

# Implementasi Intelligent IoT Gateway Sebagai Pengendali Jarak Jauh Pada Raspberry Pi Berbasis OpenWrt

Dhito Aji Tri Prabowo, Marsekal Dadrian Setyayudha, Renny Sari Dewi

Fakultas Teknologi Industri dan Kreatif, Sistem Informasi, Universitas Internasional Semen Indonesia, Gresik, Indonesia  
Email: <sup>1</sup>me@dhaito.tk, <sup>2</sup>myudha27898@gmail.com, <sup>3</sup>renny.dewi@uisi.ac.id

Submitted 03-01-2020; Accepted 15-01-2020; Published 15-02-2020

## Abstrak

*Internet of Things* merupakan sebuah konsep dimana suatu objek tertentu memiliki fungsi untuk mentransfer data melalui jaringan wireless dan internet sebagai pengendali jarak jauh, aktifitas ini tidak memerlukan interaksi dari manusia ke manusia atau manusia ke komputer. Semua sudah dijalankan secara otomatis dengan program. *Internet of Things* biasa disebut dengan IoT. Dengan menerapkan IoT di rumah diharapkan dapat meningkatkan efisiensi dan kenyamanan bagi penghuni rumah. Sistem *smart home* yang akan dibuat memiliki dua unsur pokok yaitu perangkat ponsel pintar berbasis android dan sistem otomatisasi yang telah dipasang pada Raspberry-Pi yang sudah ter-install *firmware* OpenWrt dan terhubung langsung dengan relay. Sistem juga akan dibekali dengan pengukur sensor arus ACS712 dan sensor tegangan ZMPT101B, dua alat tersebut digunakan untuk mendapatkan data tegangan listrik lalu tersimpan pada penyimpanan di Raspberry-Pi. Setelah melakukan uji coba menggunakan AVOMeter menghasilkan perbedaan data yang relatif kecil. Tingkat kesalahan dari pengukur tegangan rata-rata 4.1% dan tingkat kesalahan dari sensor arus rata-rata 6.2%. Data dari beban listrik akan ditampilkan pada aplikasi di perangkat ponsel pintar berbasis Android. Hasil dari penelitian ini adalah sebuah sistem kendali jarak jauh terhadap perangkat yang terhubung dengan listrik dan memberikan informasi untuk menganalisa penggunaan beban listrik dirumah, sehingga dapat memilih perangkat yang sebaiknya dimatikan oleh sistem untuk penghematan energi listrik dan biaya yang harus dikeluarkan setiap bulannya.

**Kata Kunci:** IoT, OpenWrt, Raspberry Pi, Smart Home, Android

## Abstract

Internet of Things is a concept where a particular object has a function to transfer data through wireless networks and the internet as a remote controller, this activity does not require interaction from human to human or human to computer. Everything has been run automatically with the program. Internet of Things is commonly called IoT. By implementing IoT at home it is hoped that it can improve efficiency and comfort for the occupants of the house. The smart home system that will be made has two main elements, namely an Android-based smart phone device and an automation system that has been installed on a Raspberry-Pi that has OpenWrt firmware installed and is connected directly to the relay. The system will also be equipped with an ACS712 current sensor and a ZMPT101B voltage sensor, the two devices are used to obtain electrical voltage data which will then be stored in storage on the router. After testing using AVOMeter, the data difference is relatively small. The error rate of the voltage gauge is 4.1% on average and the error rate of the current sensor is 6.2%. Data from the electricity load will be displayed on the application on an Android-based smart phone device. The results of this study are a remote control system for devices connected to electricity and provide information to analyze the use of electrical loads at home, so they can choose devices that should not be activated by the system so as to cut monthly electricity costs.

**Keywords:** IoT, OpenWrt, Raspberry Pi, Smart Home, Android

## 1. PENDAHULUAN

Semakin majunya perkembangan teknologi maka kebutuhan energi listrik juga akan semakin meningkat. Hal tersebut dikarenakan pemakaian listrik yang besar pada rumah, gedung, maupun industri dimana listrik digunakan untuk dapat menjalankannya. Selain itu, pemakaian energi listrik untuk skala rumah tangga juga sangat boros. Hal ini disebabkan karena kurang memperhatikan besar kecilnya daya atau watt perangkat elektronik yang digunakan [1].

Hemat energi merupakan suatu cara pemanfaatan energi secara rasional dan efisien tanpa mengurangi penggunaan energi yang memang benar-benar diperlukan. Penghematan energi pada rumah bila dilaksanakan secara sistematis akan berpotensi mengurangi konsumsi energi yang besar. Akan tetapi pada prakteknya program hemat energi tersebut masih belum maksimal. Salah satu masalah yang dihadapi dalam melakukan penghematan energi adalah sistem pengontrolan perangkat elektronik pada rumah saat ini masih tergolong konvensional yaitu dengan menggunakan saklar atau relatif hanya menggunakan prinsip pengontrolan jarak dekat (manual), sehingga untuk menghidupkan dan mematikan perangkat elektronik harus dengan mendekati saklar terlebih dulu [2].

