

Pengembangan Pengaturan Lampu Lalu Lintas Berbasis IoT

Daniel¹, Toibah Umi Kalsum², Riska³

¹Mahasiswa, Program Studi Rekayasa Sistem Komputer Fakultas Ilmu Komputer Universitas Dehasen Bengkulu
Jalan Meranti Raya No.32 Sawah Lebar Telp. (0736) 22027, 26957 Fax. (0736) 341139;
e-mail: danielstiven704@gmail.com

^{2,3}Dosen Tetap Program Studi Rekayasa Sistem Komputer Fakultas Ilmu Komputer Universitas Dehasen Bengkulu
Jl. Meranti Raya No. 32 Kota Bengkulu 38228 Telp. (0736) 22027, 26957 Fax. (0736) 341139;
e-mail: cicik.umie@gmail.com, riska.iskandar@unived.ac.id

(Received: Nopember 2024, Revised : Februari 2024, Accepied : April 2024)

Abstract-Traffic lights are lights that are used to regulate the smooth flow of traffic at a road intersection. The traffic light system in the city of Bengkulu currently still uses a time system that is already installed in the system and does not have a feature for setting the switching time. This system means that operators cannot change the flashing time of traffic lights to suit road conditions and vehicle density at each intersection. In this research, an IoT-based traffic light control system is developed so that it can be controlled centrally using Arduino Mega, Arduino Uno, ENC28J60 as a traffic light machine and using a web-based application as a centralized control system. The research method used is the prototype method, where by using this method there are several processes that must be carried out, including analysis, planning, building a prototype, evaluating initial use, improving the prototype, and implementing and maintaining. The results of this research are that traffic lights can be controlled centrally using a web-based application, where the response time from the Arduino Mega only takes a matter of seconds. From tests carried out on one of the buttons, a response time of 1010.4 ms (milliseconds). Apart from that, the system being built can also display status messages when selecting the status button on the web page based on the selected intersection, then the web application will display information on the intersection status, intersection mode and intersection location.

Keywords: Traffic Lights, Arduino, IoT, Web Control, Prototype.

Information:

Intisari-Lampu lalu lintas merupakan lampu yang digunakan untuk mengatur kelancaran arus lalu lintas pada suatu persimpangan jalan. Sistem lampu lalu lintas di kota Bengkulu saat ini masih menggunakan sistem waktu yang sudah terpasang pada sistemnya dan tidak mempunyai fitur untuk pengaturan waktu penyalannya. Sistem tersebut, menyebabkan operator tidak dapat mengubah waktu nyala lampu lalu lintas menyesuaikan dengan kondisi jalan dan kepadatan kendaraan di setiap persimpangan. Pada penelitian ini dikembangkan sistem kendali lampu lalu lintas berbasis IoT sehingga dapat dikendalikan secara terpusat dengan menggunakan Arduino Mega, Arduino Uno, ENC28J60 sebagai mesin lampu lalu lintas dan menggunakan aplikasi berbasis web sebagai sistem kendali terpusat. Metode penelitian yang digunakan adalah metode prototipe, dimana dengan menggunakan metode ini ada

beberapa proses yang harus dilakukan antara lain analisis, perencanaan, pembuatan prototipe, evaluasi penggunaan awal, perbaikan prototipe, serta implementasi dan pemeliharaan. Hasil dari penelitian ini adalah lampu lalu lintas dapat dikontrol secara terpusat menggunakan aplikasi berbasis web, dimana waktu respon dari Arduino Mega hanya membutuhkan waktu hitungan detik, dari pengujian yang telah dilakukan pada salah satu tombol diperoleh waktu respon sebesar 1010.4 ms (milliseconds). Selain itu sistem yang dibangun juga dapat menampilkan pesan status pada saat pemilihan tombol status pada halaman web berdasarkan simpang yang dipilih, kemudian aplikasi web akan menampilkan informasi status simpang, mode simpang dan lokasi simpang

Kata Kunci: Lampu Lalu Lintas, Arduino, IoT, Web Kontrol, Prototipe.

I .PENDAHULUAN

Sistem lampu lalu lintas yang ada di kota Bengkulu saat ini masih menggunakan sistem waktu yang sudah terpasang pada sistemnya dan tidak memiliki fitur pengaturan waktu penyalannya / *fixed time*. Metode tersebut menyebabkan operator tidak dapat mengubah waktu nyala lampu lalu lintas, untuk menyesuaikan kondisi jalan dan kepadatan kendaraan yang ada di setiap persimpangan. Hal tersebut merupakan salah satu kekurangan dari pengendalian lampu lalu lintas yang ada pada saat ini. Berdasarkan hasil wawancara yang telah penulis lakukan oleh seksi lalu lintas jalan dinas perhubungan Provinsi Bengkulu mengatakan bahwa pernah akan diterapkan sistem ATCS (*Area Traffic Control System*), tetapi karena penerapan ATCS ini cukup mahal dinas perhubungan Provinsi Bengkulu membatalkan untuk penerapan sistem ATCS. Sistem tersebut merupakan suatu sistem pengendalian lalu lintas berbasis teknologi informasi pada suatu kawasan, yang bertujuan untuk mengoptimalkan kinerja jaringan jalan melalui optimasi dan kordinasi pengaturan lampu lalu lintas di setiap persimpangan untuk mengatur waktu penyalan lampu lalu lintas dan, dalam keadaan tertentu memberikan waktu hijau pada kendaraan yang memiliki prioritas (Pemadam

Kendaraan, Ambulance, VVIP, Konvoi, Dll). Menurut Purba, Sulistyorini, Sadnowo dan, Ilhami (Seminar Nasional AVoER IX 2017) penerapan ATCS ini cukup mahal dan rentan terhadap gangguan seperti perilaku pengemudi serta ketidakstabilan jaringan listrik. Hal tersebut di atas yang mendasari penulis ingin merancang dan membuat sistem pengaturan lampu lalu lintas berbasis IoT. Sehingga dengan adanya sistem tersebut, lampu lalu lintas dapat dikontrol waktu penyalanya tanpa harus datang kelokasi untuk mengubah waktu penyalanya. Oleh karena keadaan yang telah diuraikan di atas, maka dari itu penulis mengangkat judul “Pengembangan Pengaturan Lampu Lalu Lintas Berbasis IoT”.

II. TINJAUAN PUSTAKA

Menurut Fahrul (2018: 7), pengembangan adalah kegiatan ilmu pengetahuan dan teknologi yang bertujuan memanfaatkan kaidah dan teori ilmu pengetahuan yang telah terbukti kebenarannya untuk meningkatkan fungsi, manfaat, serta aplikasi ilmu pengetahuan dan teknologi yang telah ada atau, menghasilkan teknologi baru.

