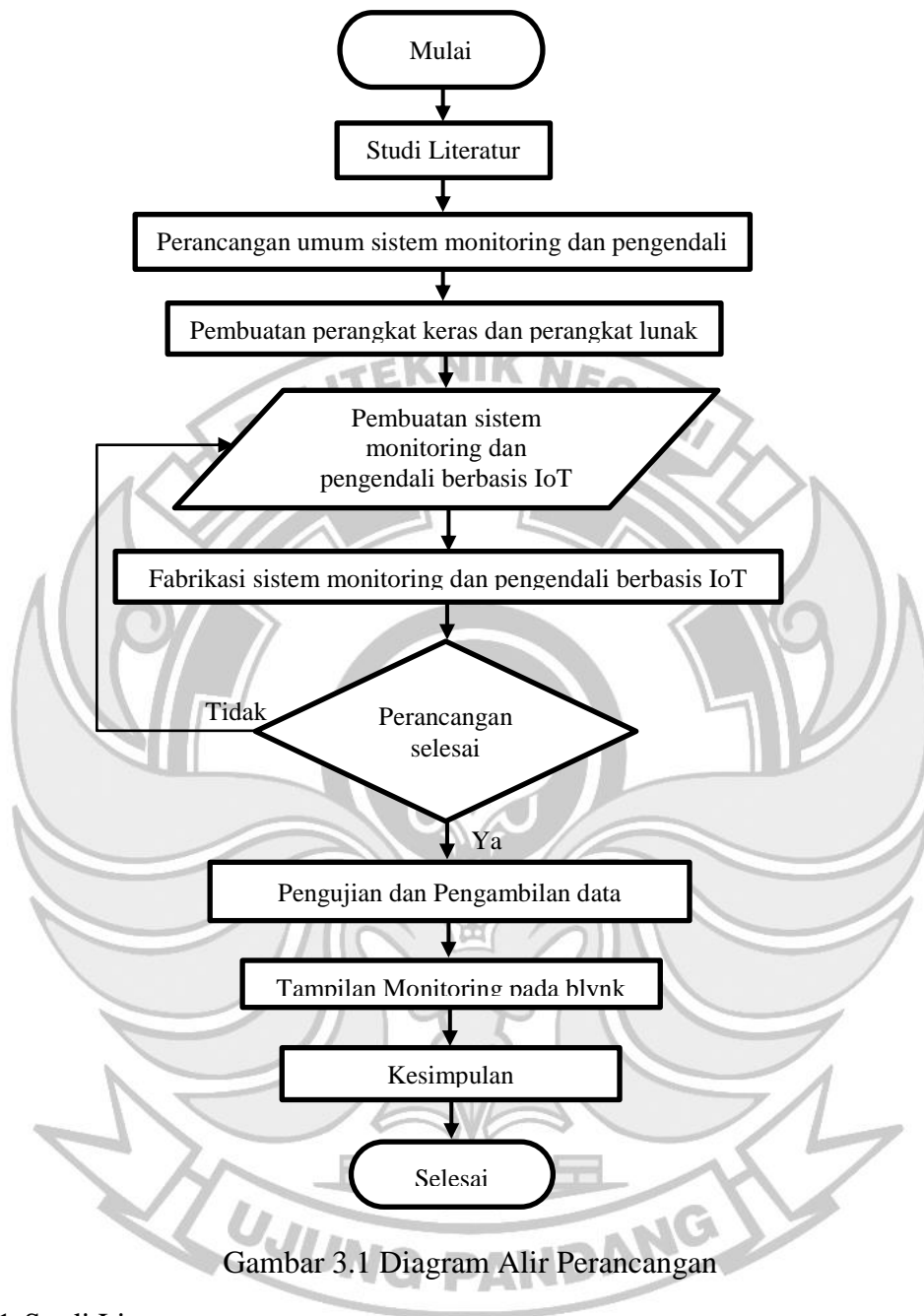
b. Perangkat keras (*hardware*)

Perangkat keras yang digunakan dalam perancangan sistem monitoring dan pengendalian daya listrik ini sebagai berikut:

Tabel 3.2 Bahan yang digunakan dalam perangkat keras

No	Nama Bahan	Jumlah
1	ESP8266	1 buah
2	ESP32	1 buah
3	Sensor PZEM-004T	1 buah
4	Relay	1 buah
5	RTC DS3231	1 buah
6	LCD	1 buah
7	Obeng	1 buah
8	Tang potong	1 buah
9	Tang jepit	1 buah
10	Stripper	1 buah
11	Solder	1 buah
12	Multimeter	1 buah
13	Smartphone	1 buah

3.3 Prosedur Perancangan

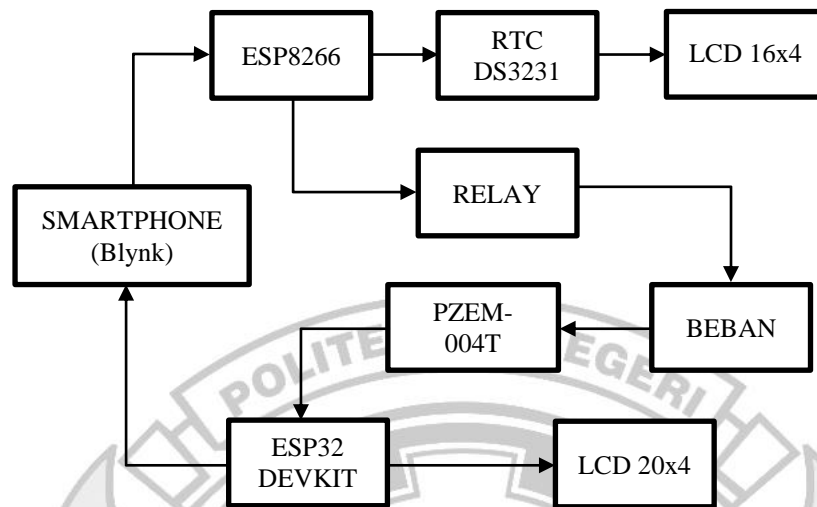


Gambar 3.1 Diagram Alir Perancangan

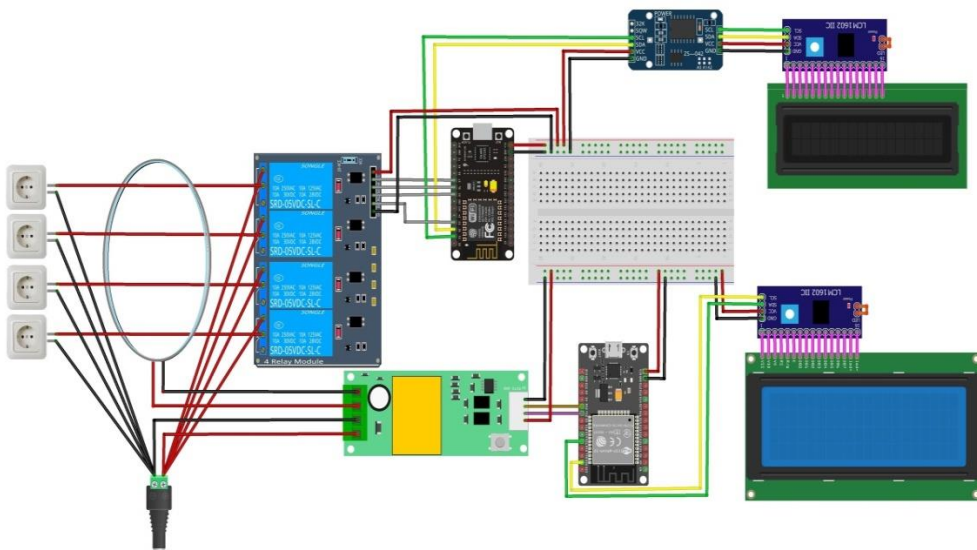
3.3.1 Studi Literatur

Dalam perancangan alat sistem monitoring konsumsi dan pengendalian daya listrik ini, langkah pertama yang harus dilakukan adalah mencari sebanyak-banyaknya data serta informasi yang relevan mengenai alat yang akan dibuat. Referensi juga diperlukan dalam penyusunan laporan yaitu: sensor PZEM-004T, ESP8266, Relay, LCD dan software Blynk.