Dengan berkembangnya *Internet of Things* [2][3] maka kontrol perangkat elektronik yang ada didalam rumah dapat dilakukan dengan menggunakan internet dan juga *smartphone* yang terhubung dengan internet, sehingga pengontrolan dan monitoring bisa dilakukan dimana saja [4].

Beberapa usaha dan penelitian sebelumnya telah banyak dilakukan, diantaranya, penelitian yang dilakukan oleh Ibrahim, Muhammad Fakhri Ali Hermin, Fariani Arafiyah, dan Ria, dengan judul *Home Automation Berbasis GSM Untuk Kendali Listrik Jarak Jauh Menggunakan Short Message Service* [5]. Mikrokontroler yang digunakan adalah Arduino UNO R3 sedangkan untuk kontrol jarak jauh digunakan ponsel berteknologi GSM dengan cara mengirimkan SMS ke server lalu diterjemahkan oleh kontroler dan diteruskan ke saklar on off di perangkat listrik rumah. Ponsel GSM harus memiliki sinyal yang cukup supaya dapat berkomunikasi antara transmitter dan receiver.

Sedangkan penelitian yang dilakukan oleh Sri Supatmi, Taufiq Nuzwir Nizar, and Reza Fahlevi, dengan judul *Sistem Kontrol Peralatan Rumah dan Monitoring Kondisi Rumah Melalui Internet Berbasis Web dan OpenWRT* [6], menggunakan perangkat access point TL-MR3020 sebagai servernya/pengganti PC server dirumah. Sistem informasi yang digunakan

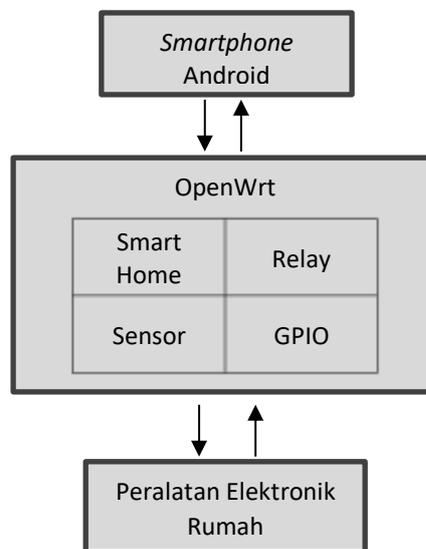
adalah sistem informasi berbasis web. Perangkat lunak yang digunakan adalah linux openWRT yang di install di server. Hasil dari penelitian ini adalah berupa sistem pengontrolan perangkat elektronik rumah menggunakan web melalui browser. Data hasil pengontrolan akan dikirim melalui email pengguna.

Melihat perkembangan tersebut timbul gagasan bagi peneliti untuk membuat sebuah sistem otomatisasi perangkat elektronik rumah yang dapat diakses dari jarak jauh dengan memanfaatkan jaringan internet. Sistem *smart home* yang akan dibuat memiliki dua unsur pokok yaitu perangkat ponsel pintar berbasis android dan sistem otomatisasi yang telah dipasang pada Raspberry Pi yang sudah ter-install firmware OpenWrt dan terhubung langsung dengan relay. Sistem juga akan dibekali dengan pengukur sensor arus ACS712 dan sensor tegangan ZMPT101B, dua alat tersebut digunakan untuk mendapatkan data tegangan listrik kemudian akan tersimpan pada penyimpanan di Raspberry Pi [7]. Sistem bisa dikendalikan secara otomatis dan juga manual melalui internet serta dilengkapi dengan penjadwalan waktu kapan perangkat elektronik rumah harus menyala dan padam, *User interface* yang lebih *frendly* akan memudahkan pengguna dalam menggunakan aplikasi tersebut, serta dilengkapi dengan sistem pengendalian jarak jauh HVAC ( *Heating, Ventilation and Air Conditioning* ) yang akan membuat pengguna merasa nyaman berada di dalam rumah. Untuk sistem informasi yang digunakan berupa sistem informasi multiplatform dan sistem informasi berbasis android [8].

## 2. METODE PENELITIAN

Perancangan sistem dilakukan dalam beberapa tahapan diantaranya perancangan aplikasi untuk pemilik rumah, perancangan aplikasi pada Raspberry Pi dengan *firmware* OpenWrt yang berada di rumah dan perancangan aplikasi berbasis android yang akan menjadi perantara antara pemilik rumah sebagai pengendali dan peralatan elektronik di rumah yang akan dikontrol dan dipantau [9].

### 2.1 Block Diagram Sistem



Gambar 1. Blok Diagram Sistem

Gambar 1 menunjukkan blok diagram sistem, Perangkat pengguna adalah perangkat yang digunakan untuk berkomunikasi yang berfungsi sebagai pengendali jarak jauh yaitu *smartphone* android yang dihubungkan melalui IP server ke Raspberry Pi dengan sistem operasi OpenWrt dengan bantuan *Dynamic Domain Name System* (DDNS). Sistem OpenWRT adalah sistem operasi distribusi GNU/Linux untuk perangkat keras. Sistem operasi ini kemudian diinstall ke Raspberry Pi yang nantinya akan mengontrol kerja dari GPIO untuk kontrol pada beban peralatan listrik. Peralatan elektronik yang akan di kontrol adalah beban seperti lampu, AC, televisi, kipas angin dan lain-lain [10]. Agar sistem ini dapat di kontrol dari jarak jauh maka sistem ini terhubung menggunakan internet. Sambungan internet bisa diambil dari modem maupun koneksi WiFi yang sudah ada. Dalam sistem telah dilengkapi *Auto Report* sehingga pengguna dapat memonitoring rumah secara real time.

