

**SISTEM MONITORING PENGGUNAAN ENERGI PERANGKAT  
LISTRIK RUMAH VIA BOT TELEGRAM**

**SKRIPSI**

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



Disusun Oleh:

**REMA SUFIATIKA**

**3332170009**

**JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS SULTAN AGENG TIRTAYASA  
2024**

## LEMBAR PERNYATAAN KEASLIAN SKRIPSI

Dengan ini saya sebagai penulis Skripsi berikut:

Judul : Sistem Monitoring Penggunaan Energi Perangkat Listrik  
Rumah Via Bot Telegram  
Nama Mahasiswa : Rema Sufiatika  
NPM : 3332170009  
Fakultas/Jurusan : Fakultas Teknik / Jurusan Teknik Elektro

Menyatakan dengan sesungguhnya bahwa skripsi tersebut di atas benar-benar hasil karya saya dan tidak memuat hasil karya orang lain, kecuali dinyatakan melalui rujukan yang benar dan dapat dipertanggungjawabkan. Apabila di kemudian 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.

Serang, 30 Maret 2024



Rema Sufiatika

NIM. 3332170009

## LEMBAR PENGESAHAN

Dengan ini ditetapkan bahwa Skripsi berikut:

Judul : Sistem Monitoring Penggunaan Energi Perangkat Listrik  
Rumah Via Bot Telegram  
Nama Mahasiswa : Rema Sufiatika  
NPM : 3332170009  
Fakultas/Jurusan : Fakultas Teknik/Jurusan Teknik Elektro

Telah diuji dan dipertahankan pada tanggal 30 Januari 2024 melalui Sidang Skripsi di Fakultas Teknik Universitas Sultan Ageng Tirtayasa Cilegon dan dinyatakan **LULUS**.

Dewan Penguji


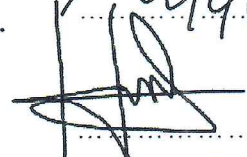

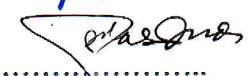
Pembimbing I : Anggoro Suryo Pramudyo, S.Kom., M.Kom.

Pembimbing II : Heri Haryanto, S.T., M.T.

Penguji I : Dr. Eng. Rocky Alfanz, S. T., M. Sc

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

Tanda Tangan

  
.....  
  
.....  
  
.....  
  
.....

Mengetahui,  
Ketua Jurusan

  
  
Dr. Eng. Rocky Alfanz, S. T., M. Sc.  
NIP. 198103282010121001

## PRAKATA

Puji dan syukur atas kehadiran Allah Subhanahu Wata'ala berkat limpahan Rahmat dan Karunia-Nya penulis dapat menyelesaikan skripsi yang berjudul “**Sistem Monitoring Penggunaan Energi Perangkat Listrik Rumah Via Bot Telegram**” sebagai salah satu syarat dalam menyelesaikan Program Sarjana (S1) dan memperoleh gelar Sarjana Teknik di Universitas Sultan Ageng Tirtayasa. Penulis menyadari, penulisan skripsi ini tidak dapat terselesaikan tanpa dukungan dan bantuan dari berbagai pihak. Oleh karena itu, penulis ingin menyampaikan rasa terima kasih kepada semua pihak yang memberikan bantuan dan dukungan dalam penulisan, penyusunan dan penyelesaian skripsi ini. Adapun penulis sampaikan banyak terima kasih yang sebesar - besarnya kepada:

1. Kedua orang tua, adik dan kerabat yang selalu memberikan doa maupun dukungan sehingga penulis dapat menyelesaikan skripsi ini.
2. Bapak Dr. Eng. Rocky Alfan, S. T., M. Sc. selaku Ketua Jurusan Teknik Elektro Universitas Sultan Ageng Tirtayasa.
3. Bapak Anggoro Suryo Pramudyo, S.Kom., M.Kom. selaku Dosen Pembimbing 1 yang telah membantu atas segala bimbingan, arahan, serta saran yang diberikan kepada penulis selama proses menyelesaikan skripsi.
4. Bapak Heri Haryanto, S.T., M.T. selaku Dosen Pembimbing 2 yang telah membantu atas segala bimbingan, arahan, serta saran yang diberikan kepada penulis selama proses menyelesaikan skripsi
5. Bapak Dr. Romi Wiryadinata, M.Eng. selaku Dosen Pembimbing Akademik yang telah membantu dalam mengikuti dan menyelesaikan studi selama masa perkuliahan di Jurusan Teknik Elektro Universitas Sultan Ageng Tirtayasa.
6. Seluruh Dosen, Staf Akademik dan teman-teman Jurusan Teknik Elektro Fakultas Teknik Universitas Sultan Ageng Tirtayasa yang telah memberikan banyak dukungan dan semangat kepada penulis.

Akhir kata, penulis menyadari bahwa penelitian ini masih jauh dari kata sempurna dan masih banyak kekurangan. Penulis berharap semoga penelitian ini akan memberi manfaat bagi pembaca dan penelitian di masa depan. Penulis memohon maaf sebesar-besarnya apabila terjadi kesalahan dalam penulisan skripsi ini. Penulis menerima kritik dan saran dari para pembaca untuk kemajuan penelitian ini.

Serang, 30 Maret 2024

Penulis

## ABSTRAK

Rema Sufiatika

Teknik Elektro

### Sistem Monitoring Penggunaan Energi Perangkat Listrik Rumah Tangga via Bot Telegram

Kenaikan biaya listrik kerap menjadi masalah di kalangan masyarakat karena penggunaan listrik sudah menjadi kebutuhan sehari-hari. Melakukan pengukuran dan pencatatan penggunaan energi listrik dapat membantu meningkatkan fleksibilitas dan efisiensi bagi penggunaannya. Penelitian skripsi ini bertujuan untuk merancang dan mengimplementasikan teknologi IoT pada perangkat sistem monitoring dan kendali penggunaan energi listrik serta estimasi biaya yang dikeluarkan oleh penggunaan peralatan listrik rumah tangga. *Bot Telegram* sebagai perantara antara perangkat dan pengguna dengan mikrokontroler menggunakan Wemos D1 R1 sebagai pusat kendalinya dan untuk instalasi pemrogramannya menggunakan Arduino IDE. Pengendalian ini dilakukan dengan bantuan koneksi internet sebagai media penghubungnya dan juga *Bot Telegram* sebagai media komunikasi perintah yang diberikan dengan jarak yang jauh dan masih terkoneksi internet antara perangkat penerima dan juga penggunanya. Hasil pengujian menunjukkan bahwa alat berhasil menampilkan nilai parameter tegangan, arus, daya, kWh, dan biaya ditampilkan pada layar LCD serta data dapat diakses dan dikendalikan melalui *Bot Telegram*. Persentase kesalahan yang dihasilkan oleh alat untuk monitoring yaitu sebesar 0,18% untuk tegangan dan 2% untuk arus. Sedangkan hasil pengujian kendali melalui *Bot Telegram* untuk mengendalikan on dan off dengan beban seperti lampu, kipas angin dan rice cooker mendapatkan rata-rata waktu respon selama 17,9 detik.

Kata Kunci: *Internet of Things* (IoT), Monitoring, Telegram, Listrik

## ***ABSTRACT***

Rema Sufiatika

*Electrical Engineering*

### *System of Monitoring Energy Usage of Household Electrical Devices via Bot Telegram*

*The rising cost of electricity is often a problem among the public because the use of electricity has become a daily necessity. Measuring and recording the use of electrical energy can help increase flexibility and efficiency for its use. This final project research aims to design and implement IoT technology on a monitoring and control system device for the use of electrical energy and estimation of costs incurred by the use of household electrical appliances. Telegram bot as an intermediary between the device and the user where Wemos D1 R1 as the control center and programming using Arduino IDE. This control is carried out with the help of an internet connection as a connecting medium and also Telegram Bot as a communication medium for commands given at a distance and still connected to the internet between the receiving device and the user. The test results show that the device successfully displays the parameter values of voltage, current, power, kWh, and cost on the LCD screen and the data can be accessed and controlled via Telegram Bot. The percentage of error generated by the device for monitoring is 0.18% for voltage and 2% for current. While the test results of control via Telegram Bot to control on and off with loads such as lights, fans and rice cookers get an average response time of 17.9 seconds.*

*Keywords: Internet of Things (IoT), Monitoring, Telegram, Electricity*

## DAFTAR ISI

<b>HALAMAN JUDUL .....</b>	<b>i</b>
<b>LEMBAR PERNYATAAN KEASLIAN SKRIPSI.....</b>	<b>ii</b>
<b>LEMBAR PENGESAHAN.....</b>	<b>iii</b>
<b>PRAKATA.....</b>	<b>iv</b>
<b>ABSTRAK.....</b>	<b>vi</b>
<b>ABSTRACT .....</b>	<b>vii</b>
<b>DAFTAR ISI.....</b>	<b>viii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xi</b>
<b>DAFTAR TABEL.....</b>	<b>xiii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Rumusan Masalah .....	4
1.3 Tujuan Penelitian.....	5
1.4 Manfaat Penelitian.....	5
1.5 Batasan Masalah.....	5
1.6 Sistematika Penulisan.....	6
<b>BAB II TINJAUAN PUSTAKA.....</b>	<b>8</b>
2.1 Energi Listrik.....	8
2.1.1 Arus Listrik.....	9
2.1.2 Tegangan.....	9
2.1.3 Daya Listrik .....	10
2.1.4 Beban Listrik .....	11
2.2 <i>Internet of Things</i> (IoT).....	12
2.2.1 Wemos D1 R1 .....	14
2.2.2 Modul ESP8266 Port Wi-Fi .....	17
2.2.3 Modul PZEM-004T .....	18



2.3 Sistem Kendali .....	19
2.4 <i>Google Sheets</i> .....	20
2.5 <i>Bot Telegram</i> .....	20
2.6 Kajian Pustaka.....	21
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>24</b>
3.1 Metode Penelitian.....	24
3.2 Instrumen Penelitian.....	25
3.3 Diagram Blok Sistem .....	26
3.4 Perancangan Perangkat Keras ( <i>Hardware</i> ).....	27
3.5 Perancangan Perangkat Lunak ( <i>Software</i> ) .....	30
3.5.1 Perancangan <i>Bot Telegram</i> .....	31
3.5.2 Perancangan <i>Google Sheets</i> sebagai <i>Database</i> .....	37
3.6 Perancangan Kendali Beban Listrik .....	41
3.7 Perancangan Monitoring Energi.....	42
<b>BAB IV HASIL DAN ANALISIS .....</b>	<b>46</b>
4.1 Hasil Perancangan Alat .....	46
4.2 Pengujian <i>Hardware</i> .....	47
4.2.1 Hasil Pengujian Relay .....	47
4.2.2 Hasil Pengujian PZEM004T.....	48
4.3 Pengujian <i>Software</i> .....	52
4.3.1 Hasil Pengujian <i>Google Sheets</i> sebagai <i>Database</i> .....	52
4.3.2 Hasil Pengujian <i>Bot Telegram</i> .....	53
4.4 Pengujian Keseluruhan .....	57
4.4.1 Hasil Pengujian Beban Listrik .....	59
4.4.2 Hasil Pengujian Kendali Beban Listrik .....	61
4.4.3 Hasil Pengujian Monitoring Energi dan Estimasi Biaya .....	63
<b>BAB V PENUTUP.....</b>	<b>67</b>
5.1 Kesimpulan.....	67
5.2 Saran.....	68

<b>DAFTAR PUSTAKA .....</b>	<b>69</b>
<b>LAMPIRAN.....</b>	<b>72</b>
<b>LAMPIRAN A.....</b>	<b>A-1</b>
<b>LAMPIRAN B.....</b>	<b>B-1</b>
<b>LAMPIRAN C.....</b>	<b>C-1</b>

## DAFTAR GAMBAR

Gambar 2. 1 Sumber Tegangan .....	10
Gambar 2. 2 Wemos D1 R1 Board. ....	14
Gambar 2. 3 Software Arduino IDE.....	15
Gambar 2. 4 Jenis-Jenis ESP8266.....	17
Gambar 2. 5 Modul PZEM-004T .....	18
Gambar 3. 1 Alur Penelitian .....	24
Gambar 3. 2 Diagram Blok Sistem .....	26
Gambar 3. 3 Flowchart Sistem Hardware .....	28
Gambar 3. 4 Rangkaian Komponen Alat Monitoring Kendali .....	29
Gambar 3. 5 Flowchart Pembuatan Akun Bot Telegram .....	32
Gambar 3. 6 Search BotFather .....	33
Gambar 3. 7 BotFather.....	33
Gambar 3. 8 Menu Perintah pada BotFather.....	34
Gambar 3. 9 Tampilan Token API pada BotFather .....	34
Gambar 3. 10 Menu /mybots pada BotFather .....	35
Gambar 3. 11 <i>Flowchart</i> Sistem <i>Bot Telegram</i> .....	36
Gambar 3. 12 <i>Flowchart</i> Perancangan <i>Google Sheets</i> sebagai <i>Database</i> .....	38
Gambar 3. 14 <i>Flowchart</i> Perancangan Kendali Beban Listrik .....	41
Gambar 3. 15 <i>Flowchart</i> Perancangan Monitoring Energi .....	43
Gambar 3. 16 Rangkaian Penggunaan Alat pada MCB.....	44
Gambar 4. 1 (a). Tampak Dalam; (b). Tampak Luar Alat Monitoring .....	46
Gambar 4. 2 Hasil Pengujian Relay .....	47
Gambar 4. 3 Perbandingan Tegangan Hasil Alat Monitoring dan Alat Ukur.....	49
Gambar 4. 4 Grafik Perbandingan Tegangan Hasil Alat Monitoring dan Alat Ukur.....	49
Gambar 4. 5 Pengamatan Perbandingan Arus Hasil	

Alat Monitoring dan Alat Ukur.....	50
Gambar 4. 6 Grafik Perbandingan Arus Hasil	
Alat Monitoring dan Alat Ukur.....	51
Gambar 4. 7 Tampilan <i>Database</i> pada <i>Google Sheets</i> .....	53
Gambar 4. 8 Tampilan Bio Akun Bot pada Aplikasi Telegram.....	54
Gambar 4. 9 Tampilan Pesan Perintah “/start” pada <i>Bot Telegram</i> .....	54
Gambar 4. 10 Tampilan Tombol Alternatif Pesan Perintah <i>Bot Telegram</i> .....	55
Gambar 4. 11 Tampilan Monitoring perparameter pada Bot Telegram.....	55
Gambar 4. 12 Tampilan Pesan Perintah Reset pada Bot Telegram .....	56
Gambar 4. 13 Tampilan Pesan Perintah Kendali pada Bot Telegram.....	56
Gambar 4. 14 Tampilan Data Monitoring pada Bot Telegram .....	57
Gambar 4. 15 Tampilan Serial Monitor pada Arduino IDE saat	
Internet telah Terhubung dengan Alat Monitoring Kendali .....	58
Gambar 4. 16 Tampilan pada <i>Bot Telegram</i> (a) dan LCD (b).....	59
Gambar 4. 17 Grafik Hasil Pengujian Beban Listrik Rumah selama 1 Bulan .....	60
Gambar 4. 18 Hasil Pengujian Kendali Beban Listrik .....	62
Gambar 4. 19 Grafik Daya dan Estimasi Biaya	
Penggunaan Energi Listrik dalam Waktu 1 Bulan .....	63
Gambar 4. 20 Grafik Perbandingan Nilai kWh PLN dan Alat Monitoring .....	65

## DAFTAR TABEL

Tabel 4. 1 Hasil Pengujian Beban Listrik (1 Jam) .....	60
Tabel A- 1 Library yang digunakan pada Arduino IDE.....	A-5
Tabel A- 2 Variabel yang digunakan pada Arduino IDE.....	A-6
Tabel B- 1 Rekapitulasi Data Pengujian selama 1 Bulan (25 Juni 2023-25 Juli 2023) .....	B-1

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Sistem tenaga listrik merupakan sistem yang paling kompleks dan memiliki arti penting dalam kehidupan modern. Listrik memiliki dampak langsung pada aspek modernisasi, ekonomi, politik dan sosial, untuk mengoperasikan sistem tersebut dalam mode stabil, diperlukan beberapa teknik kendali dan keamanan. Namun, sistem *modern* dilengkapi dengan beberapa skema perlindungan dengan tujuan untuk menghindari kejadian tak terduga dan pemadaman listrik [1]. Pemantauan kelistrikan pada suatu instalasi sangat penting karena merogoh porsi dominan (73%) dari penggunaan energi di dunia dalam bentuk listrik. Di Eropa, sektor perumahan, tersier, dan bangunan komersial bertanggung jawab atas lebih dari 50% konsumsi listrik. Sebagian besar masyarakat saat ini menghabiskan 90% waktunya di dalam ruangan dengan mengandalkan listrik untuk sebagian besar aktivitasnya seperti penerangan dan AC untuk kenyamanan ruangan [2].

Sektor rumah tangga merupakan penyumbang terbesar persentase konsumsi listrik dari total konsumsi listrik Indonesia pada tahun 2021, Listrik yang terjual pada tahun 2021 sebesar 257,634.25 GWh, meningkat sebesar 4,94% dibandingkan dengan tahun lalu. Penjualan listrik untuk basis pelanggan rumah tangga 115,370.05 GWh menyumbang 44,78% dari total persentase [3]. PT. PLN Persero di Indonesia mewajibkan setiap konsumen listrik memiliki satu *kiloWatt hour* (kWh) meter. kWh meter digunakan sebagai alat ukur dalam transaksi kelistrikan listrik [4]. kWh meter hanya memberikan informasi umum untuk konsumsi listrik di sebuah rumah.

Perangkat listrik sudah menjadi kebutuhan pokok masyarakat saat ini. Sejak Maret 2020 hingga saat ini, terjadinya pandemi virus corona berdampak pada peningkatan konsumsi listrik. PT Perusahaan Listrik Negara (PLN) menyatakan ada

tiga hal yang menyebabkan tagihan listrik meroket di tengah penerapan Pembatasan Sosial Berskala Besar (PSBB), yakni kebijakan *Work from Home* (WFH), Ramadan, dan kebijakan pencatatan meter rata-rata. Kenaikan tarif energi listrik bagi rumah tangga dan industri kecil dapat menyebabkan penghapusan subsidi energi berdasarkan Undang-Undang Nomor 30 Tahun 2007 tentang Energi dan Undang-Undang Nomor 30 Tahun 2009 tentang Ketenagalistrikan. Pemerintah mengalokasikan dana subsidi untuk membantu kelompok masyarakat yang kurang mampu [5].

Kenaikan penggunaan daya listrik bergantung pada pemakaian dalam rumah tangga. Ketika semakin banyak perangkat elektronik yang digunakan, terjadi peningkatan konsumsi energi yang dapat menyebabkan beban berlebih pada sistem kelistrikan. Penelitian ini bertujuan untuk membantu beberapa pengguna listrik mengetahui penggunaan energi listrik pada perangkat listrik yang digunakan di rumah, maka diperlukan alat monitoring dan kendali yang dapat memantau pemakaian energi listrik pada perangkat listrik yang dianggap cukup boros pemakaiannya.

Energi listrik adalah salah satu sektor terpenting di mana sistem *Internet of Things* (IoT) dapat mengendalikan, mendeteksi, dan menginformasikan kesalahan, memonitoring data dan meningkatkan efisiensi dari penggunaan energi listrik [6]. Model pengendalian dan pemantauan penggunaan energi listrik dapat membantu menentukan konsumsi energi listrik, perhitungan penggunaan jarak jauh, dan kendali, serta menentukan daya yang disimpan dan diproses secara *real-time* dengan menerapkan metode kecerdasan buatan yaitu *Internet of Things* (IoT) [7][8]. Beriringan dengan penggunaan *smartphone* yang semakin meningkat, aplikasi *instant messenger* sudah menjadi kebutuhan bagi para pengguna *smartphone*. Ada banyak aplikasi *instant messenger* gratis yang tersedia sekarang dengan perkembangan teknologi yang telah menunjukkan peningkatan besar di dalamnya. Layanan ini memudahkan dalam berkomunikasi dengan teman melalui pesan teks, panggilan telepon, video, berbagi dokumen, grup, dan tersedia botnya dalam aplikasi tersebut [9].

Beberapa penelitian terdahulu telah menyajikan perancangan monitoring energi listrik dalam bentuk digital dan terintegrasi dengan Internet of Things. Penelitian yang

sudah dilakukan selanjutnya adalah menerapkan sistem analitik energi real-time untuk bangunan pintar yang menggunakan platform IoT terbuka [2]. Penelitian selanjutnya adalah mengukur kesalahan pada kWh meter, mengevaluasi akurasi dan ketelitiannya, membandingkan hasil peneraan dengan standar kWh meter induk, dan mengidentifikasi faktor-faktor yang dapat mempengaruhi kesalahan pengukuran kWh meter [4]. Penelitian selanjutnya adalah sistem monitoring dan kendali perangkat listrik berbasis IoT menggunakan NodeMCU ESP8266 dengan bantuan Artificial Intelligence Chatbot dengan menggunakan Telegram Messenger. Telegram Messenger digunakan sebagai media komunikasi pada penelitian ini [10]. Penelitian selanjutnya melakukan menerapkan teknologi Internet of Things (IoT) untuk mengendalikan alat elektornik dan memantau daya listrik terpakai dari jarak jauh dengan menggunakan Wemos D1 [8]. Penelitian lainnya mengembangkan sistem pemantauan penggunaan listrik rumah tangga secara real-time berbasis web. Penelitian menggunakan sensor PZEM-004T [11].

Pada skripsi ini memanfaatkan teknologi *Internet of Things* (IoT) dengan fokus pada bagaimana kemampuan IoT dalam melakukan pemantauan konsumsi listrik rumah dari jarak jauh menggunakan *Bot Telegram*. Metode pengukuran alat monitoring ini yang adalah dengan cara mengukur besaran tegangan dan arus yang mengalir menuju beban yang nantinya akan diolah pada Modul Wemos D1 R1 [12], yang berfungsi sebagai pusat pengolahan data. Sensor PZEM004T digunakan untuk mengukur energi listrik. Modul PZEM004T merupakan sebuah modul sensor multifungsi yang berfungsi untuk mengukur daya, tegangan, arus dan energi yang terdapat pada sebuah aliran listrik. Modul ini sudah dilengkapi sensor tegangan dan *clamp* untuk arus (CT) yang sudah terintegrasi. Dalam penggunaan alat ini, beban yang terpasang tidak diperbolehkan melebihi daya yang sudah ditetapkan [11]. Hasil pengolahan data tersebut akan ditampilkan dengan parameter tegangan, arus, daya, kWh, dan biaya pada layar LCD 20x4.

Wemos D1 R1 sudah berbasis ESP8266 yaitu modul Wi-Fi, CH340 USB *to serial interface* dan selain itu mikrokontroler ini juga sudah kompatibel dengan beragam



Arduino. Sistem ini juga berbasis konsep *Internet of Things* (IoT), pada sistem ini berfungsi untuk pengiriman data antar perangkat menggunakan jaringan internet. Data dapat diakses pada aplikasi Telegram dengan diawali beberapa perintah bot yang dikirimkan oleh pengguna bot tersebut. Penelitian ini menggunakan *Google Sheets* sebagai penyimpanan data alat monitoring. *Google Sheets* terintegrasi dengan *Google Docs* yang memiliki banyak manfaat. *Google Sheets* memiliki sejumlah fitur dan keunggulan signifikan. Salah satu keunggulannya adalah kemudahan aksesibilitas, yang memungkinkan pengguna untuk mengakses dan mengelola data mereka dari berbagai lokasi melalui penyimpanan berbasis *cloud Google*. Keunggulan ini memudahkan mobilitas pengguna yang sering berpindah tempat kerja atau memiliki kebutuhan untuk mengakses data dari berbagai perangkat yang berbeda [12][13].

Aplikasi Telegram dipilih sebab gratis, ringan, serta lintas platform. Aplikasi ini bisa diunduh di perangkat seluler, tablet, atau komputer. Telegram mempunyai beberapa fitur keamanan diantaranya enkripsi *end-to-end* pada seluruh jenis pesan, *selfdestructing messages*, dan lebih mudah penggunaannya untuk *multiple devices* [15]. API (*Application Programming Interface*) bot pada telegram yang relatif lengkap serta berkembang yang memungkinkan penggunanya menghasilkan bot yang bisa merespon pesan sesuai kebutuhannya [10]. Dengan penelitian ini diharapkan mempermudah para pengguna listrik rumah tangga untuk memantau, mengendalikan serta melakukan kalkulasi beban listrik rumah tangga dalam kehidupan sehari-hari.

## 1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan sebelumnya, maka dapat ditentukan persamaan masalah yaitu bagaimana

1. Kurangnya pemantauan penggunaan energi perangkat listrik pada rumah tangga.
2. Tidak adanya pengendalian penggunaan beban listrik dari jarak jauh pada kWh meter konvensional sehingga mengurangi fleksibilitas pengguna dalam mengendalikan beban listrik yang terpasang.

### 1.3 Tujuan Penelitian

Adapun tujuan yang akan dicapai dalam penelitian ini adalah penulis dapat merancang alat untuk memonitoring dan mengendalikan perangkat listrik rumah tangga sehingga dapat memantau penggunaan energi listrik berbasis internet dengan menampilkan tegangan, arus, daya, kWh, dan biaya pada LCD dan dapat mengaksesnya melalui *bot Telegram*.