3.3.2 Perancangan Sistem *Hardware*



Gambar 3.2 Blok Diagram Sistem



Gambar 3.3 Rangkaian keseluruhan sistem

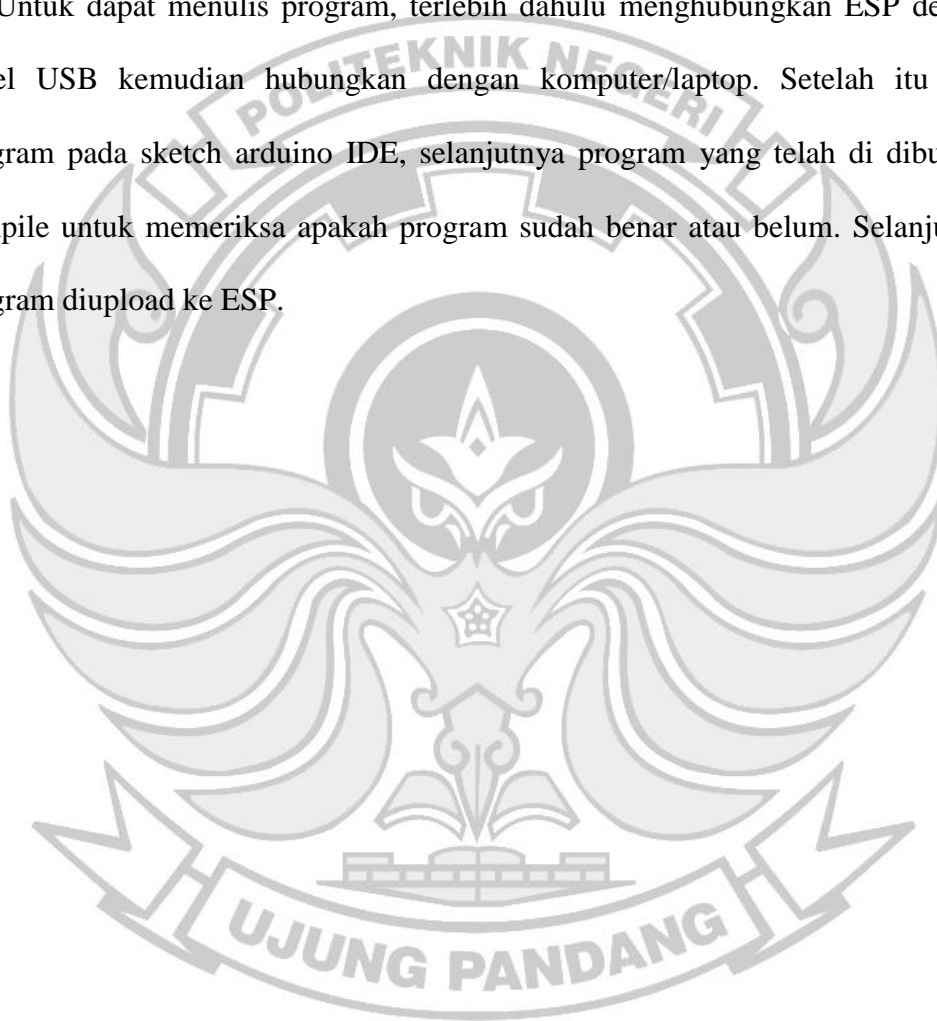
Adapun prinsip kerja pada rancangan sistem hardware diatas, smartphone bekerja untuk mengendalikan dan memonitoring tegangan, arus, daya serta energi yang dikonsumsi oleh peralatan elektronik. Untuk melakukan pekerjaan tersebut, smartphone terlebih dahulu harus terhubung dengan modul ESP8266. Fungsi dari ESP8266 ini yaitu sebagai media transmisi yang akan mengirim data atau

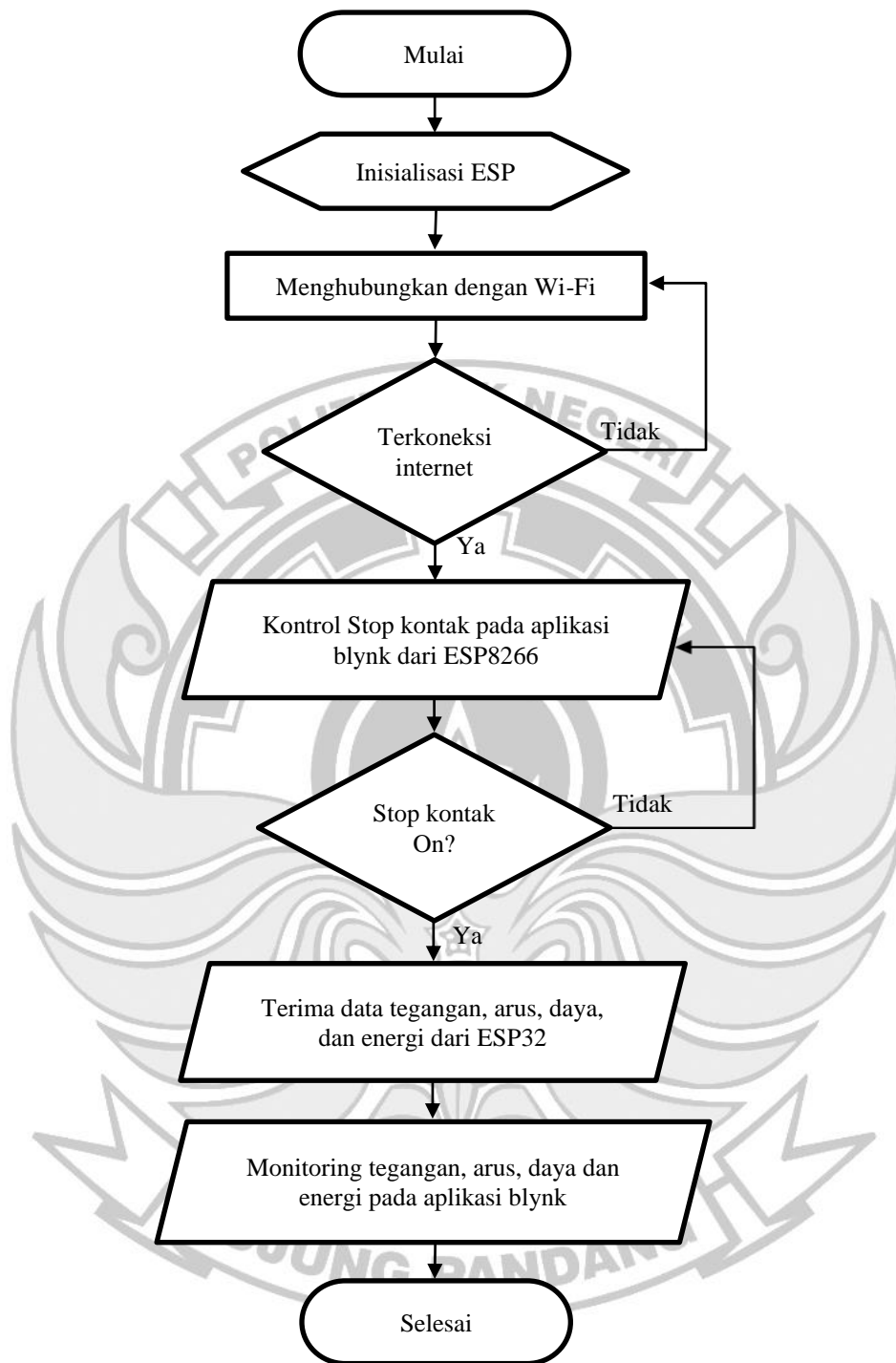
instruksi antara smartphone dengan relay yang telah terhubung dengan ESP8266.

Dalam mengontrol 4 beban, smartphone akan mengirimkan instruksi melalui aplikasi blynk yang akan diterima oleh ESP8266 ke modul relay yang berfungsi sebagai saklar yang dapat mengontrol keempat beban tersebut.

3.3.3 Perancangan sistem *software*

Untuk dapat menulis program, terlebih dahulu menghubungkan ESP dengan kabel USB kemudian hubungkan dengan komputer/laptop. Setelah itu buat program pada sketch arduino IDE, selanjutnya program yang telah di buat di compile untuk memeriksa apakah program sudah benar atau belum. Selanjutnya program diupload ke ESP.





Gambar 3.4 *Flowchart software*

Pada gambar 3.4 menunjukkan *flowchart software* proses keseluruhan sistem yang akan dibuat. Dimulai dari inisialisai/pengenalan ESP yang berfungsi sebagai pengendali utama yang terkoneksi dengan internet. Selanjutnya, hubungkan ESP dengan Wi-Fi, apabila ESP belum terkoneksi dengan Wi-Fi, ulangi proses sebelumnya, yakni menghubungkan ESP dengan Wi-Fi. Apabila ESP telah terkoneksi dengan Wi-Fi, kontrol stop kontak melalui aplikasi pada smartphone dan cek status stop kontak, apabila stop kontak dalam keadaan OFF, ulangi proses kontrol stop kontak. Jika status stop kontak sudah dalam keadaan ON maka data tegangan, kuat arus, daya dan energi dari sensor PZEM-004T akan dikirimkan dan dibaca oleh ESP32 DEVKIT V1, selanjutnya data tersebut akan ditampilkan pada LCD sistem rancangan yang dibuat serta data tersebut juga akan dikirimkan ke aplikasi Blynk yang ada pada *smartphone* melalui jaringan internet. Terakhir, data tegangan, arus, daya dan energi yang dikirim oleh ESP32 DEVKIT V1 dapat dimonitor pada aplikasi Blynk yang ada pada *smartphone*.

3.4 Pengambilan Data

Untuk melakukan pengukuran diperlukan adanya data terhadap sebuah sistem, untuk itu penulis melakukan pengambilan data dari beberapa beban dalam kurun waktu 1 jam untuk setiap bebannya. Adapun beban-beban tersebut, yaitu:

1. *Charger Laptop*
2. Kipas Angin
3. *Charger Smartphone*
4. Setrika

Sensor PZEM-004T menjadi media untuk pengambilan data berupa nilai tegangan, arus, daya, energi, frekuensi, dan faktor daya



