

RANCANG BANGUN SISTEM KENDALI LAMPU MENGGUNAKAN SENSOR SUARA BERBASIS *ARDUINO* DENGAN APLIKASI PEMANTAUAN PADA *SMARTPHONE ANDROID*

^[1]Eka Yogi Prananda, ^[2]Dedi Triyanto, ^[3]Suhardi

Jurusan Sistem Komputer, Fakultas MIPA Universitas Tanjungpura
Jalan Prof. Dr. H. Hadari Nawawi, Pontianak

Telp./Fax.: (0561) 577963

e-mail: ^[1]ekayogiprananda@gmail.com, ^[2]dedi.triyanto@siskom.untan.ac.id
^[3]suhardi@siskom.untan.ac.id

ABSTRAK

Pada penelitian ini telah dibuat sebuah sistem pengendali lampu listrik yang digunakan untuk menyalakan maupun mematikan dengan perintah suara serta dilengkapi dengan sebuah aplikasi *Android* sebagai media antarmuka pemantauan status lampu serta dapat juga digunakan untuk mengendalikan lampu dari jarak jauh menggunakan *SMS gateway*. Sistem dibuat menggunakan *Arduino Mega 2560* sebagai modul pengendali utama. Pesan atau data yang dikirim dari modul suara *EasyVR* akan diterima dan diproses *Arduino Mega* untuk mengendalikan rangkain *relay*. *Relay* berfungsi sebagai saklar elektromekanik untuk menyalakan dan mematikan lampu. Laporan pengendalian akan dikirimkan ke *Arduino* untuk diteruskan ke *smartphone Android* melalui modul *GSM SIM900* dalam bentuk notifikasi *graphic* yang berfungsi untuk mengetahui status lampu dalam keadaan menyala atau mati, notifikasi ini diberikan setiap terjadi perubahan kondisi lampu. Sistem juga memberikan alternatif pilihan pengendalian lampu secara manual. Dari hasil penelitian didapatkan rata-rata waktu proses kendali menggunakan *SMS* dari sejak pertama kali perintah diberikan sampai lampu menyala adalah 10,3 detik, untuk rata-rata waktu penerimaan pesan balasan dari sejak awal perintah diberikan adalah 22,7 detik. Selanjutnya untuk rata-rata waktu proses perintah suara dari sejak pertama kali perintah diberikan sampai lampu A menyala adalah 4,3 detik, lampu B 4,5 detik dan semua lampu 4,4 detik, sedangkan rata-rata waktu penerimaan pesan balasan dari sejak awal perintah diberikan adalah lampu A 12,7 detik, lampu B 15,8 detik dan semua lampu 19,8 detik. Kemudian rata-rata waktu proses perintah suara dari sejak pertama kali perintah diberikan sampai lampu A mati adalah 4,8 detik, lampu B 4,4 detik dan semua lampu 4,4 detik, sedangkan rata-rata waktu penerimaan pesan balasan dari sejak awal perintah diberikan adalah lampu A 14,2 detik, lampu B 17 detik dan semua lampu 16,4 detik. Rata-rata waktu notifikasi *error* adalah 13,7 detik. Hasil akhir penelitian ini adalah sebuah sistem kontroler yang dapat menyalakan dan mematikan lampu dengan tiga bentuk pengendalian yaitu kendali suara, sms berbasis aplikasi *Android* dan manual menggunakan *switch*.

Kata Kunci: Arduino Mega 2560, EasyVR, GSM SIM900, SMS gateway, Android

1. PENDAHULUAN

Kemajuan zaman dan 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, salah satunya adalah sistem pengendalian lampu rumah. Dalam mengendalikan lampu manusia seringkali harus bergerak berpindah tempat menuju titik beradanya saklar lampu kemudian melakukan kontak fisik untuk menekan

saklar yang ada, hal ini menjadi hal yang biasa dilakukan oleh kebanyakan orang normal, tapi akan menjadi hal yang tidak mudah untuk sebagian orang yang menderita tunadaksa atau lansia dengan kategori susah bergerak dan berdiri. Setiap akan mengendalikan lampu harus berupaya menjangkau saklar atau meminta bantu orang lain. Masalah lain ketika harus melakukan kontak langsung dengan saklar adalah resiko keamanan seperti tersengat listrik.

Dari permasalahan tersebut maka perlu dibuat suatu sistem yang dapat

menghilangkan kendala saat ingin mengendalikan lampu karena harus melakukan kontak fisik dengan saklar. Sistem kendali lampu dengan sensor suara merupakan sistem yang dapat membantu manusia terutama penyandang tunadaksa atau lansia dalam mengendalikan lampu rumah tanpa menyentuh saklar dan aman dari resiko tersengat listrik.

Beberapa penelitian dalam upaya menjawab permasalahan ini telah dilakukan, salah satunya adalah penelitian yang berjudul “PENGENDALIAN LAMPU RUMAH BERBASIS MIKROKONTROLER ARDUINO UNO MENGGUNAKAN SMARTPHONE ANDROID”[1]. Penelitian ini mampu menjawab permasalahan yang timbul dengan menawarkan sebuah sistem pengendalian lampu nirkabel menggunakan perintah suara, namun sistem ini memiliki kelemahan yaitu *ouput* dari sensor suara yang nantinya akan dikirim ke mikrokontroler untuk diproses lebih lanjut memiliki sistem *input* yang sangat bergantung pada kestabilan jaringan internet, karena sensor yang digunakan adalah sebuah *smartphone Android* sebagai *microphone*-nya dan *Google Voice* sebagai sistem *online* pengolah *input* perintah dari *user*, sehingga ketika jaringan internet terganggu maka akan menghambat eksekusi perintah pada sistem. Komunikasi *smartphone* dan mikrokontroler juga menggunakan *bluetooth* yang memiliki jangkauan akses terbatas, sehingga dirasa perlu dilakukan pengembangan yang lebih lanjut terhadap sistem pengendalian seperti ini.