### 2.2 Prinsip Kerja Sistem

Cara kerja dari sistem ini berawal dari perintah yang dilakukan oleh pengguna melalui *Graphical User Interface* (GUI) pada aplikasi android. Melalui perintah pada aplikasi tersebut kemudian dikonversi oleh sistem menjadi kode biner 0 dan 1. Kode biner kemudian dikirim ke Raspberry Pi dan di proses oleh sistem OpenWrt yang nantinya akan memberikan logika On dan OFF pada (*general-purpose input/output*) GPIO. Hasil dari ON dan OFF GPIO diteruskan ke *relay* yang nantinya akan terhubung pada perangkat elektronik. Pada kutub input Arus AC pada *relay* disambungkan pada sensor tegangan dan arus ACS712 dan ZMPT101B yang nantinya akan terhubung pada GPIO Raspberry Pi [11], lalu output dari relay akan dihubungkan ke perangkat. Hasil dari pembacaan sensor dan konversi akan dikembalikan ke aplikasi pengguna agar tampil

di GUI pada aplikasi android. Prinsip kerja dari sistem Smart Home diawali dari pengguna yang telah memasang aplikasi ke smartphone android yang telah terintegrasi dengan sistem *Smart Home* yang telah dibuat. Pada proses pertama ini, pengguna akan mendapatkan alamat IP yang didapatkan saat pertama kali terhubung pada jaringan lokal dengan menggunakan Wi-Fi. Alamat IP yang didapat oleh pengguna adalah alamat yang merupakan satu Subnet pada jaringan Smart Home. Agar pengguna mendapatkan alamat IP sesuai dengan urutannya, maka pada OpenWRT harus diaktifkan layanan yang bernama DHCP (Dynamic Host Configuration Protocol) dimana layanan ini merupakan layanan yang ada dalam sistem OpenWRT. Jika terhubung ke internet maka pengguna perlu memasukan hostname atau IP server secara manual dan pada OpenWrt diharuskan untuk memasang modul DDNS [12]. Saat pengguna telah terhubung pada jaringan, maka pengguna diminta memasukkan Password agar bisa terhubung dan bisa mengontrol seluruh sistem dari Smart Home. Jika semua sudah selesai dilakukan maka sistem Smart Home sudah siap digunakan.

### 2.3 Perancangan Perangkat Keras Raspberry Pi

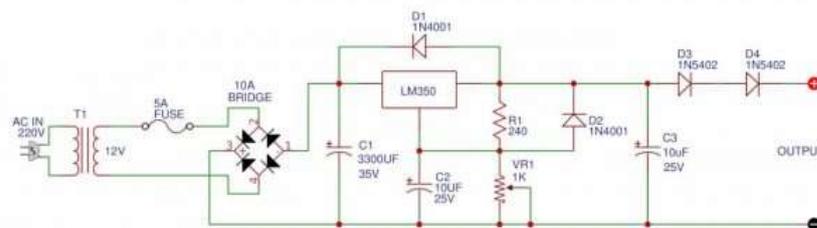
Untuk dapat berjalan sebagaimana mestinya, sistem ini memerlukan sebuah modul *micro computer* yang didalamnya sudah diinstall sistem operasi OpenWRT yang nantinya akan ditanam sistem kerja dari smart home tersebut. Dalam perangkat ini sudah memiliki [13]:

- |   |   |
|---|---|
| a. SoC : Broadcom BCM2837                         | f. Bluetooth : Bluetooth 4.1 Classic          |
| b. CPU : 4× ARM Cortex-A53, 1.2GHz                | g. Storage : microSD                          |
| c. GPU : Broadcom VideoCore IV                    | h. GPIO : 40-pin header, populated            |
| d. RAM : 1GB LPDDR2 (900 MHz)                     | i. Ports : HDMI, 3.5mm audio jack, 4× USB 2.0 |
| e. Networking : 10/100 Ethernet, 802.11n wireless |   |

Dari semua spesifikasi yang ada pada Raspberry Pi, GPIO mempunyai peranan penting yang berfungsi sebagai pendukung dari sistem Smart Home yang dibuat. Raspberry Pi ini juga sudah terintegrasi dengan wireless WiFi dan menggunakan Protokol TCP/IP.

### 2.4 Perancangan Power Supply

*Schematic* rangkaian power supply dapat dilihat Pada Gambar 4. Rangkaian tersebut merupakan jenis power supply variable dengan menggunakan IC tipe LM350. IC ini memiliki 3 terminal *adjustable regulator* tegangan positif yang dapat menyediakan lebih dari 3A dengan output variabel dari 1,2V hingga 12V DC. IC LM350 memerlukan 2 resistor eksternal untuk mengatur tegangan output yaitu R1 dan R2 [14].