### 1.4 Manfaat Penelitian

Manfaat dari penelitian ini adalah:

1. Pengguna dapat melihat penggunaan energi listrik saat itu juga.
2. Pengguna dapat mengidentifikasi peralatan atau kebiasaan yang mengonsumsi energi secara berlebihan, mengatur jadwal penggunaan energi, atau bahkan mengambil tindakan otomatis untuk mengoptimalkan penggunaan energi.
3. Melalui pemantauan yang memanfaatkan IoT konsumsi listrik dapat dipantau lebih mudah, pengguna dapat melakukan pengaturan dan penyesuaian untuk mengurangi biaya listrik, sehingga pengguna dapat menghindari pemakaian energi pada puncak tarif listrik yang lebih mahal.
4. Pengguna dapat mengendalikan penggunaan beban energi perangkat listrik rumah tangga dengan jaringan internet, seperti mengatur peralatan untuk beroperasi secara otomatis berdasarkan jadwal atau pesan perintah yang dikirim dari *Bot Telegram*.

### 1.5 Batasan Masalah

Solusi untuk menyelesaikan masalah yang dilakukan agar tidak menyimpang dari ruang lingkup yang ditentukan, maka akan dilakukan pembatasan masalah. Adapun Batasan masalah ini ialah sebagai berikut:

1. Beban yang dimonitoring hanya penggunaan listrik rumah tangga dengan sumber tegangan AC.
2. Hanya memudahkan para pengguna untuk menerima informasi tanpa merubah perangkat yang dipasang oleh PLN.
3. Menggunakan mikrokontroler Wemos D1 R1.
4. Menggunakan sensor PZEM004T untuk sensor tegangan dan arus serta relay berfungsi sebagai modul kendalinya.
5. Arus maksimal yang dapat diukur sebesar 10A.
6. Data akan dikirimkan pada *Bot Telegram* apabila pengguna mengirimkan pesan perintah pada obrolan tersebut
7. Pengendalian beban terpasang hanya dapat bekerja apabila pengguna mengirimkan pesan perintah pada obrolan tersebut

## 1.6 Sistematika Penulisan

Skripsi ini terdiri atas lima bab dengan sistematika penulisan sebagai berikut:

Bab I adalah pendahuluan yang berisikan uraian singkat mengenai segala sesuatu yang penulis lakukan dalam penelitian, yaitu meliputi latar belakang masalah dalam penelitian ini, pepersamaan masalah, tujuan penulisan, manfaat penelitian, dan batasan masalah.

Bab II adalah tinjauan pustaka. Bab ini berisikan teori-teori yang dibutuhkan dalam penelitian, yang diantaranya mengenai *Internet of Things* (IoT), arus listrik, daya listrik, kualitas daya Arduino, serta membahas mengenai perangkat dan program digunakan saat di gunakan. Teori-teori dasar yang digunakan sebagai acuan, karakteristik dari komponen yang akan digunakan, sifat dan struktur komponen. Pada bagian akhir dari bab ini terdapat kajian pustaka, yaitu berisikan mengenai penelitian terdahulu.

Bab III adalah metodologi penelitian, dimana berisi deskripsi prinsip kerja yang mencakup perancangan *hardware* yaitu skematik alat dan intalasi *software* sebagai

pengolah data dan juga berisi data-data yang diperlukan dalam penelitian. Pada bab ini penulis juga membahas mengenai diagram alir dalam penelitian ini.

Bab IV adalah bab hasil dan pembahasan. Data-data yang telah didapat lalu dipakai dalam simulasi program. Data-data yang sama juga dipakai dalam perhitungan manual. Pengujian dan pengukuran dilakukan dengan beberapa tahap. Hasil dari pengujian tersebut dianalisis dan dibandingkan dengan penelitian sebelumnya.

Bab V adalah bab penutup. Bab ini berisi hasil akhir yang didapat dari penelitian yang penulis lakukan. Data-data hasil penelitian tersebut ditulis dalam bentuk kesimpulan. Di akhir bab ini, penulis memberikan saran yang dapat dipakai untuk pengembangan dalam penelitian yang sama

## BAB II

### TINJAUAN PUSTAKA

#### 2.1 Energi Listrik

Bila pada ujung-ujung suatu kawat penghantar yang hambatannya ( $R$ ) terdapat beda potensial ( $V$ ), maka di dalamnya mengalir arus ( $I$ ) berbanding lurus dengan tegangan ( $V$ ) dan berbanding terbalik dengan resistansi ( $R$ ). Untuk mengalirkan arus ini sumber arus mengeluarkan energi. Sebagian dari energi ini berubah menjadi kalor yang menyebabkan kawat penghantar menjadi panas. Hal ini terjadi karena elektron-elektron bebas dalam kawat atom-atom kawat yang dilaluinya. Besarnya kalor yang timbul ditentukan oleh faktor-faktor:

- a. besarnya hambatan kawat yang dilalui arus
- b. besarnya arus yang mengalir
- c. waktu atau lamanya arus mengalir.

Besarnya energi yang dikeluarkan oleh sumber arus untuk mengalirkan arus listrik ditulis dengan persamaan 2-1 dan 2-2 [16].

$$W = V \times I \times t = I^2 \times R \times t \quad (2-1)$$

$$W = \frac{V^2}{R} \times t = P \times t \quad (2-2)$$

Keterangan:

$W$  = Energi listrik (Joule)

$V$  = Tegangan (Volt)

$I$  = Arus listrik (Ampere)

$P$  = Daya listrik (Watt)

$R$  = Hambatan (Ohm,  $\Omega$ )

$t$  = waktu (sekon)

### 2.1.1 Arus Listrik

Arus listrik adalah aliran muatan listrik atau muatan listrik yang mengalir tiap satuan waktu. Arus listrik diukur dalam *Ampere* (A), yang diambil dari nama Andre Ampere [17]. Pada dasarnya, dua kabel paralel dengan arus listrik yang mengalir melaluinya akan menghasilkan gaya tarik-menarik satu sama lain.

Kuat arus listrik yang mengalir berbanding lurus dengan beda potensial. Titik yang memiliki potensial tinggi melepaskan muatan ke titik yang memiliki potensial rendah. Kuat arus yang mengalir berbanding lurus dengan beda potensial antara dua titik atau  $I \propto V$  [16], yang dapat ditulis seperti pada persamaan 2-3.

$$I = \frac{1}{R} V \quad (2-3)$$

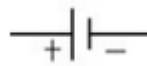
Arus yang bergerak secara konstan dalam satu arah disebut arus searah, karena listrik mengalir hanya dalam satu arah. Arus searah sering disebut arus DC. Banyak sumber tenaga listrik, seperti bendungan pembangkit listrik tenaga air yang ditunjukkan di awal bab ini, menghasilkan arus bolak-balik. Bolak-balik, di mana arah arus bolak-balik. Arus bolak-balik sering disebut arus AC. Bolak-balik bolak-balik pada interval waktu yang teratur. Arus bolak-balik yang berasal dari stopkontak-stopkontak normal tidak secara tiba-tiba berganti arah. Sebaliknya, arus ini meningkat dengan lancar hingga mencapai arus maksimum dan kemudian dengan lancar menurun kembali ke nol. Kemudian bertambah lagi, tetapi dalam arah yang berlawanan sampai mencapai maksimum yang sama nilai. Setelah itu, arus akan menurun secara mulus kembali ke nol, dan siklus dimulai lagi[17].

### 2.1.2 Tegangan

Tegangan atau *Voltage* adalah kerja yang dilakukan untuk menggerakkan satu muatan (sebesar satu *coulomb*) pada elemen atau komponen dari suatu terminal/kutub lainnya, atau pada kedua terminal/kutub akan mempunyai beda potensial jika kita menggerakkan/memindahkan muatan sebesar satu coulomb dari satu terminal ke terminal lainnya. Keterkaitan antara kerja yang dilakukan sebenarnya adalah energi yang

dikeluarkan, sehingga pengertian diatas dapat dipersingkat bahwa tegangan adalah energi per satuan muatan

Sebuah lampu yang dihubungkan dengan sumber potensial listrik (sumber tegangan) berupa baterai. Selain baterai, sumber tegangan juga dapat dihasilkan oleh aki, sel surya dan lain-lain. Perbedaan potensial pada titik yang berbeda dapat terjadi apabila pada rangkaian dipasang sumber potensial listrik yang dikenal juga dengan istilah GGL (gaya gerak listrik). Arus listrik akan mengalir dari dari titik kutub positif ke kutub negatif [16]. Sumber tegangan direpresentasikan pada Gambar 2.1 berikut



Gambar 2. 1 Sumber Tegangan [16]

Tegangan adalah perkalian antara arus listrik dengan hambatan listrik terdapat pada persamaan 2-4 :

$$V = I \times R \quad (2-4)[17]$$

Keterangan :

V = Tegangan (Volt)

I = Arus (Ampere)

R = Resistansi (Ohm)

### 2.1.3 Daya Listrik

Daya Listrik adalah jumlah energi yang diserap atau dihasilkan dalam sebuah sirkuit/rangkaian. Sumber energi seperti tegangan listrik akan menghasilkan daya listrik sedangkan beban yang terhubung dengannya akan menyerap daya listrik tersebut. Dengan kata lain, daya listrik adalah tingkat konsumsi energi dalam sebuah sirkuit atau rangkaian listrik [18]. Daya listrik adalah energi listrik tiap waktu. Persamaan besar daya listrik dapat dilihat pada persamaan 2.5 [16].

$$P = \frac{W}{t} = V \times I \quad (2-5)$$

Keterangan :

P = Daya listrik (Watt)

W = Energi listrik (Joule)

V = Tegangan (Volt)

I = Arus listrik (Ampere)

t = waktu (sekon)

Daya listrik ada 3 macam yaitu daya nyata (P), daya reaktif (Q), dan daya semu (S).

#### 1. Daya Nyata

Daya nyata atau daya aktif digunakan secara umum oleh konsumen. Satuan daya nyata dinyatakan dalam Watt.

#### 2. Daya Reaktif

Daya reaktif adalah daya yang mengakibatkan terjadinya kerugian-kerugian daya, atau daya yang mengakibatkan terjadinya penurunan nilai faktor daya ( $\cos \phi$ ). Satuan daya reaktif adalah VAR.

#### 3. Daya Semu

Daya semu adalah daya yang dihasilkan dari perhitungan-perhitungan listrik sebelum dibebani dengan beban listrik. Satuan daya semu adalah VA.

### 2.1.4 Beban Listrik

Beban listrik adalah jumlah daya listrik yang dikonsumsi oleh suatu sistem atau peralatan listrik pada suatu waktu tertentu. Beban daya listrik dinyatakan dalam satuan Watt (W) atau kiloWatt (kW) dan dapat bervariasi tergantung pada jenis peralatan listrik dan penggunaan yang dilakukan [19]. Pada kondisi beban seimbang jumlah daya yang dibangkitkan oleh generator tiga fasa atau daya yang dikonsumsi oleh beban tiga fasa, diperoleh dengan menjumlahkan daya dari setiap fasa. Dalam sistem yang seimbang, dayanya sama untuk setiap fasa, sehingga daya totalnya sama dengan tiga kali daya fasa.



Secara umum daya listrik untuk alat listrik rumah tangga menggunakan satuan Watt. Nilai 1kW adalah 1000W dan 1w adalah 1/1000. Energi adalah daya listrik yang diserap oleh beban selama selang waktu tertentu. Perhitungan Energi (kWh) ditunjukkan pada persamaan 2-6.

$$kWh = kW \times t \quad (2-6)$$

Satuan kWh adalah energi, kW adalah daya dan t adalah waktu (jam). Rumus konversi dari kWh ke satuan rupiah dengan persamaan 2-7.

$$\text{Pemakaian Listrik}(Rp) = \text{Jumlah Pemakaian}(kWh) \times \text{Harga}(Rp) \text{ per kWh} \quad (2-7)$$

## 2.2 *Internet of Things (IoT)*

*Internet of Things* atau sering disebut IoT adalah sebuah gagasan dimana semua benda di dunia nyata dapat berkomunikasi satu dengan yang lain sebagai bagian dari satu kesatuan sistem terpadu menggunakan jaringan internet sebagai penghubung. IoT sering disebut sebagai "*internet of everything*," sebuah konsep yang menghubungkan dunia fisik dan digital melalui berbagai teknologi seperti sensor, komputer, internet, identifikasi frekuensi radio (RFID), *smart grid*, dan teknologi komunikasi [20]. IoT terdiri dari berbagai hal yang saling terhubung yang mampu mendeteksi, menggerakkan, dan berkomunikasi di antara mereka sendiri dan lingkungan (seperti *smart things* atau *smart objects*) sambil memberikan kemampuan untuk berbagi informasi dan bertindak secara otomatis terhadap kejadian dunia nyata/fisik dan dengan memicu proses dan menciptakan layanan dengan atau tanpa campur tangan manusia secara langsung [21].

Cara kerja IoT mengacu pada 3 elemen utama, yakni: barang fisik yang dilengkapi modul IoT, perangkat koneksi ke internet seperti modem dan *router wireless speedy* dan *Cloud Data Center* tempat untuk menyimpan aplikasi beserta *database*. Dalam perkembangannya sebuah benda dapat diberi pengenal berupa *IP address* dan menggunakan jaringan internet untuk bisa berkomunikasi dengan benda lain yang memiliki pengenal *IP address*. *Internet of Things* memanfaatkan sebuah argumentasi pemrograman yang dimana tiap-tiap perintah argumennya itu menghasilkan sebuah interaksi antara sesama mesin yang terhubung secara otomatis tanpa campur tangan manusia dan dalam jarak berapa pun. Internet

yang menjadi penghubung di antara kedua interaksi mesin tersebut, sementara manusia hanya bertugas sebagai pengatur dan pengawas bekerjanya alat tersebut secara langsung. Secara sederhana mikrokontroler dapat diartikan sebagai suatu sistem komputer yang dikemas dalam IC, dimana sebelum digunakan harus diisi suatu program atau perintah terlebih dahulu sehingga mikrokontroler hanya dapat berjalan bila telah diisi suatu perintah atau program terlebih dahulu.

Perangkat IoT yang digunakan pada penelitian skripsi ini ialah Wemos D1 R1 yang dimanfaatkan sebagai mikrokontroler dan pusat pengolahan data. Data tersebut kemudian diolah agar dapat ditampilkan dengan parameter tegangan, arus, Watt, kWh dan biaya pada layar LCD. Wemos D1 R1 sudah berbasis ESP8266 sebagai modul Wi-Fi dan CH340 USB *to serial interface*, selain itu mikrokontroler ini juga sudah kompatibel dengan beragam Arduino. Wemos D1 R1 sama halnya seperti arduino yang merupakan platform pembuatan *prototype* elektronik yang bersifat *opensource hardware* yang berdasarkan pada perangkat keras dan perangkat lunak yang fleksibel dan mudah digunakan. Wemos D1 R1 terdiri dari *Board* arduino, *shield*, bahasa pemrograman, dan arduino *Integrated Development Environment* (IDE). *Board* Wemos D1 R1 merupakan *board* yang dikembangkan berdasarkan ESP8266 yang merupakan IC komunikasi Wi-Fi yang dirancang menyerupai Arduino Uno. *Shield* adalah sebuah papan yang dapat dipasang di atas *Board* arduino untuk menambah kemampuan dari *Board* Wemos D1 R1. Bahasa pemrograman Wemos D1 R1 adalah Bahasa pemrograman yang digunakan untuk membuat perangkat lunak yang ditanamkan pada *Board* Wemos D1 R1. *Board* Wemos D1 R1 terdiri dari *hardware*/modul mikrokontroler yang siap pakai dan *software* IDE yang digunakan untuk memprogram sehingga kita bisa belajar dengan mudah. Kelebihan dari Wemos D1 R1 yaitu kita tidak direpotkan dengan rangkaian minimum sistem dan programmer karena sudah built in dalam satu *board* dan dari sisi spesifikasi, sebenarnya jauh lebih unggul Wemos D1 R1, salah satunya dikarenakan inti dari Wemos D1 R1 adalah ESP8266EX yang memiliki prosesor 32 bit.

### 2.2.1 Wemos D1 R1

Mikrokontroler adalah sebuah perangkat komputasi kecil yang terdiri dari sebuah IC dengan *microprocessor*, *Read Only Memory* (ROM) dan *Random Access Memory* (RAM) yang dapat mengatur input/output dari alat lain. Mikrokontroler Wemos D1 R1 sudah berbasis ESP8266 yaitu modul Wi-Fi, CH340 USB *to serial interface* dan selain itu mikrokontroler ini juga sudah kompatibel dengan beragam Arduino *shield*. Sistem ini juga berbasis konsep *Internet of Things* (IoT), pada sistem ini berfungsi untuk pengiriman data antar perangkat menggunakan jaringan internet.

Wemos D1 R1 merupakan *board* yang menggunakan ESP8266EX sebagai modul Wi-Fi dan dirancang menyerupai Arduino Uno. Kelebihan dari Wemos D1 R1 ini adalah bersifat *open source*, kompatibel dengan Arduino, dapat diprogram menggunakan Arduino IDE, *pinout* yang kompatibel dengan Arduino Uno, dapat berdiri sendiri tanpa menggunakan mikrokontroler lain, memiliki prosesor 32-bit dengan kecepatan 80 MHz, *High Level Language*. Berikut ini bentuk fisik dari Wemos D1 R1 dapat dilihat pada Gambar 2.2.



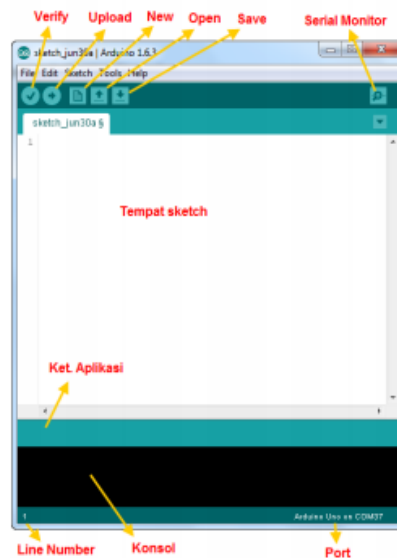
Gambar 2. 2 Wemos D1 R1 Board [22].

Adapun spesifikasi Wemos D1 R1, ialah sebagai berikut[22]:

- |                                   |                            |
|-----------------------------------|----------------------------|
| a. <i>Microcontroller</i>         | : ESP8266 Tensilica 32-bit |
| b. <i>Serial to USB Converter</i> | : CH340G                   |
| c. <i>Operating Voltage</i>       | : 3.3V                     |
| d. <i>Input Voltage</i>           | : 7-12V                    |

- e. *Digital I/O Pins* : 11
- f. *PWM I/O Pins (Shared with Digital I/O)* : 10
- g. *Analog Input Pins* : 1 (10-bit)
- h. *DC Current per I/O Pin* : 12mA (Max)
- i. *Hardware Serial Ports* : 1
- j. *Flash Memory* : 4 Mbytes
- k. *Instruction RAM* : 64 Kbytes
- l. *Data RAM* : 96 Kbytes
- m. *Clock Speed* : 80MHz
- n. *Network* : IEEE 802.11 b/g/n WiF
- o. *Built-in LED* : Attached to digital pin 13
- p. *USB Connector Style* : Micro-B Female
- q. *Board Dimensions (PCB)* : 69 x 53mm (2.7 x 2.1")
- r. *Datasheet* : ESP8266EX

Dalam mengoperasikan arduino harus menggunakan *software* untuk memasukkan program, program yang digunakan pada arduino adalah bahasa C. Berikut fungsi dan penampakan *software* Arduino IDE dilihat pada gambar 2.3 dibawah ini:



Gambar 2. 3 *Software* Arduino IDE [23]

Untuk memprogram *Board* Wemos D1R1, dibutuhkan aplikasi Arduino IDE. Aplikasi ini berguna untuk membuat, membuka, dan mengedit *source code* Arduino atau dapat juga disebut *Sketches*. *Sketch* merupakan *source code* yang berisi logika dan algoritma yang akan diupload ke dalam IC mikrokontroler. *Interface* Arduino IDE tampak seperti gambar 2.2. Dari kiri ke kanan dan atas ke bawah, bagian-bagian IDE Arduino terdiri dari [23]:

- a. *Verify*: versi sebelumnya dikenal dengan istilah *Compile*. Sebelum aplikasi diupload 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.
- b. *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.
- c. *New Sketch*: Membuka *window* dan membuat *sketch* baru
- d. *Open Sketch*: Membuka *sketch* yang sudah pernah dibuat. *Sketch* yang dibuat dengan Arduino IDE akan disimpan dengan ekstensi file.ino
- e. *Save Sketch*: menyimpan *sketch*, tapi tidak disertai meng-*compile*.
- f. *Serial Monitor*: Membuka *interface* untuk komunikasi serial
- g. *Keterangan Aplikasi*: pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal "*Compiling*" dan "*Done Uploading*" ketika meng-*compile* dan mengupload *sketch* ke *board* Wemos D1R1
- h. *Konsol*: Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang *sketch* akan muncul pada bagian ini, seperti ketika aplikasi meng-*compile* atau ketika ada kesalahan pada *sketch* yang kita buat, maka informasi *error* dan baris akan diinformasikan di bagian ini.
- i. *Baris Sketch*: bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada *sketch*.

- j. Informasi *Port*: bagian ini menginformasikan *port* yang dipakai oleh *Board* Wemos D1R1.

### 2.2.2 Modul ESP8266 Port Wi-Fi

Pada Wemos D1R1 tertanam modul ESP8266. ESP8266 adalah sebuah chip yang sudah lengkap dimana di dalamnya sudah termasuk *processor*, memori dan juga akses ke GPIO. Hal ini menyebabkan ESP8266 dapat secara langsung menggantikan Arduino dan ditambah lagi dengan kemampuannya untuk mensupport koneksi Wi-Fi secara langsung.

IoT (*Internet of Things*) semakin berkembang seiring dengan perkembangan mikrokontroler, module yang berbasis Ethernet maupun Wi-Fi semakin banyak dan beragam dimulai dari Wiznet, Ethernet shield hingga yang terbaru adalah Wi-Fi module yang dikenal dengan ESP8266. Ada beberapa jenis ESP8266 yang dapat ditemui dipasaran, namun yang paling mudah didapatkan di Indonesia adalah type ESP-01,07 dan 12 dengan fungsi yang sama perbedaannya terletak pada GPIO pin yang disediakan. Berikut beberapa tipe ESP8266. Gambar 2.4 berikut merupakan bentuk beberapa tipe dari ESP8266.



Gambar 2. 4 Jenis-Jenis ESP8266 [24]

### 2.2.3 Modul PZEM-004T

Modul PZEM-004T adalah sebuah modul sensor multifungsi yang berfungsi sebagai pengukur daya AC listrik, mampu mengukur (arus, tegangan, daya aktif, dan energi), mampu menyimpan energi yang diukur sebelum daya mati, dan berisi fungsi komunikasi serial TTL (*Transistor-Transistor Total Consumption of Logic*) sehingga modul mampu untuk berkomunikasi dengan mikrokontroler yang berbeda. Sensor ini mampu mendeteksi rentang tegangan AC dari 80-260 V AC, frekuensi operasi 45-65 Hz, dan rentang yang dapat diterima untuk arus pengukuran adalah 0-100A AC [25]. Gambar 2. 5 menunjukkan gambar modul sensor PZEM004T. Modul ini sudah dilengkapi sensor tegangan dan sensor arus (CT) yang sudah terintegrasi. Dalam penggunaannya, alat ini khusus untuk penggunaan dalam ruangan (*indoor*) dan beban yang terpasang tidak diperbolehkan melebihi daya yang sudah ditetapkan.



Gambar 2. 5 Modul PZEM-004T [26]

Prinsip kerja CT pada modul PZEM-004T adalah sebatang penghantar yang dialiri arus yang dilewatkan melalui cincin toroid maka akan menimbulkan medan magnet, sehingga memiliki fluks magnet yang melingkar kemudian ditangkap oleh lilitan [27]. CT tersebut terdiri dari kumparan sekunder dan kumparan primer yang dililitkan pada suatu inti magnet. Arus yang akan dideteksi dialirkan ke kumparan primer. Arus ini menghasilkan medan magnet yang mengalir ke kumparan sekunder. Inti magnetik pada sensor berfungsi membuat agar fluks magnet yang dihasilkan oleh kumparan primer menembus kumparan sekunder. Perubahan fluks yang dihasilkan oleh arus primer mengakibatkan timbulnya tegangan listrik induksi pada kumparan sekunder. CT

memiliki tahanan beban internal, yang berfungsi untuk menghasilkan keluaran dari sensor berupa tegangan.

### 2.3 Sistem Kendali

Sistem kendali adalah sistem yang dirancang untuk mempertahankan atau mengubah variabel sistem sesuai dengan keinginan. Variabel sistem yang dikendalikan disebut sebagai variabel *output*. Variabel sistem yang digunakan untuk mengendalikan disebut sebagai variabel *input*. Dalam sistem kendali dua posisi atau *on/off* elemen penggerak hanya memiliki dua posisi tetap, yang dalam banyak kasus, hanya hidup dan mati. Kendali *on/off* relatif sederhana dan murah sehingga sangat banyak digunakan dalam sistem kendali rumah tangga dan industri.

