

# PERANCANGAN PROTOTIPE *REALTIME* AKUISISI DATA DAN KONTROL STOP KONTAK MENGGUNAKAN ESP 32 BERBASIS WEB

Naufal Qodari<sup>\*)</sup>, Yuli Christyono, dan Sukiswo

Departemen Teknik Elektro, Fakultas Teknik, Universitas Diponegoro  
Jl. Prof. Sudharto, SH, Kampus UNDIP Tembalang, Semarang 50275, Indonesia

<sup>\*)E-mail: qodarinaufal@gmail.com</sup>

## Abstrak

Listrik merupakan kebutuhan primer manusia. Sebuah bangunan hampir pasti memiliki peralatan listrik didalamnya, Stop kontak adalah antarmuka yang digunakan untuk **supply** listrik ke alat elektronik. Model stop kontak telah dikembangkan cukup banyak, mulai dari model manual, otomatis, sampai dengan yang terkontrol. Penelitian ini mencoba untuk merancang sebuah stop kontak yang mampu untuk melakukan kontrol peralatan listrik melalui web pada komputer atau smartphone sehingga penggunaan listrik sebuah bangunan dapat lebih efisien. Selain itu sistem stop kontak yang kami buat dapat menampilkan besaran listrik tegangan, arus, daya dan faktor daya secara realtime sehingga dapat terpantau keadaan kelistrikan disebuah bangunan tersebut dengan baik. Sistem stop kontak tersebut menggunakan **ESP32** sebagai mikrokontroler, **ZMCT103C** sebagai sensor arus, **ZMPT101B** sebagai sensor tegangan, dan relay 10 A sebagai aktuator, serta dimmer untuk mengatur daya masukan. Hasil pengujian menunjukkan sistem stop kontak mampu mengendalikan relay sehingga **socket** pada stop kontak dapat dilakukan kontrol on-off. Sistem stop kontak juga mampu menampilkan besaran listrik arus, tegangan, daya, dan faktor daya secara realtime.

Kata kunci : *Sistem stop kontak, Mikrokontroler, Arduino ESP32, Kontrol Relay*

## Abstract

*Electricity is a primary human need. A building almost certainly has electrical equipment in it. Electric socket is the interface used to supply electricity to electronic devices. The stop contact model has been developed quite a lot, ranging from manual, automatic, to controlled models. This research tries to design a contact stop that can control electrical equipment via the web on a computer or smartphone so that the use of electricity in a building can be more efficient. Also, the socket system that we made can display the amount of electrical voltage, current, power, and power factors in real-time so that the electricity can be monitored in a building well. The stop system uses ESP32 as a microcontroller, ZMCT103C as a current sensor, ZMPT101B as a voltage sensor, and a 10 A relay as an actuator, and a dimmer to regulate the input power. The test results show the power outlet system can control the relay so that the socket on the socket can be controlled on-off. The power outlet system is also capable of displaying the amount of electric current, voltage, power, and power factor in real-time.*

*Keywords: Stop contact system, Microcontroller, Arduino ESP32, Relay Control*

## 1. Pendahuluan

Penggunaan energi yang semakin meningkat menyebabkan ketersediaan dari sumber energi yang ada semakin menurun. Salah satunya adalah penggunaan dari sumber energi listrik. Pemakaian energi listrik pada rumah tangga sebagian besar adalah untuk penerangan, sisanya digunakan untuk keperluan lainnya seperti TV, AC, kipas angin, Kulkas, dispenser, mesin air dan lain sebagainya. Namun penggunaan secara tidak teratur dan bahkan secara berlebihan dapat menyebabkan pemborosan pada sumber energi dan kerusakan pada perangkat sumber tenaga listrik<sup>[1]</sup>. Berdasarkan perumusan masalah tersebut maka dibuat prototipe sistem stop kontak terkontrol. Konsep dari

prototipe ini adalah sebuah alat yang dapat membantu masyarakat untuk menghemat penggunaan listrik pada rumah yang dapat dikontrol dengan jarak jauh serta dapat meningkatkan efisiensi konsumsi daya pada perangkat elektronik. Perkembangan teknologi saat ini mendorong manusia untuk terus berpikir kreatif, tidak hanya menggali penemuan-penemuan baru, tapi juga memaksimalkan kinerja teknologi yang ada untuk meringankan kerja manusia dalam kehidupan sehari-hari seperti pemantauan penggunaan arus listrik agar tidak digunakan secara berlebihan dikarenakan rentan menyebabkan terjadinya hubungan pendek arus listrik yang dapat berakibat fatal seperti kebakaran pada rumah<sup>[2]</sup>.

