

2018

Perancangan dan Pembuatan Sistem Kendali Lampu Melalui Android dan Wifi Berbasis Mikrokontroler Arduino Atmega328

Sibagariang, Morris Egmon

Universitas Sumatera Utara

<http://repositori.usu.ac.id/handle/123456789/10491>

Downloaded from Repositori Institusi USU, Universitas Sumatera Utara

**PERANCANGAN DAN PEMBUATAN SISTEM KENDALI
LAMPU MELALUI ANDROID DAN WIFI BERBASIS
MIKROKONTROLER ARDUINO ATMEGA328**

SKRIPSI

**MORRIS EGMON SIBAGARIANG
131421103**



**PROGRAM STUDI S1 ILMU KOMPUTER
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI
INFORMASI UNIVERSITAS SUMATERA UTARA
MEDAN**

PERSETUJUAN

Judul : PERANCANGAN DAN PEMBUATAN SISTEM
KENDALI LAMPU MELALUI ANDROID DAN
WIFI BERBASIS MIKROKONTROLER ARDUINO
ATMEGA328

Kategori : SKRIPSI

Nama : MORRIS EGMON SIBAGARIANG

Nomor Induk Mahasiswa : 131421103

Program Studi : SARJANA (S-1) ILMU KOMPUTER

Departemen : ILMU KOMPUTER

Fakultas : FAKULTAS ILMU KOMPUTER DAN
TEKNOLOGI INFORMASI UNIVERSITAS
SUMATERA UTARA

Diluluskan di
Medan, Agustus 2018

Komisi Pembimbing :
Pembimbing 2



Drs. Dahlan Sitompul, M.Eng
NIP. 196707252005011002
Diketahui/disetujui oleh
Program Studi S-1 Ilmu Komputer
Ketua,



Dr. Poltak Sihombing, M.Kom.
NIP. 196203171991031001

Pembimbing 1



Dr. Poltak Sihombing, M.Kom.
NIP. 196203171991031001

PERNYATAAN

PERANCANGAN DAN PEMBUATAN SISTEM KENDALI LAMPU
MELALUI ANDROID DAN WIFI BERBASIS MIKROKONTROLER
ARDUINO ATMEGA328

SKRIPSI

Saya mengakui bahwa skripsi ini adalah hasil karya saya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing telah disebutkan sumbernya.

Medan, Juli 2018

Morris Egmon Sibagariang
131421103

UCAPAN TERIMA KASIH

Puji dan syukur penulis ucapkan kepada Tuhan Yang Maha Esa, karena rahmat dan izin-Nya penulis dapat menyelesaikan skripsi ini. Banyak bantuan berupa uluran tangan, budi baik, buah pikiran dan kerjasama yang telah penulis terima selama menempuh studi sampai dengan penyelesaian studi (skripsi) ini. Oleh karena itu, penulis menyampaikan ucapan terima kasih kepada pihak-pihak yang telah membantu.

Ucapan terima kasih penulis sampaikan kepada:

1. Bapak Prof. Dr. Runtung, SH, M.Hum selaku Rektor Universitas Sumatera Utara.
2. Bapak Prof. Dr. Opim Salim Sitompul selaku Dekan Fasilkom-TI USU.
3. Bapak Dr. Poltak Sihombing, M.Kom. selaku Ketua Program Studi S-1 Ilmu Komputer Universitas Sumatera Utara dan Dosen Pembimbing Iyang telah memberikan bimbingan dan dukungan kepada penulis.
4. Bapak Drs.Dahlan Sitompul ,M.Eng selaku Dosen Pembimbing II yang telah memberikan bimbingan dan dukungan kepada penulis.
5. Yang teristimewa, Ibunda Anny N Nainggolan beserta dan kakak Ruth Lamtiur Sibagariang dan Desi Arisandi Sibagariang yang selalu memberikan kasih sayang dan dukungan kepada penulis.
6. Seluruh tenaga pengajar dan pegawai di Fakultas Ilmu Komputer dan Teknologi Informasi USU, terkhususnya di Program Studi S-1 Ilmu Komputer.
7. Semua pihak yang terlibat langsung atau tidak langsung yang penulis tidak dapat tuliskan satu per satu.

Semoga Tuhan melimpahkan berkah dan kasih sayang-Nya kepada semua pihak yang telah memberikan bantuan, semangat, dukungan dan perhatian kepada penulis dalam menyelesaikan skripsi ini. Semoga skripsi ini bermanfaat bagi penulis, pendidikan, masyarakat, organisasi atau negara.

Medan, Juli 2018

Penulis,

Morris Egmon Sibagariang

PERANCANGAN DAN PEMBUATAN SISTEM KENDALI LAMPU MELALUI ANDROID DAN WIFI BERBASIS MIKROKONTROLER ARDUINO ATMEGA328

ABSTRAK

Internet of Things semakin banyak bermunculan, sehingga membuat Perangkat IoT yang murah meriah akan terus bermunculan. Berbagai layanan platform IoT pun semakin berkembang dan saling berkompetisi. Persaingan yang semakin meluas ini berdampak pada membludaknya jumlah penggiat IoT untuk membuat berbagai kreasi. Pada skripsi ini, Saya akan menunjukkan satu kasus implementasi IoT membuat perangkat yang dapat menyalakan dan mematikan lampu melalui komputer atau handphone yang terhubung ke wifi. ESP8266 merupakan modul wifi yang berfungsi sebagai perangkat tambahan mikrokontroler seperti Arduino agar dapat terhubung langsung dengan wifi dan membuat koneksi TCP/IP. Modul ini adalah salah satu modul favorit dalam smart home. Sistem kendali pada alat yang akan dibangun menggunakan aplikasi yang akan dibuat bersamaan dengan alat yang akan dibuat pada smartphone android, dimana proses pengkoneksian antara wifi dan ATMEGA 328 beserta smartphone dilakukan oleh ESP8266. Sehingga kita bisa menghidupkan dan mematikan lampu menggunakan smartphone.