Sinyal digital merupakan sinyal yang memiliki nilai berbentuk digital, yaitu hanya 0 dan 1. Sinyal digital dapat berupa gelombang pulsa atau kode. Sinyal digital digunakan untuk mempelajari berbagai aspek dari sinyal digital, mulai dari representasi, pengolahan, transmisi, hingga penyimpanan. Sinyal digital dapat diaplikasikan ke dalam banyak hal, antara lain:

- a. Komunikasi digital, seperti transmisi data, suara, dan gambar
- b. Pengolahan sinyal digital, seperti kompresi data, pengenalan pola, dan filter digital
- c. Komputer digital, seperti aritmatika biner dan representasi data
- d. Sistem digital, seperti sistem kendali digital dan sistem sinyal

Sistem bilangan biner adalah sistem yang menggunakan dua bit, yaitu 0 dan 1. Sistem bilangan biner merupakan dasar dari semua sistem bilangan berbasis digital. Nilai bilangan biner ditentukan oleh posisi bitnya. Bit yang paling kanan disebut bit paling tinggi, sedangkan posisi bit paling kiri disebut bit paling rendah. Dalam sistem bilangan biner, bit 0 dan bit 1 mewakili dua nilai yang berbeda. Bit 0 mewakili nilai *false* atau tidak ada. Sedangkan bit 1 mewakili nilai *true* atau ada.



## 2.4 *Google Sheets*

*Google Sheets* terintegrasi dalam ekosistem *Google Document*. Salah satu keunggulannya adalah kemudahan aksesibilitas, yang memungkinkan pengguna untuk mengakses dan mengelola data mereka dari berbagai lokasi melalui penyimpanan berbasis *cloud Google*[13]. Keunggulan ini membuka peluang bagi mobilitas pengguna yang sering berpindah tempat kerja atau memiliki kebutuhan untuk mengakses data dari berbagai perangkat yang berbeda. Dengan penyimpanan yang otomatis pengguna dapat dengan mudah berpindah antara aplikasi, berbagi dokumen dengan pengguna lain dan menggunakan berbagai alat ini secara bersamaan untuk meningkatkan produktivitas, terutama dalam hal proyek yang melibatkan berbagai jenis data atau informasi. Pengguna dapat mengedit dan melihat dokumen secara bersamaan, meningkatkan produktivitas dan kerja sama.

## 2.5 *Bot Telegram*

Telegram adalah aplikasi pesan yang dapat berjalan pada system operasi android dan windows. Seperti *Whatsapp*, telegram juga memiliki sistem keamanan *end-to end* sehingga memberikan jaminan kemanan yang tinggi kepada *user*. Telegram telah diunduh lebih dari 1 miliar *user* di *google playstore* oleh seluruh pengguna *smartphone* di seluruh dunia.

*Bot* telegram adalah aplikasi pihak ketiga yang berjalan pada aplikasi Telegram. Fiturnya yang fleksibel dan mudah digunakan membuatnya menjadi pilihan yang tepat bagi pengembang dan pengguna. Pengguna dapat berinteraksi dengan *bot* dengan mengirimkan pesan berupa perintah [11]. Beberapa contoh penggunaan *Bot Telegram* antara lain:

- a. *Bot customer service*: Bot dapat digunakan untuk memberikan dukungan pelanggan dan memproses permintaan pelanggan secara otomatis.
- b. Bot untuk keperluan bisnis: Bot dapat digunakan untuk mengirimkan notifikasi tentang perubahan harga, mengirimkan daftar produk, dan membantu dalam proses penjualan.

- c. Bot untuk keperluan pribadi: Bot dapat digunakan untuk membantu mengatur jadwal, mengingatkan tentang janji, dan memberikan informasi tentang cuaca atau berita terbaru [28].

Cara kerja *bot* secara umum yaitu pertama user mengirimkan pesan kepada *bot*. *Bot* tidak diijinkan untuk mengirim pesan terlebih dahulu untuk mencegah *spamming*. Setelah menerima pesan, *bot* akan menyimpan data pesan yang sudah dimasukkan oleh user dan *bot* akan mengirimkan ke *engine* yang dituju. Setelah itu, *engine* akan memproses pesan yang sudah dikirimkan melalui *bot* dan menyiapkan jawaban yang nantinya akan dikirimkan oleh *bot* tersebut. *Bot* menerima pesan dari *engine* dan akan mengirimkan balasan atau respon kepada pengguna.

## 2.6 Kajian Pustaka

Rancang Bangun Monitoring Penggunaan Energi Perangkat Listrik Rumah Via Telegram merupakan suatu sistem yang memudahkan pengguna dalam memantau konsumsi listrik dan mengendalikan perangkat listrik dari jarak jauh melalui *bot Telegram*. Berikut adalah beberapa kajian pustaka terkait rancang bangun monitoring penggunaan energi perangkat listrik rumah via telegram:

1. Penelitian [2] ini mengusulkan sistem analitik energi *real-time* untuk bangunan pintar yang menggunakan platform IoT terbuka. Sistem ini terdiri dari tiga bagian utama: lapisan akuisisi, lapisan transportasi, dan lapisan aplikasi. Lapisan akuisisi mengumpulkan data listrik dari sensor dan mengirimkannya ke lapisan transportasi. Lapisan transportasi mengirimkan data ke lapisan aplikasi, yang memvisualisasikan dan menganalisis data. Sistem ini diuji coba di sebuah gedung kantor dan menunjukkan kinerja yang baik. Sistem ini dapat membantu manajer gedung untuk mengendalikan dan menjaga kualitas daya. Perangkat yang digunakan adalah PZEM-004T V3.0 dan mikrokontroler ESP8266. Data energi listrik yang dikumpulkan divisualisasikan menggunakan ChartJS.

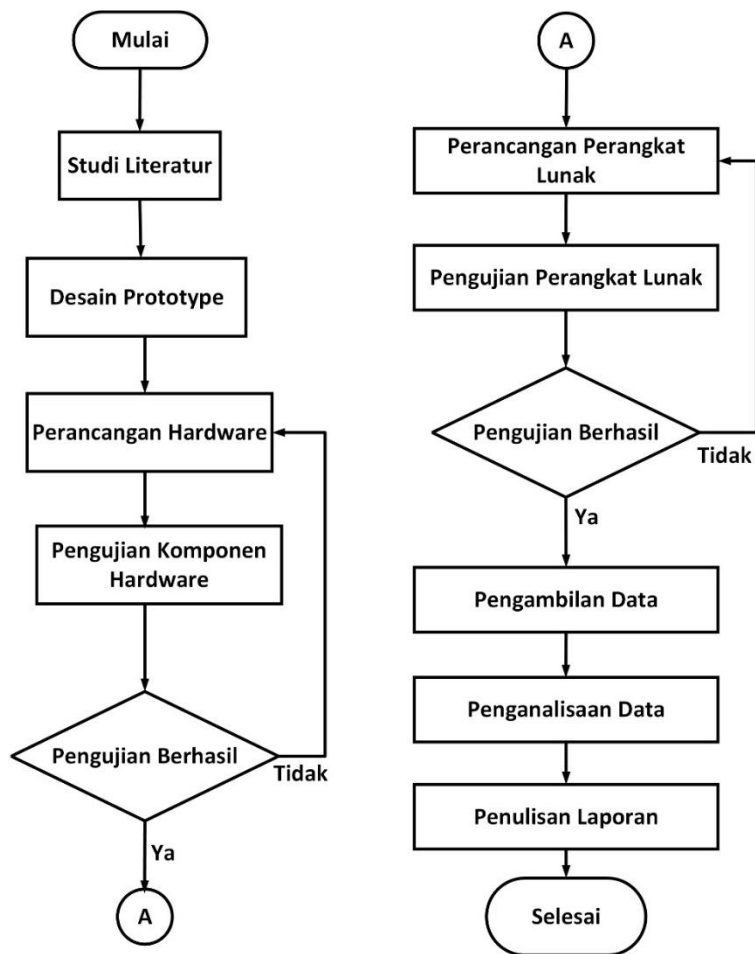
2. Penelitian [4] ini bertujuan untuk memeriksa kondisi mekanis dan alat pendukung kWh meter, serta melakukan pemanasan pada kWh meter sebelum dilakukan peneraan. Tujuan lainnya adalah mengukur kesalahan pada kWh meter, mengevaluasi akurasi dan ketelitiannya, membandingkan hasil peneraan dengan standar kWh meter induk, dan mengidentifikasi faktor-faktor yang dapat mempengaruhi kesalahan pengukuran kWh meter. Penelitian ini menunjukkan kesalahan pada peneraan berkisar antara 0 hingga 2,5%. Dalam pelaksanaan peneraan, perlu diperhatikan beberapa hal, seperti mengatur rangkaian arus dan tegangan agar sesuai dengan alat pendukungnya, serta melakukan pemanasan pada alat ukur kWh meter.
3. Penelitian [6] bertujuan untuk memonitoring sistem listrik berbasis jaringan sensor nirkabel (WSN) yang memantau dan mengendalikan variabel listrik berbasis IoT otomatis yang dapat mendeteksi suhu dan kelembapan, serta pencahayaan, konsumsi listrik, dan gangguan kabel. Dalam sistem pemantauan listrik berbasis IoT, mikrokontroler yang pada penelitian ini menggunakan NodeMCU yang telah berbasis ESP8266, yang memiliki kemampuan untuk terhubung ke internet melalui Wi-Fi. Penelitian ini menggunakan beberapa sensor yaitu sensor api dan cahaya, sensor DHT11, sensor ACS712 dan tegangan DC. Data yang dikumpulkan oleh sensor-sensor ditampilkan secara *real-time* pada halaman web dan disimpan dalam *database* server lokal.
4. Penelitian [10] merancang sistem monitoring dan kendali perangkat listrik berbasis IoT menggunakan NodeMCU ESP8266. Penelitian ini mengendalikan beberapa perangkat elektronik pada ruang kerja melalui internet, seperti lampu, kipas angin, pengunci laci, stop kontak, dan pengecekan suhu. Penelitian ini dilakukan dengan membangun sebuah perangkat kendali jarak jauh dengan ESP8266 untuk mengakses kendali lokal dengan bantuan *Artificial Intelligence Chatbot* dengan menggunakan *Telegram Messenger*. Pengujian ini dilakukan untuk memudahkan karyawan dalam mengendalikan perangkat elektronik pada ruang kerja mereka melalui smartphone atau PC, tanpa harus berpindah-pindah secara manual.

5. Penelitian yang dilakukan oleh [8] bertujuan mengkaji penerapan teknologi *Internet of Things* (IoT) untuk mengendalikan alat elektornik dan memantau daya listrik terpakai pada alat tersebut dari jarak jauh melalui Internet. Sistem diimplementasikan menggunakan Wemos D1, sensor arus ACS712, relay, dan aplikasi *Blynk* sebagai antarmuka sistem di smartphone. Penelitian ini dapat mengendalikan alat elektronik yang terpasang pada alat dan dapat memantau daya listrik yang terpakai tiap detiknya melalui aplikasi *Blynk*.
6. Penelitian ini [11] mengembangkan sistem pemantauan penggunaan listrik rumah tangga secara *real-time* berbasis web. Penelitian ini melibatkan pengembangan perangkat keras dan konfigurasi kabel untuk sistem pemantauan. Penelitian menggunakan sensor PZEM-004T untuk mengukur konsumsi listrik peralatan rumah tangga dan modul NodeMCU ESP8266 untuk mengirimkan data *website*.

## BAB III METODOLOGI PENELITIAN

### 3.1 Metode Penelitian

Metode penelitian pada skripsi ini digambarkan dengan diagram alir pada Gambar 3.1. Diagram alir penelitian berfungsi sebagai panduan untuk memastikan kesinambungan dan keterhubungan setiap tahapannya.



Gambar 3. 1 Alur Penelitian

Berdasarkan Gambar 3.1 penelitian skripsi ini diawali dengan studi literatur dengan mengumpulkan beberapa referensi untuk memahami kerangka kerja penelitian. Setelah

pemahaman konseptual diperoleh maka dilanjutkan dengan menyusun *prototype* alat monitoring kendali. Susunan ini meliputi identifikasi dan penentuan komponen-komponen yang dibutuhkan dalam penyusunan alat monitoring kendali. Perancangan *hardware* dengan merakit komponen-komponen sesuai rancangan yang telah dibuat sebelumnya. Pengujian komponen dilakukan untuk memastikan bahwa setiap modul dapat berfungsi dengan baik. Perancangan *software* dengan melakukan instalasi program sistem yang sesuai dengan kebutuhan alat monitoring kendali. Pengujian *software* dilakukan untuk menjamin bahwa antara sistem dan alat dapat terintegrasi dengan baik sesuai yang diinginkan.

Selanjutnya melakukan pengambilan data, dengan alat monitoring kendali diimplementasikan untuk mengumpulkan informasi terkait konsumsi energi listrik pada ruang lingkup rumah tangga. Data yang terkumpul kemudian dianalisis. Proses akhir metode penelitian yaitu penulisan laporan dengan hasil analisis yang disusun secara sistematis.

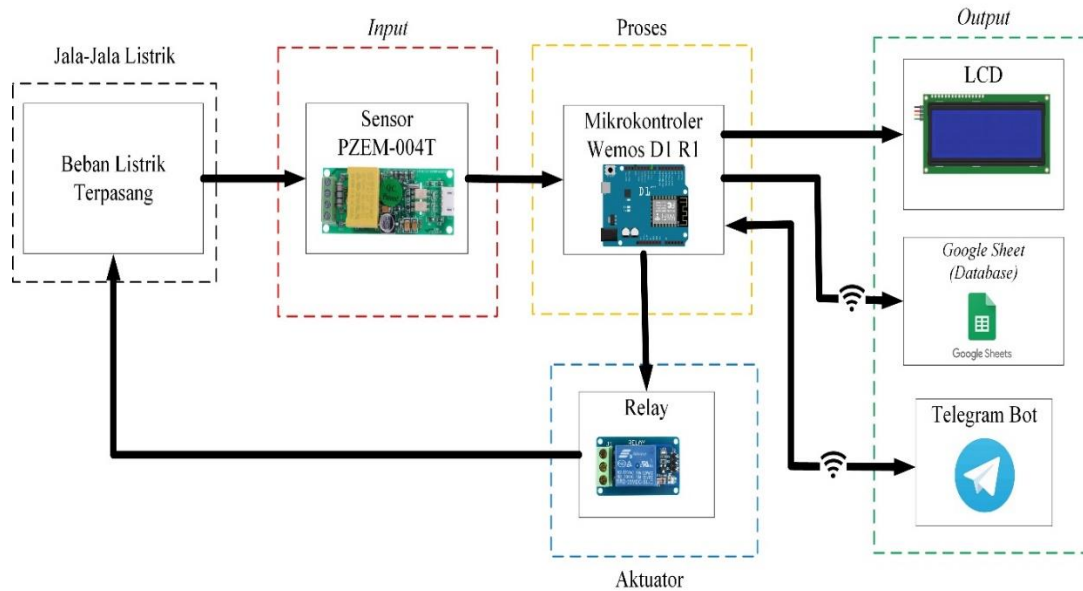
### **3.2 Instrumen Penelitian**

Perangkat keras, perangkat lunak dan fasilitas penunjang dalam penelitian ini adalah:

1. Aduino IDE
2. Visio sebagai perangkat lunak yang digunakan untuk menyusun rangkaian gambar komponen alat monitoring kendali.
3. Wemos D1 R1 berfungsi sebagai mikrokontroler.
4. Sensor PZEM-004T berfungsi sebagai input Wemos D1R1
5. Relay
6. LCD 20×4 dengan I2C berfungsi sebagai output tampilan data monitoring.
7. Stop kontak
8. Multimeter/tang amper
9. Beban listrik yang termasuk berupa peralatan listrik rumah tangga

### 3.3 Diagram Blok Sistem

Diagram blok sistem dibutuhkan untuk mempermudah memahami alur kerja dari sistem yang dirancang.



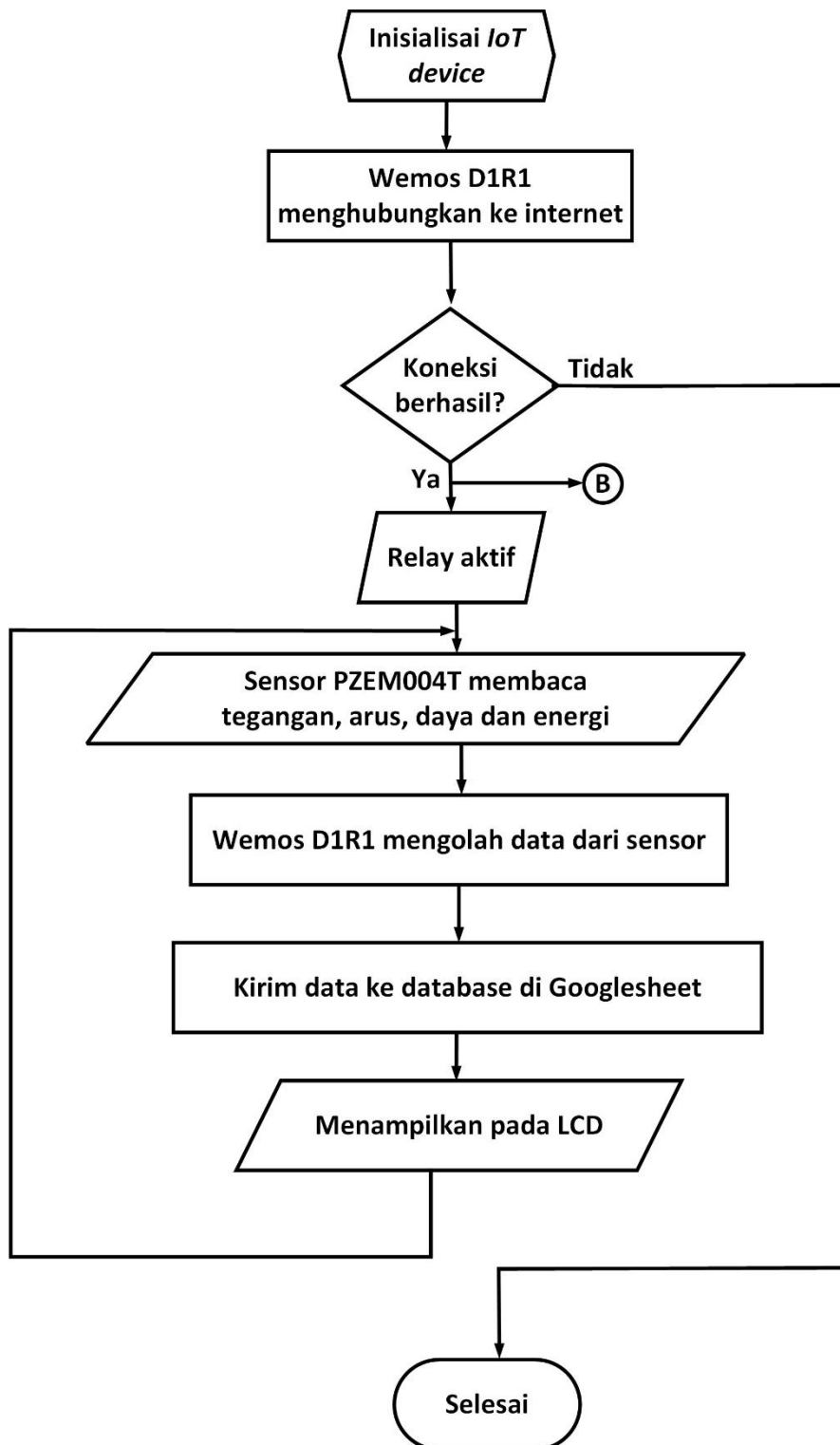
Gambar 3. 2 Diagram Blok Sistem

Cara kerja dari diagram blok berdasarkan Gambar 3.2 yaitu sensor PZEM004T sebagai sensor yang membaca nilai tegangan serta arus dari beban listrik terpasang yang telah dialiri tegangan dan arus dari jala-jala listrik rumah kemudian dilakukan proses konversi nilai secara *hardware* dan *software* (di dalam Wemos D1 R1 sebagai pusat kendali) sehingga menghasilkan nilai-nilai tegangan, arus, daya, kWh dan biaya konsumsi energi listrik yang ditampilkan pada layar LCD. Setelah data aktual didapatkan selanjutnya ESP8266 yang terdapat pada modul Wemos D1R1 mengirim dan meznympan data ke *Google Sheets* yang difungsikan sebagai *database* dan data monitoring dapat diakses melalui *bot Telegram*, untuk mengendalikan beban listrik yang berupa perangkat listrik rumah tangga *Bot Telegram* menyediakan pesan perintah *ON/OFF* untuk relay. Komunikasi antara alat monitoring kendali, *Telegram* dan *Google Sheets* dibutuhkan jaringan internet yang stabil.

### 3.4 Perancangan Perangkat Keras (*Hardware*)

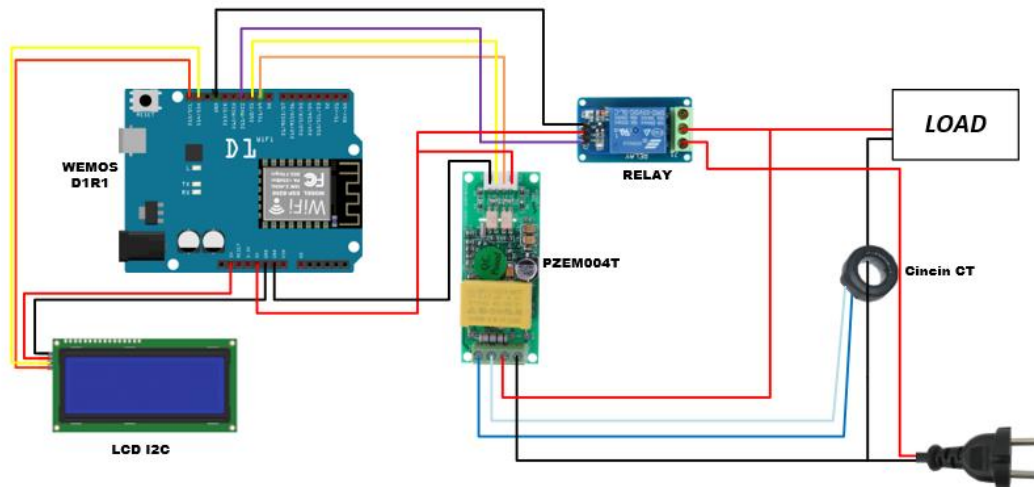
Perangkat keras (*Hardware*) untuk sistem monitoring dan kendali pada alat ini terdiri dari modul Wemos D1 R1, sensor PZEM004T, relay, LCD I2C dan stop kontak. Wemos D1 R1 berfungsi sebagai mikrokontroler, PZEM004T berfungsi sebagai sensor yang mengukur nilai tegangan, arus, daya, dan kWh, LCD I2C berfungsi menampilkan nilai tegangan, arus, daya, kWh serta biaya konsumsi listrik, relay sebagai kendali jarak jauh dan stop kontak berfungsi untuk menghubungkan alat ke beban listrik. Alat ini menggunakan *Bot Telegram* untuk mengirim perintah ke modul Wemos D1R1. Perintah tersebut kemudian diteruskan ke relay untuk menghidupkan atau mematikan arus yang mengalir ke beban listrik melalui stop kontak. Alat ini menggunakan *Google Sheets* untuk menyimpan data monitoring secara *realtime*. Data monitoring dapat ditampilkan secara *realtime* melalui *Google Sheets* dan *Bot Telegram*, untuk *Bot Telegram* data yang ditampilkan sesuai kode perintah yang dikirimkan.





Gambar 3. 3 *Flowchart* Sistem Hardware

*Flowchart* pada Gambar 3.3 menjelaskan tentang alur kerja alat secara *hardware*. Perangkat Wemos D1R1 melakukan inialisasi data untuk sensor PZEM004T. Data yang diinisialisasi mencakup tegangan, arus, daya, kWh, dan biaya terkait penggunaan energi listrik. Sensor PZEM004T merupakan modul sensor yang mengukur tegangan dan arus listrik. Data dari sensor ini menjadi dasar penelitian ini. Selanjutnya, data yang terkumpul diproses dan dikirim ke *database* di *Google Sheets* dan layar LCD. *Google Sheets* digunakan untuk penyimpanan data jangka panjang, memungkinkan analisis pola konsumsi energi. Sementara itu, layar LCD memberikan tampilan *realtime* terhadap pengguna terkait konsumsi listrik seperti arus, tegangan, daya, kWh dan biaya. Pengguna dapat mengakses data melalui *Bot Telegram*, yang berfungsi sebagai antarmuka komunikasi. Pengguna dapat memantau dan mengendalikan penggunaan beban listrik terhubung melalui pesan perintah yang dikirim ke *Bot Telegram*.



Gambar 3. 4 Rangkaian Komponen Alat Monitoring Kendali

Gambar 3.4 merupakan susunan rangkaian komponen dan modul alat monitoring kendali dengan Wemos D1 R1 sebagai mikrokontrolernya. Berdasarkan Gambar 3.4 pin 5V pada Wemos D1 R1 digunakan untuk menghubungkan sumber tegangan 5V ke modul lainnya yaitu PZEM004T, relay dan LCD 20x4 I2C, sedangkan pin GND digunakan untuk menghubungkan ground ke modul lainnya. Komunikasi serial antara mikrokontroler dan

sensor dengan menghubungkan pin D11 pada Wemos D1R1 dengan pin RX PZEM004T dan pin D12 pada Wemos D1R1 dihubungkan dengan pin TX PZEM004T. Wemos D1R1 dapat mengendalikan relay melalui pin D13. Ketika pin D13 *HIGH*, relay akan *ON*, dan ketika *LOW*, relay akan *OFF*. Data monitoring kemudian ditampilkan pada LCD dengan menghubungkan Wemos D1R1 melalui pin D14 ke SDA (Serial Data) LCD I2C dan pin D15 ke SCL (Serial Clock) LCD I2C. cincin/*clamp* CT berperan sebagai sensor arus. Steker pada susunan rangkaian berfungsi untuk menghubungkan sumber tegangan (jala-jala listrik) ke PZEM004T sehingga nilai tegangan yang akan terbaca oleh sensor. Steker juga menyalurkan arus listrik ke stop kontak. Pemasangan perangkat listrik rumah tangga sebagai beban listrik dapat melalui stop kontak seperti pada Gambar 3.4, sehingga beban listrik yang terpasang pada stop kontak dapat dikendalikan melalui bot telegram.