Berdasarkan pengertian para ahli dapat disimpulkan bahwa pengembangan merupakan perluasan atau pendalaman suatu materi pembelajaran sehingga menghasilkan suatu produk baru.

Lampu Lalu Lintas

Pengertian Umum Lampu Lalu Lintas

Menurut Herdian (2019: 23), pengaturan lalu lintas di persimpangan jalan dimaksudkan untuk mengatur pergerakan kendaraan pada masing-masing kelompok pergerakan kendaraan agar dapat bergerak secara bergantian sehingga tidak saling mengganggu antar-arus yang ada. Berdasarkan pengertian pengertian para ahli dapat disimpulkan bahwa lampu lalu lintas ini yang menandakan kapan kendaraan harus berjalan dan berhenti secara bergantian dari berbagai arah.

Tujuan Adanya Lampu Lalu Lintas

- a. Menghindari hambatan karena adanya perbedaan arus jalan bagi pergerakan kendaraan.
- b. Memfasilitasi persimpangan antara jalan utama untuk kendaraan dan pejalan kaki dengan jalan sekunder sehingga kelancaran arus lalu lintas dapat terjamin.
- c. Mengurangi tingkat kecelakaan yang diakibatkan oleh tabrakan karena perbedaan arus jalan

Mikrokontroler

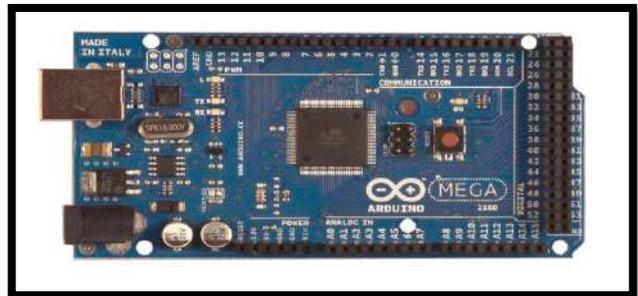
Menurut Fauzi (2019: 12), mikrokontroler adalah sebuah chip yang berfungsi sebagai pengontrol atau pengendali rangkaian elektronik dan umumnya dapat menyimpan program di dalamnya, program yang ada di mikrokontroler bisa dihapus dan ditulis ulang.

Arduino

Arduino adalah perangkat keras terbuka yang ditujukan kepada siapa saja yang ingin membuat purwarupa peralatan elektronik interaktif berdasarkan *hardware* dan *software* yang *fleksibel* dan mudah digunakan. Pada aplikasi Arduino, Mikrokontroler diprogram menggunakan bahasa pemrograman C atau C++, dengan pustaka khas Arduino (Wikipedia Indonesia, 2023)

Arduino Mega 2560

Arduino Mega 2560 merupakan pengembangan dari papan Arduino Mega sebelumnya. Pada awalnya Arduino Mega menggunakan chip Atmega1280 yang kemudian diubah menjadi chip Atmega2560, karena penggantian nama tersebut maka sekarang lebih dikenal dengan nama Arduino Mega 2560. Sampai saat ini Arduino Mega 2560 telah sampai pada revisi yang ke 3 (Eko 2021 : 6), Menurut Saputro (2018 : 17), Arduino Mega adalah papan pengembangan mikrokontroler yang berbasis Arduino dengan menggunakan *chip* ATmega2560. *Board* ini memiliki pin I/O yang cukup banyak, sejumlah 54 buah digital I/O pin (15 pin diantaranya adalah PWM), 16 pin analog input, 4 pin UART (*serial port hardware*). Arduino Mega 2560 dilengkapi dengan sebuah oscillator 16 Mhz, sebuah port USB, *power jack* DC, ICSP *header*, dan tombol *reset*. *Board* ini sudah sangat lengkap, sudah memiliki segala sesuatu yang dibutuhkan untuk sebuah mikrokontroler.



Gambar 1 Arduino Mega 2560

Tabel 1. Spesifikasi Arduino Mega 2560

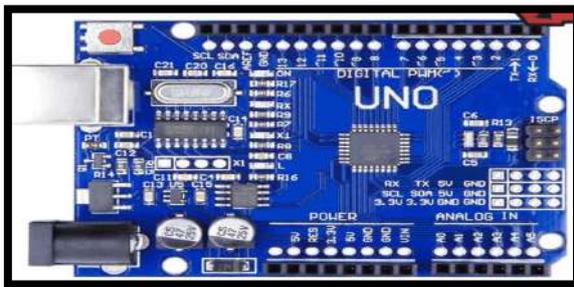
Komponen	Keterangan
Chip mikrokontroler	Atmega 2560
Tegangan operasi	5V
Tegangan input (direkomendasikan, via jack DC)	7V - 12V
Tegangan input (limit, via jack DC)	6V - 20V
Digital I/O pin	54 buah, 6 diantaranya menyediakan PWM output
Analog Input pin	16 buah

Arus DC pin	3.3V 50 mA
Arus DC per pin I/O	20 mA
Memori Flash	256 KB, 8 KB telah digunakan untuk bootloader
SRAM	8 KB
EEPROM	4 KB
Clock speed	16 Mhz
Dimensi	101.5 mm x 53.4 mm
Berat	37

Arduino Uno

Arduino Uno adalah mikrokontroler berbasis ATmega328. Memiliki 14 pin input dari output digital dimana 6 pin input tersebut dapat digunakan sebagai output PWM dan 6 pin input analog, 16 MHz osilator kristal, dan tombol *reset*. Untuk mendukung mikrokontroler agar dapat digunakan, cukup hanya menghubungkan Arduino Uno ke komputer dengan menggunakan kabel USB UART atau adaptor-DC 5v atau baterai 4,7v untuk menjalankannya (*Arduino Uno Datasheet.pdf*).