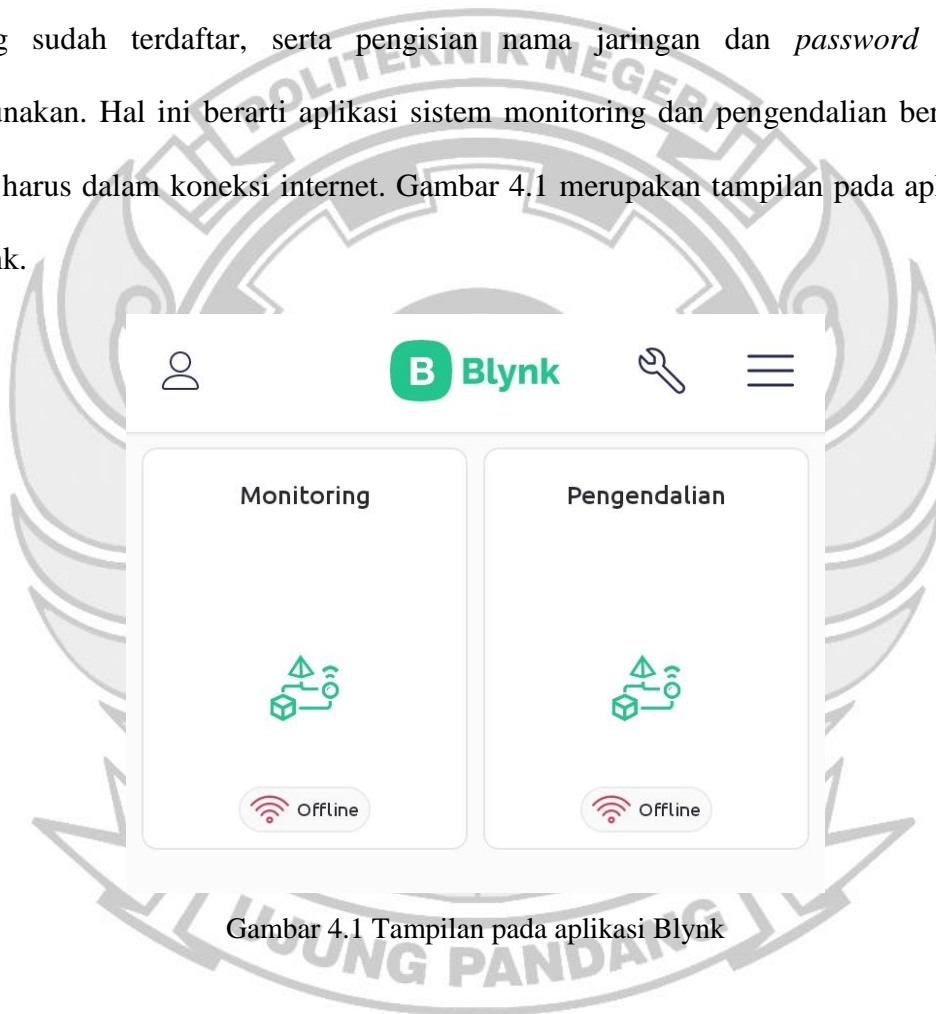
Gambar 3.5 Proses pengukuran beban

Pengujian pada sensor tegangan dan arus dilakukan untuk mengetahui sensor PZEM-004T telah bekerja dengan baik dan hasil dari pembacaan PZEM-004T dengan pengukuran. Selanjutnya, melakukan pengujian RTC DS3231 yang terlihat pada LCD 16x4 pengujian ini bertujuan untuk mengetahui bahwa RTC DS3231 dapat bekerja dengan baik yakni dapat menampilkan menit, jam, hari, tanggal, bulan dan tahun sesuai dengan program yang telah ditentukan sebelumnya. Kemudian, melakukan pengujian kontrol ON/OFF stop kontak melalui aplikasi Blynk pada smartphone. Terakhir, pengujian tingkat kesalahan sistem saat bekerja terhadap pengukuran.

BAB IV HASIL DAN PEMBAHASAN

4.1 Data Hasil Penelitian

Pengujian pertama yang dilakukan adalah koneksi ESP32. Pengujian ini dilakukan untuk mengetahui konektifitas antara ESP32 dengan aplikasi blynk. Dalam program terdapat sebuah nomor token yang didapatkan dari akun email yang sudah terdaftar, serta pengisian nama jaringan dan *password* yang digunakan. Hal ini berarti aplikasi sistem monitoring dan pengendalian berbasis IoT harus dalam koneksi internet. Gambar 4.1 merupakan tampilan pada aplikasi blynk.



Gambar 4.1 Tampilan pada aplikasi Blynk



Gambar 4.2 Tampilan LCD pada sistem rancangan

Untuk menguji apakah alat rancangan yang dibuat dapat bekerja dengan baik kami memasang alat monitoring daya listrik dengan beberapa jenis beban yang berbeda yaitu, Setrika, kipas angin, charger laptop dan charger handphone dalam satu waktu. Pada tampilan LCD menunjukkan 7 variable yang ditunjukkan yaitu biaya, tegangan, arus, daya, energi, frekuensi, serta faktor daya yang digunakan oleh perangkat elektronik.

4.2.1 Pengujian (Pengamatan) untuk Setiap Beban (Peralatan Elektronik)

(Skema 1)

Semua data yang didapatkan dengan cara memasang beban-beban ke alat monitoring dalam kurun waktu 1 jam dan setiap 10 menit akan di catat hasil pembacaannya.

Adapun hasil pembacaan alat monitoring daya listrik sebagai berikut :

Beban A (Charger Laptop Lenovo IdeaPad S340)

Tegangan : 100-240 V

Arus : ~1.7 A

Frekuensi : 50-60 Hz

Tabel 4.1 Pengujian pada Beban A (Charger Laptop Lenovo IdeaPad S340)

Waktu (menit)	Hasil Pengamatan (Tampilan LCD)								
	Tegangan (Volt)	Arus (A)	Daya (Watt)	Nilai kWh	Frekuensi (Hz)	PF (%)	Biaya (Rupiah)	Δ kWh	Δ Biaya (Rupiah)
	V	A	P	E	F				
0	218,9	0,567	68,5	1,130	49,9	55	1527,76	0	0
5	217,8	0,563	68,5	1,136	50	56	1535,87	0,006	8,11
10	217,7	0,566	68,4	1,142	49,9	56	1543,98	0,006	8,11
20	217,3	0,565	68,4	1,153	50	56	1558,86	0,011	14,88
30	218,1	0,558	68,2	1,164	50	56	1573,73	0,011	14,87
40	217,2	0,483	59,2	1,176	50	56	1589,95	0,012	16,22
50	219,7	0,315	37	1,183	49,9	53	1599,42	0,007	9,47
60	219,9	0,213	23,9	1,188	49,9	51	1606,18	0,005	6,67

Beban B (Charger Smartphone Asus Max Pro M2)

Tegangan : 100-240 V

Arus : 0,3 A

Frekuensi : 50-60 Hz

Tabel 4.2 Pengujian pada Beban B (*Charger Smartphone Asus Max Pro M2*)

Waktu (menit)	Hasil Pengamatan (Tampilan LCD)								
	Tegangan (Volt)	Arus (A)	Daya (Watt)	Nilai kWh	Frekuensi (Hz)	PF (%)	Biaya (Rupiah)	Δ kWh	Δ Biaya (Rupiah)
	V	A	P	E	F				
0	219,9	0,098	11,7	1,189	50	54	1607,53	0	0
5	219,7	0,072	8,3	1,189	50	52	1607,53	0	0
10	217,7	0,057	6,3	1,190	49,9	51	1608,88	0,001	1,35
20	218,7	0,048	5,3	1,191	50	50	1610,23	0,001	1,35
30	219	0,053	6	1,192	50	52	1611,58	0,001	1,35
40	218,9	0,052	6	1,193	50	53	1612,94	0,001	1,36
50	222,2	0,063	7,5	1,195	49,9	54	1615,64	0,002	2,7
60	226,2	0,057	6,7	1,196	50	52	1616,99	0,001	1,35

Beban C (Kipas Angin Mitochiba)

Tegangan : 220 V

Arus : ~0,227 A (50 Watt)

Frekuensi : 50 Hz

Tabel 4.3 Pengujian pada Beban C (Kipas Angin Mitochiba)

Waktu (menit)	Hasil Pengamatan (Tampilan LCD)								
	Tegangan (Volt)	Arus (A)	Daya (Watt)	Nilai kWh	Frekuensi (Hz)	PF (%)	Biaya (Rupiah)	Δ kWh	Δ Biaya (Rupiah)
	V	A	P	E	F				
0	223	0,162	36	1,198	49,8	100	1619,7	0	0
5	224,5	0,157	35,1	1,201	49,9	100	1623,75	0,003	4,05
10	223,9	0,154	34,5	1,205	49,9	99	1629,16	0,004	5,41
20	222,5	0,149	32,8	1,210	49,8	99	1635,92	0,005	6,76
30	222,8	0,147	32,4	1,216	50	99	1644,03	0,006	8,11
40	222	0,145	31,9	1,221	50	99	1650,79	0,005	6,76
50	222,2	0,145	31,9	1,226	49,8	99	1657,55	0,005	6,76
60	221,4	0,144	31,6	1,232	50	99	1665,66	0,006	8,11

Beban D (Setrika)

Tegangan : 220 V

Arus : ~1,59 A (350 Watt)

Frekuensi : 50 Hz

Tabel 4.4 Pengujian pada Beban D (Setrika)

Waktu (menit)	Hasil Pengamatan (Tampilan LCD)								
	Tegangan (Volt)	Arus (A)	Daya (Watt)	Nilai kWh	Frekuensi (Hz)	PF (%)	Biaya (Rupiah)	Δ kWh	Δ Biaya (Rupiah)
	V	A	P	E	F				
0	219	1,53	334,9	1,234	50,1	100	1668,57	0	0
5	221,5	1,547	342,7	1,250	50	100	1690	0,016	21,43
10	219,5	1,533	336,5	1,257	49,9	100	1699,46	0,007	9,46
20	219,3	1,532	336	1,271	50	100	1718,39	0,014	18,93
30	219,6	1,533	336,6	1,284	50	100	1735,97	0,013	17,58
40	217,4	1,518	329,9	1,297	49,9	100	1753,54	0,013	17,57
50	217,9	1,522	331,6	1,318	49,9	100	1781,94	0,021	28,4
60	218,5	1,527	333,5	1,346	49,9	100	1819,79	0,028	37,85

4.2.2 Pengujian (Pengamatan) Beban (Peralatan Elektronik) Secara Bersamaan (Skema 2)

Keterangan: Beban A, B, C dan D dihubungkan secara Bersamaan pada perangkat

Tabel 4.5 Pengujian beban A, B, C, dan D dihubungkan secara bersama-sama

Waktu (menit)	Hasil Pengamatan (Tampilan LCD)								
	Tegangan (Volt)	Arus (A)	Daya (Watt)	Energi (kWh)	Frekuensi (Hz)	PF (%)	Biaya (Rupiah)	Δ kWh	Δ Biaya (Rupiah)
	V	A	P	E	F				
0	216,8	2,065	433,8	1,398	49,8	97	1890,10	0	0
5	218,3	2,075	439,9	1,413	49,9	97	1910,38	0,015	20,28
10	220,6	0,682	106,4	1,429	49,9	71	1932,01	0,016	21,63
20	221,9	0,409	70,4	1,459	49,9	78	1972,57	0,03	40,56
30	222,3	0,458	76,7	1,486	49,9	75	2009,07	0,027	36,5
40	223,3	0,526	87,3	1,515	50,1	74	2048,28	0,029	39,21
50	219,4	1,861	404,5	1,542	50	99	2084,78	0,027	36,5
60	221	1,828	401,5	1,567	49,9	99	2118,58	0,025	33,8