Berdasarkan dua kelemahan tersebut maka penulis melakukan penelitian tentang sistem kendali lampu menggunakan sensor suara yang tidak terkendala dengan kestabilan jaringan internet, kemudian penulis juga menggantikan *bluetooth* dengan *SMS gateway* sehingga sistem tidak lagi terhambat jarak dalam melakukan pemantauan status maupun mengendalikan lampu dari jauh. Harapan dengan adanya sistem ini bisa memberikan kemudahan untuk pengguna terutama penyandang tunadaksa atau lanjut usia agar dapat

dengan mudah dalam mengendalikan lampu di dalam rumah.

2. LANDASAN TEORI

2.1. Sistem Kontrol

Sistem kontrol adalah suatu sistem yang digunakan pada nilai masukan tertentu sebagai pengendali untuk keluaran dengan nilai tertentu, memberikan urutan kejadian tertentu atau memunculkan suatu kejadian jika beberapa kondisi terpenuhi.

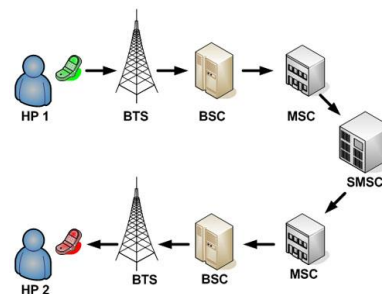
2.2. SMS

2.2.1. Pengertian SMS

SMS (Short Message Service) adalah aplikasi standar yang dimiliki oleh *handphone*, berfungsi untuk mengirim dan menerima pesan singkat dalam bentuk huruf maupun angka [2]. *SMS* merupakan sebuah layanan yang banyak diaplikasikan pada sistem komunikasi tanpa kabel, memungkinkan dilakukannya pengiriman pesan antara terminal pelanggan dengan sistem eksternal seperti *e-mail*, *voice*, *mail* dan lain-lain.

2.2.2. Cara Kerja SMS

Teknologi pada *SMS* memanfaatkan gelombang mikro dan pengiriman sinyal yang dibagi berdasarkan waktu dan frekuensi, sehingga sinyal informasi yang dikirim akan sampai pada tujuan. *SMS* yang dikirimkan dari suatu dari suatu ponsel akan masuk *SMS Center*, kemudian diteruskan ke nomor tujuan *SMS* tersebut [3]. Gambar 1 adalah alur cara kerja dari *SMS*.

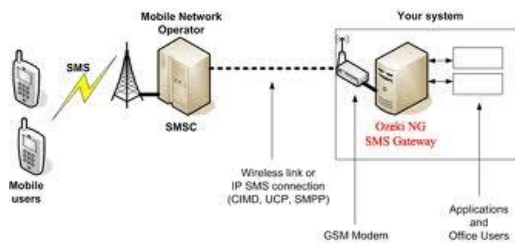


Gambar 1. Alur Cara Kerja SMS

2.2.3. SMS Gateway

SMS gateway adalah suatu *platform* metode untuk mengirimkan dan menerima pesan SMS tanpa menggunakan perangkat *mobile* atau ponsel. *SMS gateway* merupakan pintu gerbang bagi penyebaran

informasi dengan menggunakan SMS [4]. Gambar 2 adalah cara kerja dari SMS gateway.



Gambar 2. Cara Kerja SMS Gateway

2.2.4. Perintah AT Command

AT Command merupakan standar command yang digunakan oleh komputer untuk berkomunikasi dengan modem/phone modem. Dengan menggunakan AT Command, dapat diperoleh informasi mengenai modem, melakukan setting pada modem, mengirim SMS dan menerima SMS pada modem GSM dan sebagainya. AT Command juga dapat digunakan sebagai perintah-perintah yang digunakan pada komunikasi serial port.

Tabel 1 adalah diantara perintah AT Command yang berhubungan dengan sistem kerja SMS gateway [5].

Tabel 1. Perintah dalam AT Command

AT Command	Keterangan
AT	Mengecek apakah handphone telah terhubung
AT+CMGF	Menetapkan format mode dari terminal
AT+CMCS	Menetapkan jenis encoding
AT+CNMI	Mendeteksi pesan SMS baru masuk secara otomatis
AT+CMGL	Membuka daftar SMS yang ada pada SIM card
AT+CMGS	Mengirim pesan SMS
AT+CMGR	Membaca pesan SMS
AT+CMGD	Menghapus pesan SMS

Setelah mengirim perintah kepada GSM modem, maka modem akan memberikan respon sebagai indikator bahwa perintah kita telah berhasil atau tidak dieksekusi. Tabel 2 adalah beberapa keterangan yang menunjukkan balasan respon modem setelah diberikan perintah.