Arduino Uno adalah papan pengendali mikro dengan sumber terbuka yang berbasis mikrokontroler Microchip ATmega328P. Pengembangnya ialah Arduino.cc dan awalnya dirilis pada tahun 2010. Papan ini dilengkapi dengan set pin masukan/keluaran (I/O) digital dan analog yang dapat dihubungkan ke berbagai kartu ekspansi (perisai) dan sirkuit lainnya (Wikipedia Indonesia 2023)



Gambar 2. Arduino Uno
Tabel 2.2 Spesifikasi Arduino Uno

Komponen	Keterangan
Mikrokontroler	Atmega 328
Tegangan Operasi	5v
Jumlah pin I/O digital	14 pin digital
Jumlah pin I/O analog	6 pin analog
Arus DC tiap Pin I/O	40 mA
Memori Flash	32 kb (Atmega 328) 0,5 kb

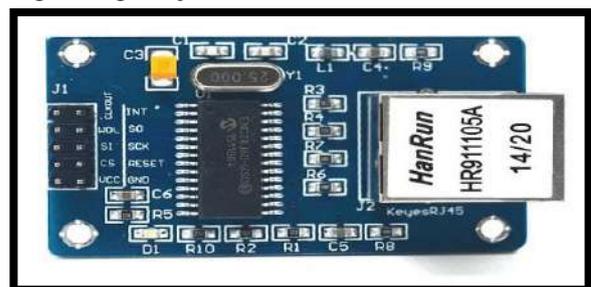
	digunakan untuk bootloader
SRAM	2 kb
EPROM	1 kb
Clock Speed	16 MHz

ENC28J60

Menurut Chandra (2020 : 11), ENC28J60 adalah sebuah modul elektronik yang bekerja sebagai *ethernet controller*. Modul ini terdiri dari chip ENC28J60 itu sendiri, konektor RJ45 female dan beberapa rangkaian komponen resistor, kapasitor dan induktor. Sebagai sebuah modul, ENC28J60 ini dapat berfungsi sebagai *gateway* Arduino (dan mikrokontroler lain) ke jaringan TCP/IP yakni bisa berupa LAN (*Local Area Network*), WAN (*Wide Area Network*) maupun MAN (*Metropolitan Area Network*). Atau lebih singkatnya dapat kita sebut jaringan intranet dan internet.

Menurut Pratama (2018 : 2) ENC28J60 merupakan *ethernet controller* yang berdiri sendiri yang bisa berkomunikasi melalui komunikasi SPI dengan mikrokontroler. IC (chips) ini yang memungkinkan mikrokontroler bisa terhubung dengan *ethernet* LAN, sehingga bisa berkomunikasi dengan perangkat network lain menggunakan protokol TCP/IP yang nantinya menjadi sebuah *web server* yang diakses dari komputer.

Berdasarkan keterangan di atas bisa disimpulkan bahwa modul ENC28J60 ini melakukan interfacing menggunakan protokol SPI (*Serial Peripheral Interface*), jadi bagi siapa saja yang mau berhubungan dengan modul ini tentunya harus support protokol komunikasi tersebut. Kalau kita kita lihat dalam keluarga Arduino yang berbasis mikrokontroler ATMEGA ternyata sudah mendukung protokol tersebut sehingga dapat langsung kita hubungkan ke pin-pin terkait. Seperti yang kita ketahui, pin-pin yang digunakan untuk berkomunikasi SPI adalah MISO (*Master In Slave Out*), MOSI (*Master Out Slave In*), SCK (*Serial Clock*) dan SS (*Slave Select*). Pin MISO/MOSI digunakan untuk bertukar data, SCK digunakan untuk *clock* sinkronisasi sedangkan pin SS dipakai untuk memilih *slave* yang akan digunakan untuk berkomunikasi (pin SS ini akan sangat berguna jika *slave* SPI lebih dari satu).



Gambar 4. ENC28J60

Wireless

Menurut Kusuma (2017 : 16), *wireless* merupakan sekumpulan komputer yang saling terhubung antara satu dengan lainnya sehingga terbentuk sebuah jaringan komputer dengan menggunakan media udara atau gelombang sebagai jalur lintas datanya. Pada dasarnya *wireless* dengan LAN merupakan sama-sama jaringan komputer yang saling terhubung antara satu dengan lainnya, yang membedakan antara keduanya adalah media jalur lintas data yang digunakan, jika LAN masih menggunakan kabel sebagai media lintas data, sedangkan *wireless* menggunakan media gelombang radio/udara.

Menurut Sundari (2018 : 10), *wireless* merupakan jaringan dengan media berupa gelombang elektromagnetik. Pada jaringan ini tidak diperlukan kabel untuk menghubungkan antar komputer karena menggunakan gelombang elektromagnetik yang akan mengirimkan sinyal informasi antar komputer jaringan.

Berdasarkan beberapa definisi di atas penulis dapat menyimpulkan bahwa jaringan *wireless* merupakan sebuah koneksi yang menghubungkan dari satu komputer ke komputer lainnya sebagai media perantara untuk mengirim data dan informasi tanpa menggunakan kabel yaitu dengan gelombang elektromagnetik.

Pemrograman

Menurut Sukrisno (2019 : 9), program adalah kata, ekspresi, atau pernyataan yang dirancang dan dirakit menjadi satu prosedur, yang terdiri dari langkah-langkah, untuk memecahkan masalah yang diimplementasikan menggunakan bahasa pemrograman yang dapat diakses komputer.

Pemrograman adalah proses menulis, menguji dan memperbaiki (*debug*), dan memelihara kode yang membangun suatu program komputer. Kode ini ditulis dalam berbagai bahasa pemrograman. Tujuan dari pemrograman adalah untuk memuat suatu program yang dapat melakukan suatu perhitungan atau pekerjaan sesuai dengan keinginan si pemrogram (Wikipedia Indonesia 2023).

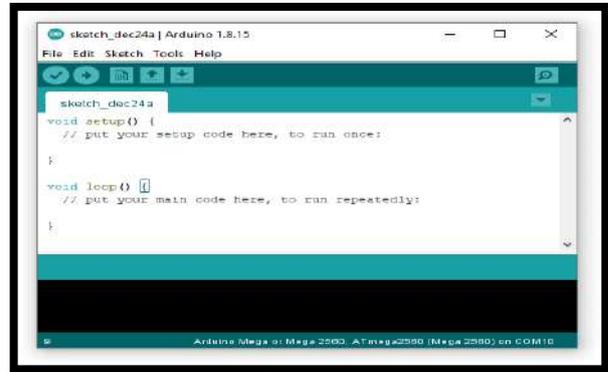
Berdasarkan keterangan di atas dapat disimpulkan bahwa pemrograman adalah proses menulis dan menguji sebuah sintak kode program untuk mendefinisikan program komputer, atau langkah untuk memecahkan masalah yang diimplementasikan menggunakan bahasa pemrograman yang dapat diakses komputer.

Arduino IDE

Menurut Eka (2020: 15), IDE (*Integrated Development Environment*) yang diperuntukan untuk membuat perintah atau *source code*, melakukan pengecekan

kesalahan, kompilasi, *upload* program, dan menguji hasil kerja Arduino melalui serial monitor.