### 3.5 Perancangan Perangkat Lunak (*Software*)

Pada tahap ini akan dilakukan penginstalan program agar antara *hardware* dan *software* saling menginisialisasi yang akan membuat keseluruhan perangkat saling terintegrasi, *Software* yang digunakan yaitu Arduino IDE sebagai *software* untuk menjalankan program arduino dan Visio untuk perancangan *prototype*, instalasi *Google Sheets* sebagai *database*, Microsoft *Office* (*Excel* 2016, *Word* 2016, *Visio*) sebagai pengolah dan perhitungan data serta pembuatan laporan dari hasil simulasi dan perhitungan. *Google Sheets* sebagai *database*. Telegram sebagai aplikasi yang digunakan untuk berkomunikasi dengan alat monitoring kendali pada penelitian ini.

Adapun cara program mikrokontroler dengan *software* Arduino IDE. Berikut ini beberapa aturan penulisan program Aduino IDE terutama yang sering digunakan dalam pemograman mikrokontroler:

- a. Pada program utama harus terdapat main rutin yang ditulis dengan nama *main*.
- b. *Statement* di dalam rutin, baik itu *main*, fungsi atau prosedur harus diawali dengan tanda kurung kurawal buka (*{*) dan diakhiri dengan tanda kurung kurawal tutup (*}*).
- c. Setiap *statement* program baik itu perintah, deklarasi variabel atau konstanta harus diakhiri dengan tanda titik koma (*;*).

- d. Komentar program diawali dengan tanda // atau ditulis diantara tanda /\* dan \*/. Komentar program adalah statemen yang tidak ikut dikompile atau tidak dikerjakan oleh mikrokontroler dan tidak terikat dengan aturan sintak yang benar.
- e. Bahasa C mendukung penggunaan preprosesor seperti *include*, *define*, *if*, *ifdef* dan sebagainya. Setiap preprosesor diawali dengan tanda # dan tidak diakhiri dengan tanda titik koma (;).
- f. Untuk deklarasi beberapa variabel sekaligus maka setiap variabel harus dipisahkan dengan tanda koma (,).
- g. Setiap *identifiser* baik itu main rutin, fungsi atau prosedur harus disertai tanda kurung sepasang () pada akhir *identifiser*, misalnya *main* (), *hitung* (), *display* () dan sebagainya.

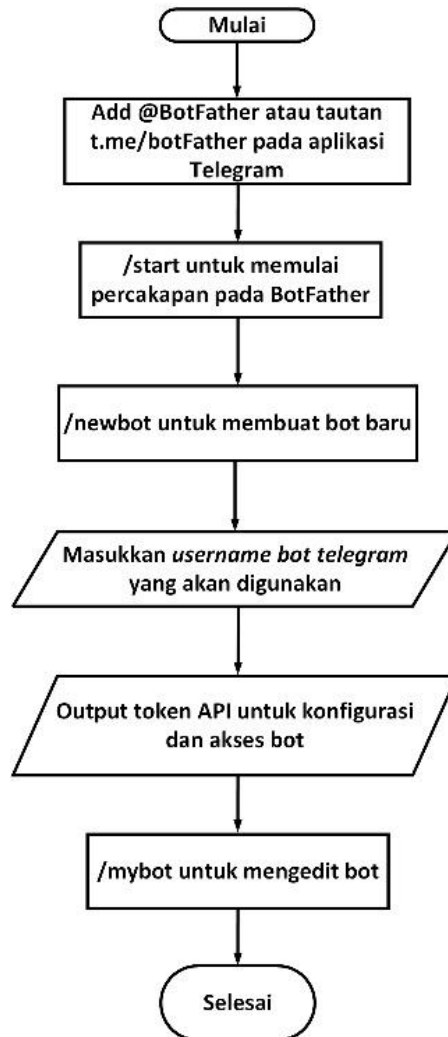
Merujuk pada lampiran C, *listing code* yang digunakan pada Arduino IDE terdiri dari beberapa bagian. Impor *library* diperlukan untuk menjalankan fungsionalitas ke modul mikrokontroler, dengan menggunakan *library*, kode untuk mikrokontroler lebih ringkas sehingga tidak perlu menulis dari awal untuk melakukan tugas-tugas yang umum. Untuk mengimpor *library* ke mikrokontroler digunakan pernyataan *#include*, pernyataan ini akan menyertakan kode *library* ke dalam mikrokontroler.

Mendefinisikan variabel/konstanta mempermudah program dalam membaca kode sehingga program lebih mudah dipahami. Mendefinisikan variabel/konstanta memudahkan perubahan nilai konstan jika diperlukan di masa depan. ketika pin yang digunakan dalam proyek ingin diganti, hanya perlu mengubah nilai variabel/konstanta di satu tempat, yaitu dalam deklarasi konstanta, daripada harus mencari dan mengganti semua nilai tersebut di seluruh kode. Kode ini dimulai dengan mendefinisikan beberapa variabel/konstanta, seperti:

### 3.5.1 Perancangan *Bot Telegram*

Perancangan ini menghubungkan aplikasi *Telegram* dan perangkat keras agar dapat berinteraksi. *Telegram* akan memberikan perintah ke perangkat keras untuk melakukan sebuah aksi seperti *on/off*. Sebuah pesan perintah dari *bot Telegram* akan dikirim ke perangkat Wemos D1 R1 sehingga alat monitoring dapat berjala. Komunikasi antara *Bot Telegram* dan alat monitoring membutuhkan koneksi internet yang stabil.

Telegram mempunyai fasilitas *bot* atau yang biasa dikenal dengan istilah *Bot Telegram*. *Bot Telegram* adalah akun khusus yang tidak memerlukan nomor telepon tambahan untuk disiapkan. Akun ini dapat digunakan untuk mengendalikan perangkat lain.



Gambar 3. 5 *Flowchart* Pembuatan Akun Bot Telegram

Flowchart pada Gambar 3.5 merupakan alur pembuatan akun bot pada aplikasi Telegram. Langkah-langkah pembuatan bot telegram diuraikan sebagai berikut:

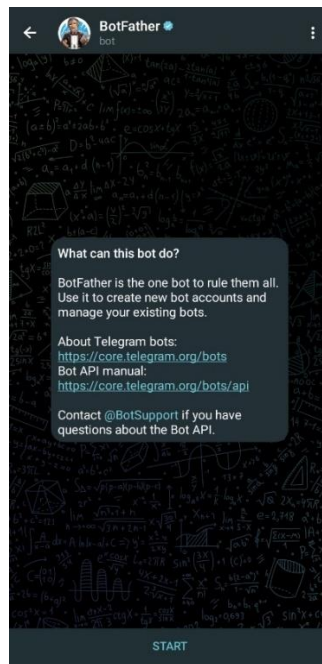
1. Buka aplikasi *Telegram* melalui *smartphone* atau PC, lalu ketik *BotFather* pada kolom pencarian seperti pada Gambar 3.6 atau langsung melalui tautan

t.me/botFather. *BotFather* sebagai saluran resmi di telegram yang bertugas menciptakan bot baru.



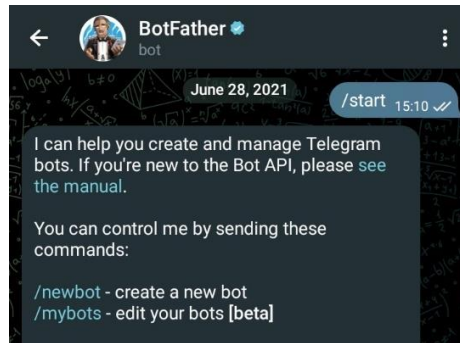
Gambar 3. 6 Search *BotFather*

2. Untuk memulai obrolan, klik *Start* pada pesan perintah otomatis yang muncul ketika membuka saluran *BotFather* seperti pada Gambar 3.7.



Gambar 3. 7 *BotFather*

3. Gunakan perintah `\newbot`` seperti pada Gambar 3.8 untuk membuat bot baru. *BotFather* akan meminta nama yang akan digunakan untuk bot.



Gambar 3. 8 Menu Perintah pada *BotFather*

4. Setelah menentukan nama bot yang akan digunakan, seperti pada Gambar 3.9 *BotFather* akan memberikan token API yang bersifat rahasia. Token ini akan digunakan untuk mengonfigurasi dan mengakses bot melalui API *Telegram*.



Gambar 3. 9 Tampilan Token API pada *BotFather*

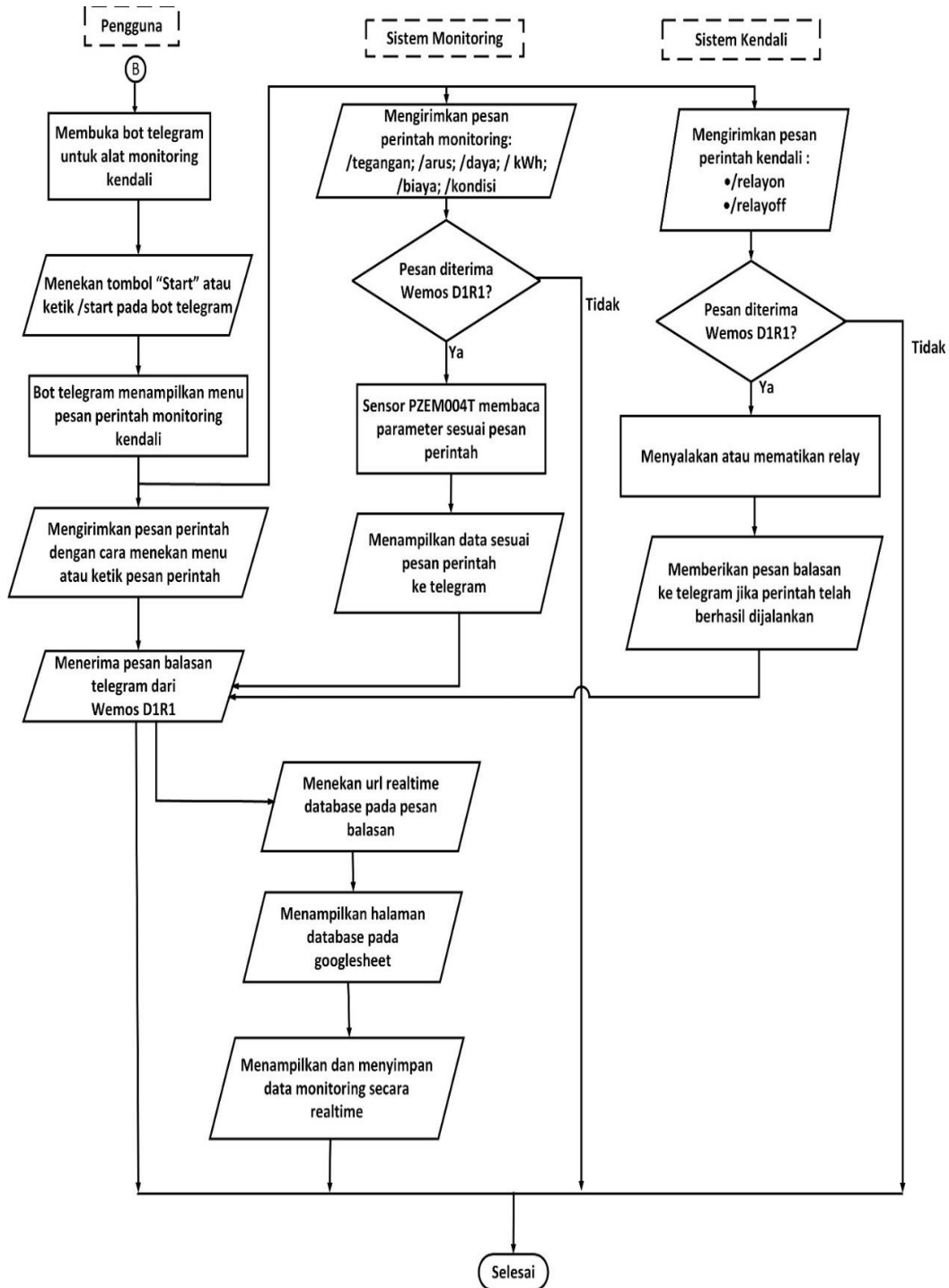
5. *BotFather* menyediakan perintah tambahan untuk mengonfigurasi bot, seperti /mybot seperti pada Gambar 3.10 untuk mengedit bot yang telah dibuat sebelumnya, seperti menambahkan deskripsi, gambar profil, atau mengonfigurasi perintah bot kostum.



Gambar 3. 10 Menu /mybots pada *BotFather*

6. Setelah selesai, *bot Telegram* yang telah dibuat siap digunakan. *Bot Telegram* diakses melalui tautan yang diberikan *BotFather* atau mencarinya langsung pada kolom pencarian di Telegram dengan nama yang telah ditentukan sebelumnya.





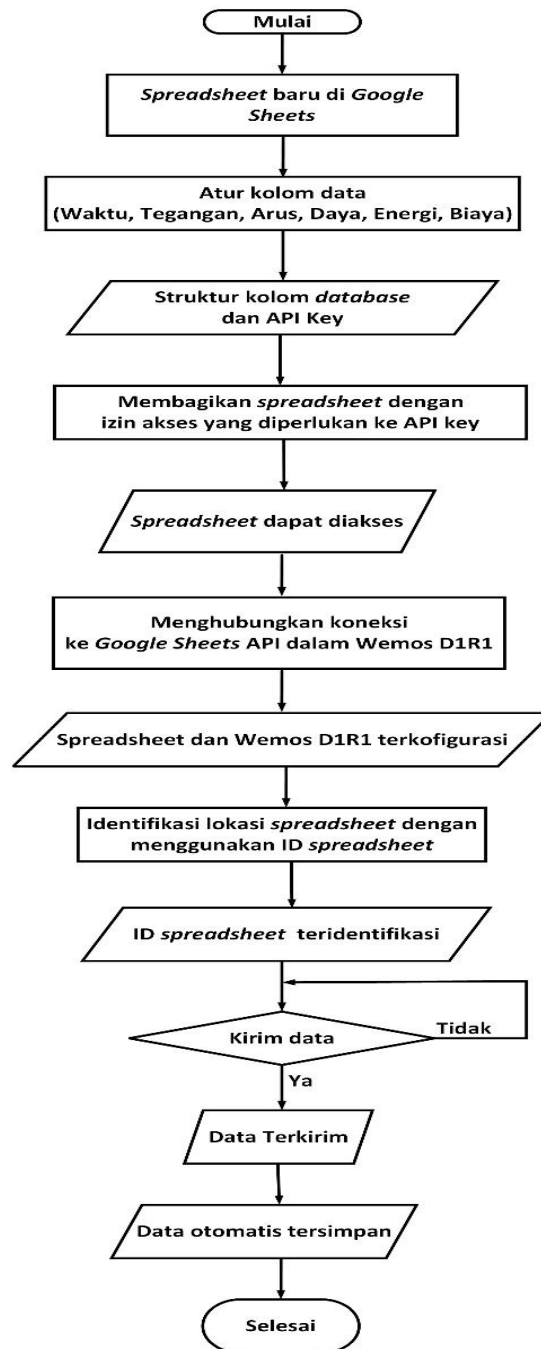
Gambar 3. 11 Flowchart Sistem Bot Telegram

*Flowchart* pada Gambar 3.11 merupakan alur sistem alat berkomunikasi dengan *bot Telegram*. *Bot Telegram* awalnya menerima pesan dari pengguna. Pesan-pesan yang dapat berupa perintah untuk menampilkan data, mengubah pengaturan serta melakukan kendali pada perangkat listrik yang terhubung dengan alat monitoring kendali. *Bot telegram* kemudian memproses pesan tersebut dengan memberi respon yang sesuai dengan perintah yang dikirimkan oleh pengguna. Jika pesan tersebut berupa perintah kendali relay, *bot Telegram* akan melakukan perintah sesuai yang dikirimkan pengguna. *Database* monitoring secara *realtime* disimpan pada *Google Sheets* yang dapat diakses dengan menekan url yang dikirimkan pada pesan balasan *bot Telegram*. Stabilitas koneksi internet mempengaruhi fungsionalitas sistem ini. Koneksi yang tidak stabil dapat mengakibatkan sistem tidak menyimpan data dan *Bot Telegram* tidak merespons, mengganggu penggunaan sistem.

### 3.5.2 Perancangan *Google Sheets* sebagai *Database*

Perancangan ini memanfaatkan *Google Sheets* sebagai *database* konsumsi energi listrik yang didapatkan dari sensor. Dengan menggunakan *Google Sheets* sebagai alat *database* konsumsi listrik, mempermudah untuk mengetahui informasi yang terkumpul secara *realtime*, memudahkan pengguna dalam pemantauan terhadap pemakaian konsumsi energi listrik. Dalam merancang *Google Sheets* sebagai *database* untuk monitoring secara *realtime* terdapat beberapa tahapan. Berdasarkan Gambar 3.12 merupakan *flowchart* perancangan *Google Sheets* sebagai *database* dimulai dengan pembuatan *spreadsheet* baru di *Google Sheets*. Setelah *spreadsheet* dibuat, langkah selanjutnya menentukan struktur kolom untuk penyimpanan data monitoring, seperti waktu, tagangan, arus, daya, energi dan biaya. Selanjutnya untuk mengamankan koneksi antara Wemos DIR1 dan *Google Sheets*, maka diperlukan *API Key*, dengan membuat proyek di *Google AppScript*, mengaktifkan *Google Sheets* API dan mendapatkan *API Key*. *API Key* ini digunakan untuk memberikan izin akses ke *spreadsheet*. Proses berikutnya yaitu membagikan *spreadsheet* dengan izin akses yang diperlukan untuk memastikan bahwa Wemos DIR1 dapat mengirimkan data ke *spreadsheet* secara otomatis dan sesuai struktur yang telah dibuat sebelumnya. Proses integrasi ke *Google Sheets* API dalam kode

program dilakukan untuk memastikan Wemos D1R1 dan *spreadsheet* saling terhubung. Berikut flowchart perancangan Google Sheet sebagai database monitoring.



Gambar 3. 12 Flowchart Perancangan Google Sheets sebagai Database

Berdasarkan Gambar 3.12 ID *Spreadsheet* berfungsi untuk mengidentifikasi lokasi *spreadsheet* yang akan diakses oleh Wemos D1R1. Proses terakhir yaitu mengirimkan data ke spreadsheet melalui Wemos D1R1 yang telah terhubung dengan PZEM004T, kemudian memastikan data yang telah terkirim sesuai dengan struktur kolom pada spreadsheet. Data yang tercatat pada spreadsheet disimpan secara otomatis.

Merujuk pada lampiran C terkait listing code pada Goofke AppScript, pertama, nilai-nilai yang akan dicatat oleh sensor PZEM004T dalam spreadsheet harus diidentifikasi seperti tegangan, arus, daya, kWh, biaya, dan waktu data pencatatannya. Nilai-nilai tersebut akan membentuk dataset yang akan dikirim ke *Google Sheets*. Dataset ini terbentuk dengan memasukkan nilai-nilai sensor ke dalam struktur data yang sesuai, dengan menggunakan fungsi ``Data_to_Sheets``. Tahap berikutnya adalah membuat identitas khusus untuk *spreadsheet* yang akan berfungsi sebagai basis data. Identitas ini terdapat pada variabel ``Sheet_GAS_ID`` dalam kode program, berfungsi sebagai kunci identitas utama yang menghubungkan alur data Wemos D1R1 ke *Google Sheets*.

Tahapan selanjutnya memulai koneksi dengan *Google Sheets* melalui fungsi khusus "Google\_Sheets\_Init". Fungsi ini berfungsi untuk memperlancar komunikasi antara perangkat keras dan *Google Sheets*. Kemudian menetapkan nama kolom sebagai parameter, agar sistem mengetahui letak setiap data harus disimpan. Setiap data yang dikumpulkan dari sensor diberi timestamp yang menunjukkan waktu pengambilan data. Fungsi waktu yang terkait dengan zona waktu memastikan bahwa data tersimpan dengan baik. Selama operasional sistem, alat ini memiliki fitur pembersihan data. Fungsi ``Google_Sheets_ClearData`` digunakan ketika reset data pada *spreadsheet* diperlukan. Memberikan informasi melalui *bot Telegram* dengan interaktif dan fleksibel.

Merujuk pada lampiran C mengenai listing program pada App Script untuk *Google Sheets*. Fungsi ``doGet`` yang terlampir merupakan bagian integral dari sistem komunikasi antara sistem monitoring IoT dengan *Google Sheets*. Saat ada permintaan *HTTP GET*, fungsi ini diaktifkan untuk menangani parameter yang dikirim melalui URL. Fungsi ini mencetak parameter pertama yang diterima dengan menggunakan ``Logger.log (JSON.stringify (e));``. Memungkinkan untuk melihat struktur data yang diterima.

Selanjutnya, fungsi menginisialisasi variabel hasil dengan fungsi ``result`` dengan asumsi keberhasilan, kemudian fungsi ``undefined`` untuk memeriksa apakah parameter diterima tidak didefinisikan, Jika tidak ada parameter yang diterima, variabel diubah menjadi 'No Parameters', menunjukkan bahwa tidak ada data yang dikirim. Jika terdapat parameter yang diterima, sistem akan membuka lembar aktif *Google Sheets* dengan menggunakan ID *spreadsheet* yang telah ditetapkan.

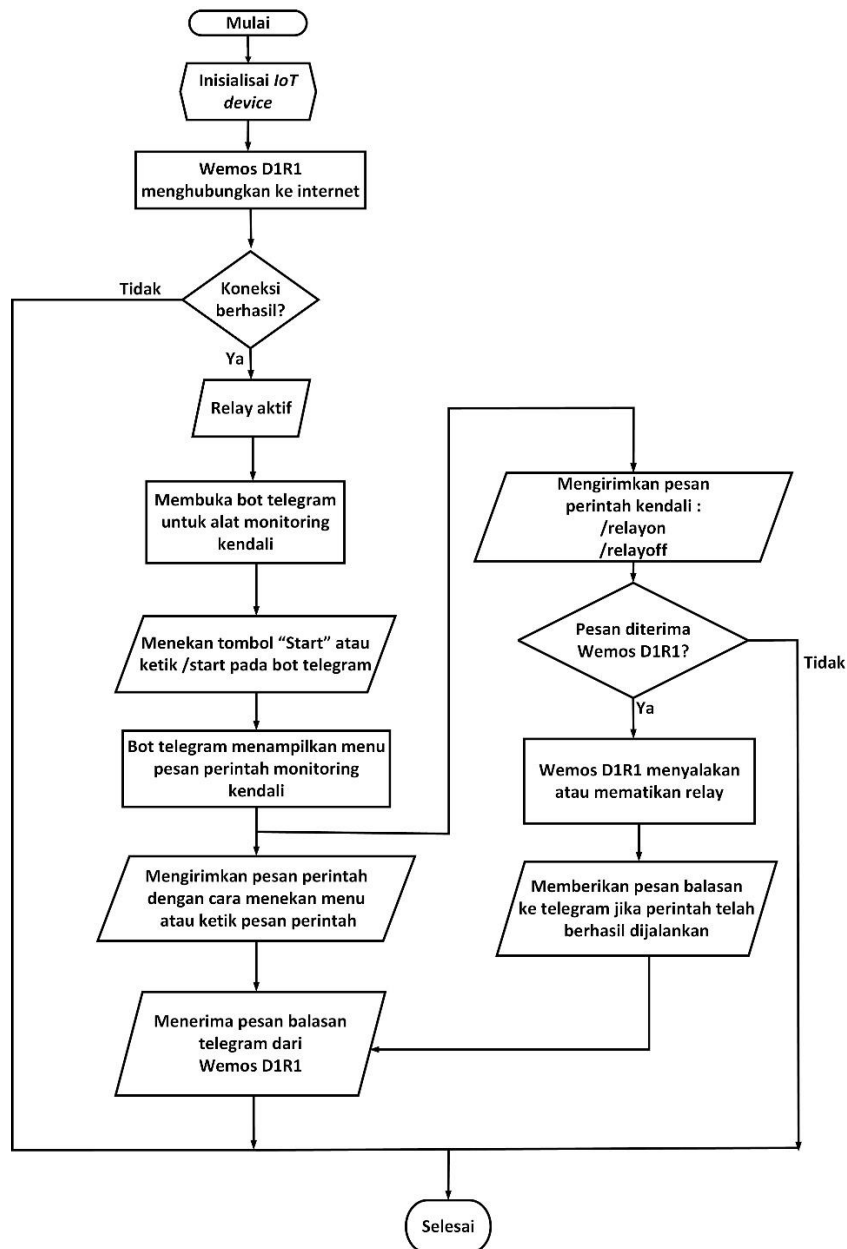
Selanjutnya fungsi membentuk baris data baru dengan timestamp dan nilai-nilai parameter yang diterima. Proses pembentukan baris ini melibatkan penggunaan perulangan ``for`` untuk mengiterasi melalui parameter yang diterima dan menyesuaikan nilai-nilai tersebut ke dalam baris data. Data baru ini disimpan dilembar *Google Sheets* pada baris terakhir dengan menggunakan ``sheet.getRange (newRow, 1, 1, rowData.length)`` dan disertakan ke dalam fungsi ``setValues ([rowData])``, Log dari data yang telah dibentuk untuk keperluan monitoring dan debugging. Hasil proses ini, baik itu berhasil atau pesan kesalahan, dikembalikan sebagai output dari fungsi ini.