Kata kunci : Arduino-Uno, *ESP8266*, *IOT*, Relay, Wifi

**DESIGN AND DEVELOPMENT OF LIGHT CONTROL SYSTEM
BASED ON ANDROID AND ARDUINO MICROCONTROLLER
ATMEGA328**

ABSTRACT

Internet of Things is increasingly emerging, thus making the cheap IoT Devices will continue to popping out. Various IoT platform services are growing and competing. This increasingly widespread competition has an impact on the increasing number of IOT activists to create various creations. In this thesis, I will show you one case of IoT implementation to create a device that can turn on and off the lights through a computer or mobile phone connected to wifi. ESP8266 is a wifi module that serves as a microcontroller enhancement such as Arduino to connect directly to wifi and create TCP / IP connections. This module is one of the favorite modules in smart home. The control system on the tool to be built using the application will be created simultaneously with the tool to be created on android smartphone, where the process of connecting between wifi and ATMEGA 328 along with smartphones is done by ESP8266. So we can turn on and off the lights using a smartphone.

Keywords : Arduino-Uno, ESP8266, IOT, Relay, Wifi

DAFTAR ISI

PENGESAHAN SKIRPSI.....	i
ABSTRAK.....	ii
ABSTRACT.....	iii
DAFTAR ISI.....	iv
DAFTAR TABEL.....	iv
DAFTAR GAMBAR.....	vi
 BAB 1 PENDAHULUAN.....	 1
1.1 Latar Belakang.....	1
1.2 Perumusan masalah.....	2
1.3 Batasan masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat penelitian.....	3
 BAB 2 IDENTIFIKASI ALAT SISTEM KENDALI LAMPU BERBASIS ANDROID	
2.1 Mikrokontroler.....	5
2.2 Mikrokontroler Board Arduino UNO.....	6
2.3 Modul WIFI ESP8266.....	9
2.4 Relay.....	11
2.5 Android.....	13
2.6 Router.....	16
 BAB 3 METODOLOGI PERANCANGAN SISTEM KENDALI LAMPU MELALUI ANDROID DAN WIFI.....	 19
3.1 Alur Kerja Penelitian.....	19
3.2 Perancangan Sistem.....	20
3.2.1 Skema Diagram Rancangan Kontroler Lampu.....	20
3.2.2 Diagram Flow Sistem.....	21
3.3 Flashing Firmware ESP 8266.....	22
3.4 Rangkaian Kontroler Lampu.....	23
3.5 Perancangan Antar-Muka Sistem Android.....	24
 BAB 4 PERANCANGAN SISTEM KENDALI LAMPU MELALUI ANDROID DAN WIFI.....	 26
4.1 Rancangan Sistem Kendali Lampu Melalui Android dan Wifi.....	26
4.1.1 Aplikasi Control Relay Pada Android.....	26
4.1.2 Sistem Kendali Lampu Melalui Android dan wifi....	26
4.2 Pemograman Sistem Kendali Lampu Melalui Android dan Wifi.....	34
4.2.1 Pemograman Arduino Uno.....	35
4.2.2 Pemograman Android dan Hasil.....	37

BAB 5 Kesimpulan dan Saran.....	47
5.1 Kesimpulan.....	47
5.2 Saran.....	47

DAFTAR PUSTAKA
LAMPIRAN

DAFTAR GAMBAR

NOMOR GAMBAR	JUDUL	HALAMAN
2.1	Mikroprosesor	5
2.2	Board Arduino UNO.....	7
2.3	Kabel USB	7
2.4	Atmega 328	8
2.5	Tampilan IDE	10
2.6	Modul ESP8266	11
2.7	Pengklasifikasian AT command	11
2.8	Gambar Dan Simbol Relay	13
2.9	Struktur Sederhana dan Relay	14
2.10	Arsitektur Android	15
2.11	Router	17
2.12	Ilustrasi Cara Kerja Router	18
3.1	Diagaram Alur Kerja Penelitian.....	19
3.2	Skema Diagram Rancangan Kontroler Lampu	20
3.3	Diagram Flow Sistem.....	21
3.4	Susunan Rangkaian Flashing Firmware.....	22
3.5	Susunan Rangkaian Kontroler Lampu	23
3.6	Rancangan Aplikasi Android	24
3.7	Code Block Aplikasi Android.....	25
4.1	Aplikasi Relay pada Android	26
4.2	Susunan Rangkaian Flahing Firmware Esp8266.....	27
4.3	Upload Sketch Pada board Arduino	28
4.4	Peroses flashing firmware Esp8266	28
4.5	Rangkaian Arduino Relay dan ESP8266	29
4.6	Hasil Perancangan Hardware.....	30
4.7	Library yang Digunakan.....	34
4.8	Kode Program untuk Menghidupkan Sakelar.....	34
4.9	Kode Pemograman Untuk Menampilkan Status koneksi.....	35
4.10	Kode untuk Mencek jika Esp8266 Mengirm pesan.....	36
4.11	Pengujian pada Software Arduino 1.8.5.....	36
4.12	Tampilan App pada smartphone.....	37
4.13	Tampilan awal aplikasi	38
4.14	Pemograman android pada tampilan awal.....	39
4.15	Tampilan Software dan Hardware ketika saklar 1 ditekan.....	40
4.16	Pemograman pada Sakelar 1.....	40
4.17	ketika Sakelar Dua Ditekan maka Modem diisi Bateriainya.....	41