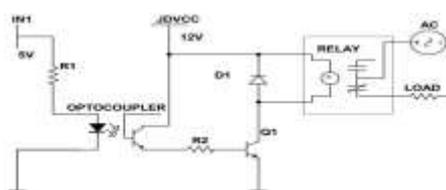


**Gambar 2.** Schematic Power Supply Variable  
 (Sumber: eleccircuit.com, Chayapol, 2014)

Rangkaian power supply variabel ini bisa dilengkapi dengan voltmeter analog untuk dapat menampilkan tegangan output dan sebuah potensiometer 5K untuk mengatur nilai tegangan R1 dan R2 sehingga mendapatkan tegangan DC variabel sesuai dengan yang diinginkan yaitu 5V DC untuk daya yang dibutuhkan Raspberry Pi.

### 2.5 Perancangan Power Supply

Relay digunakan untuk mengalirkan atau memutuskan arus listrik alternating current (AC), yang dikontrol oleh pin GPIO pada Board Raspberry Pi. Misalkan bila pin IN1 pada controller board diberi *logic low* maka *optocoupler* akan aktif dan mengakibatkan transistor (Q1) dan kumparan pada relay menjadi tidak aktif, sehingga peralatan listrik juga tidak aktif. Sebaliknya bila pin IN1 controller board diberi *logic high* maka *optocoupler* tidak aktif dan mengakibatkan transistor (Q1) dan kumparan pada relay menjadi aktif, sehingga peralatan listrik juga aktif.

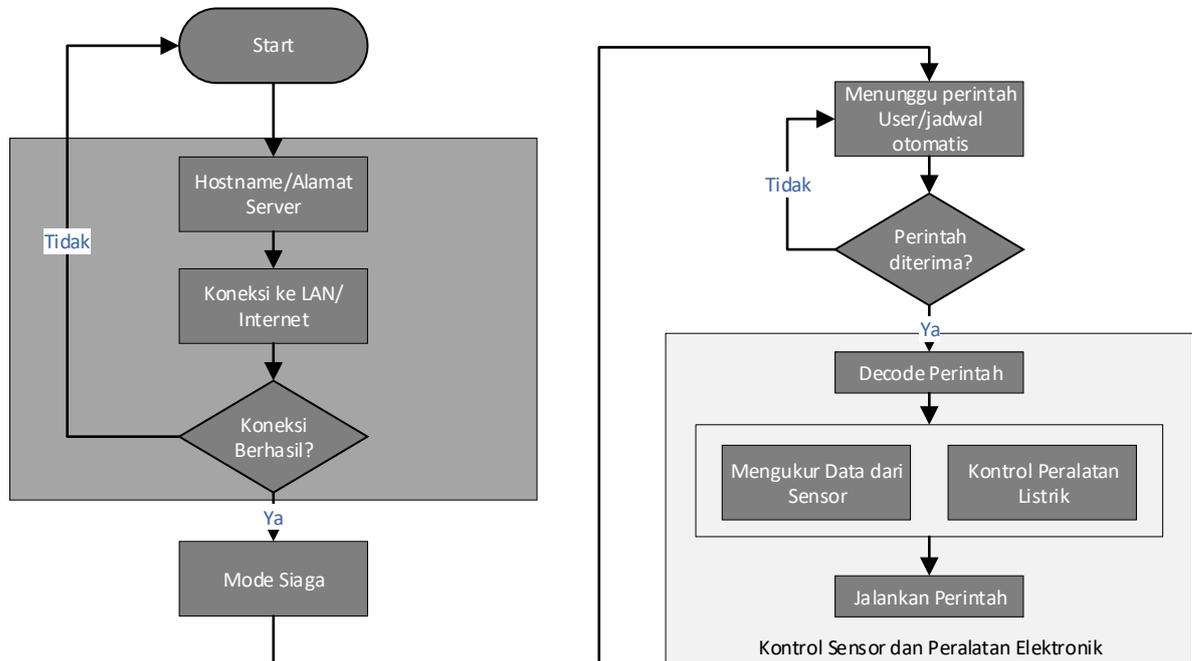


**Gambar 3.** Rangkaian Relay  
 (Sumber: Instructables.com, 2017)

Pada rangkaian relay Gambar 5 terdapat Optocouple yang berfungsi untuk mengamankan tegangan tinggi dan arus pendek yang dapat terjadi pada modul relay dengan mengisolasi sumber daya antara *controller board* yakni Raspberry Pi dengan modul relay. Dioda pada modul relay dipergunakan untuk mencegah terjadinya arus balik listrik, yang dapat berasal dari induksi medan magnet yang dihasilkan oleh kumparan pada relay.

### 2.6. Perancangan Perangkat Lunak

Pada Gambar 6 memperlihatkan diagram alir dari perancangan perangkat lunak pengendali berbasis android terhadap sensor dan GPIO pada Raspberry Pi sebagai pendukungnya.