Tabel 2. Respon Komunikasi Data

Respon	Keterangan
OK	Command execute, no error
ERROR	Invalid command or command line too long
RING	Ring detected
NO CARRIER	Link not established or disconnected
NO DIALTONE	No dial tone, dialing impossible, wrong mode
BUSY	Remote station busy

2.3. Sistem Operasi Android

Android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, middleware, dan aplikasi. Pada awalnya dikembangkan oleh Android Inc., sebuah perusahaan pendatang baru yang membuat perangkat lunak untuk ponsel yang kemudian dibeli oleh Google Inc. Untuk pengembangannya, dibentuklah OHA (Open Handset Alliance) konsorsium atau pembiayaan bersama dari 34 perusahaan perangkat keras, perangkat lunak, dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia [6]. Android memungkinkan penggunaannya untuk memasang aplikasi pihak ketiga, baik yang diperoleh dari toko aplikasi seperti Google Play, Amazon Appstore, ataupun dengan mengunduh dan memasang berkas APK dari situs pihak ketiga.

2.4. Arduino Mega

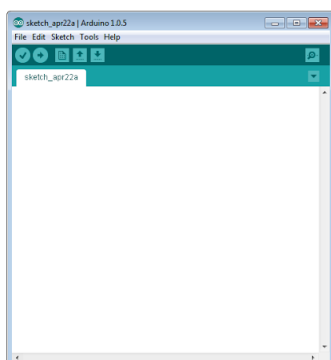
Arduino Mega 2560 adalah pengendali mikro single board berbasis mikrokontroler pada ATmega2560 yang bersifat open source. Arduino menggunakan bahasa C yang

disederhanakan dengan bantuan pustaka-pustaka (*libraries*) *Arduino*. *Arduino* juga menyederhanakan proses bekerja dengan mikrokontroler [7]. *Arduino Mega* memiliki 54 *digital input* atau *output*, dimana 14 pin digunakan untuk *PWM output* dan 16 pin digunakan sebagai *analog input*, 4 pin serial, 16 MHz osilator kristal, koneksi *USB*, *power jack ICSP header* dan tombol *reset*. Modul ini memiliki segala yang dibutuhkan untuk memprogram mikrokontroler seperti kabel *USB* dan catu daya melalui adaptor atau baterai. Semua ini diberikan untuk mendukung pemakaian mikrokontroler *Arduino* hanya terhubung ke komputer dengan kabel *USB* atau listrik dengan adaptor dari *AC-DC* atau baterai untuk memulai pemakaian. *Arduino Mega* kompatibel dengan *shield* yang dirancang untuk *Arduino Duemilanove*, *Deimila* maupun *UNO*. Gambar 3 menunjukkan bentuk fisik dari model board *Arduino Mega*.



Gambar 3. Bentuk *Arduino Mega*

Arduino juga menggunakan *software processing* yang digunakan untuk menulis program kedalam *Arduino*. *Processing* merupakan penggabungan antara bahasa C++ dan Java. Gambar 2.8 menunjukkan tampilan utama perangkat lunak *Arduino*. Gambar 4 menunjukkan tampilan utama perangkat lunak *Arduino*.



Gambar 4. Tampilan Perangkat Lunak *Arduino IDE*

2.5. Relay

Relay adalah saklar elektronik yang dapat membuka atau menutup rangkaian dengan menggunakan kontrol dari rangkaian elektronik lain. Sebuah *relay* tersusun atas kumparan, pegas, saklar yang terhubung pada pegas dan dua kontak elektronik *NC* dan *NO* [8].

- NC* (*Normally close*) adalah saklar yang terhubung dengan kontak saat kondisi *relay* tidak aktif.
- NO* (*Normally open*) adalah saklar yang terhubung dengan kontak saat kondisi *relay* aktif.

Gambar 5 adalah bentuk fisik dari *relay*.



Gambar 5. Bentuk *Relay*

2.6. Modul *GSM SIM900*

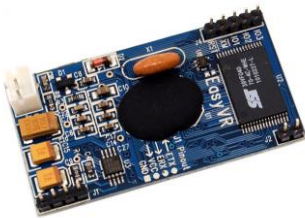
Modul *GSM SIM900* adalah sebuah perangkat elektronik yang berfungsi sebagai alat pengirim dan penerima pesan *SMS*. Modul ini merupakan modul *Quad-band GSM/GPRS* berbentuk *SMT* terbuat dari sebuah prosesor canggih *ARM926EJ-S*, sehingga ukurannya kecil (24mm x 24mm x 3mm) dan merupakan solusi yang efektif sebagai modul komunikasi. *GSM SIM900* sudah menerapkan antarmuka standar industri dalam menyediakan fitur komunikasi *GSM/GPRS* 850/900/1800/1900MHz untuk *voice*, *SMS*, data dan *fax*.



Gambar 6. Modul *GSM SIM900*

2.7. Modul *Easyvr*

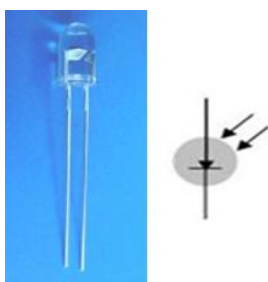
Easyvr merupakan modul *voice recognition* multi-fungsi. Dapat digunakan pada banyak aplikasi pengontrolan yang membutuhkan pendeteksian bukan hanya suara melainkan percakapan. *Easyvr* merupakan generasi penerus setelah kesuksesan generasi pertamanya di pasaran yaitu *VRBot*. Modul ini dapat digunakan/dihubungkan dengan *board* mikrokontroler *Arduino*. Sangat cocok digunakan untuk beragam aplikasi, seperti *home automation* (dimana modul dapat membantu mengontrol nyala lampu, kunci pintu, televisi, atau perangkat lainnya hanya dengan perintah suara) atau sebagai modul pelengkap sensor pendengaran robot yang dibuat sebagaimana robot-robot canggih yang dijual di pasaran yang harganya luar biasa mahal.