Menurut Alamsyah (2019 : 5), Arduino IDE (*Integrate Development Enviroment*) adalah *software* yang dipakai untuk membuat, mengedit suatu kode program, memverifikasi, dan mengunggah kode program ke Arduino. Arduino menggunakan bahasa pemrograman sendiri.



Gambar 5. Arduino IDE

Pada Gambar, Arduino IDE memiliki *toolbars* IDE yang memberikan akses instan ke fungsi fungsi yang penting, yaitu :

- Tombol *Verify*, untuk mengkompilasi program yang saat ini dikerjakan.
- Tombol *Upload*, untuk mengkompilasi program dan mengunggah ke papan Arduino.
- Tombol *New*, menciptakan lembar kerja baru.
- Tombol *Open*, untuk membuka program yang ada di file sistem.
- Tombol *Save*, untuk menyimpan program yang dikerjakan.
- Tombol *Stop*, untuk menghentikan serial number yang sedang dijalankann.

III. METODOLOGI PENELITIAN

Metode Penelitian

Metode penelitian yang dipaparkan disini adalah penelitian rancang bangun dengan menggunakan metode prototipe. Berikut tahap-tahap pada metode prototipe yang digunakan dalam pengembangan sistem yang digunakan pada penelitian ini:

- Analisa

Merupakan tahapan awal dari model prototipe guna mengidentifikasi permasalahan-permasalahan yang ada, serta informasi-informasi lain yang diperlukan untuk membangun sistem.
- Perencanaan

Pembuatan desain sederhana yang akan memberi gambaran singkat tentang sistem yang ingin dibuat. Tentunya berdasarkan diskusi dari langkah 1 diawal.
- Pebangunan Prototipe

Setelah perencanaan selanjutnya adalah pembangunan prototipe sebenarnya yang akan

dijadikan rujukan untuk pembuatan alat atau program aplikasi.

4. Evaluasi Penggunaan Awal

Di tahap ini, sistem yang telah dibuat dalam bentuk prototipe di presentasikan pada klien untuk di evaluasi. Selanjutnya klien akan memberikan komentar dan saran terhadap apa yang telah dibuat.

5. Perbaiki Prototipe

Jika klien tidak mempunyai catatan revisi dari prototipe yang dibuat, maka tim bisa lanjut pada tahapan 6, namun jika klien mempunyai catatan untuk perbaikan sistem, maka fase 4-5 akan terus berulang sampai klien setuju dengan sistem yang akan dikembangkan.

6. Implementasi dan Pemeliharaan

Pada fase akhir ini, produk akan segera dibuat oleh berdasarkan prototipe akhir, selanjutnya sistem akan diuji dan diserahkan pada klien. Selanjutnya adalah fase pemeliharaan agar sistem berjalan lancar tanpa kendala.

IV. HASIL DAN PEMBAHASAN

Hasil

Hasil Rancangan Alat

Pada bab ini akan menjelaskan hasil dari penelitian yang yang sudah diajukan sebelumnya, dimana penelitian yang dilakukan adalah pengembangan pengaturan lampu lalu lintas berbasis IoT. Adapun hasil dari penelitian yang sudah dibuat adalah sebagai berikut.



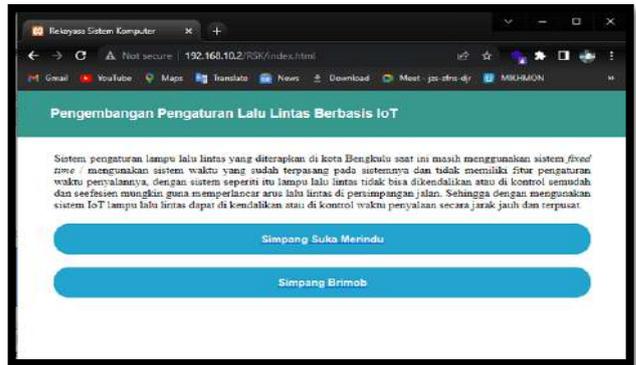
Gambar 6 Alat Pengaturan Lampu Lalu Lintas Berbasis IoT

Pada gambar 6 dapat dilihat sistem lampu lalu lintas sudah berjalan sesuai dengan rancangan yang sudah dilakukan sebelumnya. Dalam pengembangan pengaturan lampu lalu lintas berbasis IoT ini, pengujian dilakukan secara simulasi menggunakan metode prototipe yang dapat dikontrol menggunakan aplikasi berbasis web yang terhubung menggunakan jaringan komputer berbasis *point to point*.

Hasil Rancangan Web Kontrol

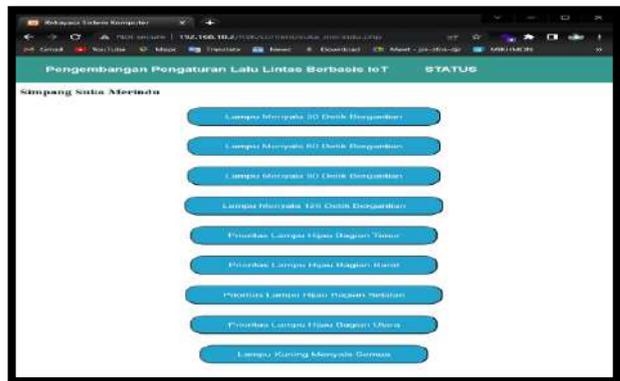
Aplkasi berbasis web kontrol dapat diakses melalui Alamat <http://192.168.10.2/rsk/>. Dimana dengan mengakses alamat ini akan ada tampilan dari aplikasi berbasis web yang dapat memilih simpang yang dapat dikontrol, sehingga untuk melakukan kontrol

terhadap lampu lalu lintas dapat dilakukan melalui kantor perhubungan Kota Bengkulu. Adapun tampilan dari aplikasi kontrol lampu lalu lintas dapat dilihat seperti berikut ini.



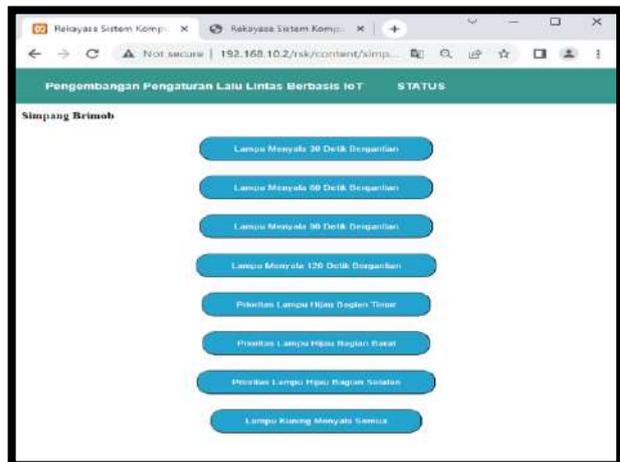
Gambar 7 Tampilan Beranda Aplikasi Web Kontrol

Pada gambar 8 diatas, dapat dilihat tampilan dari halaman index ataupun beranda dari aplikasi berbasis web. Untuk mengontrol lampu lalu lintas simpang suka merindu dapat memilih tombol simpang suka merindu, sehingga tampilan dari aplikasi kontrol lampu lalu lintas suka merindu dapat dilihat seperti gambar berikut.