4.18	Pemograman pada Sakelar 2.....	42
4.19	ketika Sakelar Tiga Ditekan Maka Lampu Hidup.....	43
4.20	Pemograman pada Sakelar 3.....	43
4.21	Ketika Sakelar Keempat Ditekan Maka Lampu Dua Hidup.....	44
4.22	Pemograman pada sakelar 4.....	45
4.23	Pemograman Ketika Hendak MengClose Aplikasi.....	46

DAFTAR TABEL

NOMOR TABEL	JUDUL	HALAMAN
4.1	Susunan Rangkaian Flashing Firmware ESP8266	27
4.2	Susunan Rangkaian Arduino	29

BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Di era perkembangan teknologi analog, pada umumnya perangkat-perangkat listrik dikendalikan secara konvensional oleh pengguna. Seseorang harus menghidupkan dan mematikan sakelar secara langsung yang terhubung ke perangkat listrik tersebut. Terkadang, ada beberapa perangkat listrik yang dijumpai masih hidup ketika tidak digunakan, hal ini dapat disebabkan oleh kelalaian pengguna untuk mematikan perangkat listrik tersebut. Jika jumlah perangkat listrik yang berada di dalam suatu rumah cukup banyak, maka akan sangat tidak efektif dan tidak nyaman untuk mematikan dan menghidupkan perangkat-perangkat listrik tersebut secara konvensional. Penggunaan energi listrik dari perangkat-perangkat tersebut juga akan tidak efisien (Husada & Iqbal, 2013).

Perkembangan teknologi digital yang pesat ikut mendorong perkembangan teknologi komputer. Sekarang ini, banyak perangkat-perangkat listrik yang bekerja secara terintegrasi dengan sistem komputer. Hal ini tentunya akan sangat membantu pekerjaan manusia dalam mengoperasikan perangkat listrik tersebut. Salah satu penelitian yang sedang berkembang sekarang ini adalah mengenai Smart Home. Perangkat Smart Home adalah sebuah perangkat yang memiliki sistem otomatisasi sangat canggih untuk mengendalikan lampu dan suhu, perangkat multi media untuk memantau dan menghidupkan sistem keamanan yang terhubung dengan pintu atau jendela dan beberapa fungsi yang lainnya (Yusron, 2014).

Smart Home memiliki beberapa manfaat seperti memberikan kenyamanan yang lebih baik, keselamatan dan keamanan yang lebih terjamin, dan menghemat penggunaan energi listrik (Husada & Iqbal, 2013). Dengan menerapkan perangkat Smart Home di rumah atau perkantoran, perangkat-perangkat listrik akan dapat bekerja secara otomatis sesuai dengan kebutuhan pengguna.

Pengguna juga dapat memantau dan mengendalikan perangkat- perangkat listrik di dalam rumah dari jarak jauh melalui suatu saluran komunikasi seperti melalui jaringan internet, Wi-Fi atau bluetooth.

Mikrokontroler adalah suatu perangkat yang digunakan untuk mengontrol suatu proses atau aspek-aspek dari lingkungan. Mikrokontroler biasanya digunakan untuk peralatan yang tidak terlalu membutuhkan kecepatan pemrosesan yang tinggi. Sifat spesial dari mikrokontroler adalah kecil dalam ukuran, hemat daya listrik serta fleksibilitasnya menyebabkan mikrokontroler sangat cocok untuk dipakai sebagai pencatat/perekam data pada aplikasi yang tidak memerlukan kehadiran operator (Purnomo, 2015).

Arduino Uno adalah papan sirkuit berbasis mikrokontroler ATmega328. IC (integrated circuit) ini memiliki 14 input/output digital (6 output untuk PWM), 6 analog input, resonator kristal keramik 16 MHz, Koneksi USB, soket adaptor, pin header ICSP, dan tombol reset. Arduino Uno sendiri menggunakan chip mikrokontroler ATmega 328 sebagai pusat pengendaliannya Chip ini dipasangkan ke header socket yang ada pada board Arduino Uno sehingga memungkinkan untuk dilepas dan dipasang ulang. Kelebihannya adalah apabila terjadi kerusakan pada chip yang diakibatkan oleh kegagalan ataupun 'cedera fisik' (jatuh atau terbanting) maka anda dapat melakukan penggantian chip yang rusak tersebut dengan chip yang baru. Jadi anda tidak perlu mengganti board Arduino Uno anda secara keseluruhan sehingga dapat menghemat biaya kerusakan.

Koneksi Wifi ESP8266 yang bersifat bersifat SOC (*System on Chip*), sehingga kita bisa melakukan programming langsung ke ESP8266 tanpa memerlukan mikrokontroller tambahan. Kelebihan lainnya, ESP8266 ini dapat menjalankan peran sebagai adhoc akses poin maupun klien sekaligus. (Sinuardino, 2016). Dengan memanfaatkan kelebihan Wifi ESP8266 memungkinkan untuk menghubungkan Arduino Uno terhadap perangkat android.

Dari penjelasan yang telah diuraikan tersebut, maka penulis berencana mengadakan penelitian tugas akhir dengan judul “PERANCANGAN DAN PEMBUATAN SISTEM KENDALAI LAMPU MELALUI PLATFORM

ANDROID DAN WIFI BERBASIS MIKROKONTROLER ARDUINO ATMEGA328”

1.2. Perumusan Masalah

Dalam era perkembangan teknologi analog, pada umumnya perangkat-perangkat listrik dikendalikan secara konvensional oleh pengguna. Sekarang ini, banyak perangkat- perangkat listrik yang bekerja secara terintegrasi dengan sistem komputer. Hal ini tentunya akan sangat membantu pekerjaan manusia dalam mengoperasikan perangkat listrik tersebut. Salah satu penelitian yang sedang berkembang sekarang ini adalah mengenai *Smarthome*. *Smarthome* memiliki beberapa manfaat seperti memberikan kenyamanan yang lebih baik, keselamatan dan keamanan yang lebih terjamin, dan menghemat penggunaan energi listrik. Untuk itu perlu dilakukan penelitian perancangan dan pembuatan sistem kendali lampu melalui platform android dan wifi berbasis mikrokontroler arduino atmega328.