Tabel 4.6 Perbandingan hasil pengamatan nilai Kwh pada skema 1 dan skema 2

Durasi Waktu Pengamatan (menit)	Hasil Pengamatan Nilai Kwh						
	Skema 1 (A+B+C+D)					Skema 2	Biaya (Rupiah)
	A	B	C	D	Biaya (Rupiah)		
0	0	0	0	0	0	0	0
5	0,006	0	0,003	0,016	33,8	0,015	20,28
10	0,006	0,001	0,004	0,007	24,33	0,016	21,63
20	0,011	0,001	0,005	0,014	41,91	0,03	40,56
30	0,011	0,001	0,006	0,013	41,91	0,027	36,5
40	0,012	0,001	0,005	0,013	41,91	0,029	39,21
50	0,007	0,002	0,005	0,021	47,32	0,027	36,5
60	0,005	0,001	0,006	0,028	54,08	0,025	33,8

Pada tabel 4.6 dapat dilihat selisih nilai kWh dan biaya ketika beban A, B, C, dan D ditambahkan dengan ketika beban A, B, C, dan D dipasang secara bersamaan, hal ini dapat disebabkan oleh beberapa faktor yaitu, pada *charger* laptop nilai dayanya berubah karena laptop sudah dalam keadaan *fully charge*. Selanjutnya pada setrika, nilai ukur dayanya berubah setiap waktu dikarenakan setrika mempertahankan panas pada tingkat tertentu, maka setiap selang waktu tertentu akan terjadi perubahan yang konstan.

Tabel 4.7 Perbandingan besar biaya antara hasil perhitungan (berdasarkan rumus) dan perangkat

Durasi Waktu Pengamatan (menit)	Berdasarkan Perhitungan (Rumus)		Berdasarkan Tampilan Pada Perangkat		% Error
	Nilai kWh	Biaya (Rupiah) (kWh x Rp 1352,-)	Nilai kWh	Biaya (Rupiah)	
0	1,398	1890,09	1,398	1890,10	0.000529
5	1,413	1910,37	1,413	1910,38	0.000523
10	1,429	1932	1,429	1932,01	0.000517
20	1,459	1972,56	1,459	1972,57	0.000506
30	1,486	2009,072	1,486	2009,07	0,000099
40	1,515	2048,280	1,515	2048,28	0
50	1,542	2084,784	1,542	2084,78	0,000191
60	1,567	2118,584	1,567	2118,58	0,000188

Untuk menghitung biaya digunakan persamaan 2.4

$$Biaya = \text{nilai kWh} \times 1352$$

$$Biaya = 1.398 \times 1352$$

$$Biaya = 1890.10$$

$$\% \text{ Kesalahan} = \left| \frac{V_{out} \text{ Perhitungan} - V_{out} \text{ Pengukuran}}{V_{out} \text{ Perhitungan}} \right| \times 100\%$$

$$\% \text{ Kesalahan} = \left| \frac{1890.10 - 1890.09}{1890.10} \right| \times 100\%$$

$$\% \text{ Kesalahan} = 0.000529$$

$$\% \text{ Kesalahan Rata - rata} = \left| \frac{\sum \% \text{ Kesalahan}}{n} \right|$$

$$\% \text{ Kesalahan Rata - rata} = 0.0003\%$$

Pada hasil pengukuran tabel 4.7, maka dapat dilihat besar perbandingan biaya antara hasil perhitungan (berdasarkan rumus) dengan tampilan pada perangkat. Berdasarkan persamaan 2.4, 2.5 dan 2.6, menunjukkan bahwa sensor perangkat dapat bekerja dengan baik dan memiliki nilai *error* rata-rata sebesar 0,0003. Nilai *error* sebesar 0,0003%, maka dapat dinyatakan bahwa termasuk dalam kategori golongan 1 yang artinya memiliki presisi yang tinggi. Naik dan turunnya tegangan sumber PLN juga mempengaruhi kestabilan alat ini dalam menghitung biaya konsumsi energi listrik. Batas digit karakter yang ditampilkan pada LCD juga menjadi faktor berbedanya nilai penulisan pada hasil pembacaan sensor.

4.2 Pengujian Sensor PZEM-004T untuk pembacaan tegangan dan arus

Pengujian sensor PZEM-004T ini dilakukan untuk mengetahui keakuratan sensor dalam menghitung tegangan arus, dan daya yang mengalir pada setiap beban listrik dan membandingkannya dengan Multimeter yang merupakan pengukur standar untuk arus listrik. Pada pengujian ini, perangkat listrik yang digunakan adalah kipas angin, setrika, *Charge Laptop* dan *Charge Handphone*. Dalam pengujian sensor terdapat nilai *error* yang berfungsi sebagai informasi tingkat keakuratan sensor.

Tabel 4.8 hasil pengujian tegangan dan arus pada perangkat

Jenis Beban	Alat ukur rancangan		Multimeter		Error (%)	
	V	A	V	A	V	A
Kipas Angin	223.9	0.1	225.4	0.209	0.006	0.52
Charge Laptop	224.9	0.245	227.4	0.138	0.01	0.77
Charge Handphone	224.9	0.056	230.3	0.131	0.02	0.57
Setrika	220.8	1.672	221.5	1.687	0.003	0.008
Error Rata-rata					0.23	

$$\% \text{ Kesalahan} = \left| \frac{223.9 - 225.4}{223.9} \right| \times 100\%$$

$$\% \text{ Kesalahan} = 0.006$$

$$\% \text{ Kesalahan Rata - rata} = \left| \frac{\sum \% \text{ Kesalahan}}{n} \right|$$

$$\% \text{ Kesalahan Rata - rata} = 0.23\%$$

Berdasarkan hasil pengukuran pada Tabel 4.8, maka dapat dilihat perbandingan pembacaan sensor PZEM-004T dengan Multimeter. Berdasarkan persamaan 2.5 dan 2.6, menunjukkan bahwa sensor PZEM-004T bekerja dengan baik dan memiliki nilai *error* rata-rata sebesar 0.23%. Nilai *error* sebesar 0.23%, maka dapat dinyatakan bahwa termasuk dalam kategori golongan 1 yang artinya memiliki presisi yang tinggi. Adapun persentase kesalahan ini dapat diakibatkan oleh resolusi pembacaan antara sensor PZEM-004T dengan alat ukur multimeter yang berbeda, dapat juga disebabkan oleh ketidakstabilan saat proses pengukuran sehingga terdapat selisih pembacaan yang masih dalam tahap wajar.

Tabel 4.9 Perbandingan spesifikasi beban dengan hasil ukur perangkat

Nama Beban	Spesifikasi		Alat Ukur Rancangan		%Error	
	(Volt)	(A)	(Volt)	(A)	(Volt)	(mA)
Charger Laptop	240	1,7	218.9	0,567	0,096	1,998
Charger Smartphone	240	0,3	219,9	0,098	0,091	2,061
Kipas Angin	220	0,227	223	0,162	0,013	0,401
Setrika	220	1,59	219	1,53	0,004	0,039

$$\% \text{ Kesalahan} = \left| \frac{240 - 218.9}{0.096} \right| \times 100\%$$

$$\% \text{ Kesalahan} = 0.000529$$

$$\% \text{ Kesalahan Rata - rata} = \left| \frac{\sum \% \text{ Kesalahan}}{n} \right|$$

$$\% \text{ Kesalahan Rata - rata} = 0.39\%$$

Pada tabel 4.9 dapat dilihat perbandingan antara spesifikasi beban dengan hasil pengukuran dari alat ukur rancangan. Berdasarkan persamaan 2.5 dan 2.6 diperoleh nilai error rata-rata yaitu sebesar 0,39% dimana 0,39% ini termasuk dalam kategori golongan kedua yaitu memiliki ketelitian dan presisi tingkat berikutnya, hal ini terjadi karena naik dan turunnya tegangan sumber PLN yang mempengaruhi kestabilan alat, hal ini akan menimbulkan error pada sensor tegangan, arus, dan frekuensi dalam membaca data nilai tegangan, arus dan frekuensi pada beban.

4.3 Pengujian Pembacaan *Timer/RTC*

Real Time Clock (RTC) merupakan komponen yang diperlukan untuk memberikan informasi mengenai waktu. Waktu yang dimaksud disini dapat berupa detik, menit, jam, hari, tanggal, bulan, dan tahun.

Pada saat pengujian RTC DS3231 terlihat pada LCD bahwa RTC DS3231 dapat bekerja dengan baik yakni dapat menampilkan menit, jam, hari, tanggal, bulan dan tahun sesuai dengan program yang telah ditentukan sebelumnya. Dapat dilihat pada gambar dibawah ini.

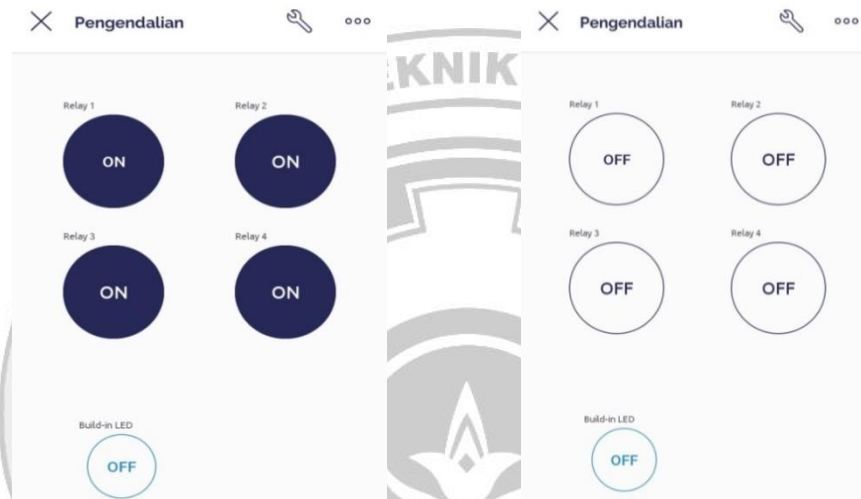


Gambar 4.3 Tampilan waktu, hari, tanggal, bulan dan tahun pada LCD

RTC DS3231 memiliki *battery* ketika *supply* terputus maka RTC dapat terus berjalan dikarenakan pada rangkaian RTC DS3231 memiliki *battery* cadangan 3V. Dari pengujian yang dilakukan ini dapat disimpulkan bahwa sistem pewaktuan bisa berfungsi dengan baik, selain itu dengan adanya *battery* cadangan sebesar 3V memberikan pewaktuan jadi lebih efektif karena apabila terjadi listrik padam, waktu dan tanggal akan terus berfungsi dengan baik.