Pemanfaatan teknologi membuat tindakan-tindakan tersebut dapat dilakukan secara semi-otomatis dan otomatis. Salah satu teknologi tersebut yaitu Internet of Things atau biasa disingkat dengan kata IoT. Internet of Things (IoT) membuat perangkat saat ini mampu bertukar informasi melalui jaringan internet<sup>[3]</sup>. Istilah Internet of Things (IoT) sendiri pertama kali digunakan oleh Kevin Ashton saat presentasi pada tahun 1999<sup>[4]</sup>.

Purwarupa dapat membaca besar penggunaan tegangan, arus, daya serta factor daya alat elektronik yang terhubung pada stop kontak.<sup>[5]</sup> Apabila terjadi arus pendek maupun gangguan arus lebih, melebihi batas yang diatur pada mikorkontroler maka relay akan melakukan trip. Prototipe Stop Kontak ini memiliki empat perangkat utama yaitu Sensor Arus ZMCT103C, Sensor Tegangan ZMPT101B, relay 10 Ampere, dan ESP32 nodeMCU<sup>[6]</sup>. Saklar listrik pada Stop Kontak menggunakan relay dan dapat dikendalikan melalui mikrokontroler berbasis jaringan internet sehingga dapat terhubung ke ponsel pintar maupun PC melalui website.

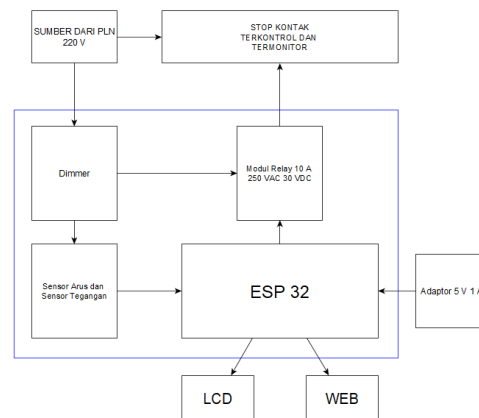
## 2. Metode Perancangan

Perancangan Tugas Akhir terbagi menjadi dua bagian, yaitu perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*). Perancangan perangkat keras berupa penyusunan komponen-komponen elektronika menjadi satu kesatuan sistem rangkaian yang bisa bekerja sesuai dengan yang diharapkan. Perancangan perangkat lunak berupa perintah-perintah dalam bahasa mesin yang membuat sistem dapat bekerja sesuai bahasa perintah tersebut.

Perancangan Tugas Akhir ini mempunyai tahapan sebagai berikut:

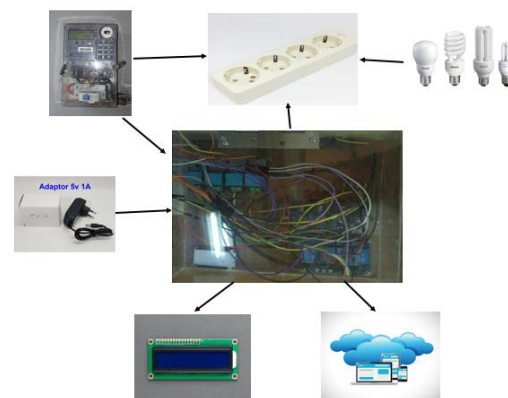
1. Menentukan komponen-komponen yang diperlukan untuk tiap rangkaian.
2. Membuat PCB untuk indicator relay.
3. Melaksanakan perakitan blok-blok rangkaian sensor, relay, dan dimmer
4. Konfigurasi pin ESP32 dengan rangkaian dan LCD
5. Pemrograman ESP32 dan Kalibrasi Sensor
6. Pengaturan koneksi alat dengan WiFi
7. Pengontrolan relay dari Web

Gambar 1 merupakan diagram blok prototipe sistem. Prototipe terdiri dari perangkat keras dan perangkat lunak. Komponen tersebut saling terhubung dan bekerja secara terus menerus sehingga didapatkan data realtime. ESP32 mendapat input data dari sensor arus dan sensor tegangan, kemudian dikirim ke LCD dan WEB. Web membalas status relay berupa code biner yang kemudian di proses untuk mengaktifkan atau menonaktifkan relay<sup>[7]</sup>.