1.3. Batasan Masalah

Karena keterbatasan waktu, sumber daya, dana dan kemampuan keilmuan penulis, maka dalam penelitian ini hanya akan dilakukan hal-hal sebagai berikut:

- a. Sistem hanya membahas tentang sistem kendali on/off lampu menggunakan smartphone android dengan memanfaatkan fasilitas wifi ESP8266.
- b. Program mikrokontroler arduino dibuat dengan bahasa pemrograman C Arduino IDE dan aplikasi android dibuat dengan software tool visual MIT App Inventor.

1.4. Tujuan Penelitian

Penelitian ini bertujuan untuk menghasilkan suatu aplikasi yang dapat mengendalikan lampu penerangan pada rumah dengan menggunakan smartphone android melalui koneksi *wifi* yang saling terhubung dalam jangkauan koneksi *wifi*.

1.5. Manfaat Penelitian

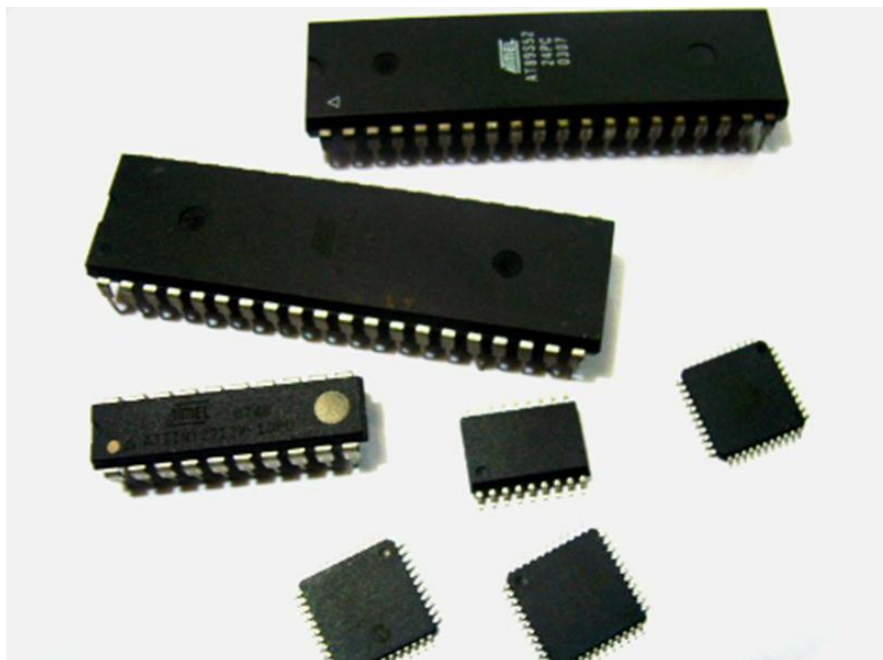
Manfaat dalam penelitian ini adalah untuk memperoleh cara baru dalam mengendalikan perangkat lampu yang terkomputerisasi.

BAB 2

TINJAUAN PUSTAKA

2.1 Mikrokontroler

Mikrokontroler adalah Suatu kontroler digunakan untuk mengontrol suatu proses atau aspek-aspek dari lingkungan. Satu contoh aplikasi dari mikrokontroler adalah untuk memonitor rumah kita. Ketika suhu naik kontroler membuka jendela dan sebaliknya. Pada masanya, kontroler dibangun dari komponen-komponen logika secara keseluruhan, sehingga menjadikannya besar dan berat. Setelah itu barulah dipergunakan mikroprosesor sehingga keseluruhan kontroler masuk kedalam PCB yang cukup kecil. Hingga saat ini masih sering kita lihat kontroler yang dikendalikan oleh mikroprosesor. Proses pengecilan komponen terus berlangsung, semua komponen yang diperlukan guna membangun suatu kontroler yang dapat dikemas dalam satu keping. Maka lahirlah komputer keping tunggal (one chip microcomputer) atau disebut juga mikrokontroler dapat dilihat pada Gambar 2.1.



Gambar 2.1. Mikroprosesor (Satria, 2013).

Mikrokontroler adalah suatu dimana semua bagian yang diperlukan untuk suatu kontroler sudah dikemas dalam satu keping, biasanya terdiri dari:

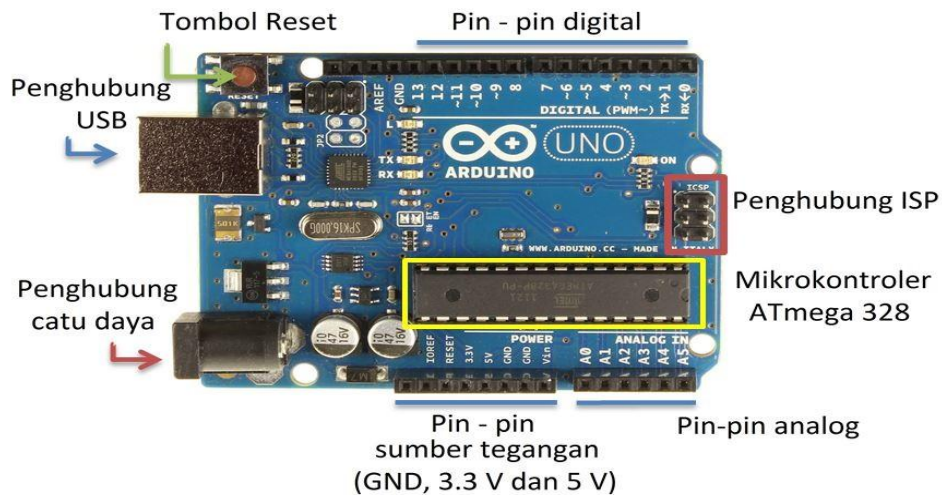
1. CPU (Central Processing Unit)
2. RAM (Random Access Memory)
3. EEPROM/EPROM/PROM/ROM
4. I/O, Serial & Parallel
5. Timer
6. Interrupt Controller

rata-rata mikrokontroler memiliki instruksi manipulasi bit, akses ke I/O secara langsung dan mudah, dan proses interrupt yang cepat dan efisien. Dengan kata lain mikrokontroler adalah " Solusi satu Chip" yang secara drastis mengurangi jumlah komponen dan biaya disain (harga relatif rendah).

Selain sebagai sistem monitor rumah seperti diatas, mikrokontroler sering dijumpai pada peralatan rumah tangga (microwave oven, TV, stereo set dll), komputer dan perlengkapannya, mobil dan lain sebagainya. Pada beberapa penggunaan bisa ditemukan lebih dari satu prosesor didalamnya. Mikrokontroler biasanya digunakan untuk peralatan yang tidak terlalu membutuhkan kecepatan pemrosesan yang tinggi. Sifat spesial dari mikrokontroler adalah kecil dalam ukuran, hemat daya listrik serta fleksibilitasnya menyebabkan mikrokontroler sangat cocok untuk dipakai sebagai pencatat/perekam data pada aplikasi yang tidak memerlukan kehadiran operator (Purnomo, 2015).

2.2 Mikrokontroler *Board* Arduino UNO

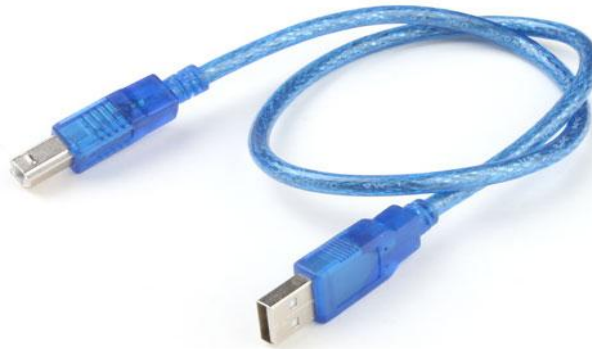
Arduino uno sebenarnya adalah salah satu kit mikrokontroler yang berbasis pada ATmega28. Modul ini sudah dilengkapi dengan berbagai hal yang dibutuhkan untuk mendukung mikrokontroler untuk bekerja, tinggal colokkan ke power suply atau sambungkan melalui kabel USB ke PC. board Arduino Uno ini sudah siap digunakan (Santoso, 2015).



Gambar 2.2. Board Arduino UNO (Purbakawaca, 2015).

Kabel berwarna biru Gambar 2.3 adalah konektor USB tipe A ke tipe B. Tipe B biasanya adalah USB yang dipakai pada printer inkjet. Board pada Gambar 2.2 mempunyai sebuah port USB female, sebuah colokan female DC power jack, penghubung ISP, pin analog, pin digital dan tombol reset.

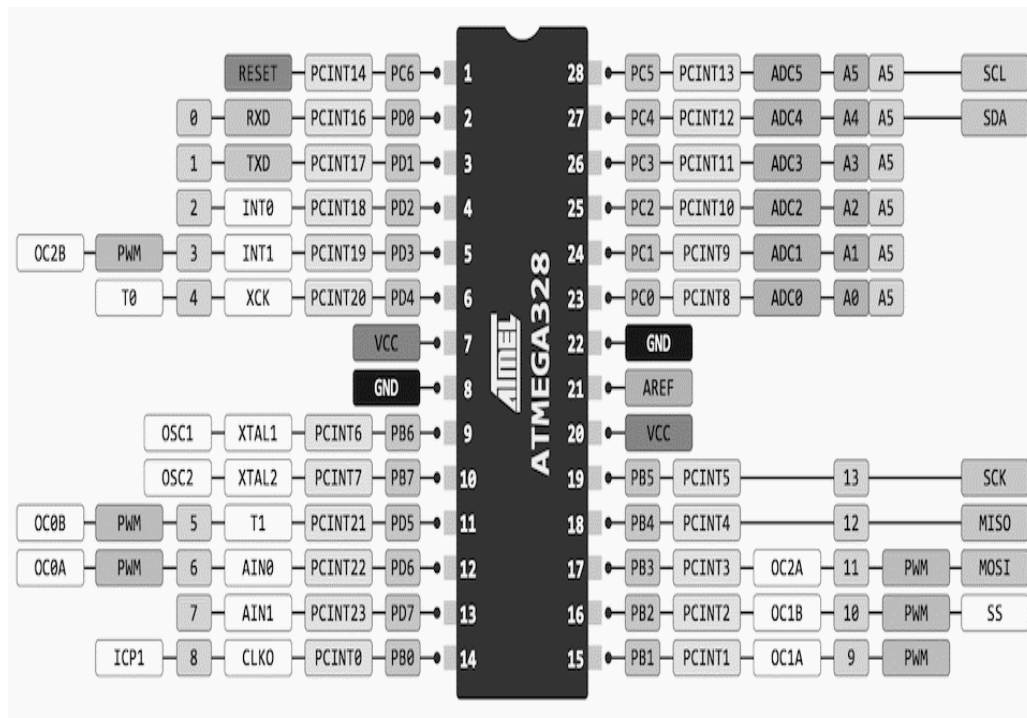
Port USB ini digunakan untuk koneksi data ke PC, DC power jack untuk catu daya eksternal. Meski bisa bekerja dari tegangan 3V – 20V, tapi disarankan untuk memakai tegangan antara 5V – 12V saja, karena jika catu kurang dari 5V maka output high pada digital output tidak akan bernilai “1” secara penuh, dan jika catu melebihi 12V maka output low akan lebih dari 0V sehingga tidak akan bernilai “0” sempurna (Seno, 2015).