4.4 Pengujian Relay/Stopkontak

Pada pengujian ini akan diuji kontrol ON/OFF stop kontak melalui aplikasi Blynk pada *smartphone*. Pengujian ini dilakukan dengan jarak yang berbeda beda untuk mengetahui apakah alat yang dirancang dapat dikontrol dari jarak jauh.



Gambar 4.4 Tampilan Pengendalian Stop kontak pada aplikasi Blynk

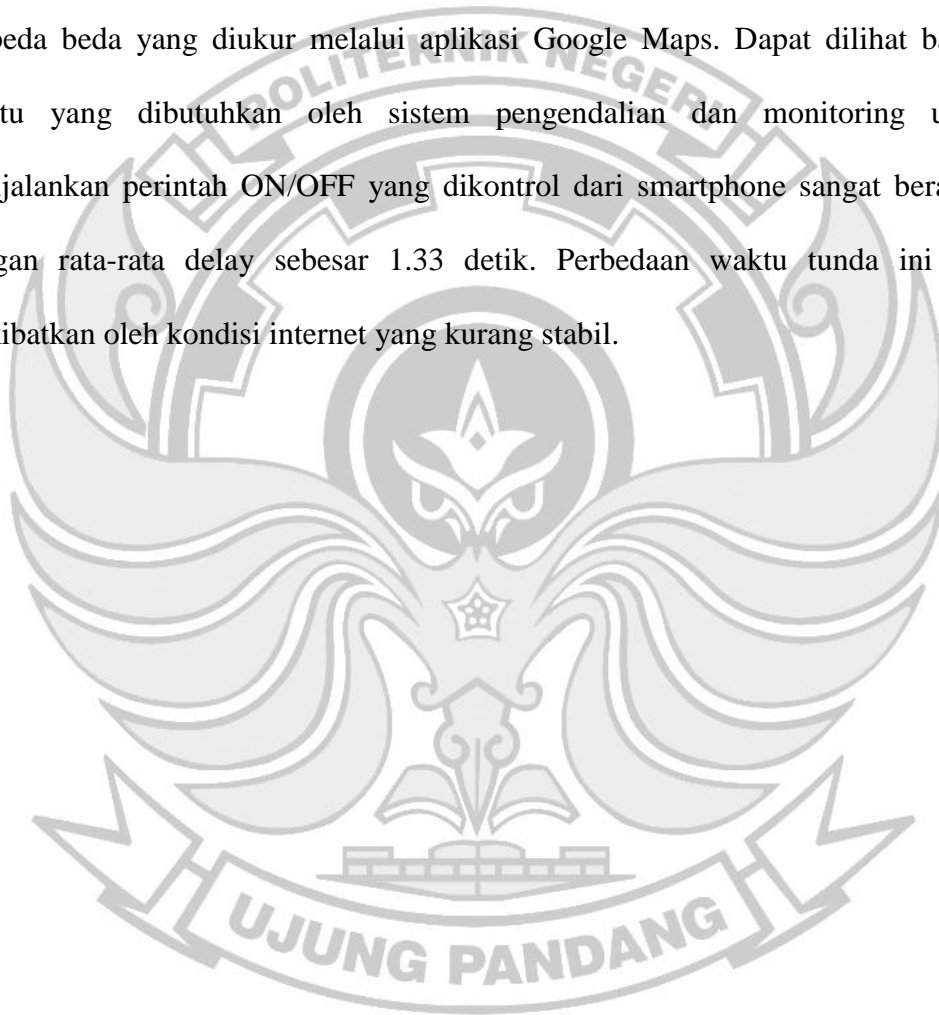
Tabel 4.10 Pengujian Data Pengontrolan Secara Jarak Jauh

No.	Kontrol	Jarak	Delay			Keterangan
			1	2	3	
1	ON/OFF	5.43 Km	1.82 detik	1.05 detik	1.52 detik	Berhasil
2	ON/OFF	13 Km	1.01 detik	1.20 detik	0.87 detik	Berhasil
3	ON/OFF	21 Km	0.98 detik	1.90 detik	1.41 detik	Berhasil
4	ON/OFF	185 Km	1.07 detik	1.41 detik	1.8 detik	Berhasil
Rata-rata			1.33 detik			

$$\% \text{ Kesalahan Rata - rata} = \left| \frac{\sum \% \text{ Kesalahan}}{n} \right|$$


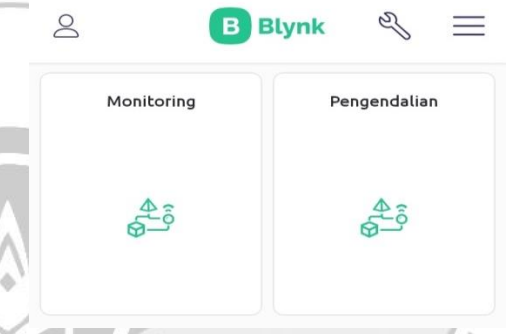
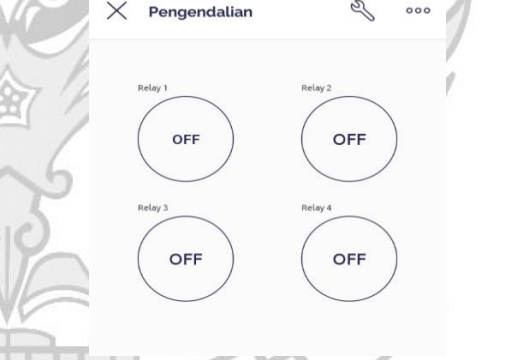

$$\% \text{ Kesalahan Rata - rata} = 1.33$$

Pada tabel diatas diperlihatkan data hasil percobaan pengontrolan jarak jauh alat melalui aplikasi Blynk. Dimana jarak antar pengontrol dan alat ukur berbeda beda yang diukur melalui aplikasi Google Maps. Dapat dilihat bahwa waktu yang dibutuhkan oleh sistem pengendalian dan monitoring untuk menjalankan perintah ON/OFF yang dikontrol dari smartphone sangat beragam dengan rata-rata delay sebesar 1.33 detik. Perbedaan waktu tunda ini bisa diakibatkan oleh kondisi internet yang kurang stabil.



4.5 Tampilan *Interface* pada aplikasi Blynk

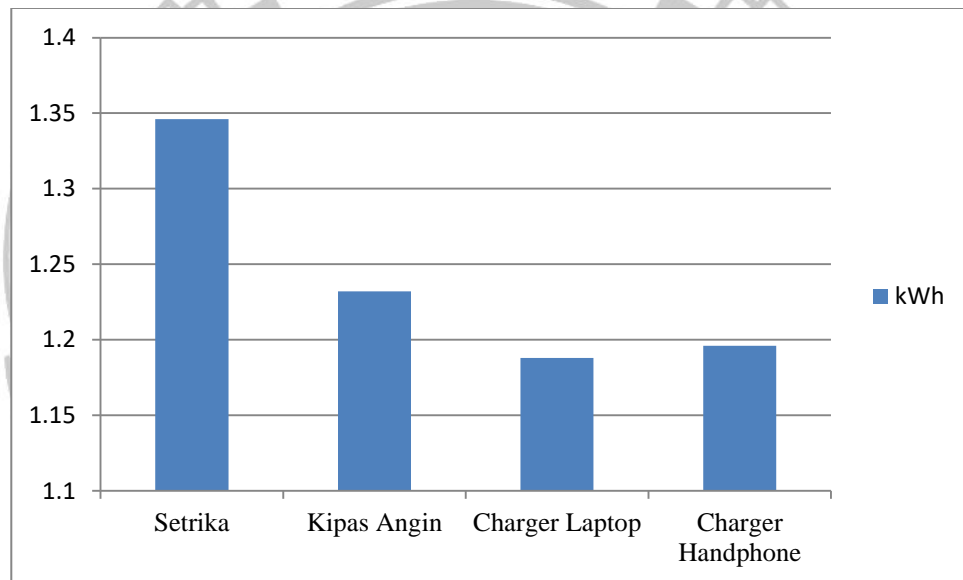
Tabel 4.11 Tampilan interface pada aplikasi Blynk

No.	Kondisi	Tampilan pada Aplikasi
1.	Tampilan awal pada aplikasi blynk dalam kondisi <i>offline</i> , disini terdapat dua device yaitu monitoring dan pengendalian.	
2.	Tampilan pada aplikasi blynk dalam kondisi <i>online</i>	
2.	Tampilan <i>device</i> pengendalian pada aplikasi blynk, disini terdapat 5 tombol, 4 tombol untuk mengendalikan relay dan satu tombol untuk men-ON-OFF lampu pada ESP	
3.	Tampilan <i>device</i> monitoring pada aplikasi blynk, disini dapat dilihat terdapat 7 variabel yaitu tegangan, kuat arus, daya, <i>energy</i> , frekuensi, faktor daya, serta biaya yang digunakan pada peralatan elektronik.	

4.6 Pembahasan

4.2.1 Monitoring kWh

Alat monitoring daya listrik ini mampu menghitung nilai kWh pada setiap bebannya. Tujuan dari monitoring kWh ini adalah para konsumen dapat menentukan estimasi biaya dari pemakaian beban-beban yang terpasang pada sumber listrik PLN. Adapun nilai kWh pada beban-beban yang berbeda bisa dilihat pada gambar 4.5.