Struktur kolom untuk database monitoring secara real-time. setiap kolom memiliki label yang sesuai dengan parameter yang diukur, yaitu "Date", "Tegangan", "Arus", "Daya", "Faktor Daya", "kWh" dan "Biaya". Kolom "Date" untuk mencatat informasi timestamp untuk setiap pengukuran, sementara kolom-kolom lainnya menyimpan data yang dihasilkan oleh beban listrik.

Dalam konteks pengambilan data ini, sistem menggunakan ``ContentService.createTextOutput (result)`` untuk mengirimkan pesan hasil proses ke pihak yang melakukan permintaan. Fungsi ``stripQuotes`` digunakan untuk menghilangkan tanda kutip ganda atau tanda kutip sati dari nilai yang diterima, memastikan kebersihan dan konsistensi data sebelum disimpan ke *Google Sheets*. Keseluruhan, fungsi ``doGet`` ini memainkan peran penting dalam proses mentransfer dan menyimpan data dari perangkat keras monitoring ke *Google Sheets*.

### 3.6 Perancangan Kendali Beban Listrik

Perancangan kendali beban listrik yaitu dengan menghubungkan alat monitoring dengan perangkat listrik yang akan diuji sebagai beban, seperti lampu, kipas angin dan *rice cooker*. Alat monitoring sudah terkoneksi ke jaringan internet dan dapat menerima perintah melalui *Bot Telegram*.



Gambar 3. 13 Flowchart Perancangan Kendali Beban Listrik

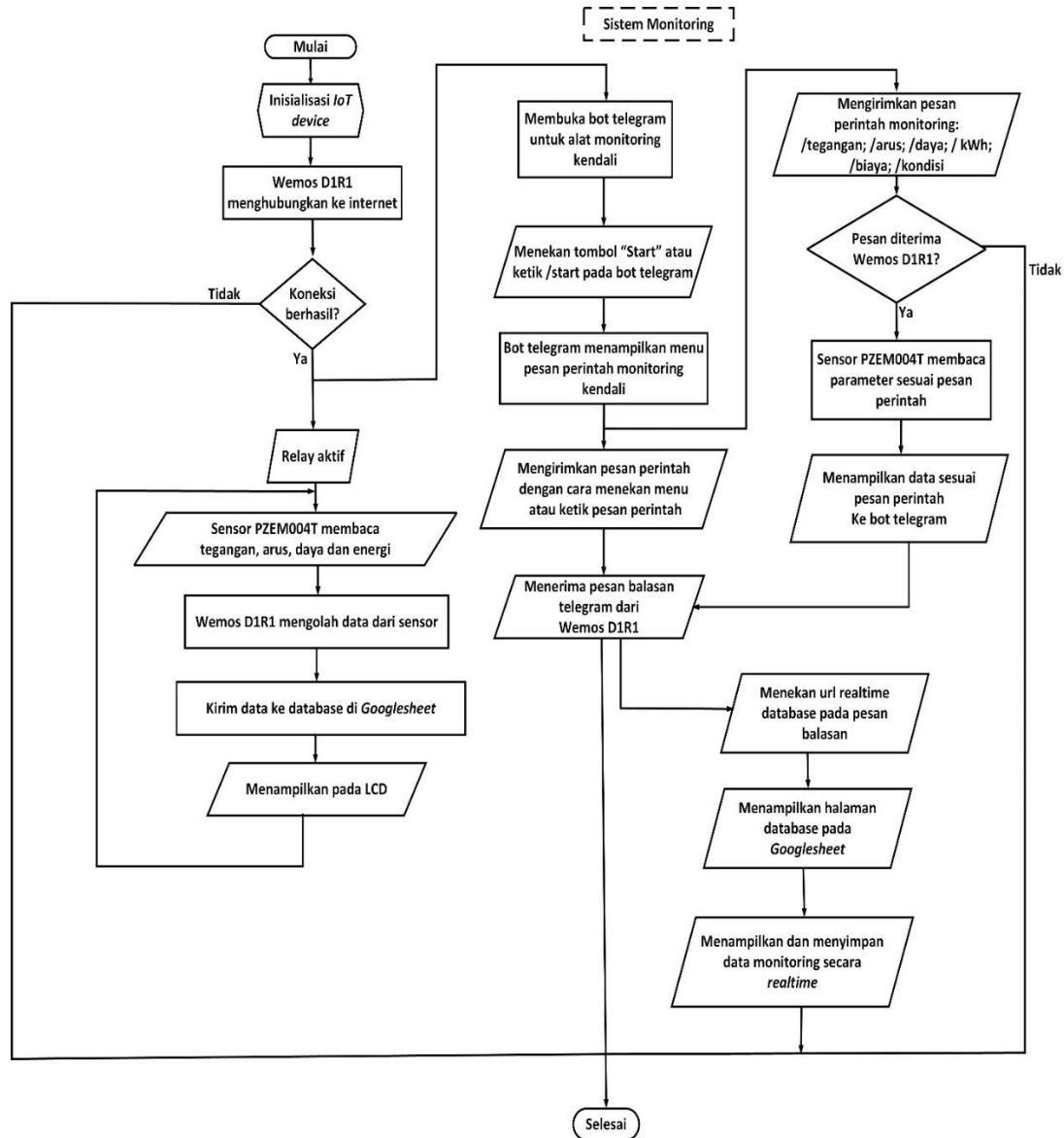
*Flowchart* pada Gambar 3.14 merupakan perancangan kendali pada beban listrik yang terpasang pada alat monitoring kendali. Pengguna mengirim kode perintah pada Bot Telegram sebagai kendali alat monitoring. Pengguna mengirim kode perintah `/relayoff` untuk mematikan perangkat listrik yang terhubung melalui *Bot Telegram*. Pengamatan dilakukan terhadap waktu yang diperlukan oleh alat monitoring untuk mengubah status perangkat listrik menjadi *"OFF"*. Selanjutnya, dilakukan pengukuran waktu respon sebanyak lima kali untuk memastikan akurasi dan konsistensi hasil. Langkah serupa juga dilakukan untuk perintah `/relayon` untuk menyalakan perangkat listrik, dengan pengukuran waktu respon yang mencatat waktu dari pengiriman perintah *"ON"* hingga status perangkat listrik benar-benar berubah menjadi *"ON"*.

Hasil pengujian waktu respon tersebut akan dianalisis dan rata-rata waktu respon akan dihitung. Tujuan utama pengujian ini adalah untuk memastikan bahwa alat monitoring mampu mengendalikan beban listrik sesuai perintah yang diberikan melalui Telegram dengan waktu respon yang dapat diterima.

### **3.7 Perancangan Monitoring Energi**

Perancangan monitoring energi bertujuan untuk mempermudah proses pemantauan serta analisis konsumsi energi listrik berbasis *Internet of Things (IoT)* yang terintegrasi dengan *bot Telegram*. Implementasi sistem ini melibatkan penerapan sensor PZEM004T, yang berfungsi untuk mengukur parameter arus, tegangan, dan daya dari sumber listrik AC 220Volt. Selain itu, relay modul 5v *1channel* sebagai bagian dari sistem ini, bertindak sebagai sensor untuk mengendalikan dan mengelola arus serta tegangan dalam menjaga stabilitas rangkaian saat terdapat potensi beban yang berlebihan. Relay tersebut dapat dikendalikan melalui bot Telegram, memungkinkan pengaksesan jarak jauh. Informasi mengenai parameter listrik yang terbaca akan ditampilkan dengan jelas pada layar LCD 20×4 yang telah terpasang dengan perangkat I2C. Melalui alat perangkat ini, pemantauan

konsumsi energi dapat dilakukan dengan lebih mudah dan interaktif, menghubungkan pengguna dengan data secara *real-time*.

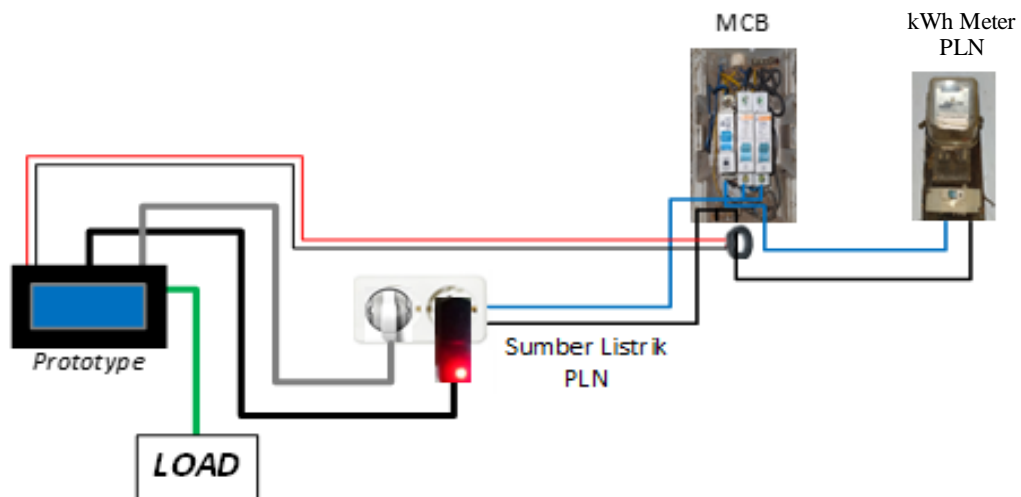


Gambar 3. 14 *Flowchart* Perancangan Monitoring Energi

Gambar 3.15 menunjukkan implementasi monitoring penggunaan energi perangkat listrik rumah tangga via *bot Telegram* dan *Google Sheets* sebagai media penyimpanan data monitoring. Berdasarkan Gambar 3.15 tahapan dimulai dengan kondisi alat monitoring



kendali sudah terhubung dengan *power supply*, ketika dijalankan maka akan dilakukan inisialisasi oleh alat, kemudian Wemos D1R1 akan menghubungkan koneksi ke internet, apabila koneksi internet tidak dapat terhubung maka alat tidak akan bisa beroperasi dengan baik, sedangkan apabila koneksi internet terhubung maka Wemos D1R1 akan menampilkan informasi berupa “*connected*” pada layar LCD. Relay akan dalam kondisi aktif. Sensor PZEM004T kemudian akan membaca nilai parameter beban listrik yang telah terpasang dengan alat monitoring kendali. Data-data yang terbaca oleh sensor secara real-time kemudian akan tercatat otomatis pada *database* di *Google sheets* dan data juga ditampilkan pada layar LCD. Setelah Wemos D1R1 sudah terhubung dengan koneksi internet, komunikasi dengan bot telegram dapat dilakukan dengan menggunakan pesan perintah yang telah tersedia. Apabila pesan perintah tersebut terkirim dan Wemos D1R1 merespon maka *bot telegram* akan mengirimkan pesan balasan sesuai pesan perintah yang diberikan. Apabila Wemos D1R1 tidak dapat merespon maka *bot telegram* tidak akan memberi pesan balasan. Tautan URL *database* pada *Google Sheet* juga dicantumkan pada pesan balasan *bot telegram* dengan mengklik atau menyalin alamat, pengguna dapat dengan cepat akan diarahkan ke halaman *spreadsheet* database untuk melihat data yang telah tercatat.



Gambar 3. 15 Rangkaian Penggunaan Alat pada MCB

Gambar 3.16 merupakan susunan rangkaian alat monitoring kendali yang dipasangkan pada instalasi MCB rumah. Alat monitoring kendali ini menggunakan adaptor 12V untuk menghubungkan dengan catu daya, catu daya dialirkan dengan menggunakan dua terminal yang telah terhubung dengan kabel fasa pada MCB. Pemasangan CT *clamp* PZEM004T diletakkan di sekitar kabel fasa yang berasal dari MCB kWh meter PLN. Data monitoring yang yang terbaca akan ditampilkan pada layar LCD dan data monitoring disimpan otomatis di *Google Sheets*.

## BAB IV

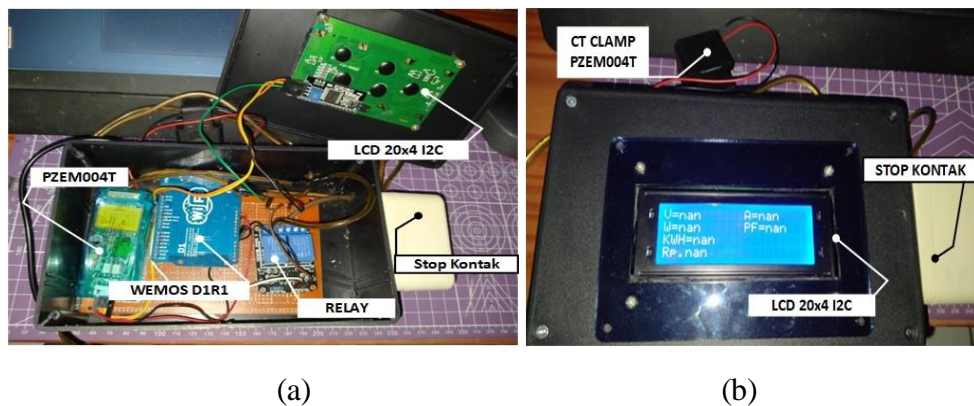
### HASIL DAN ANALISIS

Bab ini berisi penjelasan prosedur hasil pengujian dari alat yang telah dirancang untuk mengetahui sistem kerja alat monitoring berjalan dengan baik sesuai tujuan. Data hasil pengujian yang diperoleh nantinya akan dianalisa untuk dijadikan acuan dalam mengambil kesimpulan. Pengujian yang dilakukan pada penelitian ini meliputi:

- a. Hasil Perancangan
- b. Pengujian *Hardware*
- c. Pengujian *Software*
- d. Pengujian Keseluruhan

#### 4.1 Hasil Perancangan Alat

Perancangan *hardware* atau perangkat keras yang telah disusun pada Gambar 3.7, diterapkan dalam bentuk yang sesuai perancangan agar memudahkan pengujian dan pengaplikasiannya. Hasil dari perancangan *hardware* yang telah di buat dapat dilihat pada Gambar 4.1.



Gambar 4. 1 (a). Tampak Dalam; (b). Tampak Luar Alat Monitoring

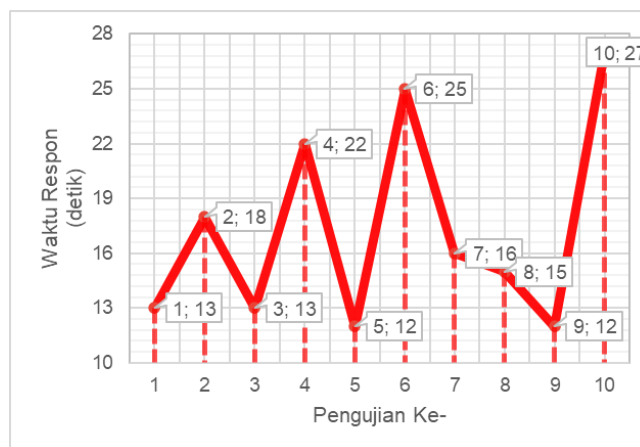
Gambar 4.1 merupakan hasil perancangan alat monitoring penggunaan energi perangkat listrik rumah via *bot Telegram* berdasarkan tampak dalam dan tampak luar.

Gambar 4.1 bagian (a) terlihat tampilan dalam alat yang terdiri dari susunan komponen-komponen yang terpasang didalam kotak alat monitoring kendali ini. Komponen yang terpasang yaitu modul Wemos D1 R1, sensor PZEM004T dan relay yang saling terhubung antara satu komponen dengankomponen lainnya. Gambar 4.1 bagian (b) terlihat tampilan luar alat monitoring yang terdiri dari LCD 20x4 yang sudah terintegrasi dengan I2C sebagai penampil informasi data secara *realtime*, kemudia stop kontak yang telah terhubung dengan relay, sehingga beban listrik terpasang dapat dikendalikan jarak jauh, CT clamp yang merupakan bagian dari sensor PZEM004T berfungsi untuk menghitung arus yang mengalir pada beban listrik terpasang. Tegangan yang digunakan untuk menyuplai Wemos D1 R1 adalah sekitar 12VDC, sehingga tegangan dari PLN yaitu sekitar 220VAC perlu diubah menjadi VDC dengan menggunakan adaptor.

## 4.2 Pengujian *Hardware*

### 4.2.1 Hasil Pengujian Relay

Pengujian ini bertujuan untuk menguji kemampuan relay dalam mengendalikan perangkat listrik yang terhubung dengan relay secara *wireless* melalui *bot Telegram*. Pengujian ini dilakukan dengan cara mengirimkan pesan perintah *ON* atau *OFF* ke relay melalui Wemos D1R1. Relay juga memiliki LED indikator yang menunjukkan status *ON* atau *OFF* dari relay.



Gambar 4. 2 Hasil Pengujian Relay

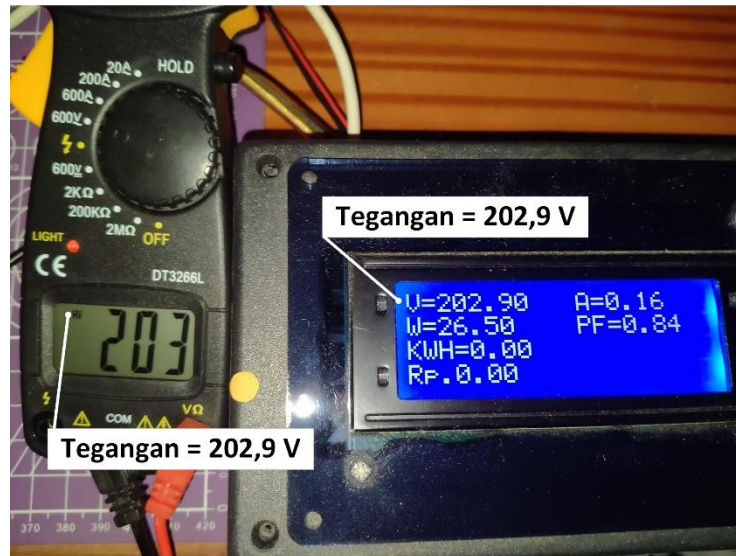
Grafik pada Gambar 4.2 merupakan hasil pengujian relay terhadap beban listrik. Relay diuji sebanyak 10 kali. Pengujian dilakukan dengan mengirimkan pesan perintah pada bot Telegram dalam kondisi *ON/OFF* Waktu respon tercepat tercatat pada pengujian ke-5, dengan waktu respon sebesar 12 detik. Waktu respon terlama tercatat pada pengujian ke-10, dengan waktu respon sebesar 27 detik. Rata-rata waktu respon yang tercatat pada pengujian relay yaitu 16,5 detik. Perbedaan waktu respon pada masing-masing pengujian dapat disebabkan oleh kondisi koneksi internet yang kurang stabil sehingga memengaruhi kecepatan pengiriman perintah dari bot telegram ke relay. Berdasarkan Gambar 4.2 menunjukkan bahwa relay dapat mengendalikan perangkat listrik dengan baik dengan respon waktu yang berbeda-beda.

#### **4.2.2 Hasil Pengujian PZEM004T**

Pengujian sensor PZEM004T bertujuan untuk memastikan bahwa sensor tersebut berhasil dalam membaca parameter yang diperlukan dalam perancangan alat monitoring ini. Pengujian sensor ini terdiri dari dua tahap yaitu pengujian tegangan dan pengujian arus.

##### **4.2.2.1 Hasil Pengujian Tegangan**

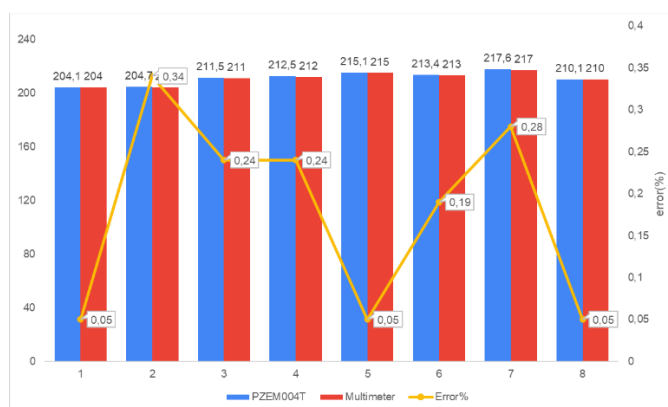
Berdasarkan hasil perancangan yang dapat dilihat pada Gambar 4.3 terlihat bahwa pengujian sensor tegangan AC pada alat monitoring ini, dilakukan dengan membandingkan pengukuran tegangan antara yang terbaca oleh alat monitoring dengan pembacaan oleh Voltmeter digital. Pengujian tegangan ini menggunakan tegangan listrik PLN yang biasa di gunakan pada setiap rumah yaitu sebesar 220V.



Gambar 4. 3 Perbandingan Tegangan Hasil Alat Monitoring dan Alat Ukur

Data yang terlihat Gambar 4.3 merupakan hasil perbandingan pada salah satu pengujian tegangan AC dengan ruang lingkup rumah tangga. Nilai tegangan yang terbaca oleh PZEM004T yaitu sebesar 202,9V, sedangkan nilai tegangan yang terbaca oleh multimeter yaitu sebesar 203V. Nilai tegangan antara PZEM004T dan multimeter memiliki selisih sebesar 0,10V.

Hasil yang didapatkan pada perbandingan pengujian tegangan yang dilakukan antara PZEM004T dengan multimeter, dapat dilihat pada Gambar 4.4



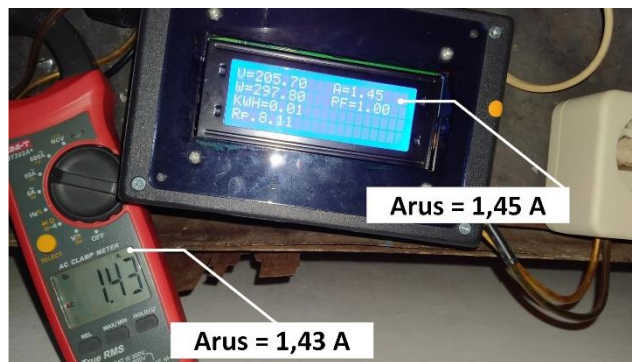
Gambar 4. 4 Grafik Perbandingan Tegangan Hasil Alat Monitoring dan Alat Ukur

Hasil pengujian pada Gambar 4.4 menunjukkan sensor tegangan pada PZEM004T dapat mencatat tegangan AC dengan nilai tegangan terendah yaitu sebesar 204,1V dan nilai tegangan tertinggi sebesar 217,6V. Rata-rata tegangan yang tercatat pada pengujian tegangan menggunakan PZEM004T yaitu 210,75V

*Error* pada pengukuran pada sensor tegangan PZEM004T dan multimeter digital memiliki rata-rata persen kesalahan tegangan sebesar 0.18%, persentase kesalahan dapat diakibatkan resolusi pembacaan antara sensor tegangan AC dengan alat ukur yang berbeda, dan dapat juga disebabkan ketidakstabilan tegangan saat proses pengukuran sehingga terdapat selisih pembacaan yang masih dalam tahap wajar. Persentase kesalahan dalam mengukur tegangan pada alat monitoring ini sebesar 0.18% dan dapat dikategorikan sebagai alat ukur cermat atau presisi.

#### 4.2.2.2 Pengujian Arus

Pengujian arus ini bertujuan untuk mendapatkan nilai arus efektif yang diambil secara berkala dan diolah untuk dilakukan perbandingan dengan hasil pembacaan oleh tang meter. Pengamatan perbandingan arus menggunakan tang amper dan alat monitoring arus dapat dilihat pada Gambar 4.5.

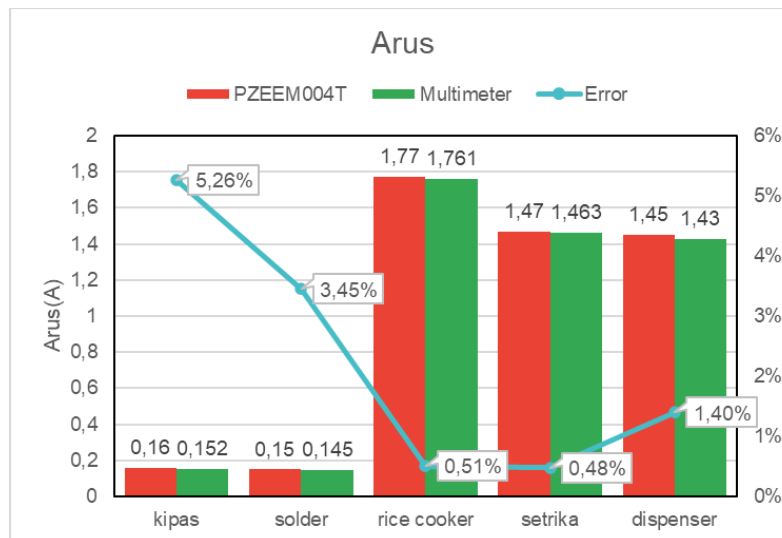


Gambar 4. 5 Pengamatan Perbandingan Arus Hasil Alat Monitoring dan Alat Ukur

Data yang terlihat Gambar 4.5 merupakan hasil perbandingan pada salah satu pengujian arus pada dispenser sebagai bebannya. Nilai tegangan yang terbaca oleh

PZEM004T yaitu sebesar 1,45A, sedangkan nilai tegangan yang terbaca oleh multimeter yaitu sebesar 1,43. Nilai tegangan antara PZEM004T dan multimeter memiliki selisih sebesar 0,02A.

Alat monitoring ini membaca arus yang dipakai oleh beban peralatan listrik rumah tangga. Grafik hasil pengujian dari pembacaan arus dengan menggunakan PZEEM004T dan tang meter dengan menggunakan beban yang berbeda-beda dapat dilihat pada Gambar 4.6. Perbandingan ini dilakukan untuk melihat seberapa besar penyimpangan pembacaan arus yang dilakukan oleh alat monitoring tersebut.