Gambar 7. Modul *EasyVR*

2.8. Sensor *Photodiode*

Photodiode adalah sebuah dioda semikonduktor yang berfungsi sebagai sensor cahaya. *Photodiode* memiliki hambatan yang sangat tinggi pada saat dibias mundur. Hambatan ini akan berkurang ketika *photodiode* disinari cahaya dengan panjang gelombang yang tepat. Sehingga *photodiode* dapat digunakan sebagai detektor cahaya dengan memonitor arus yang mengalir melaluinya [9].



Gambar 8. Sensor *Photodiode*

3. METODOLOGI PENELITIAN

3.1. Studi Pustaka

Melakukan studi pustaka dengan mempelajari buku-buku yang relevan dengan teori-teori sistem kontrol, *Arduino*, sensor suara, *SMS Gateway*, bahasa pemrograman yang akan digunakan serta sistem-sistem pendukung lainnya. Sumber-sumber lain yang dapat digunakan selain buku adalah jurnal ilmiah, dan berbagai artikel yang terdapat di *blog*, *website*, dan forum di internet.

3.2. Analisa Kebutuhan

Analisa kebutuhan meliputi kebutuhan akan perangkat keras dan perangkat lunak.

3.3. Perencanaan Penelitian

3.3.1. Perancangan Sistem

Prinsip kerja pada sistem ini dimulai dari perintah *user* yang diberikan ke modul *Easyvr*. Jika modul mendapat input suara, maka *Arduino* akan memerintahkan modul *Easyvr* untuk mendeteksi ucapan *user* kemudian membandingkan atau mencocokkan dengan rekaman suara yang ada di *database* modul. Jika ditemukan data yang sama dengan ucapan *user* maka *output* dari sensor akan diolah kembali oleh *Arduino* yang kemudian menerjemahkannya menjadi kode sehingga dapat memberikan perintah ke *relay* untuk mengendalikan lampu. Status lampu dikirimkan *Arduino* ke *smartphone Android* dengan bantuan modul *GSM SIM900*. *Arduino* juga dapat menerima dan mengolah pesan dari *smartphone Android* untuk memberikan perintah kepada *relay* sesuai dengan kode *SMS* yang diberikan.

3.3.2. Perancangan Perangkat Keras

Perancangan perangkat keras dilakukan dengan merangkai berbagai komponen seperti *Arduino Mega*, modul *Easyvr*, modul *GSM SIM900*, *LCD*, *relay*, rangkaian lampu, *photodiode* serta elemen pendukung lainnya seperti kabel penghubung, adaptor dan lain-lain.

3.3.3. Perancangan Perangkat Lunak

Bahasa pemrograman yang digunakan adalah bahasa pemrograman C *Arduino IDE* dan pemrograman *APP Inventor* untuk membuat aplikasi *interface* pada *Android*. Urutan atau algoritma

pembuatan program dirancang mengacu pada alur kerja sistem, yang mana program ini memungkinkan *Arduino* untuk memproses sinyal dari perangkat *input* dan memberikan respon serta tindakan ke perangkat *output*.

3.4. Integrasi

Pada tahap ini, hasil dari perencanaan dan analisa kebutuhan dieksekusi untuk dijadikan sebuah sistem secara keseluruhan. Tahap integrasi dibutuhkan guna merealisasikan sistem atau sistem kedalam bentuk riil dengan melakukan integrasi perancangan sistem perangkat keras dan perangkat lunak.

3.5. Pengujian

Tahap ini dilakukan untuk mencoba kinerja dari semua sistem yang telah dibangun. Tahapan pengujian melibatkan uji coba terhadap perangkat keras dan perangkat lunak.

3.6. Penerapan

Pada tahap penerapan merupakan tahap akhir setelah dilakukan serangkaian pengujian terhadap sistem.

3.7. Analisa dan Kesimpulan

Analisa dilakukan untuk mendapatkan kebutuhan-kebutuhan guna dalam pengembangan sistem lebih lanjut. Disamping itu juga analisa dilakukan untuk mempelajari dan memahami gambaran umum tentang aplikasi yang menerapkan metode yang digunakan dalam proses pengenalan karakter. Setelah dilakukan analisa, selanjutnya ditarik kesimpulan terhadap keberhasilan sistem yang dibuat.

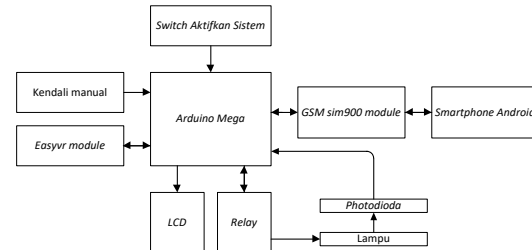
4. PERANCANGAN DAN PEMBUATAN SISTEM

Perancangan yang dilakukan dalam penelitian ini meliputi perancangan perangkat keras dan perancangan perangkat lunak. Perancangan perangkat keras dimulai dengan merancang diagram blok dan prinsip kerja sistem, kemudian dilanjutkan merancang rangkaian sistem dengan menggabungkan keseluruhan perangkat menjadi sebuah sistem terkendali.

4.1. Diagram Blok dan Prinsip Kerja Sistem

4.1.1. Diagram Blok Sistem

Untuk memudahkan perancangan sistem diperlukan sebuah diagram blok sistem. Gambar 9 merupakan diagram blok dari rancang bangun sistem kendali lampu menggunakan sensor suara berbasis *Arduino* dengan aplikasi pemantauan pada *smartphone Android*.