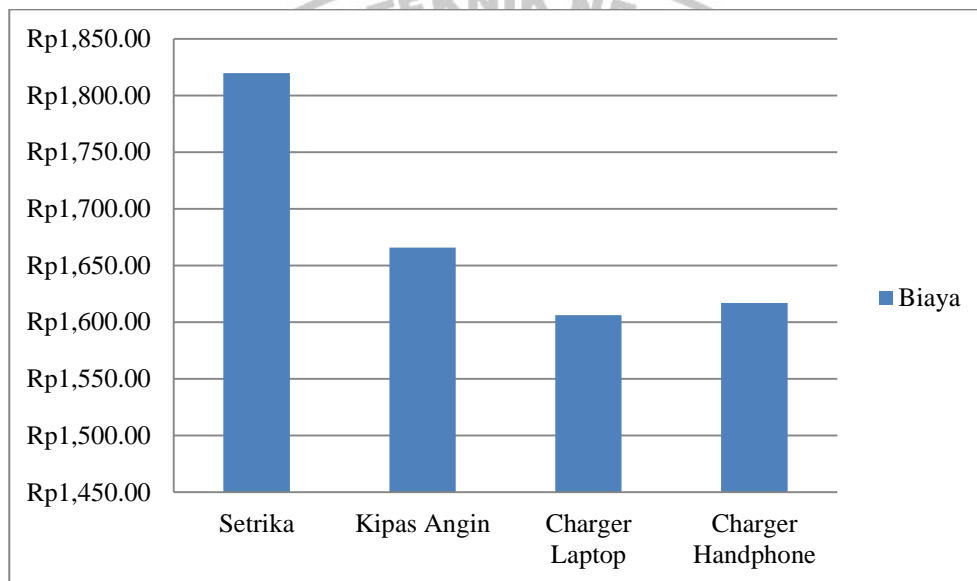


Gambar 4.5 Nilai kWh pada Beban

Pada gambar 4.5 dapat dilihat nilai kWh yang termonitoring dari beban yang terpasang pada perangkat dalam waktu 1 jam. Dari hasil monitoring selama 1 jam dapat dilihat bahwa beban yang memiliki nilai kWh terbesar adalah setrika yaitu 1,346 kWh dan beban yang memiliki nilai kWh terkecil adalah charger laptop yaitu sebesar 1.188 kWh.

4.2.2 Monitoring Biaya

Alat monitoring daya listrik ini menampilkan biaya pemakaian dari beban-beban yang terhubung pada sumber listrik PLN. Perhitungan tarif biaya pemakaian listrik dilakukan dengan cara nilai kWh pada beban dikalikan dengan tarif biaya konsumsi listrik yang telah ditetapkan pemerintah per-kWh nya. Adapun biaya pemakaian beban dapat dilihat pada gambar 4.6.



Gambar 4.6 Biaya Pemakaian Beban

Pada gambar 4.6 menunjukkan biaya pemakaian energi listrik terbesar adalah setrika dan biaya terendah adalah *charger* laptop. Pemakaian beban yang menyerap energi listrik paling besar akan menghasilkan nilai kWh yang besar juga, dan menyebabkan biaya konsumsi akan mengikuti besar dan kecilnya nilai kWh. Oleh sebab itu, penghematan energi harus dilakukan dengan meminimalisir pemakaian energi listrik dengan cara menggunakan energi listrik seperlunya saja dan mematikan peralatan listrik yang tidak digunakan.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan tujuan dan hasil pengujian serta data yang telah diperoleh, maka dapat disimpulkan bahwa:

1. Hasil rancangan sistem monitoring dan pengendalian daya listrik berbasis IoT dapat bekerja sesuai dengan yang diharapkan dengan delay rata-rata sebesar 1.33 detik.
2. Nilai-nilai yang ditampilkan pada sistem monitoring merupakan besarnya tegangan, arus, daya, dan energi serta biaya yang digunakan pada peralatan elektronik dengan nilai error rata-rata sebesar 0.23% dan 0.39%.

5.2 Saran

Penelitian ini tentunya memiliki keterbatasan yang memerlukan perbaikan dan pengembangan dalam penelitian selanjutnya. Keterbatasan dalam penelitian ini diharapkan dapat diperbaiki oleh peneliti selanjutnya.

Sistem monitoring dan pengendalian ini baru diimplementasikan pada skala kecil terbatas monitoring dan pengendalian 4 unit alat elektronik, pada penelitian selanjutnya diharapkan dapat diimplementasikan dalam skala yang lebih besar misalnya keseluruhan alat elektronik dalam rumah tangga atau mungkin bisa lebih besar lagi.

DAFTAR PUSTAKA

- Agus Ardiansyah. 2020. “Monitoring Daya Listrik Berbasis IoT (*Internet of Things*)”
- Aidil Azhari. 2020. Perancangan Sistem Proteksi Dan Peringatan Dini Terhadap Arus Beban Lebih Pada Ruko Berbasis Sms Dan Mikrokontroler, (Online) Fakultas Teknik Universitas Muhammadiyah Sumatera Utara Medan.
- Elga Aris Prasetyo. 2019. “Arsitektur dan Fitur ESP32 (Module ESP32) IoT”, (Online) (<https://www.edukasiaelektronika.com/2019/07/arsitektur-dan-fitur-esp32-module-esp32.html>) Diakses pada 10 Juni 2022.
- Inovator Guru. 2021. Modul Komunikasi AC PZEM-004T V3.0 Panduan Pengguna, (Online) (<https://manuals.plus/id/inovatorsguru/modul-komunikasi-ac-pzem-004t-v3-0-manual#axzz7bcM4hYiN>) Diakses pada 15 Juni 2022.
- Kusumandarut. 2015. Daya Listrik (Daya Aktif, Daya Reaktif, dan Daya Semu), (Online), (<https://kusumandarutp.blogspot.com/2015/08/daya-listrik-daya-aktif-daya-reaktif.html>), Diakses pada 8 Januari 2022.
- Kompas. 2022. “Daftar Harga Listrik Per kWh 2022 untuk Golongan Tarif Non-subsidi”, (Online) (<https://money.kompas.com/read/2022/07/03/130130526/daftar-harga-listrik-per-kwh-2022-untuk-golongan-tarif-non-subsidi?page=all>) Diakses 11 September 2022
- Nusa, T., Sompie, S. R., & Rumbayan, M. 2015. Sistem monitoring konsumsi energi listrik secara real time berbasis mikrokontroler,(Online) Jurnal Teknik Elektro dan Komputer, 4(5), 19-26.

- Robith Muhammad Adani. 2020. Mengenal Apa Itu Internet of Things dan Contoh Penerapannya, (Online) (<https://www.sekawanmedia.co.id/blog/pengertian-internet-of-things/>), Diakses pada 11 Januari 2022.
- Robby Yuli Endra, Ahmad Cucus, Freddy Nur Affandi, dan M. Bintang Syahputra. 2019. “Model Smart Room dengan menggunakan Mikrokontroler Arduino untuk Efisiensi Sumber Daya”
- Rosmauli Situngkir. 2021. “Rancang Bangun Pengendalian Jarak Jauh Perangkat Listrik serta Monitoring Penggunaan Daya Listrik Berbasis IoT”
- Taufiqullah. 2019 “Akurasi dan Kalibrasi Alat Ukur” (Online). (<https://www.tneutron.net/elektro/akurasi-dan-kalibrasi-alat-ukur/>) Diakses Pada 12 September 2022.
- Vera Febriani. 2013. “Monitoring, Pengertian dan Tujuan Menurut Para Ahli”, (Online) (<http://veyranazyha1207.blogspot.com/2013/03/monitoring-pengertian-dan-tujuan.html>) Diakses pada 29 Agustus 2022.
- Widiyaman Tresna. 2016. Pengertian Modul Wifi ESP8266, (Online) (<https://www.warriornux.com/pengertian-modul-wifi-esp8266/>). Diakses pada 13 Januari 2022.

LAMPIRAN



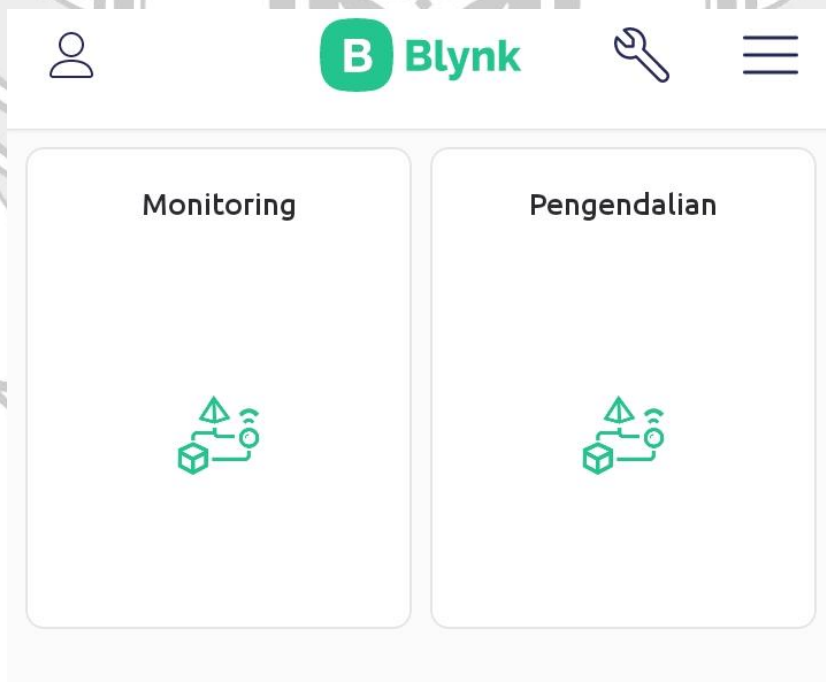
Gambar 1. Alat Rancangan



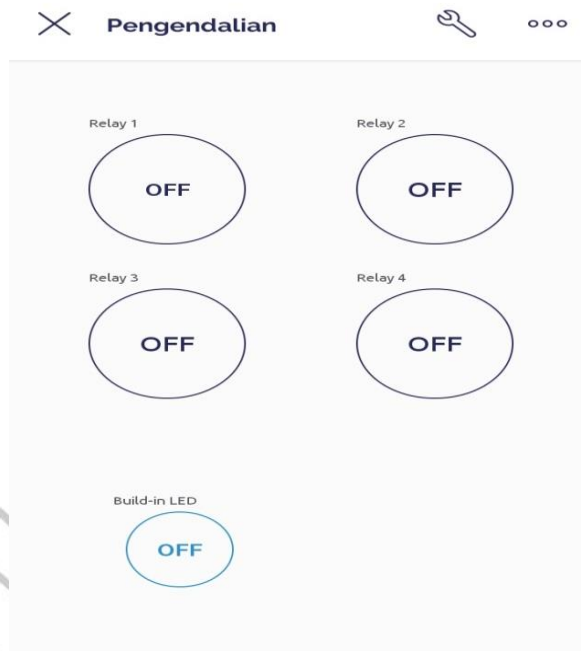
Gambar 2. Tampilan LCD pada Alat Rancangan