Gambar 4. 6 Grafik Perbandingan Arus Hasil Alat Monitoring dan Alat Ukur

Gambar 4.6 menampilkan grafik hasil pengujian sensor arus menggunakan PZEM004T dan Multimeter kemudian dilakukan perbandingan hasil data dari keduanya. Hasil pengujian pada Gambar 4.6 menunjukkan sensor tegangan pada PZEM004T dapat mencatat arus perangkat listrik rumah tangga dengan nilai arus terendah yaitu berasal dari solder dengan nilai arus sebesar 0,15A dan nilai arus tertinggi yaitu berasal dari *rice cooker* dengan nilai arus sebesar 1,47A. Nilai rata-rata *error* perbandingan nilai arus menggunakan PZEM004T dan multimeter yaitu sebesar 2%, pada hasil pengujian



PZEEM004T diatas masih terdapat selisih nilai, hal ini terjadi disebabkan terbatasnya sensitifitas pembacaan dari sensor maupun alat ukur yang digunakan.

### 4.3 Pengujian *Software*

Pengujian dilakukan dengan menghubungkan Wemos D1R1 ke internet melalui *software* Arduino IDE dengan dilengkapi instalasi *library* yang digunakan untuk alat monitoring. *Library Firebase* pada program ini bertujuan untuk menggunakan 2 sumber internet yaitu Wi-Fi dan *hotspot* yang akan terhubung dengan alat monitoring. *Library Google Sheets* difungsikan untuk menggunakan *Google Sheets* sebagai penyimpanan data alat monitoring, sedangkan untuk menjalankan *Bot Telegram* pada alat monitoring ini menggunakan *library Universal Bot Telegram*. Komunikasi antara *Bot Telegram* dan alat monitoring menggunakan beberapa perintah sesuai informasi yang dibutuhkan.

Pengujian *software* ini terdiri dari pengujian *Google Sheets* sebagai *Database* dan pengujian *bot Telegram*, dalam pengujian *Google Sheets*, dilakukan pengecekan terhadap kemampuan *Google Sheets* dalam menyimpan data dari alat monitoring. Sedangkan dalam pengujian *bot Telegram*, bertuju pada kemampuan interaksi antara *bot Telegram* dan alat monitoring, serta kemampuan *bot Telegram* dalam memberikan informasi dan menerima perintah dari pengguna melalui *bot Telegram*.

#### 4.3.1 Hasil Pengujian *Google Sheets* sebagai *Database*

Pengujian ini bertujuan untuk mengimplementasikan *Google Sheets* sebagai penyimpanan data alat monitoring. Pengujian ini dilakukan untuk mengetahui kemampuan *Google Sheets* sebagai *platform* yang menangani aliran data secara terus-menerus dari alat monitoring, pengujian ini juga bertujuan untuk mengidentifikasi potensi kendala atau hambatan yang timbul dalam penggunaan *Google Sheets* sebagai database. Selama pengujian, data konsumsi energi listrik rumah tangga telah dikumpulkan menggunakan sensor PZEM004T. Data ini mencakup besaran tegangan, arus, daya, kWh dan biaya yang terukur secara berkala. Setiap hasil pengukuran

disimpan dalam *Google Sheets* dengan interval pengukuran yang telah diatur sebelumnya, yaitu setiap 30 detik.



	A	B	C	D	E	F	G	H
1	Date	Time	Tegangan	Arus	Daya	PF	kWh	Biaya
11472	25/07/2023	18.09.14	202.60	1.34	235.70	0.87	0.13	174.41
11473	25/07/2023	18.09.57	201.60	1.35	236.70	0.87	0.13	178.46
11474	25/07/2023	18.10.41	202.10	1.36	237.70	0.87	0.14	182.52
11475	25/07/2023	18.11.24	206.50	1.40	251.80	0.87	0.14	186.58
11476	25/07/2023	18.12.04	205.70	1.50	265.50	0.86	0.14	190.63
11477	25/07/2023	18.12.47	202.80	1.41	245.00	0.86	0.14	194.69
11478	25/07/2023	18.13.29	202.00	1.39	241.10	0.86	0.15	198.74
11479	25/07/2023	18.14.11	202.90	1.37	240.20	0.86	0.15	202.80
11480	25/07/2023	18.14.53	199.70	1.36	232.00	0.86	0.15	206.86
11481	25/07/2023	18.15.37	201.40	1.37	236.40	0.86	0.16	210.91
11482	25/07/2023	18.16.21	202.80	1.63	291.00	0.88	0.16	214.97
11483	25/07/2023	18.17.04	202.80	1.54	276.10	0.88	0.16	219.02
11484	25/07/2023	18.17.46	198.80	1.62	282.60	0.88	0.17	224.43
11485	25/07/2023	18.18.27	198.10	1.56	272.40	0.88	0.17	228.49
11486	25/07/2023	18.19.09	198.20	1.64	284.90	0.88	0.17	232.54
11487	25/07/2023	18.19.49	199.50	1.66	290.20	0.88	0.17	236.60
11488	25/07/2023	18.20.31	197.70	1.51	264.50	0.88	0.18	240.66

Gambar 4. 7 Tampilan *Database* pada *Google Sheets*

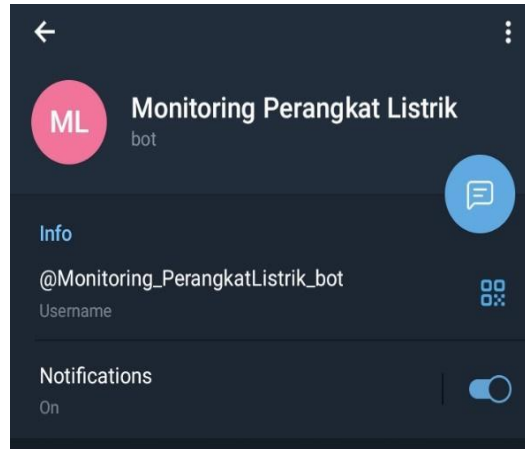
Data yang terlihat Gambar 4.7 merupakan hasil pengujian *Google Sheets* sebagai *database* monitoring konsumsi energi listrik pada rumah tangga. Data terdiri dari beberapa parameter yaitu Date sebagai tanggal pencatatan data, Time sebagai waktu data tercatat, arus, daya, fakyor daya, kWh serta estimasi biaya konsumsi energi listrik.

Hasil pengujian menunjukkan bahwa *Google Sheets* dapat mengintegrasikan dan menyimpan data monitoring dengan baik. *Google Sheets* juga mampu mempertahankan konsistensi data selama seluruh periode pengujian. pengujian *Google Sheets* untuk penyimpanan data monitoring ini telah berhasil dilaksanakan.

#### 4.3.2 Hasil Pengujian *Bot Telegram*

*Bot Telegram* akan menjalankan perintah dengan baik apabila *device mobile* dan alat monitoring telah terkoneksi dengan internet, jika salah satu tidak terkoneksi maka

perintah dari bot tidak dapat dijalankan. Pengguna dapat mencari akun bot pada aplikasi telegram dengan menggunakan username bot yang tertera pada bio akun bot seperti pada Gambar 4.8 berikut.



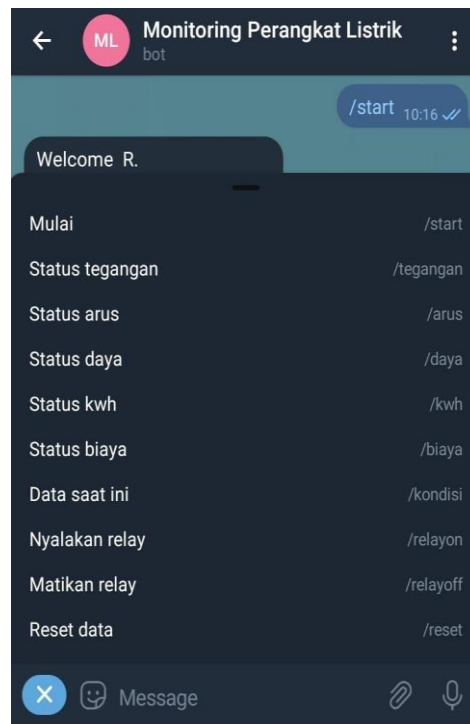
Gambar 4. 8 Tampilan Bio Akun Bot pada Aplikasi Telegram

*Bot Telegram* juga memiliki fitur menu yang dapat membantu pengguna untuk memilih perintah yang diinginkan. Pengguna bot menggunakan beberapa pesan perintah untuk mendapatkan informasi data monitoring terkini. Gambar 4.9 merupakan perintah-perintah yang digunakan dalam penelitian ini.



Gambar 4. 9 Tampilan Pesan Perintah “/start” pada *Bot Telegram*

*Bot Telegram* menyediakan tombol untuk memudahkan pengguna menjalankan perintah bot dengan alat monitoring. Pada Gambar 4.10 berikut merupakan tombol-tombol yang digunakan sebagai perintah bot.



Gambar 4. 10 Tampilan Tombol Alternatif Pesan Perintah *Bot Telegram*



Gambar 4. 11 Tampilan Monitoring perparameter pada *Bot Telegram*

Gambar 4.11 merupakan tampilan respon atau pesan balasan *bot Telegram* dari pesan perintah /tegangan untuk status/informasi tegangan dan /arus untuk status/informasi arus saat alat terpasang pada beban listrik dan terhubung dengan koneksi internet.



Gambar 4. 12 Tampilan Pesan Perintah Reset pada *Bot Telegram*

Gambar 4.12 merupakan tampilan respon atau pesan balasan bot telegram dari pesan perintah /reset. Pesan perintah /reset digunakan untuk mengatur ulang data yang telah tercatat menjadi kembali seperti awal, sehingga alat dapat mencatat ulang data yang dibutuhkan.



Gambar 4. 13 Tampilan Pesan Perintah Kendali pada *Bot Telegram*

Gambar 4.13 merupakan tampilan respon atau pesan balasan *bot Telegram* dari pesan perintah /relayon dan /relayoff. Pesan perintah tersebut digunakan untuk mengaktifkan atau menonaktifkan relay. Jika relay diaktifkan maka *bot Telegram* akan mengirimkan pesan balasan “Relay ON” dan sebaliknya jika relay dinonaktifkan maka *bot Telegram* akan mengirimkan pesan balasan “Relay OFF”.



Gambar 4. 14 Tampilan Data Monitoring pada *Bot Telegram*

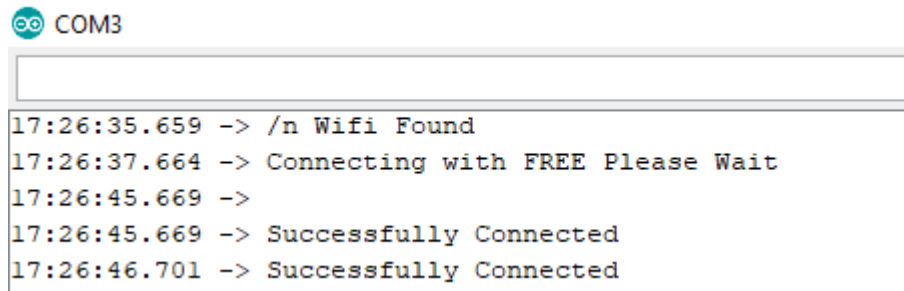
Gambar 4.14 merupakan tampilan respon atau pesan balasan *bot Telegram* dari pesan perintah /kondisi. Pesan perintah /kondisi digunakan untuk mengirimkan informasi data monitoring seluruh parameter serta disediakan tautan yang menuju ke *database* di *Google Sheets*. Berdasarkan Gambar 4.15 data yang tercatat pada bot telegram yaitu nilai tegangan 199V, nilai arus 0,95A, nilai daya 158W, nilai kWh 175, 15 dan estimasi biaya yaitu Rp. 236.797, dalam pesan balasan dilampirkan tautan ke halaman *database* pada *Google Sheets*. Hal ini menunjukkan bahwa *Bot Telegram* mampu memonitoring dan mengendalikan dari jarak jauh. Koneksi internet sangat mempengaruhi pengoperasian sistem. Respons dan kinerja sistem tergantung pada kualitas koneksi internet. Koneksi yang lambat dapat mengakibatkan penundaan atau bahkan ketidakresponsifan sistem terhadap perintah.

#### 4.4 Pengujian Keseluruhan

Pengujian keseluruhan merupakan pengujian yang mencakupi *hardware* dan *software*. Pengujian ini dilakukan untuk mengetahui apakah alat monitoring yang terhubung dengan

telegram sudah berkerja dengan baik dan sesuai yang diharapkan. Pada pengujian ini alat monitoring dihubungkan dengan sumber arus listrik dengan beban listrik AC yang berupa peralatan listrik rumah tangga seperti kipas angin, lampu, dispenser dan setrika. Beban dihubungkan melalui stop kontak yang terdapat pada alat monitoring sehingga sistem dapat membaca nilai-nilai tegangan, arus, daya, energi serta biaya yang terpakai.

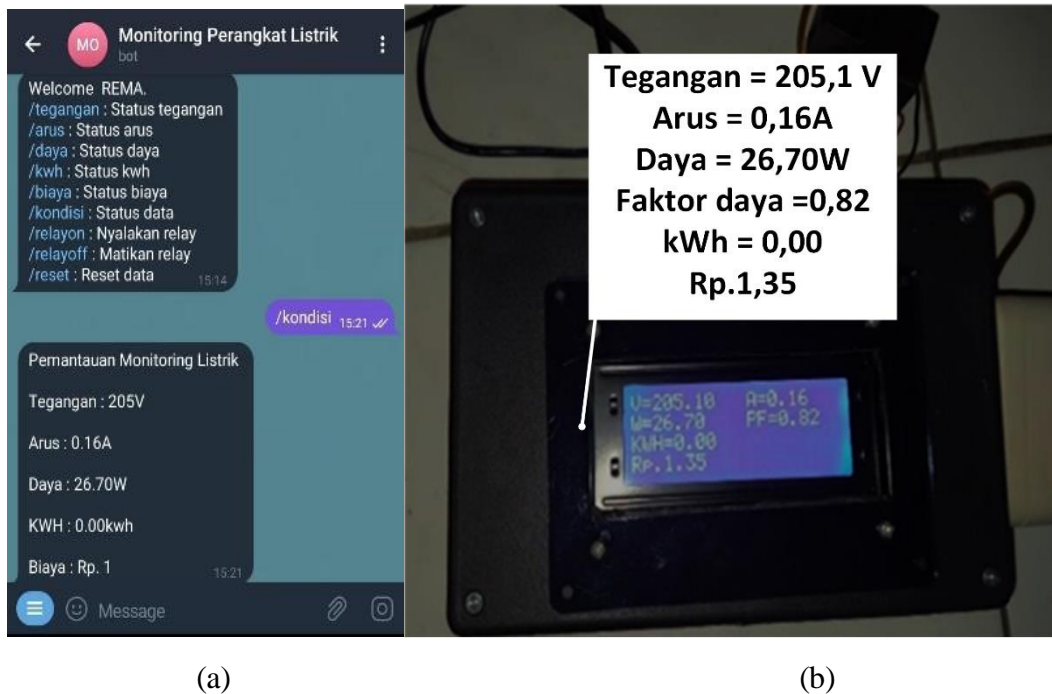
Pengujian ini dimulai dengan memastikan sistem telah terhubung dengan jaringan internet yang dapat dipastikan dengan melihat LCD yang menampilkan kata “*Connected*” atau dapat memberikan perintah *Bot Telegram* pada *smartphone*. Saat sistem telah terhubung internet, perintah bot yang dikirimkan akan mendapatkan respon dengan mengirimkan balasan sesuai perintahnya namun apabila sistem tidak terhubung internet maka perintah yang telah dikirimkan sebelumnya tidak akan mendapatkan respon.



```
COM3
17:26:35.659 -> /n Wifi Found
17:26:37.664 -> Connecting with FREE Please Wait
17:26:45.669 ->
17:26:45.669 -> Successfully Connected
17:26:46.701 -> Successfully Connected
```

Gambar 4. 15 Tampilan Serial Monitor pada Arduino IDE saat Internet telah Terhubung dengan Alat Monitoring Kendali

Tampilan serial monitor pada Arduino IDE di Gambar 4.15 merupakan tampilan awal ketika alat monitoring kendali telah terhubung dengan koneksi internet setelah perangkat dinyalakan. Pengguna tetap dapat memantau penggunaan energi listrik yang terpakai melalui tampilan dari LCD ketika koneksi tidak terhubung dengan baik. Tampilan dari LCD dan aplikasi Telegram dapat dilihat pada Gambar 4.16.



Gambar 4. 16 Tampilan pada *Bot Telegram* (a) dan LCD (b)

Gambar 4.16 (a) merupakan tampilan respon atau pesan balasan bot telegram dari pesan perintah /kondisi. Pengujian ini dilakukan untuk mengetahui keakuratan alat dalam memantau dan berkomunikasi dengan *bot Telegram*. Gambar 4.16(b) menunjukkan data yang tercatat *bot Telegram* yaitu tegangan sebesar 205V, arus 0,16A, daya 26,7W, 0 kW, sedangkan Gambar 4.16(b) menunjukkan data yang tercatat pada LCD yaitu tegangan sebesar 205,1V, arus 0,16, daya 26,7W, dan 0 kWh. Data yang terlihat pada *bot Telegram* ditampilkan dalam bentuk bilangan bulat. Hal ini menunjukkan bahwa *bot Telegram* mampu. Koneksi internet sangat mempengaruhi pengoperasian sistem. Respons dan kinerja sistem tergantung pada kualitas koneksi internet. Koneksi yang lambat dapat mengakibatkan penundaan atau bahkan ketidakresponsifan sistem terhadap perintah.

#### 4.4.1 Hasil Pengujian Beban Listrik

Pengujian ini dilakukan terhadap beberapa beban listrik berupa peralatan listrik rumah tangga. Data yang diperoleh dari pengujian ini adalah tegangan, arus, daya kWh,

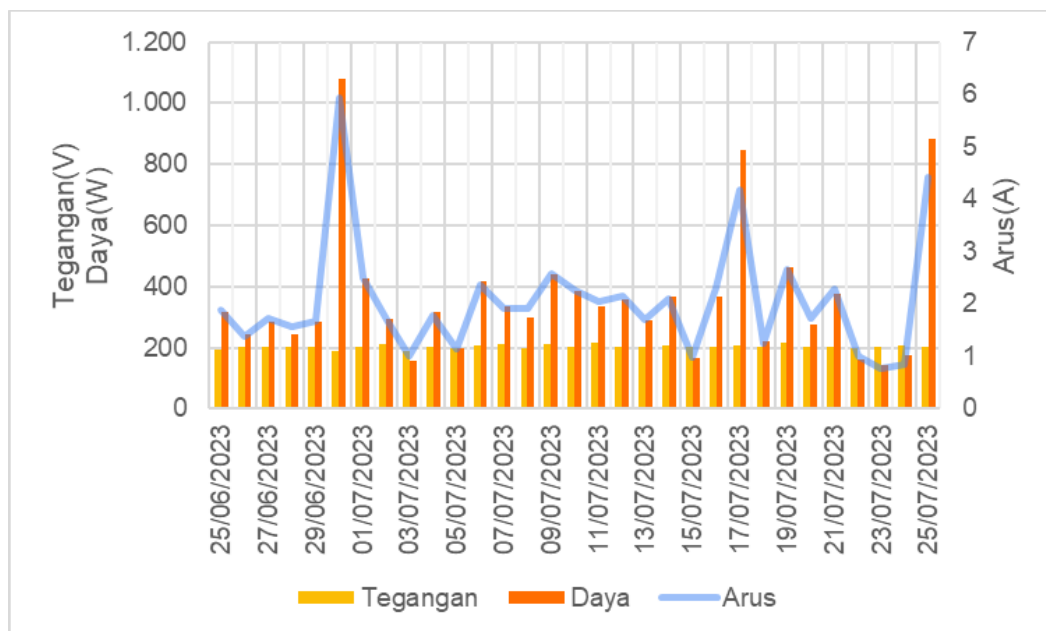


dan faktor daya pada setiap beban. Hasil pengujian dapat dilihat pada Tabel 4.1 sebagai berikut.

Tabel 4. 1 Hasil Pengujian Beban Listrik (1 Jam)

No.	Perangkat	V	I	Cos $\phi$	P
1	Kipas	202,2	0,15	0,86	26,4
2	Lampu	202,6	0,04	0,34	3
3	Setrika	205,2	1,69	1	345,9
4	Rice Cooker	198,8	0,07	0,9	12,8

Tabel 4.1 menampilkan hasil pengujian beban listrik beberapa perangkat rumah tangga yang umum digunakan sehari-hari. Waktu masing-masing perangkat rumah tangga dalam pengujian ini yaitu selama 1 jam. Data yang tercantum mencakup tegangan, arus, faktor daya dan daya. Tabel 4.1 diketahui bahwa beban lampu memiliki faktor daya terendah dengan nilai sebesar 0,34. Sedangkan faktor daya terbaik dengan nilai 1. Nilai faktor daya yang baik terdapat pada jenis beban yang bersifat induktif karena hal ini disebabkan oleh induktor yang lebih dominan.



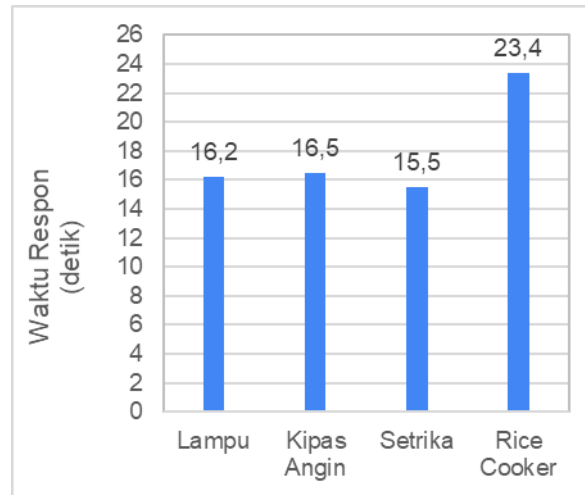
Gambar 4. 17 Grafik Hasil Pengujian Beban Listrik Rumah selama 1 Bulan

Gambar 4. 17 menampilkan hasil pengujian beban listrik pada rumah tangga selama 1 bulan dengan pemasangan alat pada MCB. Data yang ditampilkan pada grafik pada Gambar 4.17 yaitu tegangan, arus dan daya. Pada tanggal 25 Juni 2023 data yang tercatat tegangan yaitu sebesar 192,08 V, arus 1,88 A dan daya 307,20W. Terjadi peningkatan data setiap harinya yang menunjukkan dinamika penggunaan beban listrik. Tanggal 30 Juni 2023 terjadi lonjakan arus yang sangat signifikan, dengan nilai daya yang tercatat mencapai 1079,90W. puncak daya ini disebabkan oleh kombinasi tegangan dan arus pada level tertinggi dalam periode pengamatan. Pada tanggal 17 Juli 2023, terlihat data yang tercatat yaitu daya sebesar 874W dengan tegangan 206,9V dan arus 4,17A. pada tanggal 25 Juli 2023 merupakan akhir pengujian monitoring beban listrik pada rumah tangga nilai yang tercatat yaitu tegangan 204,1V, arus 4,42A dan daya 884W. Lonjakan nilai daya dapat dipengaruhi oleh penggunaan peralatan listrik rumah tangga dengan daya tinggi atau aktivitas lain yang membutuhkan daya besar pada saat pengamatan.

#### 4.4.2 Hasil Pengujian Kendali Beban Listrik

Pengujian dilakukan dengan cara pengguna mengirim perintah melalui *Bot Telegram* untuk mengendalikan atau memantau sistem, pada pengujian ini, evaluasi dilakukan terhadap kemampuan alat monitoring untuk mengendalikan perangkat listrik seperti lampu, kipas angin, setrika, dan *rice cooker*. Hasil pengujian menunjukkan bahwa alat monitoring mampu mengendalikan perangkat listrik sesuai dengan perintah yang diberikan melalui Telegram.

Selama pengujian, dicatat waktu yang diperlukan dari saat perintah dikirimkan melalui Telegram hingga perangkat listrik benar-benar berubah status (*ON* atau *OFF*). Berikut adalah lama delay yang diukur dalam detik:



Gambar 4. 18 Hasil Pengujian Kendali Beban Listrik

Gambar 4.18 merupakan grafik hasil pengujian kendali beban listrik dengan menggunakan *bot Telegram*. Beban listrik yang digunakan pada pengujian ini yaitu perangkat listrik rumah tangga seperti lampu, kipas angin, setrika, dan *rice cooker*. Waktu respon tercepat yaitu pada pengujian setrika dengan waktu selama 15,5 detik, dan waktu respon terlama yaitu *rice cooker* dengan waktu selama 23,4 detik. Rata-waktu respon yang tercatat adalah 17,9 detik pada pengujian ini wajar dan dapat diterima untuk kendali perangkat listrik dalam rumah tangga.