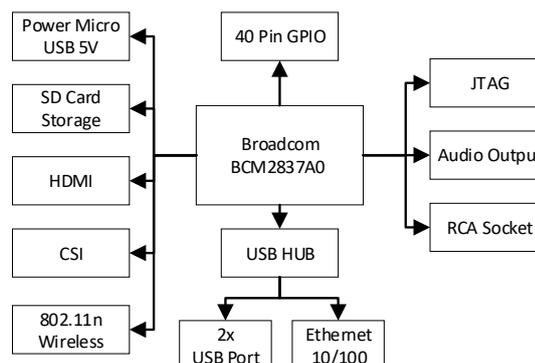
Gambar 4. Diagram Alir Perangkat Lunak

Pada pembuatan aplikasi pengendali ini menggunakan bahasa pemrograman Java berbasis android dimana pengguna diharuskan untuk memasang aplikasi pada smartphone yang dimilikinya. Sehingga output dari hardware akan sesuai dengan perintah yang diinginkan oleh pengguna dari kontrol aplikasi android yang sudah dibuat [1].

## 3. ANALISA DAN PEMBAHASAN

### 3.1 Pengujian Modul Micro Computer Raspberry Pi

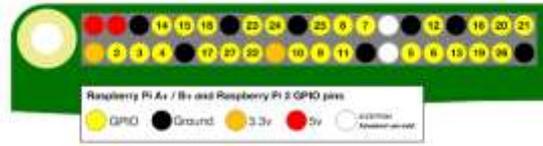
Pengujian pada modul micro computer Raspberry Pi ini akan dilakukan dengan memberikan tegangan sebesar 5V DC kemudian menghubungkan pin GPIO dengan lampu LED. Hal ini dilakukan untuk mengetahui karakteristik dari GPIO pada Raspberry Pi tersebut.



Gambar 5. Skema Raspberry Pi

Raspberry Pi diberikan output tegangan 5V DC dan output pada GPIO diukur dengan menggunakan AVOMeter digital untuk analisa output tegangan pada GPIO. Pengujian dilakukan dengan memberikan kode 1 dan 0 pada serial board Raspberry Pi

untuk mengaktifkan port GPIO Board tersebut. Setelah mendapatkan informasi data nilai tegangan yang keluar dari GPIO maka ditentukan modular yang nantinya akan mengaktifkan *relay* yang terhubung ke perangkat elektronik. GPIO Output akan mengeluarkan tegangan sebesar 3.2V apabila diberi logika 1, jika diberikan logika 0 maka akan mengeluarkan tegangan 0.6V. Pada Gambar 8 memperlihatkan datashet dari pin GPIO dari Raspberry Pi, pin berwarna kuning menunjukkan pin GPIO yang dapat kita modifikasi, pin berwarna hitam adalah Ground (GND), Pin warna orange adalah arus 3,3V, Pin warna Merah adalah arus 5V, sedangkan Pin warna putih I/O EEPROM [13].



**Gambar 6.** Datashet Pin GPIO Raspberry Pi  
 (Sumber: raspberrypi.org, 2017)

Dari hasil Tabel 1 dihasilkan bahwa ada pin GPIO pada raspberry Pi yang tidak merespon ketika mendapatkan logika 1 atau 0. Tidak diketahui secara pasti apa penyebabnya, hal ini mungkin terjadi karena kesalahan dari penulisan program atau memang bermasalah dari perangkat Raspberry Pi itu sendiri.

**Tabel 1.** Hasil Tes Logika PIN GPIO.

GPIO	Logika	Tegangan	ON/Off	Ket.
GPIO 2	1	3.2	ON	Normal
GPIO 2	0	0.6	OFF	Normal
GPIO 3	1	3.2	ON	Normal
GPIO 3	0	0.6	OFF	Normal
GPIO 4	1	3.2	ON	Tidak respon
GPIO 4	0	0.6	OFF	Tidak respon
GPIO 5	1	3.2	ON	Normal
GPIO 5	0	0.6	OFF	Normal
GPIO 6	1	3.2	ON	Normal
GPIO 6	0	0.6	OFF	Normal
GPIO 7	1	3.2	ON	Normal
GPIO 7	0	0.6	OFF	Normal
GPIO 8	1	3.2	ON	Normal
GPIO 8	0	0.6	OFF	Normal
GPIO 9	1	3.2	ON	Normal
GPIO 9	0	0.6	OFF	Normal
GPIO 10	1	3.2	ON	Normal
GPIO 10	0	0.6	OFF	Normal

Raspberry Pi membutuhkan tegangan sebesar 5V DC dan pada sistemnya terdapat Regulator penurun tegangan menjadi 3.3V 100mA yang digunakan untuk menyuplai arus Processor dan Memori dari board tersebut. Tegangan yang keluar dari GPIO tersebut akan sama dengan regulator penurun tegangan namun terjadi penurunan arus menjadi 50mA. Tegangan ini hanya mampu untuk menyalakan lampu led saja. Sehingga untuk menyalakan modul *relay* maka dibutuhkan transistor penguat. Pada kasus ini dipilihlah Transistor PNP karena Output GPIO Raspberry Pi adalah positif. Sistem kerja dari transistor ini adalah ketika basis pada transistor diberikan arus atau logika 1, maka output akan menjadi aktif, dan ketika arus arus terhenti atau logika 0 maka output menjadi tidak aktif. Output dari transistor inilah yang akan mengontrol Relay yang nantinya akan terhubung dengan beban.