Gambar 9. Diagram Blok Perancangan Perangkat Keras

4.1.2. Prinsip Kerja Sistem

Gambar 9 menunjukkan keseluruhan sistem atau perangkat keras yang akan digunakan. Terdapat beberapa perangkat masukan seperti modul suara *Easyvr*, modul *GSM*, rangkaian *switch* dan *photodiode*. Sedangkan perangkat keluaran yaitu *LCD*, modul suara *EasyVR*, modul *GSM* dan rangkaian *relay* yang terhubung dengan lampu. Sistem ini akan bekerja berdasarkan perancangan secara menyeluruh dan terintegrasi sesuai dengan fungsi dan tugas dari masing-masing rangkaian.

4.2. Perancangan Perangkat Keras

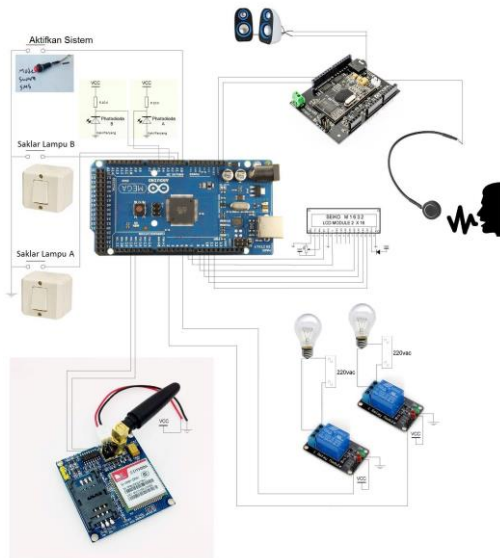
Perangkat keras adalah komponen utama dan terpenting dalam pembuatan sistem ini. Adapun perancangan perangkat keras dalam penelitian ini antara lain:

- Perancangan rangkaian *LCD*
- Perancangan rangkaian *switch*
- Perancangan rangkaian *relay* dan lampu
- Perancangan rangkaian modul *Easyvr*
- Perancangan rangkaian modul *GSM*
- Perancangan rangkaian *photodiode*

4.2.1. Perancangan Keseluruhan Perangkat Keras

Setelah dilakukan perancangan terhadap masing-masing perangkat keras, maka tahap selanjutnya adalah tahap perancangan untuk keseluruhan perangkat keras. Perangkat keras seperti *LCD*, *switch*, *relay*, modul *Easyvr*, modul *GSM* dan *photodiode* dihubungkan menjadi satu

kesatuan dengan *Arduino* melalui *port-port* yang telah ditentukan. Pada tahapan ini, keseluruhan unit perangkat keras atau rangkaian sistem diwujudkan menjadi sebuah prototipe sehingga dapat disimulasikan secara langsung. Gambar 10 adalah rangkaian skematik untuk keseluruhan perangkat keras yang digunakan pada penelitian ini.



Gambar 10. Rangkaian Keseluruhan Perangkat Keras

4.3. Perancangan Perangkat Lunak

Perangkat lunak sangat diperlukan sebagai protokol antara *Arduino* dengan komponen-komponen perangkat keras lainnya. Berdasarkan konsep pada perancangan *hardware*, maka program yang dirancang diharapkan bisa mengolah informasi yang akan digunakan pada proses kontrol dan pemantauan yang dikirimkan dari *smartphone* ke mikrokontroler ataupun sebaliknya. Pada penelitian ini perancangan perangkat lunak dibagi menjadi dua, yaitu perancangan pada pemrograman *Arduino* dan perancangan pemrograman pada aplikasi *Android*. Perancangan dan pembuatan perangkat lunak pada *Arduino* menggunakan *Arduino IDE*, sedangkan untuk aplikasi *Android* menggunakan program *APP Inventor*.

4.3.1. Perancangan Algoritma Pemrograman *Arduino Mega*

Perancangan algoritma program bertujuan untuk menentukan alur program sebelum program ditulis dan dimasukkan ke

dalam *Arduino*. Perancangan algoritma akan mempermudah saat penulisan program dan membuat penulisan program lebih terarah. Algoritma program akan mendefinisikan tindakan yang akan diambil oleh *Arduino* seperti menerima sinyal masukan dan memberikan sinyal keluaran pada perangkat keras. Algoritma pemrograman ini berfungsi mendefinisikan variabel yang digunakan untuk penulisan program. Berikut merupakan algoritma pemrograman yang digunakan pada *Arduino*.

- Mulai
- Perintah kontrol dibagi menjadi tiga, melalui perintah suara, sms *smartphone Android* dan secara manual.
- Jika *switch* yang dipilih adalah *switch* aktifkan sistem, maka perintah kendali dapat dilakukan dengan menggunakan suara atau *sms*. Urutan eksekusi program dimulai dari kendali suara terlebih dahulu baru mengeksekusi kendali *sms*.
- Jika dideteksi ada suara masuk sesuai kode yang diprogram, maka sinyal atau data yang diterima dari modul *Easyvr* oleh *Arduino* akan diproses dan dikirimkan ke *relay*.
- Jika kode perintah suara yang diterima oleh *arduino* berisikan perintah *ON*, maka *relay* dalam posisi aktif dan lampu akan menyala, di *LCD* akan ditampilkan info "Lampu ON", *Arduino* akan mengirimkan status "Lampu ON" ke *smartphone* melalui modul *GSM*.
- Jika kode perintah suara yang diterima oleh *Arduino* berisikan perintah *OFF*, maka *relay* dalam posisi tidak aktif dan lampu tidak menyala, di *LCD* akan ditampilkan info "Lampu OFF", *arduino* akan mengirimkan status "Lampu OFF" ke *smartphone*.
- Jika tidak ada suara yang masuk maka selanjutnya sistem akan mengecek ada atau tidaknya *SMS*

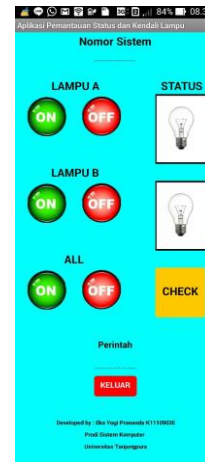
masuk yang dikirimkan *smartphone Android*.