Gambar 8 Tampilan Aplikasi Web Kontrol Simpang Suka Merindu

Sebaliknya untuk mengontrol lampu lalu lintas simpang brimob dapat dilakukan dengan mengklik tombol simpang brimob. Adapun tampilan dari web kontrol simpang brimob adalah sebagai berikut.



Gambar 9 Tampilan Aplikasi Web Kontrol Simpang Brimob

Dalam mengontrol lampu lalu lintas ini terdapat dua *rules* yang dijalankan yaitu *rules* yang akan berjalan setelah menyelesaikan *rules* yang sedang berjalan. *Rules* ini diterapkan pada kontrol penyalaan lampu 30 detik secara bergantian, penyalaan lampu 60 detik secara bergantian, penyalaan lampu 90 detik secara bergantian, dan penyalaan lampu 120 detik secara bergantian. Sedangkan *rules* yang kedua dapat dijalankan secara langsung mengeksekusi program yang diterapkan pada tombol prioritas lampu hijau bagian timur, prioritas lampu hijau bagian barat, prioritas lampu hijau bagian selatan, prioritas lampu hijau bagian utara, dan lampu kuning menyala semua. Selain itu dalam penelitian ini juga dilakukan pengujian terhadap waktu respon dari prototipe lampu lalu lintas dalam menerima perintah yang dilakukan oleh aplikasi berbasis web dengan mengamati waktu yang diterima oleh Arduino Mega melalui *Packet Internet Groper* (PING), saat proses pengiriman data akan ada lonjakan waktu yang dibutuhkan dalam menunggu *reply* dari Arduino Mega seperti yang terlihat pada gambar berikut

```

C:\Windows\system32\cmd.exe - ping 192.168.10.100 -t
Reply from 192.168.10.100: bytes=32 time=2ms TTL=64
Reply from 192.168.10.100: bytes=32 time=2ms TTL=64
Reply from 192.168.10.100: bytes=32 time=3ms TTL=64
Reply from 192.168.10.100: bytes=32 time=3ms TTL=64
Reply from 192.168.10.100: bytes=32 time=5ms TTL=64
Reply from 192.168.10.100: bytes=32 time=1ms TTL=64
Reply from 192.168.10.100: bytes=32 time=5ms TTL=64
Reply from 192.168.10.100: bytes=32 time=1224ms TTL=64
Reply from 192.168.10.100: bytes=32 time=8ms TTL=64
Reply from 192.168.10.100: bytes=32 time=2ms TTL=64
Reply from 192.168.10.100: bytes=32 time=4ms TTL=64
Reply from 192.168.10.100: bytes=32 time=2ms TTL=64
Reply from 192.168.10.100: bytes=32 time=2ms TTL=64
Reply from 192.168.10.100: bytes=32 time=5ms TTL=64
Reply from 192.168.10.100: bytes=32 time=4ms TTL=64
    
```

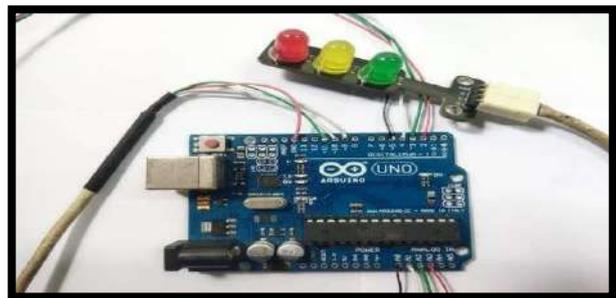
Gambar 10. Tampilan Waktu Respon Oleh Arduino Mega

Pada gambar 9 diatas dapat dilihat adanya lonjakan waktu yang dibutuhkan oleh Arduino Mega yang disebabkan oleh adanya respon dari Arduino Mega terhadap perintah yang diberikan oleh aplikasi web kontrol sebesar 1224 ms. Pengujian waktu respon ini dilakukan sebanyak lima kali pengujian pada masing tombol untuk memberikan perintah yaitu pengujian tombol lampu menyala 30 detik bergantian dengan waktu respon 1010,4 ms (*milliseconds*), tombol lampu menyala 60 detik bergantian dengan waktu respon 1324,6 ms, tombol lampu menyala 90 detik bergantian dengan waktu respon 1298 ms, tombol lampu menyala 120 detik bergantian dengan waktu respon 1190,4 ms, tombol prioritas lampu hijau bagian timur dengan waktu respon 1281,2 ms, tombol prioritas lampu hijau bagian barat dengan waktu respon 1196,2 ms, tombol prioritas lampu hijau bagian selatan dengan waktu respon 1291 ms, tombol prioritas lampu hijau bagian utara dengan waktu respon 1099,4 ms, dan tombol lampu kuning menyala semua dengan waktu respon 1085,2 ms.

B. Pembahasan Perancangan Alat

A. Rangkaian Arduino Uno

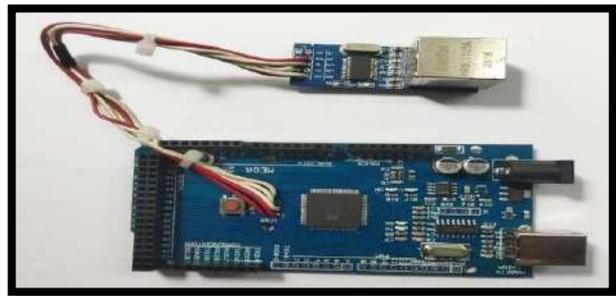
Pada satu buah lampu lalu lintas terdiri dari satu buah Arduino Uno yang berfungsi untuk mengontrol lampu lalu lintas. Apabila di suatu persimpangan terdiri dari empat ruas jalan maka diperlukannya empat buah Arduino Uno untuk mengontrol empat buah lampu lalu lintas. Dimana empat buah Arduino Uno ini saling terhubung membentuk sikul lingkaran. Rangkaian Aduino Uno menggunakan pin A0 s/d A5 sebagai pin analog *input*, pin 2 s/d 5 sebagai digital *output*, lampu led terhubung dengan Arduino di pin 9,10,11 (merah, kuning, hijau), dan setiap pin memiliki fungsi berbeda menyesuaikan kode program yang telah ditanamkan.