### 3.1. Pengujian Perangkat Lunak OpenWRT

Pengujian selanjutnya adalah pada perangkat lunak. Sistem operasi yang akan dimasukkan ke Raspberry Pi adalah OpenWRT v18.06.0. Sistem operasi ini merupakan system operasi Linux. Linux yang digunakan adalah linux yang menggunakan kernel 4.9 yang mampu bekerja dalam perangkat berspesifikasi rendah seperti Raspberry Pi dan router wireless tertentu. Setelah pemasangan OpenWrt berhasil maka langkah selanjutnya akan diinstall Packet GPIO Controller. GPIO Controller ini menggunakan bahasa C untuk pemrogramannya. Dalam bahasa C akan dimasukan Library HI yang berarti memiliki Logika 1 dan LOW yang berarti memiliki arti logika bernilai 0. GPIO Controller ini akan terhubung langsung ke GUI yang akan digunakan oleh Pengguna.

**Tabel 2.** Pengujian Output GPIO

Data GPIO	Hasil Pantauan	
	V/A	Keterangan
Output HI	3,28V/50A	Nyala

Output LOW	0,64V/50A	Mati
Output LOW	0,71V/50A	Mati
Output HI	3,30V/50A	Nyala
Output HI	3,27V/50A	Nyala

Dari data pada Tabel 2 menunjukkan bahwa informasi yang muncul pada GUI sama sekali tidak terjadi kesalahan, sehingga pengguna dapat membaca kondisi relay yang sedang aktif dan tidak aktif.

### 3.2 Pengujian Sensor ACS712 dan ZMPT101B

Pengujian arus dilakukan dengan membandingkan dengan AVOMeter, diambil data dari beberapa kali pengukuran dengan beban perangkat elektronik yang berbeda dengan membandingkan dua alat pengukuran yaitu ZMPT101B dan AVOMeter. Hasil pengujian sensor arus ZMPT101B dapat dilihat pada Tabel 3.

Tabel 3. Data Pengujian Output GPIO

Peralatan Elektronik	Hasil Pengukuran (A)		Selisih	Error (%)
	Sensor	AvoMeter		
Air Conditioner 800W	2,25	2,19	0,06	2,74
	2,21	2,27	0,06	2,86
Mesin Cuci 300W	1,39	1,33	0,06	4,51
	1,31	1,36	0,05	3,82
Kipas Angin 100W	1,06	1,01	0,06	5,94
	1,07	1,02	0,06	5,88
Televisi 110W	1,08	1,01	0,07	6,80
	1,08	1,03	0,06	5,83
Pompa Air 250W	1,18	1,12	0,06	5,36
	1,11	1,16	0,05	4,55
Vacum Cleaner 400W	1,86	1,80	0,06	3,33
	1,85	1,79	0,06	3,35
Microwave 900W	2,78	2,72	0,05	1,84
	2,77	2,71	0,06	2,21
<b>Total</b>				59
<b>Rata-Rata</b>				4,2

$$\begin{aligned}
 \text{Error} &= \frac{\text{Selisih Pengukuran}}{\text{Pengukuran Alat Standart}} \times 100 \% \\
 &= \frac{2,25 - 2,19}{2,19} \times 100 \% \\
 &= 2,74 \%
 \end{aligned}$$

$$\begin{aligned}
 \text{Error Rata-Rata} &= \frac{\text{Total Error}}{\text{Banyaknya Pengukuran}} \\
 &= \frac{59}{14} \\
 &= 4,2 \%
 \end{aligned}$$

Dari hasil perhitungan, diketahui nilai eror pengukuran menggunakan sensor arus ZMPT101B sebesar 4,2% untuk pengukuran arus. Hal ini menunjukkan pengukuran sensor arus sudah bekerja dengan baik, karena nilai sebesar 4,2% merupakan tingkat error pengukuran yang relatif kecil sehingga pengujian arus tidak mengalami kendala berarti. Selain pengujian arus ampere, pengujian tegangan juga dilakukan pada modul ini. Hasil pengujian tegangan AC dapat dilihat pada Tabel 2, hanya saja pengujian dilakukan dengan mengubah-ubah tegangan agar diperoleh tegangan yang berbeda. Output dari resistor variabel dihubungkan dengan AVOMeter standar dan kemudian dibandingkan dengan hasil sensor tegangan yang dibuat. Setelah itu dihitung tingkat eror dan rata-rata eror dari hasil pengujian tegangan yang dilakukan. Hasil pengukuran tegangan menggunakan alat ukur standart AVOMeter dan sensor tegangan litrik menggunakan ACS712 yang dibuat dapat dilihat pada Tabel 4.