Gambar 2.3. Kabel USB (Seno, 2015).

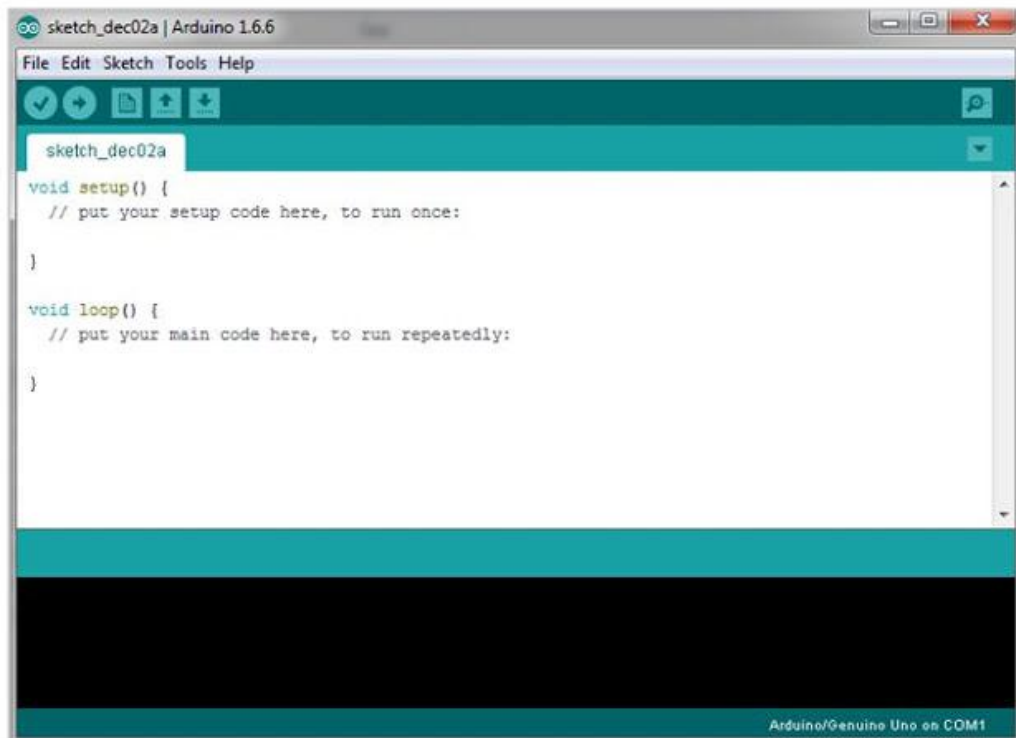
Tombol reset pada Gambar 2.2 adalah tombol yang difungsikan untuk mengatur ulang (reset) Arduino ke program terakhir apabila terjadi 'keanehan' pada eksekusi yang dilakukan. Pin analog digunakan untuk menerima keluaran sensor analog. Pada pin-pin ini terdapat ADC (Analog Digital Converter) yang berfungsi mengubah nilai analog menjadi digital. Pin analog ini diperlukan karena proses yang terjadi dalam Arduino adalah proses digital. Pin digital digunakan untuk menerima keluaran dari sensor digital. Pada pin digital terdapat dua kondisi yaitu 1 (high) dan 0 (low). (Purbakawaca, 2015)

Arduino Uno sendiri menggunakan chip mikrokontroler ATmega 328 sebagai 'otak' pusat pengendaliannya. Chip ini dipasangkan ke header socket yang ada pada board Arduino Uno sehingga memungkinkan untuk dilepas dan dipasang ulang. Kelebihannya adalah apabila terjadi kerusakan pada chip yang diakibatkan oleh kegagalan ataupun 'cedera fisik' (jatuh atau terbanting) maka anda dapat melakukan penggantian chip yang rusak tersebut dengan chip yang baru. Jadi anda tidak perlu mengganti board Arduino Uno anda secara keseluruhan sehingga dapat menghemat biaya kerusakan. Anda dapat melihat ilustrasi chip ATmega 328 dan konfigurasinya pada Gambar 2.4.



Gambar 2.4. ATmega 328 (Purbakawaca, 2015).

Pemrograman arduino dapat dilakukan dengan *Arduino softwareIntegrated Development Environment*(IDE).IDE berfungsi untuk menulis dan mengunggah kode dari computer ke Arduino.Adapun bahasa pemrograman yang dipakai oleh arduino IDE ini adalah bahasa pemrograman *C* dan *C++*. Berikut tampilan *software* IDE dari arduino dapat dilihat pada Gambar 2.5.