Gambar 11. Rangkaian Arduino Uno

B. Rangkaian Arduino Mega

Rangkaian Arduino Mega terdiri dari modul ENC28J60 yang terhubung dengan Arduino Mega pada pin ICSP (*In-Circuit Serial Programming*). Rangkaian Arduino Mega berfungsi untuk mengontrol empat buah Arduino Uno melalui pin digital analog 2 s/d 12, dan A0. Rangkaian Arduino Mega terhubung pada pin analog masing-masing Arduino Uno sesuai dengan fungsi masing-masing pin yang telah ditentukan.



Gambar 12. Rangkaian Arduino Mega

Infrastruktur Jaringan

Infrastruktur jaringan adalah alat yang menghubungkan rangkaian pengaturan lampu lalu lintas berbasis IoT dengan *server* aplikasi web kontrol agar lampu lalu lintas dapat dikontrol secara terpusat. Perangkat yang digunakan untuk menghubungkan rangkaian pengaturan lampu lalu lintas berbasis IoT dengan *server* aplikasi web adalah *wireless*. Berikut

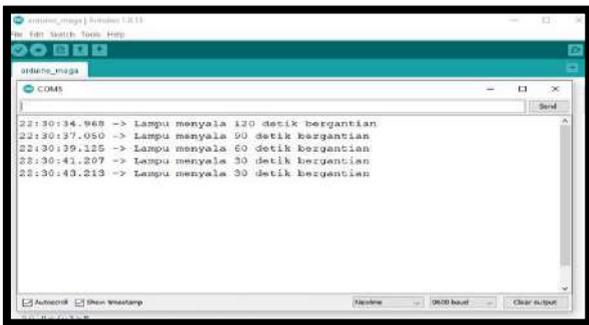
Pengujian Waktu Penyalaan Lampu Lalu Lintas

Dalam pengujian ini dilakukan skenario waktu penyalaan lampu pada salah satu simpang yang akan di kontrol penyalaan lampunya melalui aplikasi berbasis web seperti yang terlihat pada gambar 4.16 diatas. Pada aplikasi tersebut ada beberapa tombol yang dapat digunakan untuk mengontrol penyalaan lampu lalu lintas seperti lampu menyala 30 detik bergantian, lampu menyala 60 detik bergantian, lampu menyala 90 detik bergantian, lampu menyala 120 detik bergantian, prioritas lampu hijau bagian timur, prioritas lampu hijau bagian barat, prioritas lampu hijau bagian selatan, prioritas lampu hijau bagian utara, dan tombol lampu kuning menyala semua.

Dari tombol kontrol yang ada, ada dua *rules* eksekusi perintah yang akan dijalankan yaitu menunggu *rules* program yang dijalankan sebelumnya selesai dan langsung eksekusi program seperti berikut.

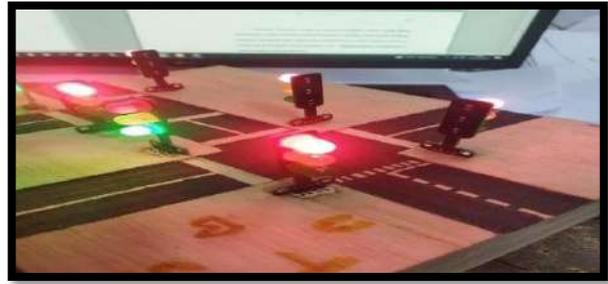
A. Menunggu *rules* program yang dijalankan sebelumnya selesai

Rules ini diterapkan pada tombol lampu menyala 30 detik bergantian, lampu menyala 60 detik bergantian, lampu menyala 90 detik bergantian, dan lampu menyala 120 detik bergantian. Proses eksekusi program yang digunakan pada *rules* ini berjalan dalam satu proses dan tidak bisa ditimpa dengan waktu yang berbeda. Dalam pengujian ini dilakukan dengan menekan tombol waktu yang diinginkan, misalnya lampu menyala 30 detik bergantian. Saat tombol lampu menyala 30 detik bergantian ditekan Arduino Mega akan menerima perintah untuk mengeksekusi penyalaan lampu selama 30 detik, begitupun juga untuk waktu – waktu sesuai dengan pemilihan tombol. Adapun hasil dari respon yang diterima oleh Arduino Mega adalah sebagai berikut.



Gambar 23. Respon Eksekusi Program Berdasarkan Waktu

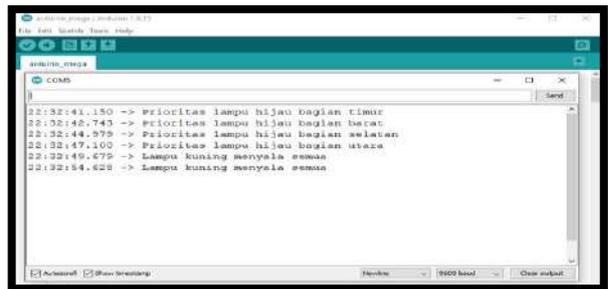
Setelah Arduino Mega menerima respon waktu yang akan dijalankan, maka eksekusi akan berjalan setelah *rules* yang sedang berjalan berhenti dan langsung menjalankan *rules* sesuai dengan waktu yang di tekan pada aplikasi web. Adapun hasil eksekusinya dapat dilihat seperti berikut.



Gambar 24. Penyalaan Lampu 30 Detik Simpang Suka Merindu

B. Langsung eksekusi program

Rules ini diterapkan pada tombol prioritas lampu hijau timur, barat, selatan, utara, dan lampu kuning menyala semua. Proses eksekusi program yang digunakan pada mode ini berjalan secara langsung tanpa harus menunggu *rules* program yang dijalankan sebelumnya selesai. Dalam pengujian ini dilakukan dengan menekan tombol lampu kuning menyala semua. Saat tombol lampu kuning menyala semua ditekan Arduino Mega akan menerima perintah untuk mengeksekusi penyalaan lampu kuning menyala semua. Adapun hasil dari respon yang diterima oleh Arduino Mega adalah sebagai berikut.



Gambar 25. Respon Eksekusi Program Langsung

Setelah Arduino Mega menerima perintah yang akan dijalankan, maka eksekusi *rules* program akan dijalankan saat itu juga. Adapun hasil eksekusinya dapat dilihat seperti berikut.