Gambar 3. Tampilan pada aplikasi blynk pada saat offline



Gambar 4. Tampilan pada aplikasis blynk pasa saat kondisi online



Gambar 5. Tampilan pada aplikasi blynk pada device pengendalian



Gambar 6. Tampilan pada aplikasi blynk pada device monitoring



Gambar 7. Pengujian pada Perangkat



Gambar 8. Tampilan Pada LCD perangkat dan pada Aplikasi blynk

Data pengukuran pada hari yang berbeda dan keadaan berbeda

Hasil Selasa 11 Oktober 2022 pukul 16.00 (Sore)

Waktu (menit)	Hasil Pengamatan (Tampilan LCD)								
	Tegangan (Volt)	Arus (A)	Daya (Watt)	Nilai kWh	Frekuensi (Hz)	PF (%)	Biaya (Rupiah)	Δ kWh	Δ Biaya (Rupiah)
	V	A	P	E	F				
0	224.90	1.906	426.2	1.610	50	99	2176.72	0	0
5	227.40	0.281	55.9	1.626	50	87	2198.35	0.016	21.63
10	225.00	1.856	416.9	1.636	49.90	100	2211.87	0.01	13.52
20	228.70	0.292	57.3	1.660	50	86	2244.32	0.024	32.45
30	228.80	0.273	54.9	1.683	50	86	2275.42	0.023	31.1
40	226.40	1.864	421.2	1.705	50	100	2306.15	0.022	30.73
50	226.80	1.832	415	1.728	50	100	2336.26	0.023	30.11
60	229.70	0.212	45.1	1.749	49.90	93	2364.65	0.021	28.39

Hasil Selasa 11 Oktober 2022 pukul 20.00 (Malam)

Waktu (menit)	Hasil Pengamatan (Tampilan LCD)								
	Tegangan (Volt)	Arus (A)	Daya (Watt)	Nilai kWh	Frekuensi (Hz)	PF (%)	Biaya (Rupiah)	Δ kWh	Δ Biaya (Rupiah)
	V	A	P	E	F				
0	226.50	1.877	423.8	1.750	49.90	100	2366	0	0
5	229.20	0.258	51.6	1.768	49.90	87	2390.34	0.018	24.34
10	229.00	0.235	48.4	1.78	50	90	2406.56	0.012	16.22
20	229.00	0.215	45.6	1.801	50	93	2434.95	0.021	28.39
30	229.60	0.237	48.8	1.822	50	90	2463.34	0.021	28.39
40	227.00	0.230	47.2	1.842	50.10	90	2490.38	0.02	27.04
50	224.50	1.801	404.2	1.864	50	100	2520.13	0.022	29.75
60	226.50	0.253	51.4	1.884	49.90	92	2547.17	0.02	27.04

Hasil Rabu 12 Oktober 2022 pukul 07.00 (Pagi)

Waktu (menit)	Hasil Pengamatan (Tampilan LCD)								
	Tegangan (Volt)	Arus (A)	Daya (Watt)	Nilai kWh	Frekuensi (Hz)	PF (%)	Biaya (Rupiah)	Δ kWh	Δ Biaya (Rupiah)
	V	A	P	E	F				
0	224.30	1.900	424.4	1.888	50	100	2552.58	0	0
5	223.60	1.889	420.5	1.903	49.90	100	2572.86	0.015	20.28
10	226.00	0.292	57.3	1.914	50	87	2587.73	0.011	14.87
20	229.0	0.257	53.0	1.939	49.90	90	2621.53	0.025	33.8
30	229.20	0.234	49.5	1.960	50	91	2649.92	0.021	28.39
40	229.30	0.232	49.1	1.985	50	92	2683.72	0.025	33.8
50	225.90	1.844	416.3	2.006	49.90	100	2712.11	0.021	28.39
60	228.70	0.222	47.5	2.028	50	94	2741.86	0.022	29.75

Hasil Rabu 12 Oktober 2022 pukul 12.00 (Siang)

Waktu (menit)	Hasil Pengamatan (Tampilan LCD)								
	Tegangan (Volt)	Arus (A)	Daya (Watt)	Nilai kWh	Frekuensi (Hz)	PF (%)	Biaya (Rupiah)	Δ kWh	Δ Biaya (Rupiah)
	V	A	P	E	F				
0	226.50	1.882	425.3	2.030	50	100	2744.56	0	0
5	229.60	0.177	39.2	2.045	50	96	2764.84	0.015	20.28
10	229.10	0.216	45.5	2.056	49.90	92	2779.71	0.011	14.87
20	228.70	0.266	47.5	2.071	49	87	2806.11	0.015	26.4
30	229.10	0.232	417.8	2.085	50	89	2830.81	0.014	24.7
40	229.0	0.241	45.9	2.103	50	100	2852.79	0.018	21.98
50	229.20	1.877	50.1	2.128	49.90	98	2877.44	0.025	24.65
60	228.80	0.211	49.8	2.139	50	87	2899.21	0.011	21.77

Pemograman

ESP 32 (MONITORING)

```
#define BLYNK_TEMPLATE_ID "TMPLLDfVFNYP"
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#define BLYNK_TEMPLATE_ID "TMPLLDfVFNYP" // 32
#define BLYNK_DEVICE_NAME "32"
#define BLYNK_AUTH_TOKEN
"YanPXLQDshREOkqRdQNX63ZDKGYvNeha"

//#define BLYNK_TEMPLATE_ID "TMPLaNSHbt4u" // BLYNK
//#define BLYNK_DEVICE_NAME "MONITORING PENGENDALIAN"
//#define BLYNK_AUTH_TOKEN
"FLwwEC8sKMcGxsRSkCn5hBhn6amzH0pE"

char auth[] = BLYNK_AUTH_TOKEN;
char ssid[] = "TA";
char pass[] = "PASSWORD";

// BATAS SESI BLYNK
// MASUK SESI PZEM
#include <PZEM004Tv30.h>
//int RX 16
//int TX 17
PZEM004Tv30 pzem(&Serial2); //12=d6 rx ,, 13=d7 tx untuk 8266,,,,,,,,, 16rx
17tx untuk 32
float Power, Energy, Voltase, Current, Frequency, PF, Harga;
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,20,4); // set the LCD address to 0x27 for a 16 chars
and 2 line display
```



```

void setup()
{
  Serial.begin(9600);
  lcd.begin();
  // lcd.init();
  // lcd.init();
  lcd.backlight();
  lcd.setCursor(0,0);
  lcd.print("  Bismillah  ");
  lcd.setCursor(0,2);
  lcd.print(" Assalamu Alaikum ");
  delay(4000);
  lcd.clear();
  lcd.setCursor(0,0);
  lcd.print("Connecting .....");delay(500);
  Blynk.begin(auth, ssid, pass);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
    lcd.setCursor(0,2);
    lcd.print(".");
  }
  lcd.clear();
  Serial.println();
  Serial.println("WiFi connected");
  lcd.setCursor(0,1);
  lcd.print("WiFi connected");
  Serial.println("IP address: ");
  lcd.setCursor(0,2);
  lcd.print("IP address: ");
  Serial.println(WiFi.localIP());
  lcd.setCursor(0,2);

```

```

lcd.print(WiFi.localIP());
Serial.println(ssid);
lcd.setCursor(0,3);
lcd.print("ssid : ");lcd.print(ssid);
delay(5000);
lcd.clear();
delay(2000);
lcd.setCursor(0,0);
lcd.print("  Monitoring");
lcd.setCursor(0,1);
lcd.print("  Dan");
lcd.setCursor(0,2);
lcd.print("  Pengendalian");
lcd.setCursor(0,3);
lcd.print("  Daya Listrik");
delay(4000);
lcd.clear();
lcd.setCursor(0,1);
lcd.print("  oleh  ");
delay(2000);
lcd.clear();
lcd.setCursor(0,0);
lcd.print(" HILDA RIANYSKA ");
lcd.setCursor(0,2);
lcd.print(" MUHAMMAD ICHSAN ");
lcd.setCursor(0,3);
lcd.print(" ARIEF MARZUKI ");
delay(4000);
lcd.clear();
}
void loop()
{

```

```

    Voltase = pzem.voltage(); //.....
TEGANGAN
    if(isnan(Voltase))
    {
        Serial.println("Gagal Membaca Voltase");
        lcd.setCursor(0,1);
        lcd.print("V: - ");
    }
    else
    {
        Serial.print("Voltase : ");
        Serial.print(Voltase);
        Serial.println("V");
        lcd.setCursor(0,1);
        lcd.print("V:"); lcd.print(Voltase,1); lcd.print("V"); lcd.print(" ");
    }
    Current = pzem.current(); //.....
KUAT ARUS
    if(isnan(Current))
    {
        Serial.println("Gagal Membaca Current");
        lcd.setCursor(10,1);
        lcd.print("I: - ");
    }
    else
    {
        Serial.print("Current : ");
        Serial.print(Current);
        Serial.println("A");
        lcd.setCursor(10,1);
        lcd.print("I:"); lcd.print(Current,3); lcd.print("A"); lcd.print(" ");
    }