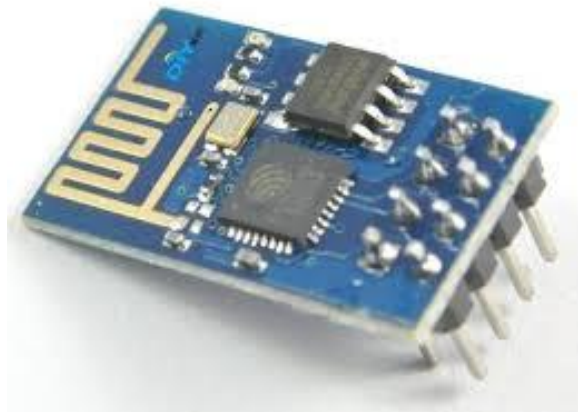


Gambar 2.5. Tampilan IDE (Purbakawaca, 2015)

Arduino IDE digunakan sebagai tempat membuat perintah (program) atau yang dikenal dengan istilah *source code*, melakukan pengecekan kesalahan, kompilasi, upload program ke Arduino, dan menguji hasil kerja Arduino melalui monitor Komputer.

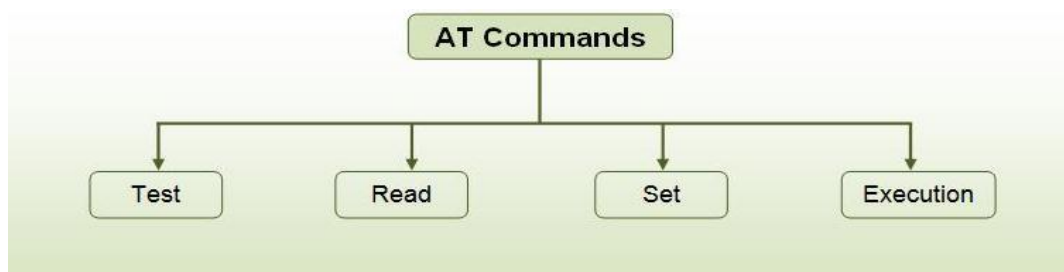
2.3 Modul WIFI ESP8266

ESP8266 adalah sebuah modul WiFi yang akhir-akhir ini semakin digemari para hardware developer dapat dilihat pada Gambar 2.6. Selain karena harganya yang sangat terjangkau, modul WiFi serbaguna ini sudah bersifat SOC (*System on Chip*), sehingga kita bisa melakukan programming langsung ke ESP8266 tanpa memerlukan mikrokontroller tambahan. Kelebihan lainnya, ESP8266 ini dapat menjalankan peran sebagai adhoc akses poin maupun klien sekaligus. (Sinuardino, 2016)



Gambar 2.6. Modul ESP8266 (Prasetyo, 2015)

ESP8266 dikembangkan oleh pengembang asal china yang bernama “espressif”. Pada ESP8266 sendiri sudah memiliki GPIO (*General Purpose Input Output*), yang artinya ESP8266 ini bisa melakukan fungsi input atau output. Salah satu kelebihan ESP8266 adalah memiliki *DEEP SLEEP MODE*, sehingga penggunaan akan lebih efisien dalam hal sumber daya. ESP-01 memiliki tiga jenis mode operasi yang perlu diketahui, yaitu *Station*, *Access Point* dan gabungan mode keduanya. jika yang dipilih adalah mode AP (*Access Point*), berarti ESP-01 difungsikan sebagai akses point wifi (memiliki SSID sendiri), sehingga perangkat lain bisa terhubung dengan ESP-01. Mode ini mirip dengan wifi tethering yang dimiliki oleh smartphone. Namun, jika mode STA yang dipilih, ESP-01 dapat terhubung dengan jaringan wifi yang tersedia oleh akses poin dari router, ataupun modem Mifi (seperti yang disediakan oleh provider bolt atau andromax, misalnya). Sehingga ESP-01 otomatis terhubung dengan jaringan internet, tentu jika ada internet aktif di jaringan tersebut. Sedangkan mode yang ketiga adalah gabungan dari AP dan STA. Konfigurasi ESP8266 sebagai *client* dan *access point* dilakukan dalam mode *AT Command*. Pengklasifikasian AT Command dapat dilihat pada Gambar 2.7



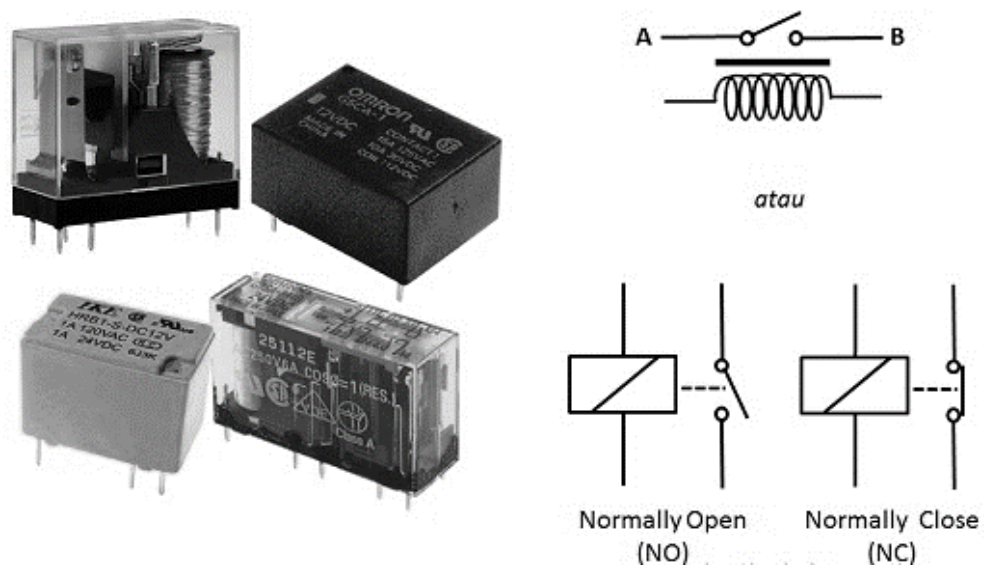
Gambar 2.7. Pengklasifikasian AT command (Prasetyo, 2015)