Gambar 1. Diagram Blok Perangkat Prototipe

### 2.1. Perancangan Perangkat Keras



Gambar 2. Hubungan Komponen Perangkat Keras

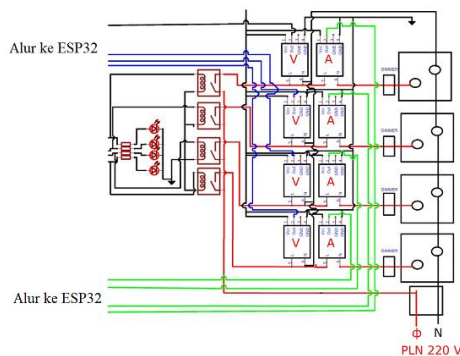
Gambar 2 menunjukkan komponen penyusun perangkat keras sebagai berikut:

#### 2.1.1. Rangkaian sensor, relay dan dimmer

Rangkaian sensor difungsikan untuk melakukan akuisisi data arus dan tegangan. Sensor berupa pengondisi sinyal yang terdiri dari sensor tegangan AC ZMPT101B dan sensor arus AC ZMCT103C. Keduanya di akuisisi data nya menggunakan ESP32<sup>[8]</sup>.

Rangkaian relay difungsikan sebagai aktuator dalam pengontrolan stop kontak. Relay yang digunakan adalah modul relay 10 A 250 VAC 30 VDC. Dimmer difungsikan untuk mengatur tegangan yang masuk dari sumber listrik ke stop kontak<sup>[9]</sup>.

Berikut Rangkaian Sensor, Relay dan Dimmer :



Gambar 3. Rangkaian Sensor, Relay, dan Dimmer

Konfigurasi Pin ESP32 dengan Rangkaian dan LCD. Pin ESP32 yang digunakan antara lain sebagai berikut :

- Sensor arus : GIOP35, GIOP34, GIOP39, GIOP36 (Pin 3-6)
- Sensor tegangan : GIOP32, GIOP33, GIOP25, GIOP26 (Pin 7-10)
- Relay : GIOP15, GIOP2, GIOP4, GIOP16 (Pin 23-27)
- LCD : I2C SDA, I2C SCL (Pin 33 dan 36)

## 2.2. Perancangan Perangkat Lunak

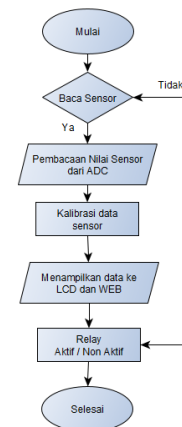
Perancangan perangkat-lunak (*software*) mencakup kedalam pembuatan program untuk ESP32 sebagai unit pegolah data dari sensor. Perangkat-lunak merupakan suatu bagian penting dalam perancangan system akuisisi data multisensor ini. Program *compiler* yang dipakai menggunakan Arduino IDE 1.8.12 yang telah *compatible* dengan bahasa Arduino sehingga lebih mudah daripada *compiler* lain. Selain hal tersebut, Arduino IDE telah menyediakan pustaka fungsi yang terdokumentasi dalam *library* yang tersedia atau dapat diunduh secara bebas di internet. Program Arduino dirancang agar dapat mengolah data pembacaan masing-masing sensor pada perangkat keras [10].

### 2.2.1. Pemrogramana ESP32

Rangkaian sensor difungsikan untuk melakukan akuisisi data arus dan tegangan. Sensor berupa pengondisi sinyal yang terdiri dari sensor tegangan AC ZMPT101B dan sensor arus AC ZMCT103C [11]. Keduanya di akuisisi data nya menggunakan ESP32. Sebelum melakukan pemrograman, dilakukan setting board terlebih dahulu pada menu tools - board manager , pilih ESP32 Dev Module. Kemudian download library – library yang diperlukan untuk pemrograman [12].

Diantaranya :

- Emonlib.h untuk perhitungan besaran listrik
- HTTPClient.h untuk menghubungkan ESP32 dengan WiFi
- LiquidCrystal\_I2C.h untuk menampilkan karakter pada LCD



Gambar 4. Flowchart Monitoring Data dan Kontrol

Gambar 4 Menunjukkan Diagram alir program pada mikrokontroller ESP 32.