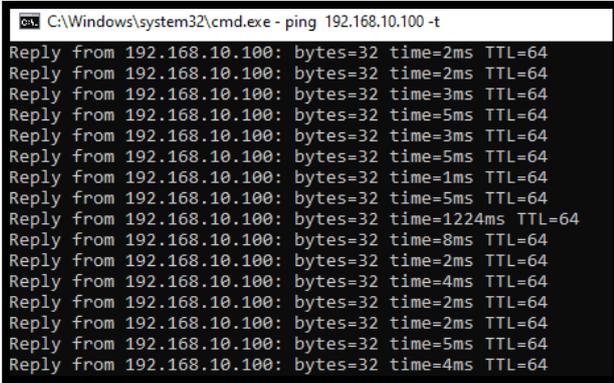


Gambar 26. Penyalaan Lampu Kuning Simpang Suka Merindu

Pengujian Waktu Respon

Pengujian waktu respon pengiriman data dari *web server* ke Arduino Mega saat memilih tombol kontrol

yang diinginkan dari aplikasi web kontrol. Dimana untuk menguji waktu respon ini dilakukan menggunakan *command prompt* (CMD) dengan menentukan *Packet Internet Groper* (PING) yang mengarah ke Arduino Mega dengan melihat lonjakan waktu yang dibutuhkan dalam merespon permintaan *reply* dari *web server*. Adapun hasil dari pengujian dapat dilihat dari gambar 4.22 dan tabel dari hasil pengujian yang telah dilakukan.



Gambar 27. Pengujian Waktu Respon Dengan CMD

Dari pengujian tersebut akan terlihat waktu yang dibutuhkan dalam melakukan reply dari permintaan oleh *web server*. Lonjakan waktu inilah yang menjadi waktu respon terhadap eksekusi program yang sedang dijalankan. Dimana dari pengujian tersebut terlihat waktu respon yang dibutuhkan sebesar 1224 ms. Pengujian ini dilakukan sebanyak lima kali pengujian dengan cara yang sama sehingga hasil dari pengujian ini dapat dilihat pada tabel berikut.

Tabel 3. Hasil Pengujian Waktu Respon

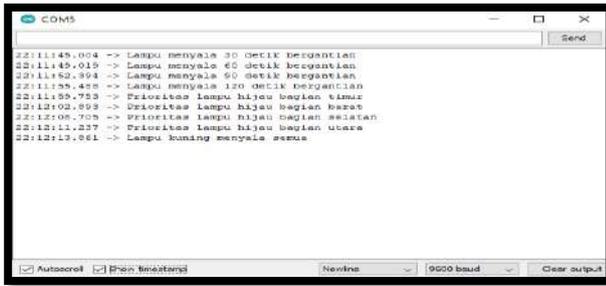
No	Kriteria Pengujian	Waktu Respon (ms)					Rata - Rata Waktu Respon (ms)
		1	2	3	4	5	
1	Lampu menyala 30 detik bergantian	12 24	14 20	1 1 0	1 5 2	1 1 0	1010,4
2	Lampu menyala 60 detik bergantian	10 07	10 33	1 3 4 1	1 7 1	1 5 3 2	1324,6
3	Lampu menyala 90 detik bergantian	14 42	94 5	1 5 4 4	1 2 3 8	1 3 2 1	1298
4	Lampu menyala 120 detik bergantian	12 76	12 65	1 2 2 2	1 1 6 3	1 0 2 6	1190,4

5	Prioritas lampu hijau bagian timur	99 2	16 75	1 3 2 5	1 1 8 7	1 2 2 7	1281,2
6	Prioritas lampu hijau bagian barat	10 90	11 33	1 0 2 7	1 4 0 5	1 3 2 6	1196,2
7	Prioritas lampu hijau bagian selatan	14 05	13 99	1 3 2 4	1 2 2 4	1 1 0 3	1291
8	Prioritas lampu hijau bagian utara	10 22	99 8	1 9 5 2	1 1 5 3	1 1 3 2	1099,4
9	Lampu kuning menyala semua	10 77	10 11	1 1 0 2	1 1 6 7	1 0 6 9	1085,2

Kesimpulan dari masing - masing kriterian pengujian nilai rata - rata waktu respon yang di dapat sebagai berikut. Pengujian tombol lampu menyala 30 detik bergantian dengan waktu respon 1010,4 ms (*milliseconds*), tombol lampu menyala 60 detik bergantian dengan waktu respon 1324,6 ms, tombol lampu menyala 90 detik bergantian dengan waktu respon 1298 ms, tombol lampu menyala 120 detik bergantian dengan waktu respon 1190,4 ms, tombol prioritas lampu hijau bagian timur dengan waktu respon 1281,2 ms, tombol prioritas lampu hijau bagian barat dengan waktu respon 1196,2 ms, tombol prioritas lampu hijau bagian selatan dengan waktu respon 1291 ms, tombol prioritas lampu hijau bagian utara dengan waktu respon 1099,4 ms, dan tombol lampu kuning menyala semua dengan waktu respon 1085,2 ms.

Pengujian Kesesuaian Status *User Interface*

Dalam pengujian ini dilakukan dengan menekan tombol waktu yang diinginkan, misalnya lampu kuning menyala semua. Saat menekan tombol kontrol penyalan lampu lalu lintas pada aplikasi web maka Arduino Mega akan menerima perintah sesuai dengan tombol yang dipilih seperti lampu menyala 30 detik bergantian, lampu menyala 60 detik bergantian, lampu menyala 90 detik bergantian, lampu menyala 120 detik bergantian, prioritas lampu hijau bagian timur, prioritas lampu hijau bagian barat, prioritas lampu hijau bagian selatan, prioritas lampu hijau bagian utara, dan lampu kuning menyala semua. Adapun hasil dari pengujian dapat dilihat pada gambar berikut.



Gambar 28 Pengujian Kesesuaian Status Pemilihan Tombol

Pengujian Status Keadaan Lampu Lalu Lintas Saat *Error*

Pengujian status keadaan lampu lalu lintas saat *error* dilakukan dengan mematikan salah satu Arduino Uno agar lampu lalu lintas mengalami keadaan *error*. Untuk mengetahui keadaan *error* operator harus menekan tombol status pada aplikasi web. Saat menekan tombol status maka akan diarahkan kehalaman baru pada aplikasi web yang memberikan informasi lokasi simpang, dan status simpang. Adapun hasil dari pengujian dapat dilihat pada gambar berikut.