Tabel 4. Pengujian Sensor Tegangan

Hasil Pengukuran (V)		Selisih	Error (%)
Sensor	AVOMeter		
90,43	92,0	1,57	1,7
40,33	41,69	1,38	3,3
60,10	61,0	0,9	1,4
52,74	56,8	4,15	7,2
81,53	87,5	6,15	6,6
53,66	54,02	0,36	0,6

150,76	151,96	1,2	0,7
197,85	196	1,89	0,9
110,52	112,22	1,2	1,0
179,91	179,47	0,44	0,2
198,60	200,93	2,33	1,1
192,2	194,87	2,67	1,3
215,80	217	1,2	0,5
210,65	212,05	1,4	0,6
219,33	221,5	2,35	1,0
222,13	221,11	1,02	0,5
<b>Total</b>			<b>28,6</b>
<b>Rata-rata</b>			<b>1,8</b>

$$\begin{aligned} \text{Error} &= \frac{\text{Nilai Sensor} - \text{Nilai Alat Ukur}}{\text{Hasil Alat Ukur}} \times 100 \% \\ &= \frac{1,57}{92} \times 100 \% \\ &= 1,7 \% \end{aligned}$$

$$\begin{aligned} \text{Error Rata-Rata} &= \frac{\text{Total Error}}{\text{Banyaknya Pengukuran}} \\ &= \frac{28,6}{16} \\ &= 1,8 \% \end{aligned}$$

Dari perhitungan rata-rata toleransi dari pengujian tegangan dihasilkan error rata-rata sebesar 1,8% untuk pengukuran tegangan. Hal ini menunjukkan alat sensor untuk mengukur tegangan sudah dapat bekerja dengan relatif baik untuk pengukuran tersebut.

### 3.3 Pengujian Perangkat Lunak Aplikasi Android

Pembuatan aplikasi pengendali sengaja dibuat dengan basis android karena smartphone android paling banyak digunakan saat ini, sehingga tidak sulit untuk dapat menguji aplikasi yang telah dibuat. Antarmuka pengguna berawal dari pengguna diharuskan untuk membuat server baru dengan memasukkan nama server, alamat *hostname* atau IP server, dan password. Server atau hostname yang dimasukkan harus sesuai dengan konfigurasi pada Raspberry Pi dengan Operasi Sistem OpenWRT yang telah dibuat. Proses menambah server baru bisa dilihat pada Gambar 9.



Gambar 7. Antarmuka Input Server Baru.

Setelah pembuatan server baru selesai dan terkoneksi dengan baik, maka selanjutnya pengguna bisa menuju ke menu Tab Saklar dan memilih perangkat elektronik yang ingin diaktifkan atau dinonaktifkan. Lebih jelasnya dapat dilihat pada Gambar 10.



Gambar 8. Antarmuka Menu Saklar

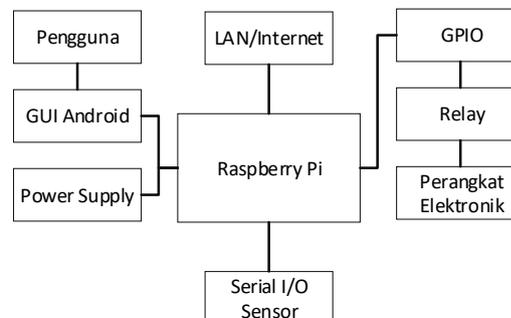
Pada Tab Jadwal, pengguna dapat membuat jadwal untuk perangkat elektronik di rumah kapan harus diaktifkan atau dinonaktifkan. Jadwal dibagi menjadi kategori harian, bulanan, atau mingguan. Menu jadwal pada antarmuka bisa dilihat pada Gambar 11.



Gambar 9. Antarmuka Menu Jadwal

### 3.3. Pengujian Secara Keseluruhan

Masing-masing komponen pada alat telah diuji, kemudian selanjutnya adalah menguji alat secara keseluruhan, Dilakukan dengan memberi perintah melalui aplikasi android yang telah terhubung ke server lokal maupun internet untuk permintaan mengaktifkan perangkat elektronik dirumah maupun memperoleh informasi total beban litrik yang digunakan pada Controller Raspberry Pi dengan sistem operasi OpenWrt melalui relay, sensor, dan GPIO.



Gambar 10. Pengujian Secara Keseluruhan

Setelah melakukan pengujian, ternyata kontrol melalui aplikasi menuju gateway ke server OpenWrt dan GPIO pada Raspberry Pi tidak terjadi secara *real time* atau terjadi *delay*. Salah satu penyebabnya adalah jaringan yang tidak stabil khususnya server melalui internet, selain itu beban dari perangkat Raspberry Pi juga mempengaruhi tenggang waktu yang terjadi. Hasil pengujian dapat dilihat pada Tabel 5.