## 2.3. Kalibrasi Sensor

Kalibrasi dilakukan dengan memberikan tegangan AC ke dengan sensor ZMPT101B. Sebelum dilakukan kalibrasi ini, dapat di cek keluaran dalam pemrograman Arduino IDE menggunakan serial plotter. Dengan terlebih dulu memastikan sinyal yang dibaca sensor sudah berbentuk sinusoidal dengan memutar-mutar potensio pada sensor zmpt101b. Jika Sudah selanjutnya adalah membaca tegangan  $V_m$  (v puncak) pada sinyal baca adc yang mempresentasikan sinyal AC asli. Anggap sebuah sinyal AC adalah sinyal sinus, maka tundaan  $\frac{1}{4}$  periodenya adalah

$$\sin\left(\theta - \frac{\pi}{2}\right) = \cos(\theta)$$

Maka apabila sinyal AC kita represntasikan dalam diagram fasor, maka theta akan berputar dengan kecepatan putar, misal di Indonesia menggunakan 50 Hz, maka theta akan berputar dengan kecepatan 50 putaran per detik. Lalu bisa sinyal AC bisa kita modelkan seperti ini [13].

$$A \sin(\theta) \rightarrow \text{adalah sinyal AC}$$

A = Amplitudo sinyal, Sedangkan tundaannya adalah

$$A \sin\left(\theta - \frac{\pi}{2}\right) = A \cos(\theta)$$

Jika menggunakan teorema pythagoras pada keduanya, maka

$$\begin{aligned} A_m^2 &= A^2 \sin^2(\theta) + A^2 \cos^2(\theta) \\ A_m^2 &= A^2 (\sin^2(\theta) + \cos^2(\theta)) \\ A_m &= A \end{aligned}$$

Tundaan  $\frac{1}{4}$  periode bisa menggunakan fungsi tunda, tergantung waktu sampling yang anda gunakan. Pada program yang saya buat saya pakai waktu sampling ( $T_s$ ) 10kHz, artinya dalam setiap periode sinyal AC 50Hz, maka

akan didapatkan 200 sampling. Maka untuk mendapatkan  $\frac{1}{4}$  periode maka ditunda sebanyak 50 kali. Jika sudah maka nilai baca ADC harus disesuaikan dengan nilai AC yang asli, misal didapat nilai magnitude ADC 400, sedangkan tegangan AC adalah 220. Maka tinggal dibagi menggunakan gain  $220/400$  [14].

#### 2.4. Menampilkan Hasil dalam LCD I2C 16x2

Koneksikan Pin LCD I2C dengan Pin dalam ESP32 yakni GPIO21 dan GPIO22 atau I2C SCA dan I2C SCL, selanjutnya sambungkan Vcc dengan Vin dan Ground dengan Ground. Kemudian Program ESP32 nya [15].

#### 2.5. Koneksi ESP32 ke Web

Diperlukan Library WiFi.h dan HTTPClient.h untuk mengkoneksikan ESP32 dengan WiFi, setelah library terinstal di Arduino IDE, koneksikan ESP32 dengan jaringan hotspot di rumah dengan memasukan command berikut:

```
const char * ssid = "Bismillah";
const char * password = "doaibupenuhberkah";
```

Setelah WiFi terkoneksi, data hasil pembacaan yang telah dikalibrasi dikirim ke Web melalui command berikut :

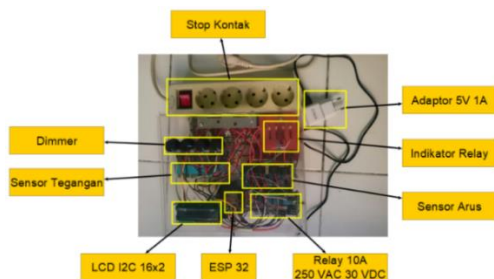
```
char*server="https://miliohm.com/ta13/data/updat
e.php?data="; HTTPClient http;
```

#### 2.6. Kontrol Relay dari Web

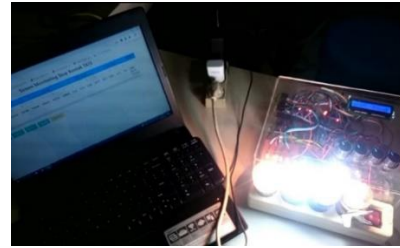
Relay dapat dikendalikan dari web dengan mengklik tombol relay pada tampilan web, ketika tombol berwarna hijau maka relay dalam kondisi off dan socket on, ketika tombol relay warna merah maka relay aktif dan socket off.