Gambar 29. Pengujian Status *Error*

V. PENUTUP

A. Kesimpulan

Berdasarkan hasil perancangan dan analisa pengujian pengembangan pengaturan lampu lalu lintas berbasis IoT dapat di ambil kesimpulan sebagai berikut:

1. Lampu lalu lintas dapat dikontrol secara terpusat menggunakan aplikasi berbasis web.
2. Waktu respon dari Arduino Mega hanya membutuhkan hitungan detik sesuai dengan pengujian yang dilakukan yaitu tombol lampu menyala 30 detik bergantian dengan waktu respon 1010,4 ms (*milliseconds*), tombol lampu menyala 60 detik bergantian dengan waktu respon 1324,6 ms, tombol lampu menyala 90 detik bergantian dengan waktu respon 1298 ms, tombol lampu menyala 120 detik bergantian dengan waktu respon 1190,4 ms, tombol prioritas lampu hijau bagian timur dengan waktu respon 1281,2 ms, tombol prioritas lampu hijau bagian barat dengan waktu respon 1196,2 ms, tombol prioritas lampu hijau bagian selatan dengan waktu respon 1291

ms, tombol prioritas lampu hijau bagian utara dengan waktu respon 1099,4 ms, dan tombol lampu kuning menyala semua dengan waktu respon 1085,2 ms..

3. Apabila terdapat *error* pada lampu lalu lintas, maka dapat dilihat melalui aplikasi web kontrol pada tombol status.

A.Saran

Berdasarkan hasil pengujian dan hasil analisa yang telah penulis lakukan maka penulis memiliki saran agar alat ini dapat di kembangkan lagi dengan sistem lampu lalu lintas menggunakan web *service* dan memiliki teknologi yang lebih canggih.

DAFTAR PUSTAKA

- [1] Alamsyah, 2019. *Penerapan IoT Untuk Kontrol Lampu Menggunakan Arduino Berbasis Web*. Malang : Universitas Muhammadiyah Malang, 60 halaman.
- [2] Ardi, 2019. *Rancang Bangun Robot Avoider*. Surakarta : Universitas Muhammadiyah Surakarta, 55 halaman.
- [3] Arduino, 2023. *Pengertian Arduino*. Diakses pada tanggal 26 September 2023 dari <https://www.arduino.cc/en/Guide/Introduction>.
- [4] Chandra, 2020, *Sistem Pengontrol Lampu Berbasis Arduino Uno dan ENC28J60*. Jawa Timur : Institut Teknologi Sepuluh Nopember, 56 halaman.
- [5] Eka, 2020. *Sistem Pendeteksi Ketinggian Dron*. Depok: UI, 40 halaman.
- [6] Eko, 2021. *Sistem Perancangan Alat Ukur Warna Menggunakan Sensor LDR Berbasis Arduino Mega 2560*. Jawa Timur : Universitas AirLangga 40 halaman.
- [7] Fahrul, 2018. *Perencanaan Pembelajaran*. Bandung : Remaja Rosdakarya, 64 Halaman.
- [8] Fajar, 2019. *Model Prototyping Pada Pengembangan Sistem Informasi*. Jakarta : Universitas BSI, 29 Halaman.
- [9] Fauzi, 2019. *Sistem Kontrol Kompor Listrik Otomatis Berbasis Mikrokontroler Arduino Uno*. Batam : Universitas Politeknik Negeri Batam, 41 halaman.
- [10] Fikri, 2018. *Prototipe Aplikasi Info Rekening Listrik PLN Berbasis Mobile*. Jakarta : Universitas Islam Negeri Syarif Hidayatullah, 57 halaman.
- [11] Hasfar & Adiwarsa, 2018. *Perancangan Lampu Lalu lintas*. Yogyakarta : Universitas Gajah Mada, 57 halaman.
- [12] Hamdani, 2018. *Pengembangan Sistem Pendidikan di Indonesia*. Bandung : Pustaka Setia, 125 halaman.

- [13] Herdian, 2019. *Analisa Perhitungan Lampu Lalu Lintas Pada Persimpangan*. Sumatera Utara : Universitas Medan Area, 74 halaman. Kurniawan, 2019. *Perancangan Dan Implementasi Load Balancing Web Server Menggunakan HAProxy*. Riau : Universitas Islam Riau, 67 halaman.
- [14] Kusuma, 2017. *Analisa Kualitas Layanan Jaringan Internet Berbasis Wireless*. Puewokerto : Institut Teknologi Telkom Purwokerto, 64 halaman.
- [15] Mayasari, 2018. *Analisa Efektivitas Lampu Lalu lintas di Kota Surakarta*. Surakarta : Universitas Sebelas Maret Surakarta, 49 halaman.
- [16] Pratama, 2018. *Aplikasi Jam NTP pada Mini Web Server untuk Kendali Jam Digital*. Jawa Timur : Universitas Politeknik Kota Malang, STIKMA Vol.07 No.1, 11 halaman.
- [17] Purba, Sulistyorini, Sadnowo dan, Ilhami, 2017. *Pengembangan Sistem Monitoring Lampu Lalu Lintas Berbasis Microcontroller Dengan Sms Jaringan Gsm*. Lampung : Universitas Lampung, Seminar Nasional AVoER IX 2017, 50 halaman.
- [18] Riski, 2016. *Sistem Pendeteksi Kekurangan Pada Air Sungai*. Surakarta : Universitas Muhammadiyah Surakarta, 53 halaman.
- [19] Saputro, 2018. *Sistem Kendali ARC Plasma Sintering (APS) Berbasis Mikrokontroler Arduino R3*. Jakarta : Bina Nusantara University, 44 halaman.
- [20] Sukrisno, 2019. *Pengembangan Aplikasi Manajemen Tugas Skripsi*. Madura : Universitas Trunojoyo Madura, 44 halaman.
- [21] Sundari, 2018. *Analisis Keamanan Jaringan Komputer pada Fasilitas wireless fidelity (wifi) Terhadap Serangan Paket sniffing*. Sulawesi Selatan : Universitas Cokroaminoto Palopo, 70 halaman.
- [22] Wikipedia, 23 Agustus 2023. *Pengertian Arduino*. Diakses pada tanggal 26 September 2023 dari <https://id.wikipedia.org/wiki/Arduino>.
- [23] Wikipedia, 23 Juli 2023. *Pengertian Arduino Uno*. Diakses pada tanggal 26 September 2023 dari <https://id.wikipedia.org/wiki/ArduinoUno>.
- [24] Wikipedia, 03 Juli 2023. *Pengertian Pemrograman*. Diakses pada tanggal 26 September 2023 dari <https://id.wikipedia.org/wiki/Pemrograman>.