Tabel 5. Pengujian Aplikasi Android

Uji Coba	Tenggang Waktu (ms)
Uji Coba 1	3000ms
Uji Coba 2	1300ms
Uji Coba 3	2350ms
Uji Coba 4	5543ms
Uji Coba 5	1020ms
<b>Total</b>	<b>13213ms</b>
<b>Rata-rata</b>	<b>2643ms</b>

## 4. KESIMPULAN

Setelah dilakukan pengujian dan analisis terhadap sistem, maka didapatkan beberapa kesimpulan sebagai berikut:

1. Setelah dilakukan penelitian, implementasi *Intelligent IoT Gateway* menggunakan Raspberry Pi dengan sistem operasi OpenWrt memiliki algoritma yang berbeda dengan Arduino. Menggunakan perintah Hi dan Low kemudian OpenWRT

- merubah ke dalam kode biner yang akan mengaktifkan GPIO yang sudah dipilih. Sehingga dari hasil kontrol GPIO tersebut kita bisa menggerakkan relay yang terhubung pada perangkat elektronik dirumah.
2. Untuk dapat melakukan kontrol yang terhubung ke perangkat elektronik dirumah bisa dilakukan menggunakan aplikasi android yang dibuat dengan bahasa pemrograman java. Kemudian diteruskan ke sistem Linux OpenWRT dan akan mengeksekusinya kedalam perintah logika 1 dan 0, dan akan mengirimkannya ke GPIO yang sesuai dengan perintah dari pengguna.
  3. Analisa yang telah dilakukan pada sensor tagangan dan arus listrik yang terpasang pada sistem dengan cara membandingkan hasil dari output sensor dengan AVOMeter. Hasil yang didapatkan dari pengukuran diperoleh tingkat kesalahan sensor arus rata-rata 4,2% dan sensor tegangan 1,8%. Karena tingkat kesalahannya relatif kecil maka bisa dikatakan memenuhi standart.
  4. Kecepatan internet dan beban dari modul Raspberry Pi akan mempengaruhi tenggang waktu atau *delay* perintah yang dikirim pengguna ke server. Dari hasil pengujian rata-rata perbedaan waktu adalah 2643ms

## REFERENCES

- [1] D. Prihatmoko, "Perancangan Sistem Monitoring Perangkat Elektronik Rumah Menggunakan Internet," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 9, no. 1, pp. 279–286, 2018.
- [2] M. Collotta, G. Scatà, and G. Pau, "A priority-based CSMA/CA mechanism to support deadline-aware scheduling in home automation applications using IEEE 802.15.4," *Int. J. Distrib. Sens. Networks*, vol. 2014, 2014.
- [3] D. Prihatmoko, Widyawan, Selo, and S. Wibowo, "Pengembangan Perangkat Lunak Gateway untuk Home Automation Berbasis IQRF TR53B Menggunakan Konsep CGI," *Semin. Nas. Sist. Inf. Indones.*, pp. 2–4, 2015.
- [4] F. Arifyanto, W. A. Syafei, and M. Somantri, "Perancangan Prototype Web-Based Online Smart Home Controlled By Smartphone," *Transient*, vol. 2, no. 3, pp. 1–8, 2014.
- [5] M. F. A. Ibrahim, F. Hermin, and R. Arafiyah, "Home Automation Berbasis GSM Untuk Kendali Listrik Jarak Jauh Menggunakan Short Message Service," *J-KOMA J. Ilmu Komput. dan Apl.*, vol. Vol 2 No., no. Vol 2 No 1 (2018): JKOMA, 2018.
- [6] R. F. Sri Supatmi, Taufiq Nuzwir Nizar, "Sistem Kontrol Peralatan Rumah Dan Monitoring Kondisi Rumah Melalui Internet Berbasis Web Dan Openwrt," *J. Tek. Komput. Unikom – Komputika – Vol. 3, No.2 - 2014 Sist.*, vol. 3, no. 2, pp. 23–28, 2015.
- [7] H. Muchtar and R. Sumanjaya, "Control Switch Otomatis pada Tegangan Energi Alternatif dan Tegangan Sumber PLN Menggunakan Raspberry Pi," vol. 1, no. 2, pp. 97–102, 2018.
- [8] N. Razzaq *et al.*, "Implementasi Sistem Monitoring Hvac Ruang Server Menggunakan Raspberry Pi Berbasis Web ( Studi kasus pada PT . Rahajasa Media Internet Surabaya )," pp. 86–92, 2019.
- [9] H. Jefton, J. Andjarwirawan, and A. Noertjahyana, "Implementasi router dengan menggunakan Raspberry Pi 3," pp. 1–6, 2017.
- [10] U. KHOLIL RM, "Mengontrol Lampu Rumah Berbasis Web Dengan Memanfaatkan Sistem General Purpose Input/Output (Gpio) Pada Router Openwrt," *J. Manaj. Inform.*, vol. 8, no. 1, pp. 37–44, 2017.
- [11] M. Dan, K. Beban, L. Berbasis, and R. Pi, "Perancangan Energi Meter Dan Analisis Karakteristik Beban Listrik Berbasis Raspberry Pi," vol. 5, no. 1, pp. 64–72, 2018.
- [12] M. Rizqy *et al.*, "Pembuatan Home Automation System Menggunakan Router OpenWRT Hg553 Berbasis IP, 2019."
- [13] C. FIOLUTION, "Tutorial Raspberry Pi 3," p. 27, 2017.
- [14] Texas Instruments, "LM150 / LM350A / LM350 3-Amp Adjustable Regulators Current Limit Constant with Temperature," no. March, 2014.