### 3. Hasil dan Analisis

Pengujian dilakukan untuk mengetahui hasil perancangan perangkat keras dan perangkat lunak yang telah dibuat dan melakukan analisis. Realisasi perangkat keras yang telah dibuat ditunjukkan pada Gambar 5. Gambar 6.



Gambar 5. Perangkat Keras yang Telah Dibuat



Gambar 6. Tampilan Web dan Prototipe yang sedang beroperasi

Perangkat keras berupa prototype yang akan memonitor secara realtime arus, tegangan, daya, dan factor daya pada beban yang terkoneksi dengan socket stop kontak dan dapat melakukan kontrol *On-Off* yakni dengan mematikan atau menghidupkan relay dari web yang beralamat <http://www.miliohm.com/ta13>.

Pengujian dari sistem prototipe ini meliputi:

1. Pengujian akuisisi data besaran listrik arus, tegangan, daya dan faktor daya pada LCD I2C yang tertanam dalam prototype.
2. Pengujian pengiriman data akuisisi dari *prototype* alat ke *web*
3. Pengujian kontrol pada web

#### 3.1. Pengujian Akuisisi Data Besaran Listrik pada LCD I2C

Pengujian dilakukan dengan beban sebagai berikut :

- Socket 1 : Lampu Pijar 5 Watt
- Socket 2 : Lampu SL 8 Watt
- Socket 3 : Lampu LED 3 Watt
- Socket 4 : Lampu Pijar 15 Watt

Parameter besaran listrik yang diuji adalah tegangan, arus, daya, dan faktor daya



Gambar 7. Perangkat Keras yang sedang beroperasi

Tabel 1. Uji sensor ZMPT101B dan ZMCT103C

Sample	Daya Nyata	Arus Terbaca (A)	Tegangan Terbaca (V)	Daya Terbaca (VA)	Faktor Daya
Lampu Pijar	5 Watt	0,0224	229,34	5,25	0,7
Lampu SL	8 Watt	0,0295	271,3	6,19	0,74
Lampu LED	3 Watt	0,013	231,39	2,84	0,76
Lampu Pijar	15 Watt	0,0657	228,35	14,39	0,89

Berdasarkan Tabel 1 hasil pengujian sensor tegangan ZMPT101B terbilang tidak cukup stabil dari sensor arus ZMCT103C yang cenderung outputnya lebih stabil.

### 3.2. Pengujian Pengiriman Data Akuisisi dari Prototipe Alat ke Web

Data diupload ke web setiap beberapa detik sekali tergantung koneksi jaringan yang terhubung dengan ESP 32, Rata-rata 5 atau 6 detik data live di web akan diperbaharui.

Database Tegangan, Arus, Daya, dan Faktor Daya

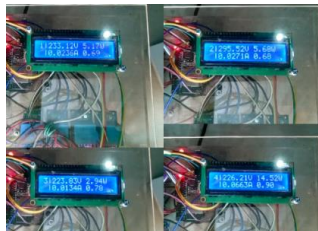
No.	Tegangan (V)				Arus (A)				Daya (watt)				Faktor Daya				Waktu
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
21.	228.12	231.1	231.16	227.47	0.0241	0.0346	0.015	0.0659	5.28	7.27	2.84	14.44	0.7	0.87	0.76	0.9	2020-06-19 17:13:16
22.	228.12	231.1	233.18	227.47	0.0241	0.0346	0.0129	0.0659	5.28	7.27	2.82	14.44	0.7	0.87	0.76	0.9	2020-06-19 17:13:11
23.	241.35	242.63	233.18	228.12	0.0228	0.0317	0.0129	0.0658	4.99	6.65	2.82	14.4	0.66	0.8	0.76	0.89	2020-06-19 17:13:06
24.	241.35	242.63	235.63	228.12	0.0228	0.0317	0.0133	0.0658	4.99	6.65	2.91	14.4	0.66	0.8	0.76	0.89	2020-06-19 17:13:01
25.	229.34	271.3	231.39	228.35	0.024	0.0295	0.015	0.0672	5.25	6.19	2.84	14.39	0.7	0.74	0.76	0.89	2020-06-19 17:12:55
26.	219.18	268.67	231.39	223.28	0.0251	0.0298	0.013	0.0672	5.5	6.25	2.84	14.71	0.73	0.75	0.76	0.91	2020-06-19 17:12:50
27.	219.18	268.67	226.27	223.28	0.0251	0.0298	0.0133	0.0672	5.5	6.25	2.9	14.71	0.73	0.75	0.77	0.91	2020-06-19 17:12:45
28.	238.14	238.62	226.27	226.47	0.0231	0.0347	0.0133	0.0662	5.06	7.28	2.9	14.51	0.67	0.88	0.77	0.9	2020-06-19 17:12:40
29.	238.14	237.63	226.1	226.47	0.0231	0.0336	0.0132	0.0662	5.06	7.06	2.89	14.51	0.67	0.85	0.77	0.9	2020-06-19 17:12:35
30.	219.48	237.63	226.66	226.37	0.0252	0.0336	0.0132	0.0663	5.51	7.06	2.9	14.51	0.73	0.85	0.77	0.9	2020-06-19 17:12:30