- h. Jika dideteksi ada *SMS* masuk sesuai kode yang diprogram, maka sinyal atau data yang diterima dari modul *GSM* oleh *Arduino* akan diproses dan dikirimkan ke *relay*.
- i. Jika format *sms* yang diterima oleh *Arduino* berisikan perintah *ON*, maka *relay* dalam posisi aktif dan lampu akan menyala, di *LCD* akan ditampilkan info “ON”, *arduino* akan mengirimkan status “Lampu ON” ke *smartphone*.
- j. Jika format *sms* yang diterima oleh *Arduino* berisikan perintah *OFF*, maka *relay* dalam posisi tidak aktif dan lampu tidak menyala, di *LCD* akan ditampilkan info “OFF”, *arduino* akan mengirimkan status “Lampu OFF” ke *smartphone*.
- k. Jika *switch* aktifkan sistem dimatikan, maka proses kendali dilakukan dengan cara manual menekan *switch on-off* yang ada.
- l. Jika posisi *switch ON*, maka relay aktif dan lampu akan menyala, di *LCD* akan ditampilkan info “ON”, *arduino* akan mengirimkan status “Lampu ON” ke *smartphone*.
- m. Jika posisi *switch OFF*, maka *relay* tidak aktif dan lampu tidak menyala, di *LCD* akan ditampilkan info “OFF”, *arduino* akan mengirimkan status “Lampu OFF” ke *smartphone*.
- n. Selesai

4.3.2. Perancangan Perangkat Lunak *Android*

Aplikasi *Android* dirancang dan dibuat sebagai media *interface* untuk membantu pengguna dalam mengendalikan sistem atau perangkat keras. Pembuatan aplikasi yaitu menggunakan program *APP Inventor*. *APP Inventor* adalah salah satu program bawaan dari penyedia aplikasi *Android* yaitu *Google* yang berfungsi sebagai program tambahan untuk media pembelajaran bagi pemula yang ingin membuat aplikasi *Android* dan masih tergolong sederhana. Gambar 11 adalah

tampilan aplikasi *interface* sistem setelah di-*install* pada *smartphone Android*.



Gambar 11. Tampilan Aplikasi pada *Smartphone Android*

5. PENGUJIAN DAN ANALISA

5.1. Pengujian Aplikasi *Android*

Pengujian ini bertujuan untuk mengetahui apakah aplikasi *Android* yang berfungsi sebagai media antarmuka bagi sistem dapat berjalan sesuai dengan yang diinginkan dan juga untuk mengetahui jenis atau tipe *smartphone Android* yang mendukung untuk aplikasi ini. Tabel 3 adalah hasil dari pengujian aplikasi yang dilakukan pada beberapa jenis *smartphone Android*.

Tabel 3. Pengujian Aplikasi pada Beberapa Jenis *Smartphone Android*

No	Tipe <i>smartphone Android</i>	Versi OS	Ukuran layar (inch)	Resolusi layar (Pixel)	Tampilan layar
1.	Asus Zenfone 5	4.3	5"	720x1280	Sesuai
2.	Lenovo S4D Gaia	4.4	4,7"	960x540	Sesuai
3.	Lenovo A390	4.0	4"	480x800	Sesuai
4.	Xiaomi Mi 4i	5.0	5"	1080x1920	Sesuai
5.	Samsung Core Duos	4.1	4,3"	480x800	Sesuai
6.	Samsung Note 2	4.4	5,5"	720x180	Sesuai
7.	Samsung Galaxy Tab 2	4.0	7"	600x1024	Sesuai
8.	Lenovo Tab A300	4.2	7"	600x1024	Sesuai
9.	Sony Xperia Z	4.1	5"	1080x1920	Sesuai
10.	Asus Zenfone 2	5.0	5,5"	1080x1920	Sesuai

5.2. Pengujian Keseluruhan Sistem

Pengujian keseluruhan ini melibatkan kinerja semua komponen guna untuk mengetahui apakah alat mampu bekerja dan dapat menghasilkan nilai keluaran yang diinginkan sesuai dengan nilai masukan yang diberikan. Pengujian keseluruhan sistem dilakukan dengan merangkai semua komponen secara terpadu.

Pengujian terbagi menjadi menjadi empat bagian yaitu:

- Pengujian mode manual dengan saklar
- Pengujian mode SMS dengan aplikasi antarmuka *Android*
- Pengujian mode suara
- Pengujian notifikasi *error*

5.2.1. Pengujian mode manual dengan saklar

Pengujian ini bertujuan untuk mengetahui apakah mode manual pengendalian lampu dalam sistem ini dapat berfungsi sesuai dengan yang diharapkan, yaitu dengan menggunakan *switch* atau saklar sebagai alat kendali. Tabel 4 adalah hasil pengujian dengan mode manual menggunakan saklar.