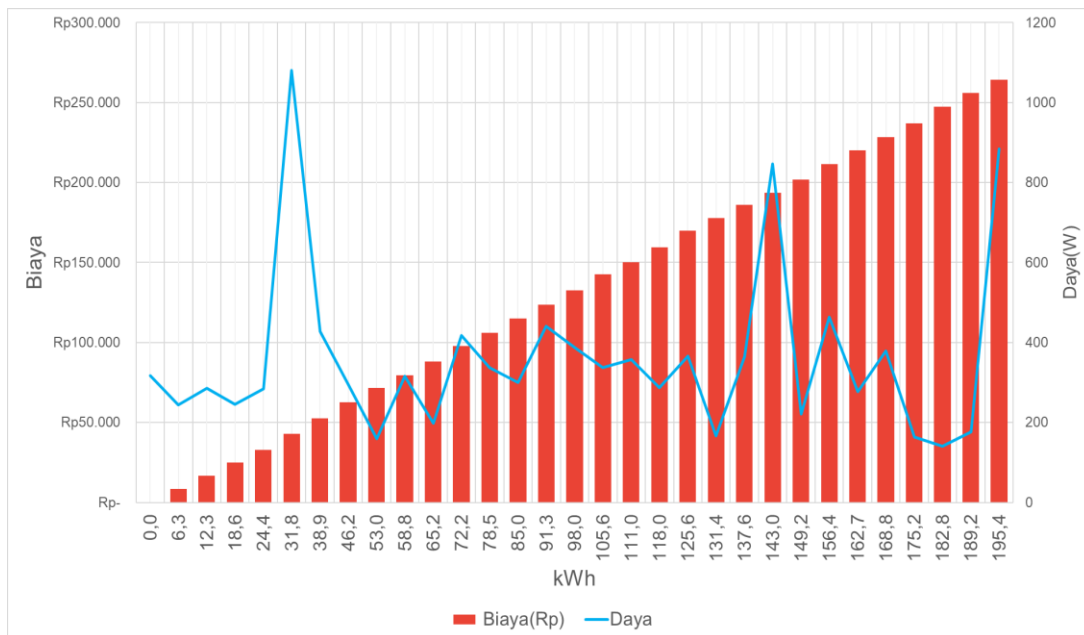
Lama *delay* yang diukur dalam pengujian kendali adalah waktu yang dibutuhkan oleh alat monitoring untuk merespons perintah dan mengendalikan perangkat listrik. *Delay* ini terjadi karena beberapa proses termasuk pengiriman perintah melalui Telegram, penerimaan perintah oleh alat monitoring, dan aktivasi relay untuk mengubah status perangkat listrik. Hal ini membuktikan pengguna dapat mengendalikan perangkat listrik jarak jauh melalui *bot Telegram*.

Peningkatan kecepatan koneksi internet dan optimalisasi perangkat keras serta perangkat lunak dapat membantu mengurangi lama *delay* ini, memastikan respon yang lebih cepat terhadap perintah yang diberikan oleh pengguna melalui *bot Telegram*.

#### 4.4.3 Hasil Pengujian Monitoring Energi dan Estimasi Biaya

Pengujian Monitoring kWh pada alat monitoring ini bertujuan untuk dapat menentukan estimasi biaya dari konsumsi daya yang digunakan pada peralatan rumah tangga. Pengujian ini dilakukan di rumah golongan tarif R-1M/TR batas daya 900VA dengan biaya per-kWh sebesar Rp.1.352/kWh. Pengamatan dalam pengujian alat monitoring ini mengumpulkan data dengan selang waktu setiap 30 detik yang akan otomatis tercatat pada *database* di *Google Sheets*.

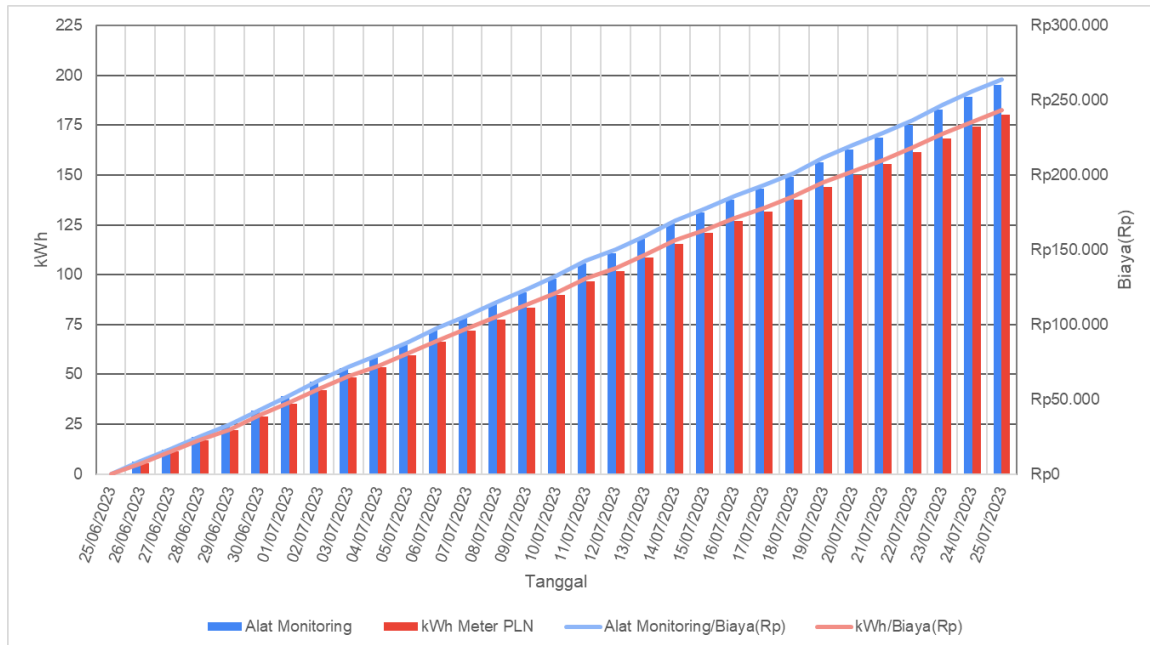
Alat monitoring ini dapat mengetahui estimasi biaya untuk konsumsi energi yang digunakan oleh peralatan listrik rumah tangga. Tarif yang digunakan dalam estimasi biaya ini sesuai dengan standar tarif PLN yang berlaku di masyarakat. Dalam pengujian ini dilakukan perhitungan estimasi biaya konsumsi daya oleh beberapa beban peralatan rumah tangga. Adapun diantaranya dapat dilihat pada Gambar 4.20.



Gambar 4. 19 Grafik Daya dan Estimasi Biaya Penggunaan Energi Listrik dalam Waktu 1 Bulan

Berdasarkan data yang diperoleh pada Gambar 4.19 merupakan hasil pengujian monitoring energi selama 1 bulan. Monitoring dilakukan dari tanggal 25/06/2023 sampai

25/07/2023. Peningkatan nilai kWh pada grafik terus meningkat setiap harinya. Konsumsi daya tertinggi pada periode ini yaitu pada tanggal 30/06/2023 dengan daya tercatat yaitu 1079,90W dan kWh tercatat yaitu 31,8. Daya tertinggi kedua yaitu pada tanggal 17/07/2023 dengan daya tercatat yaitu 847W dan kWh tercatat yaitu 143,03. Konsumsi daya terendah pada periode ini yaitu pada tanggal 03/07/2023 dengan daya tercatat yaitu 158,9W dan kWh tercatat yaitu 53. Konsumsi daya terendah juga terjadi pada tanggal 23/07/2023 dengan daya tercatat yaitu 141W dan kWh tercatat yaitu 182,8. Grafik menunjukkan bahwa pada tanggal 25/06/2023, konsumsi energi adalah 0 kWh, sebagai waktu awal pemasangan alat monitoring, namun, terjadi peningkatan konsumsi energi secara bertahap hingga mencapai puncaknya pada tanggal 25/07/2023. Hal ini mungkin disebabkan oleh peningkatan aktivitas rumah tangga dan penggunaan perangkat selama periode ini. Estimasi biaya yang berbeda pada setiap harinya menunjukkan variasi dalam penggunaan energi selama hari. Peningkatan konsumsi energi seiring berjalannya waktu dapat berdampak pada biaya listrik yang dibayarkan oleh pengguna. Terlihat bahwa biaya listrik meningkat seiring dengan peningkatan konsumsi energi. Grafik pada Gambar 4.20 menyatakan bahwa terjadi peningkatan nilai kWh seiring berjalannya waktu. Hal ini disebabkan karena konsumsi daya yang dilakukan secara terus-menerus dengan sifat konsumsi daya yang dinamis. Selama adanya konsumsi daya pada beban peralatan rumah tangga, maka alat monitoring ini akan mencatat nilai kWh. Beban yang mengkonsumsi daya besar maka akan menghasilkan kWh yang sebanding lurus dengan tarif yang harus dibayarkan oleh pelanggan. Penggunaan peralatan rumah tangga yang boros energi akan memperbesar biaya tagihan.



Gambar 4. 20 Grafik Perbandingan Nilai kWh PLN dan Alat Monitoring

Gambar 4.20 memperlihatkan grafik perbandingan nilai kWh dan estimasi biaya antara alat monitoring dan kWh meter milik PLN, selama 1 bulan. Grafik menunjukkan terdapat selisih nilai kWh antara alat monitoring kendali dan kWh meter milik PLN. Pada Pemantauan dilakukan dari tanggal 25 Juni 2023 hingga 25 Juli 2023 dengan pemasangan alat monitoring kendali pada MCB rumah. Nilai awal saat pengukuran yaitu 0 kWh dengan biaya Rp. 0, pada tanggal-tanggal berikutnya terdapat perbedaan nilai kWh dan estimasi biaya antara kedua alat tersebut. Nilai *error* berkisar antara 0,08 hingga 0,10, yang menunjukkan perbedaan yang cukup kecil dalam pengukuran. Alat monitoring menunjukkan nilai kWh dan estimasi biaya yang berbeda dengan kWh meter PLN, seperti pada tanggal 26 Juni 2023 dengan nilai kWh 6,34 dan estimasi biaya Rp. 8573 untuk alat monitoring, sedangkan untuk kWh meter PLN mencatat 5,3 dengan estimasi biaya RP.7166. Perbedaan terjadi dalam rentang tanggal pengukuran berikutnya, menunjukkan variasi hasil antar kedua alat tersebut dalam membaca penggunaan energi listrik rumah tangga.

Meskipun *error* ini relatif kecil dalam persentase, penting untuk diingat bahwa dalam jumlah besar, perbedaan ini dapat berdampak signifikan pada tagihan listrik. Oleh karena itu, pemeliharaan dan kalibrasi yang teratur pada alat monitoring menjadi penting untuk memastikan akurasi pengukuran yang konsisten. Penyebab dari perbedaan kecil ini, dapat berasal dari karakteristik sensor pada alat monitoring, yang dapat berbeda dengan karakteristik sensor pada kWh Meter PLN. Selain itu, faktor lingkungan seperti suhu dan kelembaban juga dapat memengaruhi hasil pengukuran. Untuk meningkatkan akurasi Alat Monitoring, langkah-langkah perbaikan perlu dipertimbangkan. Ini dapat mencakup kalibrasi yang lebih sering, pemantauan lingkungan sekitar yang lebih ketat, dan pemeliharaan rutin pada perangkat tersebut.

Analisis data ini menunjukkan bahwa alat ini memiliki nilai selisih yang cukup kecil dengan kWh meter tradisional, alat ini dapat dianggap sebagai solusi alternatif untuk pemantauan energi pada listrik rumah tangga. Evaluasi lebih lanjut dan pemeliharaan yang tepat diperlukan untuk memastikan bahwa alat ini dapat memenuhi berjalan dengan baik dan lancar.

## **BAB V**

### **PENUTUP**

#### **5.1 Kesimpulan**

Setelah melalui tahap perancangan, pengujian dan pembahasan hasil pengujian Secara umum dari skripsi ini dapat ditarik beberapa kesimpulan sebagai beriku.

1. Alat monitoring telah berhasil mengatasi kurangnya pemantauan penggunaan konsumsi energi listrik di rumah tangga. Alat ini memudahkan pengguna dalam memantau penggunaan energi listrik secara *realtime*. Penggunaan LCD pada alat untuk melihat data secara langsung, serta *Google Sheets* sebagai databasenya. Berdasarkan data pada Tabel 4.7, persen kesalahan yang diperoleh dari perbandingan antara alat monitoring dan kWh meter milik meter PLN yaitu sebesar 10%.
2. Alat monitoring berbasis IoT berhasil meringankan pengguna dalam memantau dan mengetahui konsumsi serta biaya listrik dalam kehidupan sehari-hari. Data rata-rata menunjukkan konsistensi dalam pengukuran, dengan tingkat rata-rata persen kesalahan sekitar 0,18% untuk tegangan dan 2% untuk arus, mengindikasikan pengukuran alat berjalan dengan baik.
3. Pengendalian beban listrik jarak jauh via telegram memberikan kemudahan kepada pengguna dalam mengelola dan mengatur pemakaian energi listrik sesuai kebutuhan yang sebelumnya tidak terdapat pada kWh meter konvensional. Data pengujian kendali *ON/OFF* pada beban listrik menunjukkan rata-rata waktu respon sekitar 17,9 detik. Hal ini meningkatkan fleksibilitas pengguna dalam mengatur dan mengelola pemakaian energi listrik sesuai kebutuhan.
4. Implementasi Telegram sebagai media untuk memberikan informasi dan kendali terkait penggunaan listrik kepada pengguna telah berhasil. Bot Telegram berkomunikasi dengan alat monitoring dengan memberikan respon sesuai pesan perintah yang diberikan.



## 5.2 Saran

Penelitian ini dapat dikembangkan lebih lanjut guna meningkatkan kemampuan alat tersebut dengan mempertimbangkan saran-saran berikut:

1. Penambahan *keypad* atau *push button* untuk kendalialat apabila tidak terhubung dengan jaringan internet.
2. Tinjau dan perbarui secara berkala *library* yang digunakan pada alat monitoring untuk memastikan ketersediaan fitur dan performa yang optimal.
3. Perluasan fungsionalitas alat monitoring dapat dipertimbangkan untuk meningkatkan kemampuan pemantauan dan kendali, seperti integrasi dengan perangkat lain dan penambahan sensor tambahan

## DAFTAR PUSTAKA

- [1] H. H. Alhelou, M. E. Hamedani-Golshan, T. C. Njenda, and P. Siano, “A Survey on Power System Blackout and Cascading Events: Research Motivations and Challenges,” *Energies*, vol. 12, no. 4, pp. 1–28, 2019, doi: 10.3390/en12040682.
- [2] M. Nasar, N. Setyawan, A. Faruq, and I. Sulistiyowati, “A Simple Real-Time Energy Analytics Model for Smart Building Using Open IoT Platforms,” *J. Elektron. dan Telekomun.*, vol. 19, no. 2, p. 83, 2019, doi: 10.14203/jet.v19.83-90.
- [3] PT. PLN (Persero), “Statistik PLN 2021 [PLN Statistic 2021],” *Jakarta*, 2021.
- [4] S. Darma, Yusmartato, and Akhiruddin, “Studi Sistem Peneraan KWH Meter,” vol. 4, no. 3, pp. 158–165, 2019.
- [5] Pemerintah Indonesia, *Undang-Undang Republik Indonesia Nomor 30 Tahun 2009 tentang Ketenagalistrikan*. Jakarta: Sekretariat Indonesia, 2009.
- [6] J. A. Hassan and B. H. Jasim, “Design and implementation of internet of things-based electrical monitoring system,” *Bull. Electr. Eng. Informatics*, vol. 10, no. 6, pp. 3052–3063, 2021, doi: 10.11591/eei.v10i6.3155.
- [7] R. R. A. Siregar, Y. K. Ningsih, P. Palupiningsih, and B. Prayitno, “Smart kWh Meter Model with Energy Control and Monitoring on Low Voltage Electricity,” vol. 198, no. Issat, pp. 227–232, 2020, doi: 10.2991/aer.k.201221.039.
- [8] M. Hayaty and A. R. Mutmainah, “IoT-Based electricity usage monitoring and controlling system using Wemos and Blynk application,” *J. Teknol. dan Sist. Komput.*, vol. 7, no. 4, pp. 161–165, 2019, doi: 10.14710/jtsiskom.7.4.2019.161-165.
- [9] Apriela Trirahma, “Telegram Bot as a Data Collection Tool for Progress Reports in Area Mapping Progress Monitoring System,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 6, pp. 1182–1192, 2021, doi: 10.29207/resti.v5i6.3654.
- [10] M. Muslih *et al.*, “Developing smart workspace based iot with artificial intelligence using telegram chatbot,” *Proc. - 2018 4th Int. Conf. Comput. Eng. Des. ICCED 2018*,

- pp. 230–234, 2019, doi: 10.1109/ICCED.2018.00052.
- [11] S. L. Zaen and A. Rozaq, “Sistem Monitoring Pemakaian Energi Listrik Rumah Tangga Berbasis Web,” *J. ELKON*, vol. 01, no. 01, pp. 2809–140, 2021.
- [12] S. U. Alam, R. Ahmed, M. S. Imam, M. Farshid, M. A. Hossain, and M. A. Islam, “Design and Implementation of Website based Energy Consumption Monitoring and Controlling,” *2019 Int. Conf. Comput. Commun. Informatics, ICCCI 2019*, pp. 1–7, 2019, doi: 10.1109/ICCCI.2019.8821978.
- [13] I. G. W. Artana, W. G. Ariastina, and I. N. S. Kumara, “Rancang Bangun Sistem Pemantau Suhu Bearing Motor Untuk Pompa Sirkulasi Air Berbasis Iot,” *J. SPEKTRUM*, vol. 9, no. 2, p. 20, 2022, doi: 10.24843/spektrum.2022.v09.i02.p3.
- [14] J. Cunha, M. Erwig, and J. Saraiva, “Automatically inferring classsheet models from spreadsheets,” *Proc. - 2010 IEEE Symp. Vis. Lang. Human-Centric Comput. VL/HCC 2010*, pp. 93–100, 2010, doi: 10.1109/VLHCC.2010.22.
- [15] A. Gupta and S. Gupta, “Telegram vs WhatsApp: A Comparison of Security Features and User Preferences,” in *Proceedings of the 4th International Conference on Internet of Things and Connected Technologies (ICIOTCT)*, 2018, pp. 973–97.
- [16] J. Siswanto, E. Susantini, and B. Jatmiko, *Fisika Dasar, Seri: Listrik Arus Searah dan Kemagnetan*. Semarang: UPGRIS Press, 2018.
- [17] P. P. Urone, R. Hinrichs, F. Gozycak, D. Pattison, and C. Tabor, *Physics High school*. Houston, Texas: Rice University, 2020.
- [18] H. Makhabbah and A. I. Agung, “Rancang Bangun Sistem Monitoring Konsumsi Daya Listrik Dan Pemutus Daya Otomatis Berbasis Internet,” 2018.
- [19] N. N. S. A. H. A. Bakar, A. I. M. Sabirin, “Classification of Reactive Power Compensation Techniques: A Review,” in *Renewable and Sustainable Energy Reviews, vol. 41*, 2015, pp. 1099–1107.
- [20] S. W. Nourildean, M. D. Hassib, and Y. A. Mohammed, “Internet of things based wireless sensor network: a review,” vol. 27, no. 1, pp. 246–261, 2022, doi: 10.11591/ijeecs.v27.i1.pp246-261.
- [21] Qusay F. Hassan, *Internet of things A to Z*. Hoboken, New Jersey: John Wiley & Sons,

Inc, 2018.

- [22] Y. Rianto, “Mendeteksi gerakan kamera menggunakan wemos d1 r1 berbasis iot,” no. 100, pp. 1–28, 2020.
- [23] H. Santoso, *Panduan Praktis Arduino untuk Pemula*. Trenggalek: Elang Sakti, 2016.
- [24] M. R. Alfariski, M. Dhandi, and A. Kiswantono, “Automatic Transfer Switch (ATS) Using Arduino Uno, IoT-Based Relay and Monitoring,” *JTECS J. Sist. Telekomun. Elektron. Sist. Kontrol Power Sist. dan Komput.*, vol. 2, no. 1, p. 1, 2022, doi: 10.32503/jtecs.v2i1.2238.
- [25] N. A. Hussien, A. A. D. Al Magsoosi, A. A. D. Al Magsoosi, H. T. Salim AlRikabi, and F. T. Abed, “Monitoring the Consumption of Electrical Energy Based on the Internet of Things Applications,” *Int. J. Interact. Mob. Technol.*, vol. 15, no. 7, pp. 17–29, 2021, doi: 10.3991/ijim.v15i07.20183.
- [26] T. B. Nguyen and T. C. Nguyen, “Design and fabrication of an IoT-based smart electrical meter for residential energy management,” vol. 30, no. 3, pp. 1259–1268, 2023, doi: 10.11591/ijeecs.v30.i3.pp1259-1268.
- [27] I. B. F. Citarsa, I. M. A. Nrartha, and R. Hidayat, “Rancang Bangun Smart Kwh Meter 3 Fase Dengan Komunikasi SMS Gateway,” *Dielektrika*, vol. 7, no. 2, p. 140, 2020, doi: 10.29303/dielektrika.v7i2.246.
- [28] Telegram Org, “Telegram Bot.” <https://telegram.org/blog/bot-revolution> (accessed Dec. 15, 2021).