```

```

Power = pzem.power(); //..... DAYA
if(isnan(Power))
{
  Serial.println("Gagal Membaca Power");
  lcd.setCursor(0,2);
  lcd.print("P: - ");
}
else
{
  Serial.print("Power : ");
  Serial.print(Power);
  Serial.println("W");
  lcd.setCursor(0,2);
  lcd.print("P:"); lcd.print(Power); lcd.print("W"); lcd.print(" ");
}

Energy = pzem.energy(); //..... ENERGY
if(isnan(Energy))
{
  Serial.println("Gagal Membaca Energy");
  lcd.setCursor(10,2);
  lcd.print("E: - ");
}
else
{
  Serial.print("Energy : ");
  Serial.print(Energy);
  Serial.println("kWh");
  lcd.setCursor(10,2);
  lcd.print("E:"); lcd.print(Energy,3); lcd.print("kWh"); //lcd.print(" ");
}

```

```

    Harga = pzem.energy()*1352; //.....
HARGA
if(isnan(Harga))
{
    Serial.println("Gagal Membaca Harga");
    lcd.setCursor(0,0);
    lcd.print("Biaya: - ");
}
else
{
    Serial.print("Harga : Rp");
    Serial.print(Harga);
    Serial.println(",-");
    lcd.setCursor(0,0);
    lcd.print("Biaya: Rp");lcd.print(Harga); lcd.print(" ");
// lcd.setCursor(17,0);lcd.print(" ");
}

    Frequency = pzem.frequency();
//.....FREKUENSI
if(isnan(Frequency))
{
    Serial.println("Gagal Membaca Frequency");
    lcd.setCursor(0,3);
    lcd.print("F: - ");
}
else
{
    Serial.print("Frequency : ");
    Serial.print(Frequency);
    Serial.println("Hz");
    lcd.setCursor(0,3);

```

```

    lcd.print("F:"); lcd.print(Frequency,1); lcd.print("Hz"); lcd.print(" ");
}
PF = pzem.pf()*100;
//.....FAKTOR DAYA
if(isnan(PF))
{
    Serial.println("Gagal Membaca PF");
    lcd.setCursor(10,3);
    lcd.print("FP: - ");
}
else
{
    Serial.print("PF : ");
    Serial.print(PF,0);Serial.print("%");
    lcd.setCursor(10,3);
    lcd.print("PF:"); lcd.print(PF,0); lcd.print("% ");
}
    Serial.println(" ");
    Serial.print(" ");
    Serial.println(" ");
// delay(1000);
    Blynk.virtualWrite(V0, Voltase);
    Blynk.virtualWrite(V13, Current);
    Blynk.virtualWrite(V14, Power);
    Blynk.virtualWrite(V15, Energy);
    Blynk.virtualWrite(V16, Frequency);
    Blynk.virtualWrite(V17, PF);
    Blynk.virtualWrite(V18, Harga);
    Blynk.run();
delay(1000);
// delay(2000);
// lcd.clear();

```

```
// delay(10);  
}
```

ESP 8266 (PENGENDALIAN)

```
#include <LiquidCrystal_I2C.h>  
LiquidCrystal_I2C lcd(0x27, 16, 2);  
#include <RTCLib.h>  
RTC_DS3231 rtc;  
char namaHari[7][12] = {"Min", "Sen", "Sel", "Rab", "Kam", "Jum", "Sab"};  
#define BLYNK_PRINT Serial  
#define BLYNK_TEMPLATE_ID "TMPLaNSHbt4u"  
#define BLYNK_DEVICE_NAME "MONITORING PENGENDALIAN"  
#define BLYNK_AUTH_TOKEN  
"FLwwEC8sKMcGxsRSkCn5hBhn6amzH0pE"  
#include <ESP8266WiFi.h>  
#include <BlynkSimpleEsp8266.h>  
char auth[] = BLYNK_AUTH_TOKEN;  
//char auth[] = "FLwwEC8sKMcGxsRSkCn5hBhn6amzH0pE";  
//char ssid[] = "Faznet";  
//char pass[] = "Mhytikromawi";  
char ssid[] = "TA";  
char pass[] = "PASSWORD";  
#define RELAY1 0  
#define RELAY2 14  
#define RELAY3 12  
#define RELAY4 13  
int status_relay1 = 0;  
int status_relay2 = 0;  
int status_relay3 = 0;  
int status_relay4 = 0;  
  
#define VIRTUAL_PIN11 V11
```

```
#define VIRTUAL_PIN12 V12
#define VIRTUAL_PIN13 V13
#define VIRTUAL_PIN14 V14
```

```
BLYNK_CONNECTED() {
  Blynk.syncVirtual(VIRTUAL_PIN11);
  Blynk.syncVirtual(VIRTUAL_PIN12);
  Blynk.syncVirtual(VIRTUAL_PIN13);
  Blynk.syncVirtual(VIRTUAL_PIN14);
}
```

```
BLYNK_WRITE(VIRTUAL_PIN11) {
  status_relay1 = param.asInt();
  digitalWrite(RELAY1, status_relay1);
}
```

```
BLYNK_WRITE(VIRTUAL_PIN12) {
  status_relay2 = param.asInt();
  digitalWrite(RELAY2, status_relay2);
}
```

```
BLYNK_WRITE(VIRTUAL_PIN13) {
  status_relay3 = param.asInt();
  digitalWrite(RELAY3, status_relay3);
}
```

```
BLYNK_WRITE(VIRTUAL_PIN14) {
  status_relay4 = param.asInt();
  digitalWrite(RELAY4, status_relay4);
}
```

```
//Reset pzem VIRTUAL PIN 49
```

```
//BLYNK_WRITE(V49) // Virtual push button is defined
as V4 of Blynk App. When the button is pushed, it will activate the codes
// {
//   if(param.asInt()==1)
```



```

//      {
//      uint16_t u16CRC = 0xFFFF;          /* declare CRC check 16
bits*/
//      static uint8_t resetCommand = 0x42;      /* reset command code*/
//      uint8_t slaveAddr =0X01;
//      u16CRC = crc16_update(u16CRC, slaveAddr);
//      u16CRC = crc16_update(u16CRC, resetCommand);
//      preTransmission();          /* trigger transmission mode*/
//      Serial.write(slaveAddr);      /* send device address in 8 bit*/
//      Serial.write(resetCommand);    /* send reset command */
//      Serial.write(lowByte(u16CRC));  /* send CRC check code
low byte (1st part) */
//      Serial.write(highByte(u16CRC));  /* send CRC check code
high byte (2nd part) */
//      delay(10);
//      postTransmission();          /* trigger reception mode*/
//      delay(100);
//      }
//      }

//
// //RESET PZEM CARA 2 virtual pin 50
//BLYNK_WRITE(V50)
//{
// currentMillis = millis();
// int pinValue = param.asInt();
// if (pinValue = 1){
// pzem.resetEnergy();
// pzem.setAddress(0x42);
// token = 0;
// Line_Notify(message8 + e + " kWh " + b + " baht ");
// }
//}

```

```

void setup()
{
  // Debug console
  Serial.begin(9600);
  // delay(5000);
  lcd.begin();
  lcd.backlight();
  rtc.begin();
  // klo baterai rusak, pake ko ini !!!
  // if (rtc.lostPower())
  // {rtc.adjust(DateTime(F(__DATE__), F(__TIME__))); }
  // ini jam komputer !!
  // aktif kan ini untuk upload PERTAMA !!!
  // rtc.adjust(DateTime(F(__DATE__), F(__TIME__)));
  lcd.setCursor(0,0);
  lcd.print(" JAM DIGITAL ");
  delay(1000);
  lcd.setCursor(0,1);
  lcd.print("REAL TIME 'CLOCK'");
  delay(3000);
  lcd.clear();
  pinMode(RELAY1, OUTPUT);
  digitalWrite(RELAY1, HIGH);
  pinMode(RELAY2, OUTPUT);
  digitalWrite(RELAY2, HIGH);
  pinMode(RELAY3, OUTPUT);
  digitalWrite(RELAY3, HIGH);
  pinMode(RELAY4, OUTPUT);
  digitalWrite(RELAY4, HIGH);
}

```

```

lcd.setCursor(0,0);
lcd.print("Connecting .....");delay(500);
Serial.print("Loading.....");
Blynk.begin(auth, ssid, pass);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
        lcd.setCursor(0,2);
        lcd.print(".");
    }
    lcd.clear();
    Serial.println();
    Serial.println("WiFi connected");
    lcd.setCursor(0,0);
    lcd.print("WiFi connected");
// delay(2000);
// lcd.clear();
// Serial.println("IP address: ");
// lcd.setCursor(0,0);
// lcd.print("IP address: ");
//
// Serial.println(WiFi.localIP());
// lcd.setCursor(0,1);
// lcd.print(WiFi.localIP());
// delay(3000);lcd.clear();
    Serial.println(ssid);
    lcd.setCursor(0,1);
    lcd.print("ssid : ");lcd.print(ssid);
    delay(3000);
    lcd.clear();
// delay(3000);
}

```

```

void loop()
{
  Blynk.run();
  DateTime now = rtc.now();
  lcd.setCursor(0,0);
  lcd.print(namaHari[now.dayOfTheWeek()]);
  lcd.print(", ");
  printAngka(now.day());
  lcd.print("-");
  printAngka(now.month());
  lcd.print("-");
  printAngka(now.year());
  lcd.setCursor(0,1);
  printAngka(now.hour());
  lcd.print(":");
  printAngka(now.minute());
  lcd.print(":");
  printAngka(now.second());
  lcd.setCursor(10,1);
  lcd.print(rtc.getTemperature(),1);
  lcd.print("C");

  // lcd.clear();
  delay(200);
}

void printAngka(int digits){
  if(digits < 10){
    lcd.print('0');
    lcd.print(digits);
  }
  else lcd.print(digits);
}

```