Gambar 8. Database Besaran Listrik Termonitor

Data diupload berdasarkan perintah dari program esp 32

```
char*server="https://miliohm.com/ta13/data/update.php?data=";
HTTPClient http;
```

Berikut ini hasil tampilan di LCD I2C dan Monitor pada Web pada tanggal 19 Juni 2020 pukul 17.11 :



Gambar 9. Tampilan LCD I2C 19 Juni 2020 pukul 17.11

No.	Tegangan (V)				Arus (A)				Daya (watt)				Faktor Daya				Waktu
	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
233.12	295.52	226.44	224.27	0.0236	0.0271	0.0132	0.0660	5.17	5.68	2.9	14.65	0.69	0.68	0.77	0.91	2020-06-19 17:11:44	
233.12	296.82	223.83	224.27	0.0236	0.0271	0.0134	0.0660	5.17	5.68	2.94	14.65	0.69	0.68	0.78	0.91	2020-06-19 17:11:38	
230.79	289.34	223.83	226.21	0.0238	0.0276	0.0134	0.0663	5.22	5.81	2.94	14.82	0.69	0.7	0.78	0.9	2020-06-19 17:11:33	
235.47	267.18	223.23	224.93	0.0234	0.0239	0.0133	0.0667	5.12	6.29	2.92	14.6	0.68	0.78	0.91	2020-06-19 17:11:28		

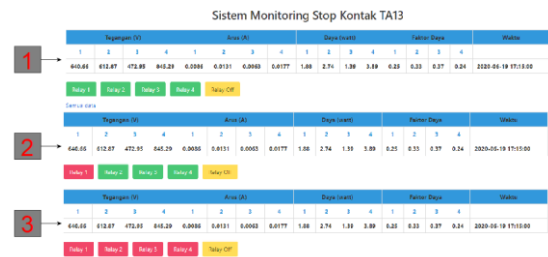
Gambar 10. Tampilan Monitor Web 19 Juni 2020 pukul 17.11

Dari gambar 9 dan gambar 10 dapat diketahui bahwa tampilan LCD I2C sesuai dengan Tampilan Monitor Web.

### 3.3. Pengujian Pengiriman Data Akuisisi dari Prototipe Alat ke Web

String dengan ditambahkan Pada web terdapat tombol Relay 1 sampai dengan Relay 4 yang berfungsi untuk menyalakan atau menonaktifkan relay dan satu tombol Relay OFF untuk menonaktifkan semua relay.

<https://ejournal3.undip.ac.id/index.php/transient>



Gambar 11. Tampilan Kontrol Relay pada Web

Pada gambar 11 nomor 1 merupakan kondisi dimana semua socket pada stop kontak beroperasi, relay relay off dan semua lampu indicator mati, pada nomor 2 relay 1 dinyalakan sehingga socket stop kontak no1 tidak teraliri arus listrik dan lampu indicator satu menyala. Sedangkan nomor 3 adalah kondisi dimana semua socket stop kontak tidak teraliri arus listrik dan relay dan lampu indicatornya on semua.

Semua button relay berwarna hijau      Semua button relay berwarna merah



Gambar 12. Tampilan Kontrol Relay pada prototype

## 4. Kesimpulan

Berdasarkan perancangan, pengujian dan analisa yang telah dilakukan maka dapat dibuat kesimpulan bahwa berdasarkan pengujian Sensor Arus ZMCT103C pada sistem bekerja cukup stabil terbukti dari tak adanya fluktuasi perubahan data yang besar. Sensor Tegangan ZMPT101B pada sistem bekerja kurang stabil terbukti dari adanya fluktuasi perubahan data yang besar. Pengujian ketepatan data antara LCD I2C dan Monitor Web berhasil. Sistem Stop Kontak dapat melakukan fungsi control relay dengan baik. Dan indikator Relay berfungsi dengan baik..