Tabel 4. Hasil Pengujian Mode Manual dengan Saklar

Switch Manual	Hasil	Keterangan
Switch A ditekan	Lampu A menyala	Berhasil
Switch A dilepas	Lampu A padam	Berhasil
Switch B ditekan	Lampu B menyala	Berhasil
Switch B dilepas	Lampu B padam	Berhasil

5.2.2. Pengujian mode SMS dengan aplikasi antarmuka *Android*

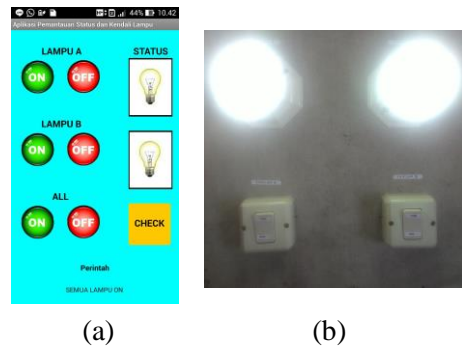
Pengujian ini bertujuan untuk mengetahui apakah proses kendali lampu dengan SMS yang didukung oleh aplikasi tambahan pada *smartphone Android* sebagai media antarmuka pemantauan sistem dapat berjalan sesuai tujuan penelitian. Keuntungan dari SMS yaitu untuk setiap proses kendali dan pemantauan lampu dapat dilakukan dari tempat manapun yang masih terjangkau oleh sinyal seluler. Tabel 5 adalah hasil pengujian

mode SMS dengan menggunakan aplikasi *Android*.

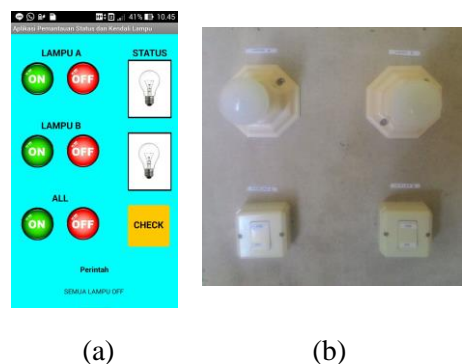
Tabel 5. Hasil Pengujian Mode SMS dengan Aplikasi *Android*

Uji ke-	Perintah	Kondisi Awal	Waktu kirim perintah sampai lampu menyala/padam (detik)	Kondisi Akhir	Waktu kirim perintah sampai menerima notifikasi (detik)	Notifikasi	Status	Ket
1	LAMPU A ON	A OFF	9	A ON	19	Lampu A Menyala	Lampu A Menyala Lampu B Mati	Berhasil
2	LAMPU A OFF	A ON	11	A OFF	24	Lampu A Mati	Lampu A Mati Lampu B Mati	Berhasil
3	LAMPU B ON	B OFF	5	B ON	13	Lampu B Menyala	Lampu A Mati Lampu B Menyala	Berhasil
4	LAMPU B OFF	B ON	14	B OFF	28	Lampu B Mati	Lampu A Mati Lampu B Mati	Berhasil
5	ALL ON	A OFF B OFF	8	A ON B ON	28	A dan B Menyala	Lampu A Menyala Lampu B Menyala	Berhasil
6	ALL OFF	A ON B ON	15	A OFF B OFF	24	A dan B Mati	Lampu A Mati Lampu B Mati	Berhasil
Jumlah			62		136			Uji Coba: 6 Berhasil: 6 Gagal: 0
Rata-rata			10,3		22,7			

Gambar 12 dan 13 adalah hasil tampilan pada pengujian mode SMS dengan aplikasi *Android*.



Gambar 12. Hasil Pengujian Perintah “SEMUA LAMPU ON” (a) Tampilan pada aplikasi *Android* (b) Tampilan pada semua lampu



Gambar 13. Hasil Pengujian Perintah “SEMUA LAMPU OFF” (a) Tampilan pada aplikasi *Android* (b) Tampilan pada semua lampu

5.2.3. Pengujian mode suara

Untuk rata-rata waktu proses kendali menggunakan suara dari sejak pertama kali perintah diberikan sampai lampu A menyala adalah 4,3 detik, lampu B 4,5 detik dan semua lampu 4,4 detik, sedangkan rata-rata waktu penerimaan pesan balasan dari sejak awal perintah diberikan adalah lampu A 12,7 detik, lampu B 15,8 detik dan semua lampu 19,8 detik. Kemudian rata-rata waktu proses kendali menggunakan suara dari sejak pertama kali perintah diberikan sampai lampu A mati adalah 4,8 detik, lampu B 4,4 detik dan semua lampu 4,4 detik, sedangkan rata-rata waktu penerimaan pesan balasan dari sejak awal perintah diberikan adalah lampu A 14,2 detik, lampu B 17 detik dan semua lampu 16,4 detik.



Gambar 14. Lampu LED Hijau Menyala Saat Mode SMS dan Suara Aktif

5.2.4. Pengujian notifikasi error

Kerusakan pada lampu dapat diketahui dengan membandingkan informasi yang ada pada relay sebagai komponen elektromekanik sistem dan lampu sebagai *ouput* akhir pengendalian dari sistem melalui *photodiode*. Jika terdapat ketidakcocokan antara informasi dari *relay* dan lampu maka hal ini memenuhi syarat untuk didefinisikan sebagai kondisi *error*, dan sistem dapat memberikan notifikasi kepada pengguna mengenai deteksi awal kerusakan pada perangkat.



Gambar 15. Lampu LED Merah Berkedip Saat Sistem Mendeteksi Adanya Error pada Lampu

5.3. Analisa Pengujian

Dari semua pengujian yang telah dilakukan, dapat dijelaskan bahwa sistem kendali lampu menggunakan sensor suara berbasis *Arduino* dengan aplikasi pemantauan pada *smartphone Android* ini berfungsi sesuai dengan yang direncanakan. Perangkat keras seperti *Arduino Mega*, *LCD*, *switch*, *relay*, modul *Easyvr*, modul *GSM*, *photodiode* dan perangkat lunak aplikasi *Android* dapat berfungsi dengan baik. Hal ini dibuktikan dari kinerja sistem yang dapat menghasilkan keluaran sesuai dengan nilai masukan yang diberikan.