## **LAMPIRAN**

**LAMPIRAN A**  
**DOKUMENTASI PENELITIAN**



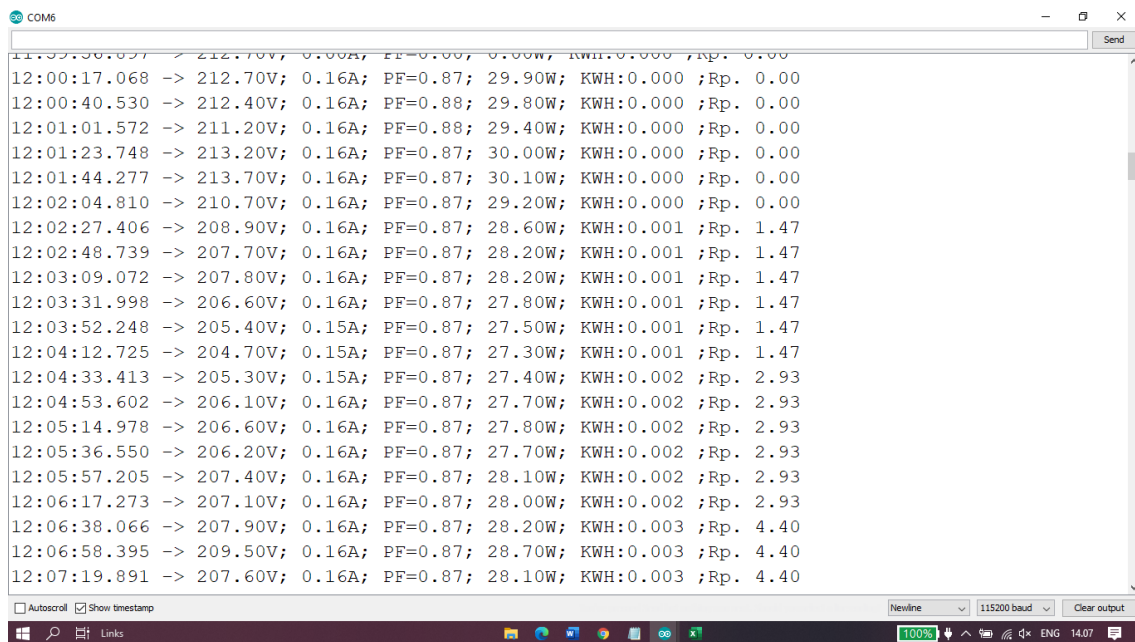
Gambar A. 1 Pengujian Wemos D1R1



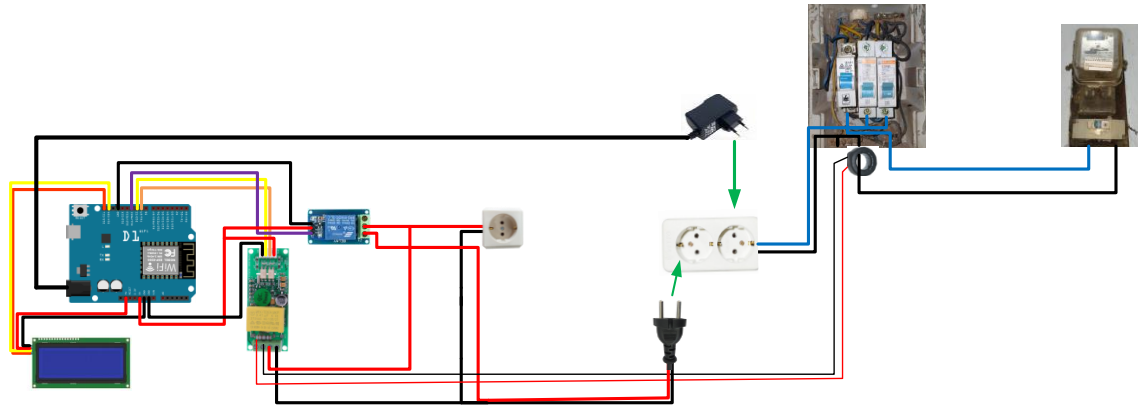
Gambar A. 2 Pengujian Relay dan LCD



Gambar A. 3 Pengujian Alat Monitoring

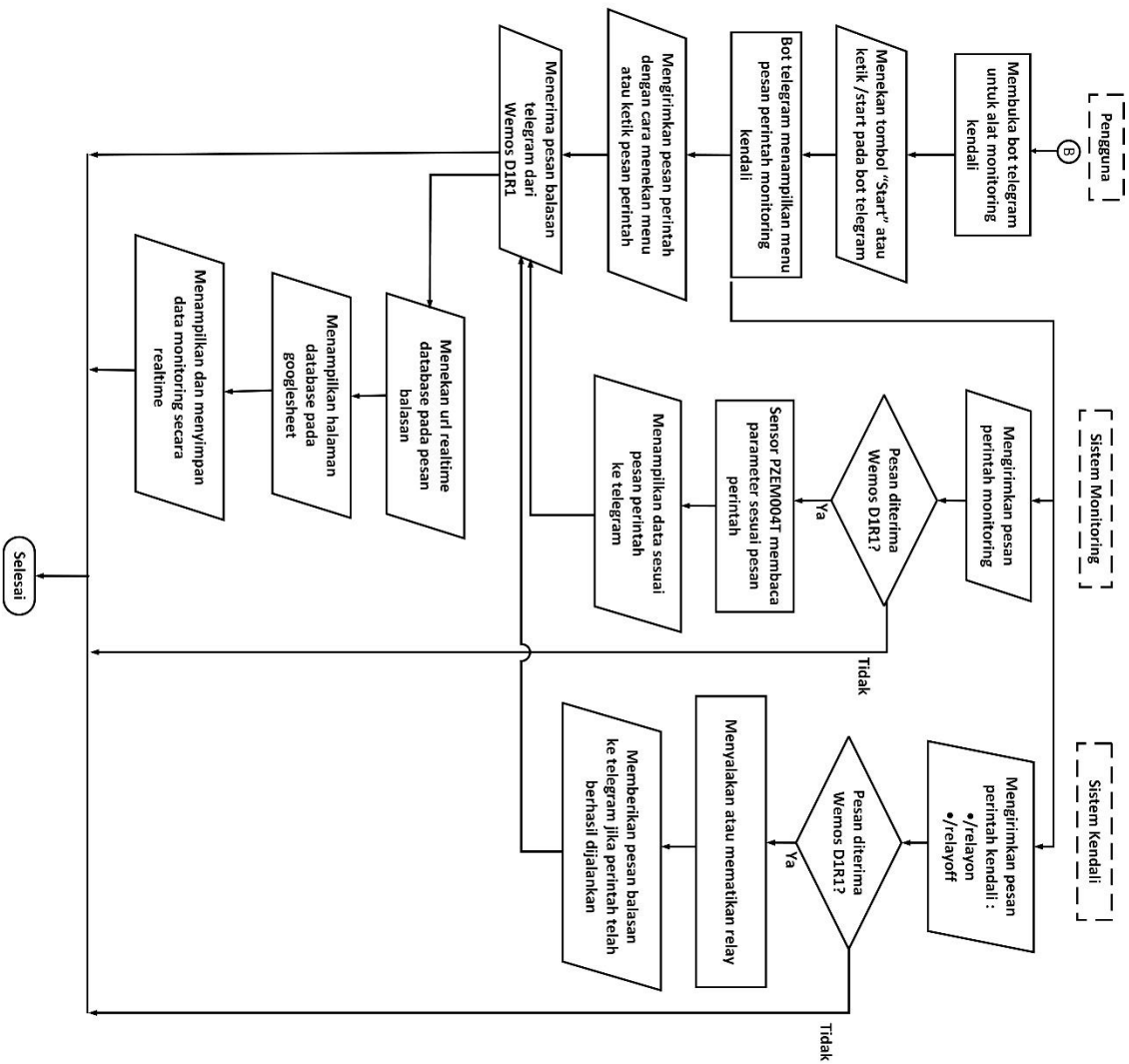
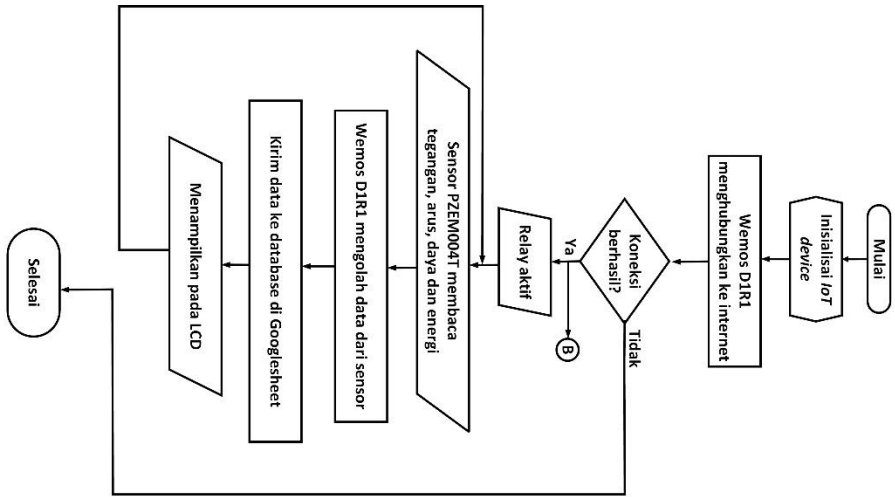


Gambar A. 4 Tampilan Data pada Arduino IDE



Gambar A. 5 Pengujian pada Beban Listrik Rumah Tangga





Tabel A- 1 Library yang digunakan pada Arduino IDE

No.	<i>Library</i>	<b>Deskripsi</b>
1	#include <TRIGGER_WIFI.h>	Mengimport <i>library</i> TRIGGER_WIFI, yang digunakan untuk mengirimkan data daya listrik ke bot Telegram dan <i>Google Sheets</i> .
2	#include <TRIGGER_GOOGLE_SHEETS.h>	Mengimport <i>library</i> TRIGGER_GOOGLE_SHEETS, yang digunakan untuk menyimpan data daya listrik di <i>Google Sheets</i> .
3	#include <ESP8266WiFi.h>	Mengimport <i>library</i> ESP8266WiFi, yang digunakan untuk koneksi WiFi.
4	#include <PZEM004Tv30.h>	Mengimport <i>library</i> PZEM004Tv30, yang digunakan untuk membaca data daya listrik dari modul PZEM004Tv30.
5	#include <LiquidCrystal_I2C.h>	Mengimport <i>library</i> LiquidCrystal_I2C, yang digunakan untuk menampilkan data daya listrik di LCD.
6	#include <Wire.h>	Mengimport <i>library</i> Wire, yang digunakan untuk komunikasi I2C.
7	#include <WiFiClientSecure.h>	Mengimport <i>library</i> WiFiClientSecure, yang digunakan untuk koneksi WiFi yang aman.
8	#include <Firebase.h>	Mengimport <i>library</i> Firebase, yang digunakan untuk menyimpan data daya listrik di Firebase.
9	#include <TimeLib.h>	Mengimport <i>library</i> TimeLib, yang digunakan untuk manajemen waktu.
10	#include <WiFiUdp.h>	Mengimport <i>library</i> WiFiUdp, yang digunakan untuk koneksi WiFi UDP.
11	#include <NTPClient.h>	Mengimport <i>library</i> NTPClient, yang digunakan untuk klien NTP.
12	#include <UniversalTelegramBot.h>	Mengimport <i>library</i> UniversalTelegramBot, yang digunakan untuk berinteraksi dengan bot Telegram.

Tabel A- 2 Variabel yang digunakan pada Arduino IDE

Konstanta/Variabel	Deskripsi
RX dan TX	Pin yang digunakan untuk komunikasi serial dengan modul PZEM004Tv30.
rl	Pin yang digunakan untuk mengendalikan relay.
FIREBASE_HOST, FIREBASE_AUTH	Kredensial yang digunakan untuk mengakses Firebase.
ssid1, password1, ssid2, dan password2	SSID dan password untuk dua jaringan WiFi yang mungkin digunakan.
ntpServer	Nama server NTP yang digunakan untuk sinkronisasi waktu.
column_name_in_sheets	Nama kolom di <i>Google Sheets</i> tempat data daya listrik akan disimpan.
Sheets_GAS_ID	ID spreadsheet <i>Google Sheets</i> yang digunakan untuk menyimpan data.
No_of_Parameters	Jumlah parameter yang akan dikirim ke <i>Google Sheets</i> .
BOTtoken	Token bot Telegram yang digunakan untuk berinteraksi dengan bot Telegram.
CHAT_ID	ID chat bot Telegram yang digunakan untuk mengirim dan menerima pesan.

**LAMPIRAN B**  
**DATA PENELITIAN**

Tabel B- 1 Rekapitulasi Data Pengujian selama 1 Bulan (25 Juni 2023-25 Juli 2023)

<b>Tanggal</b>	<b>Tegangan</b>	<b>Arus</b>	<b>Daya</b>	<b>kWh</b>	<b>Biaya (Rp)</b>
25/06/2023	192.80	1.88	317.20	0,00	Rp0
26/06/2023	201.20	1.38	243.50	6,34	Rp8.573
27/06/2023	203.60	1.74	285.30	12,27	Rp16.592
28/06/2023	204.30	1.57	245.60	18,56	Rp25.099
29/06/2023	204.10	1.68	283.70	24,41	Rp33.008
30/06/2023	187.90	5.94	1079.90	31,80	Rp42.990
01/07/2023	200.70	2.51	428.30	38,91	Rp52.601
02/07/2023	211.80	1.71	295.50	46,23	Rp62.507
03/07/2023	188.50	01.01	158.90	53,01	Rp71.671
04/07/2023	201.20	1.79	315.80	58,78	Rp79.467
05/07/2023	200.10	1.14	198.20	65,21	Rp88.157
06/07/2023	205.40	2.38	417.90	72,24	Rp97.663
07/07/2023	209.80	1.91	336.70	78,50	Rp106.135
08/07/2023	197.30	1.91	299.30	84,96	Rp114.861
09/07/2023	209.70	2.59	440.10	91,28	Rp123.405
10/07/2023	202.40	2.26	387.30	97,97	Rp132.460
11/07/2023	214.30	02.05	336.80	105,56	Rp142.710
12/07/2023	204.50	2.16	357.40	110,96	Rp150.019
13/07/2023	204.00	1.70	287.20	117,98	Rp159.500
14/07/2023	205.80	2.10	365.30	125,60	Rp169.804
15/07/2023	201.60	0.99	166.30	131,40	Rp177.647
16/07/2023	202.70	2.32	365.10	137,56	Rp185.982
17/07/2023	206.90	4.17	847.00	143,03	Rp193.375
18/07/2023	203.10	1.25	221.50	149,18	Rp201.693
19/07/2023	214.20	2.65	463.30	156,42	Rp211.477
20/07/2023	202.10	1.73	276.40	162,70	Rp219.976
21/07/2023	201.80	2.30	378.30	168,77	Rp228.177
22/07/2023	198.40	1.00	163.00	175,16	Rp236.811
23/07/2023	204.30	0.78	141.00	182,84	Rp247.197
24/07/2023	205.20	0.86	176.30	189,18	Rp255.769
25/07/2023	204.10	4.42	884.00	195,37	Rp264.142

## LAMPIRAN C

### LISTING PROGRAM

a. Listing program pada *AppScript Google Sheets*

```
function doGet (e) {
    Logger.log ( JSON.stringify (e) ); // view parameters
    var result = 'Ok'; // assume success
    if (e.parameter == 'undefined') {
        result = 'No Parameters';
    }
    else {
        var sheet_id = 'kodeunikpadalinkwebsite'; // Spreadsheet ID
        var sheet = SpreadsheetApp.openById (sheet_id).getActiveSheet
(); // get Active sheet
        var newRow = sheet.getLastRow () + 1;
        var rowData = [];
        d=new Date ();
        rowData[0] = d; // Timestamp in column A
        rowData[1] = d.toLocaleTimeString (); // Timestamp in column A

        for (var param in e.parameter) {
            Logger.log ('In for loop, param=' + param);
            var value = stripQuotes (e.parameter[param]);
            Logger.log (param + ':' + e.parameter[param]);
            switch (param) {
                case 'value1': //Parameter 1, It has to be updated in Column in
Sheets in the code, orderwise
                    rowData[2] = value; //Value in column A
                    result = 'Written on column A';
                    break;
                case 'value2': //Parameter 2, It has to be updated in Column in
Sheets in the code, orderwise
                    rowData[3] = value; //Value in column B
                    result += ' Written on column B';
                    break;
            }
        }
    }
}
```

```

        case 'value3': //Parameter 3, It has to be updated in Column in
Sheets in the code, orderwise
            rowData[4] = value; //Value in column C
            result += ' Written on column C';
            break;
        case 'value4': //Parameter 1, It has to be updated in Column in
Sheets in the code, orderwise
            rowData[5] = value; //Value in column A
            result = 'Written on column D';
            break;
        case 'value5': //Parameter 2, It has to be updated in Column in
Sheets in the code, orderwise
            rowData[6] = value; //Value in column B
            result += ' Written on column E';
            break;
        case 'value6': //Parameter 3, It has to be updated in Column in
Sheets in the code, orderwise
            rowData[7] = value; //Value in column C
            result += ' Written on column F';
            break;
        default:
            result = "unsupported parameter";
    }
}
Logger.log (JSON.stringify (rowData));
// Write new row below
var newRange = sheet.getRange (newRow, 1, 1, rowData.length);
newRange.setValues ([rowData]);
}
// Return result of operation
return ContentService.createTextOutput (result);
}
function stripQuotes ( value ) {
    return value.replace (/^["']|['"]$/g, "");
}
}

```

b. Listing program pada Arduino

```
//Library
#include <TRIGGER_WIFI.h> // Mengimpor library TRIGGER_WIFI yang
diperlukan
#include <TRIGGER_GOOGLESHEETS.h> // Mengimpor library
TRIGGER_GOOGLESHEETS yang diperlukan

#include <ESP8266WiFi.h> // Mengimpor library ESP8266WiFi untuk
koneksi WiFi

#include <PZEM004Tv30.h> // Mengimpor library PZEM004Tv30 untuk sensor
arus dan tegangan
#include <LiquidCrystal_I2C.h> // Mengimpor library LiquidCrystal_I2C
untuk tampilan LCD
#include <Wire.h> // Mengimpor library Wire untuk komunikasi I2C

#include <WiFiClientSecure.h>
#include <Firebase.h>

#include <TimeLib.h> // Mengimpor library TimeLib untuk manajemen
waktu
#include <WiFiUdp.h> // Mengimpor library WiFiUdp untuk koneksi WiFi
UDP
#include <NTPClient.h> // Mengimpor library NTPClient untuk klien NTP

#include <UniversalTelegramBot.h> // Mengimpor library
UniversalTelegramBot untuk berinteraksi dengan bot Telegram

#define RX 12
#define TX 13
#define rl 14

#define FIREBASE_HOST "internetorwifi-default-rtdb.firebaseio.com"
#define FIREBASE_AUTH "RsPOnJqa2AEAFpR6ciV74oGfg4EyMwBnC4GCAbJZ"
```

```

LiquidCrystal_I2C lcd(0x27, 20, 4);
PZEM004Tv30 pzem(12, 13); // (RX,TX) connect to TX,RX of PZEM
float voltage, current, power, pf, energy, cost;
float harga = 1352;

int currentDay = 0;
int currentMonth = 0;
bool isEndOfMonth = false; // Flag untuk pengecekan akhir bulan
bool isStartOfMonth = false; // Flag untuk pengecekan awal bulan

const char *ssid1 = "namawifil";
const char *password1 = "passwordwifil";

const char *ssid2 = " namawifi ";
const char *password2 = " passwordwifi2";

const char* ntpServer = "pool.ntp.org";

char column_name_in_sheets[][20] = {"value1", "value2", "value3",
"value4", "value5", "value6"};
const char* Sheets_GAS_ID = "kodeGASIDnya";
int No_of_Parameters = 6;

const char* BOTtoken = "Bot Token Telegram";
const int CHAT_ID = Nomor _ID_Akun_Telegram;
const String SENDER_ID = " Nomor ID Akun Telegram ";

UniversalTelegramBot bot(BOTtoken, client);

unsigned long lastTimeBotRan = 0;
const unsigned long botRequestDelay = 1000;

bool sendData = false;

WiFiUDP ntpUDP;

```



```

NTPClient timeClient(ntpUDP, ntpServer);

void Google_Sheets_ClearData() {
    // Implementasi untuk menghapus data pada spreadsheet
    // Kirim pesan ke bot Telegram
    String message = "Data pada spreadsheet telah direset.";
    bot.sendMessage("Nomor ID Akun Telegram", message);
    Serial.println("Reset Terkirim");
}

void handleNewMessages(int numNewMessages, String chat_id, String
message_sender_id) {
    for (int n = 0; n < numNewMessages; n++) {
        //String chat_id = String(bot.messages[n].chat_id);
        String text = bot.messages[n].text;
        String from_name = bot.messages[n].from_name;

        if (chat_id == String(CHAT_ID))
        {
            if (from_name == "")from_name = "Guest";
        }
        else
        {
            String errorMessage = "Anda tidak diizinkan mengakses bot ini.";
            bot.sendMessage(String(chat_id).c_str(), errorMessage);
        }

        if (message_sender_id == String(SENDER_ID)){
            //Cek Pembacaan PZEM
            if (text == "/tegangan")
            {
                int v = pzem.voltage();
                if (v < 0.0) v = 0.0;

                String tegangan = "Tegangan : ";
                tegangan += int(v);
            }
        }
    }
}

```

```

    tegangan += "\n";

    bot.sendMessage(chat_id,tegangan, "");
}

if (text == "/arus")
{
    float i = pzem.current();
    if (i < 0.000) i = 0.000;

    String arus = "Arus : ";
    arus += float(i);
    arus += "A";

    bot.sendMessage(chat_id, arus, "");
}

//Cek Pembacaan Sensor DHT11
if (text == "/daya")
{
    int p = pzem.power();
    if (p < 0.0) p = 0.0;
    String daya = "Daya : ";
    daya += int(p);
    daya += "W";

    bot.sendMessage(chat_id, daya, "");
}

if (text == "/kwh")
{
    float e = pzem.energy();
    String kwh = "kWh : ";
    kwh += float(e);
}

```

```

    bot.sendMessage(chat_id,kwh, "");
}

if (text == "/biaya")
{
    float cost = (pzem.energy()*1352);
    String biaya = "Rp. ";
    biaya += float(cost);
    biaya += "";

    bot.sendMessage(chat_id,biaya, "");
}

//Kontrol relay
if (text == "/reset") {
    digitalWrite(rl, HIGH);
    pzem.resetEnergy();
    bot.sendMessage(chat_id, "Reset", "");
}
if (text == "/resetdata") {
    digitalWrite(rl, HIGH);
    Google_Sheets_ClearData();
    bot.sendMessage(chat_id, "Reset OK", "");
}

if (text == "/relayoff") {
    digitalWrite(rl, HIGH);
    bot.sendMessage(chat_id, "Relay OFF", "");
}
if (text == "/relayon") {
    digitalWrite(rl, LOW);
    bot.sendMessage(chat_id, "Relay ON", "");
}
}

```

```

//Cek Command untuk kondisi semua
if (text == "/kondisi") {
    int v = pzem.voltage();
    float i = pzem.current();
    float p = pzem.power();
    float e = pzem.energy();
    int cost = (e*1352);
    String kondisi;
    kondisi+= (String)"Pemantauan Monitoring Listrik\n\n" +
        (String)"Tegangan : " + v + (String)"V\n\n"+
        (String)"Arus : " + i + (String)"A\n\n"+
        (String)"Daya : " + p + (String)"W\n\n"+
        (String)"KWH : " + e + "\n\n"+
        (String)"Biaya : Rp. " + cost + (String)"\n\n"+
        "KLIK DISINI: link Google Sheetsnya ";
    bot.sendMessage(chat_id, kondisi);
}

//Cek Command untuk setiap aksi
if (text == "/start") {
    String welcome = "Welcome " + from_name + ".\n";
    welcome += "/tegangan : Status tegangan\n";
    welcome += "/arus : Status arus\n";
    welcome += "/daya : Status daya\n";
    welcome += "/kwh : Status energi\n";
    welcome += "/biaya : Status biaya\n";
    welcome += "/kondisi : Data saat ini\n";
    welcome += "/relayon : Nyalakan relay\n";
    welcome += "/relayoff : Matikan relay\n";
    welcome += "/reset : Reset data\n";
    bot.sendMessage(chat_id, welcome, "Markdown");
}

}

}

```

```

}

void setup() {
  pinMode(rl, OUTPUT);
  digitalWrite(rl, LOW);
  Serial.begin(115200);
  lcd.begin();
  lcd.backlight();

  while (WiFi.status() == WL_DISCONNECTED) {
    myNetwork();
  }

  Serial.println("Successfully Connected ");

  Google_Sheets_Init(column_name_in_sheets,          Sheets_GAS_ID,
No_of_Parameters);

  timeClient.begin();
  timeClient.setTimeOffset(7 * 3600);

  currentDay = day();
  currentMonth = month();
}

void myNetwork() {
  int flag = 1;
  Serial.println("Searching Wifi.....");
  lcd.print("Connecting...");
  int network = WiFi.scanNetworks();          //5
  for (int i = 0 ; i < network; i++) {
    switch (flag) {
      case 1:
        flag =2;
        if (WiFi.SSID(i) == ssid1) {

```

```

        WiFi.begin(ssid1, password1);
        Serial.println("/n Wifi Found");
        delay(2000);
        Serial.println("Connecting with Wifi Please Wait ");
        delay(8000);
        Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
        break;
    }
    case 2:
        flag =1;
        if (WiFi.SSID(i) == ssid2) {
            WiFi.begin(ssid2, password2);
            Serial.println("/n Wifi Found");
            delay(2000);
            Serial.println("Connecting with Hostpot Please Wait ");
            delay(8000);
            Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
            break;
        }
    }
}

client.setInsecure();
Serial.println("");
Serial.println("Successfully Connected");
lcd.clear();
lcd.setCursor(0,0);
lcd.print("WiFi Connected");
delay(1000);
lcd.clear();
}

void loop() {
    timeClient.update();
    setTime(timeClient.getEpochTime());
}

```

```

voltage = pzem.voltage();
current = pzem.current();
power = pzem.power();
pf = pzem.pf();
energy = pzem.energy();
if (isnan(energy)) energy = 0.0;
int harga = 1352;
float cost = energy * harga;
if (isnan(cost)) {
    cost = 0.0;
}

lcd.clear();

Serial.print(voltage); Serial.print("V; ");
lcd.setCursor(0, 0);
lcd.print("V= ");
lcd.setCursor(2, 0);
lcd.print(voltage);

Serial.print(current); Serial.print("A; ");
lcd.setCursor(11, 0);
lcd.print("I= ");
lcd.setCursor(13, 0);
lcd.print(current);

Serial.print(power); Serial.print("W; ");
lcd.setCursor(11, 1);
lcd.print("P= ");
lcd.setCursor(13, 1);
lcd.print(power);

Serial.print(pf); Serial.print("");

Serial.print(energy); Serial.print("kWh; ");
lcd.setCursor(0, 2);

```

```

lcd.print("E= ");
lcd.setCursor(2, 2);
lcd.print(energy);

Serial.print("Rp; "); Serial.println(cost);
int roundedCost = round(cost);
lcd.setCursor(0, 3);
lcd.print("Rp.");
lcd.setCursor(3, 3);
lcd.print(roundedCost);

int currentDay = day();
int currentMonth = month();
int currentYear = year();

// Dapatkan waktu saat ini
int currentHour = hour();
int currentMinute = minute();
int currentSecond = second();

Data_to_Sheets(No_of_Parameters, voltage, current, power, pf,
energy, cost);

Serial.println();
delay(30000);

// Pengecekan waktu untuk pengiriman otomatis
if (currentHour == 17 && currentMinute == 50 && !sendData) {
// Kirim data ke Telegram
String sekarang = "Data hari ini\n";
sekarang += "Tegangan: " + String(voltage) + "V\n\n";
sekarang += "Arus: " + String(current) + "A\n\n";
sekarang += "Daya: " + String(power) + "W\n\n";
sekarang += "Faktor Daya: " + String(pf) + "\n\n";
sekarang += "Energi: " + String(energy) + "kWh\n\n";
}

```



```

    sekarang += "Biaya: Rp " + String(cost)+ "\n\n";
    sekarang += "KLIK DISINI: link Google Sheetsnya ";

    bot.sendMessage(String(CHAT_ID), sekarang);
    sendData = true; // Setel flag untuk mencegah pengiriman berulang
kali pada hari yang sama
    Serial.println("Sended");
}
delay(1000);

// Pengecekan ulang flag setiap hari baru
if (currentHour == 18 && currentMinute == 1) {
    sendData = false; // Setel flag kembali ke false pada hari yang
baru
}
Serial.println("OK");
delay(1000);

// Pengecekan akhir bulan
if (currentDay == 25 && currentHour == 17 && currentMinute == 55 &&
!isEndOfMonth) {
    isEndOfMonth = true;
    String message1 = "Akhir catat:\n\n";
    message1 += "Energi: " + String(energy) + "\n\n";
    message1 += "Biaya: Rp " + String(cost);
    bot.sendMessage(String(CHAT_ID), message1);
    // Reset energi PZEM004Tv30 di akhir bulan
    pzem.resetEnergy();
    Serial.println("Direset");
}

// Pengecekan awal bulan
if (currentDay == 25 && currentHour == 18 && currentMinute == 0 &&
!isStartOfMonth) {

```

```

    isStartOfMonth = true;
    String message2 = "Awal catat:\n\n";
    message2 += "Energi: " + String(energy) + "kWh\n\n";
    message2 += "Biaya: Rp " + String(cost);
    bot.sendMessage(String(CHAT_ID), message2);
    Serial.println();

}

if (millis() > lastTimeBotRan + botRequestDelay)
{
    int numNewMessages = bot.getUpdates(bot.last_message_received +
1);

    while (numNewMessages) {
        String chat_id = String(bot.messages[0].chat_id);
        String message_sender_id = String(bot.messages[0].from_id);
        Serial.println("Got response");
        handleNewMessages(numNewMessages, chat_id, message_sender_id);
        numNewMessages = bot.getUpdates(bot.last_message_received + 1);
    }

    lastTimeBotRan = millis();
    delay(1000);
}
}

```