## Referensi

- [1]. V. Masinambow, M. E. I. Najoran, and A. S. M. Lumenta, "Pengendali Saklar Listrik Melalui Ponsel Pintar Android," *E-Journal Tek. Elektro Dan Komput.*, vol. 3, no. 1, pp. 27–35, 2014, doi: 10.35793/jtek.3.1.2014.3772.
- [2]. R. A. Dalimunthe, "Pemantau Arus Listrik Berbasis Alarm Dengan Sensor Arus Menggunakan Mikrokontroler Arduino Uno," *Semin. Nas. R.* 2018, vol. 9986, no. September, 2018.
- [3]. Francisco, A. R. L. (2013).. *Journal of Chemical Information and Modeling*, 53(9), "," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013, doi: 10.1017/CBO9781107415324.004.

- [4]. F. Xia, L. T. Yang, L. Wang, and A. Vinel, *Internet of things*, vol. 25, no. 9. 2012.
- [5]. S. Bipasha Biswas and M. Tariq Iqbal, "Solar Water Pumping System Control Using a Low Cost ESP32 Microcontroller," *Can. Conf. Electr. Comput. Eng.*, vol. 2018-May, no. October, 2018, doi: 10.1109/CCECE.2018.8447749.
- [6]. Espressif, "ESP32 Series Datasheet," *ESP32 Ser. Datasheet*, pp. 1–61, 2020, [Online]. Available: [https://www.espressif.com/sites/default/files/documentat ion/esp32\\_datasheet\\_en.pdf](https://www.espressif.com/sites/default/files/documentat ion/esp32_datasheet_en.pdf).
- [7]. A. Maier, A. Sharp, and Y. Vagapov, "Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things," *2017 Internet Technol. Appl. ITA 2017 - Proc. 7th Int. Conf.*, no. November, pp. 143–148, 2017, doi: 10.1109/ITECHA.2017.8101926.
- [8]. I. Abubakar, S. N. Khalid, M. W. Mustafa, H. Shareef, and M. Mustapha, "Calibration of ZMPT101B voltage sensor module using polynomial regression for accurate load monitoring," *ARN J. Eng. Appl. Sci.*, vol. 12, no. 4, pp. 1076–1084, 2017.
- [9]. Nafi' Zaim Nashif, "Sistem Monitoring ATS AMF Berbasiskan IoT Cloud," Tugas Akhir. Politeknik Negeri Madiun. 2018.
- [10]. D. D. Yudhistira, M. D. Ramadhan, N. Augusta, and S. Agustini, "Proses Kalibrasi Sensor E-Tape Sebagai Instrumen Pengukuran Tinggi Muka Air Di Lapang." pp. 1–7.
- [11]. G. Agung Ayu Putri and N. Made Ika Marini Mandenni, "Desain Saklar Otomatis Untuk Kontrol Peralatan Listrik di Bangunan," *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, vol. 7, no. 1, p. 12, 2019, doi: 10.24843/jim.2019.v07.i01.p02.
- [12]. A. Amir, A. Marwanto, and D. Nugroho, "Rancang Bangun Purwarupa Alat Monitoring Dan Kontrol Beban Satu Fasa Berbasis Iot (Internet of Things)," *Transmisi*, vol. 20, no. 1, p. 29, 2018, doi: 10.14710/transmisi.20.1.29-33.
- [13]. A. Prastiantari, F. Hermin, and M. Mulyono, "Skopin (Stop Kontak Pintar) Pengendali Arus Listrik Menggunakan Timer Pada Stop Kontak Berbasis Arduino," pp. 21–28, 2012.
- [14]. K. Ogata, "Modern Control Engineering, Third Edition," ORSVH95] Sam Owre, John Rushby, Natarajan Shankar, and Friedrich von Henke. p. 20, 1997.
- [15]. A. B. Muljono, I. M. A. Nrartha, I. M. Ginarsa, and I. M. B. Sukmadana, "Rancang Bangun Smart Energy Meter Berbasis UNO dan Raspberry Pi," *J. Rekayasa Elektr.*, vol. 14, no. 1, pp. 9–18, 2018, doi: 10.17529/jre.v14i1.8718.