6. KESIMPULAN DAN SARAN

6.1. Kesimpulan

- Sistem ini dapat digunakan untuk menyalakan maupun mematikan lampu dengan cara *switch* manual, melalui *SMS* berbasis aplikasi *Android* serta dengan perintah suara, dan juga dapat memantau status lampu dari jarak jauh melalui aplikasi *Android* yang sama, dengan memanfaatkan *Arduino Mega* sebagai modul pengendali utama.
- Rata-rata waktu proses kendali menggunakan *SMS* dari sejak pertama kali perintah diberikan sampai lampu menyala adalah 10,3 detik, sedangkan rata-rata waktu penerimaan pesan balasan dari sejak awal perintah diberikan adalah 22,7 detik.
- Rata-rata waktu proses kendali menggunakan suara dari sejak pertama kali perintah diberikan sampai lampu A menyala adalah 4,3 detik, lampu B 4,5 detik dan semua lampu 4,4 detik, sedangkan rata-rata waktu penerimaan pesan balasan dari sejak awal perintah diberikan adalah lampu A 12,7 detik, lampu B 15,8 detik dan semua lampu 19,8 detik.
- Rata-rata waktu proses kendali menggunakan suara dari sejak pertama kali perintah diberikan sampai lampu A mati adalah 4,8 detik, lampu B 4,4 detik dan semua lampu 4,4 detik, sedangkan rata-

rata waktu penerimaan pesan balasan dari sejak awal perintah diberikan adalah lampu A 14,2 detik, lampu B 17 detik dan semua lampu 16,4 detik.

- e. Rata-rata waktu lampu mulai mengalami *error* sampai menerima notifikasi yaitu 13,7 detik.

6.2. Saran

- a. Penggunaan metode berbasis *web* maupun *cloud computing* untuk komunikasi antara *smartphone android* dengan *Arduino* atau otak utama pengendali sistem.
- b. Menambah jumlah lampu dan atau jenis peralatan listrik yang dikendalikan.
- c. Perlu dilakukan pengembangan yang lebih lanjut seperti desain dan konsep aplikasi antarmuka *Android* agar lebih kaya fitur tetapi tetap mempertahankan kemudahan dan kecepatan dalam penggunaan.
- d. Perlu dilakukan pengembangan aplikasi antarmuka di sistem operasi yang berbeda-beda seperti *iOS*, *Windows*, *RIM* dan lain sebagainya.

DAFTAR PUSTAKA

- [1] Setiawan, E. T. (2015). Pengendalian Lampu Rumah Berbasis Mikrokontroler *Arduino* Menggunakan *Smartphone Android*. Pangkalpinang: STMIK Atma Luhur.
- [2] Oetomo, Budi Sutedjo Dharma., dan Yosi Handoko. (2003). Telekses *Database* Pendidikan Berbasis Ponsel. Yogyakarta: Penerbit Andi.
- [3] Mubarak, Abdul Gofal Al. (2014). Pengendalian Peralatan Listrik Rumah Tangga Melalui *Short Message Service (SMS)* dengan Berbasis Mikrokontroller AT89S51. Diakses Juli 18, 2016. http://s3.amazonaws.com/academia.edu/documents/34818299/Jurnal_Abdul_Goffar-libre.pdf.
- [4] Hikmawan, Agung Tri. (2011). Pengembangan Sistem *Sms Gateway Berbasis Web Service* Untuk Penyebaran Informasi Antar Anggota Perusahaan Dengan Metode *Sms Grouping*. Diakses Agustus 9, 2016. <http://digilib.its.ac.id/public/ITS-Undergraduate-15874-Paper-pdf>.
- [5] Lesmana, Dony Hendra. (2014). Rancang Bangun Sistem *Monitoring* Kecepatan Kendaraan Berbasis *GPS* dengan *SMS* sebagai Media Pengiriman Data. Diakses 16 Juli, 2016. <http://elektro.studentjournal.ub.ac.id/index.php/teub/article/download/165/128>.
- [6] Ichwan, M. (2011). Pengukuran Kinerja *Goodreads Application Programming Interface (Api)* Pada Aplikasi *Mobile Android* (Studi Kasus Untuk Pencarian Data Buku). Vol 2, No 2, Halaman 13-21. Diakses Agustus 9, 2016. <http://lib.itenas.ac.id/kti/wp-content/uploads/2013/10/No.-2-Vol.-2-Mei-Agustus-2011-2.pdf>.
- [7] Kadir, Abdul. (2013). Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemrogramannya Menggunakan *Arduino*. Yogyakarta: Penerbit ANDI.
- [8] Nugroho, Ichsan Dwi. (2012). Alat Pengatur Lampu Dan Pembalik Telur Otomatis Pada Blok Penetasan Telur Berbasis Mikrokontroler *Atmega 16* Dilengkapi *Uninterruptible Power Supply*. Diakses Juni 31, 2016.
- [9] Rudiyanto, Hariz Bafdal. 2010. Rancang Bangun Robot Pengantar Surat Menggunakan Mikrokontroler *At89s51*. Diakses Juli 18, 2016. http://www.gunadarma.ac.id/library/articles/graduate/industrial-technology/2010/Artikel_10405805.pdf