AT Command digunakan untuk berkomunikasi dengan terminal melalui port pada Komputer, dan penggunaan *AT command* pada ESP8266 dapat memberi kemudahan untuk mengetahui:

- Mengetahui versi *Firmware* (AT+GMR).
- Menampilkan *List Access Point* (AT+CWLAP).
- Mengubungkan dengan *Access Point* (AT+CWLAP="SSID", "PASSWORD").
- Memutuskan hubungan dengan *Access point* (AT+CWQAP).
- Mendapatkan dan mengetahui *IP Address* (AT+CIFSR).

2.4 Relay

Relay adalah Saklar (Switch) yang dioperasikan secara listrik dan merupakan komponen Electromechanical (Elektromekanikal) yang terdiri dari 2 bagian utama yakni Elektromagnet (Coil) dan Mekanikal (seperangkat Kontak Saklar/Switch). Relay menggunakan Prinsip Elektromagnetik untuk menggerakkan Kontak Saklar sehingga dengan arus listrik yang kecil (low power) dapat menghantarkan listrik yang bertegangan lebih tinggi. Sebagai contoh, dengan Relay yang menggunakan Elektromagnet 5V dan 50 mA mampu menggerakkan Armature Relay (yang berfungsi sebagai saklarnya) untuk menghantarkan listrik 220V 2A.

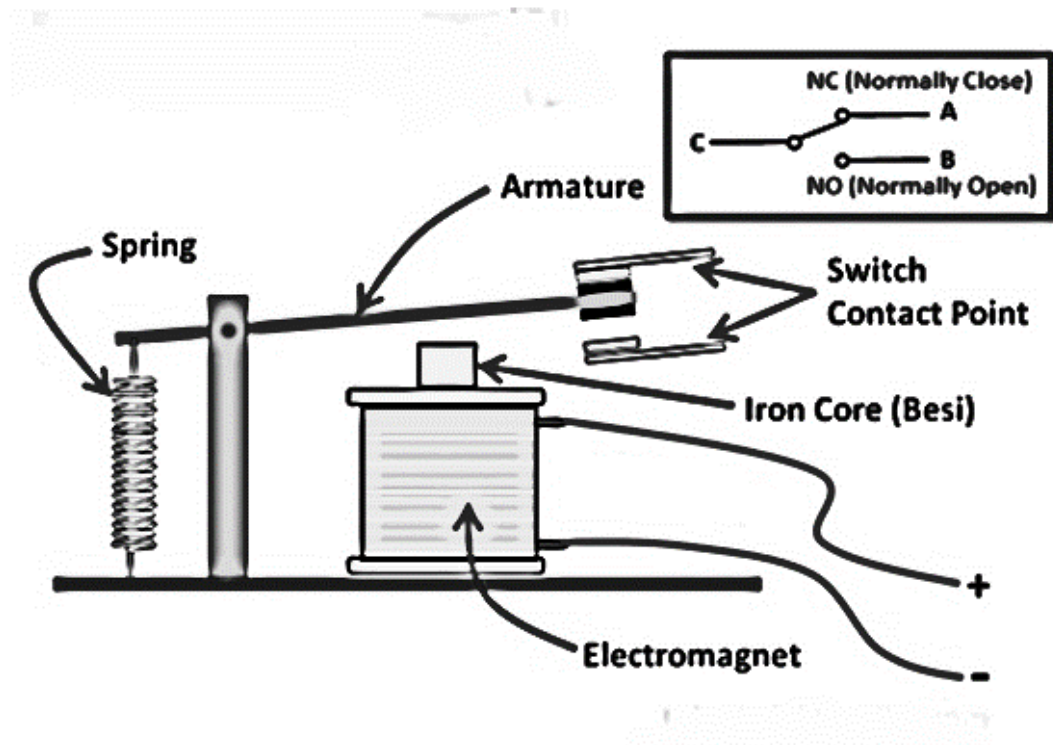


Gambar 2.8. Gambar Dan Simbol Relay

Kontak point pada relay ada dua yaitu *Normally open* yaitu keadaan dimana kondisi awal sebelum diaktifkan *open* (terbuka). dan *Normally close* yaitu kondisi awal sebelum diaktifkan berada pada posisi *close* (tertutup).

Pada dasarnya, Relay terdiri dari 4 komponen dasar yaitu :

1. Electromagnet (Coil)
2. Armature
3. Switch Contact Point (Saklar)
4. Spring.



Gambar 2.9. Struktur Sederhana Relay

Berdasarkan Gambar 2.9 diatas, sebuah Besi (Iron Core) yang dililit oleh sebuah kumparan Coil yang berfungsi untuk mengendalikan Besi tersebut. Apabila Kumparan Coil diberikan arus listrik, maka akan timbul gaya Elektromagnet yang kemudian menarik Armature untuk berpindah dari Posisi sebelumnya (NC) ke posisi baru (NO) sehingga menjadi Saklar yang dapat menghantarkan arus listrik di posisi barunya (NO). Posisi dimana Armature tersebut berada sebelumnya (NC) akan menjadi OPEN atau tidak terhubung. Pada saat tidak dialiri arus listrik, Armature akan kembali lagi ke posisi Awal (NC). Coil yang digunakan oleh Relay untuk menarik Contact Poin ke Posisi Close pada umumnya hanya membutuhkan arus listrik yang relatif kecil.

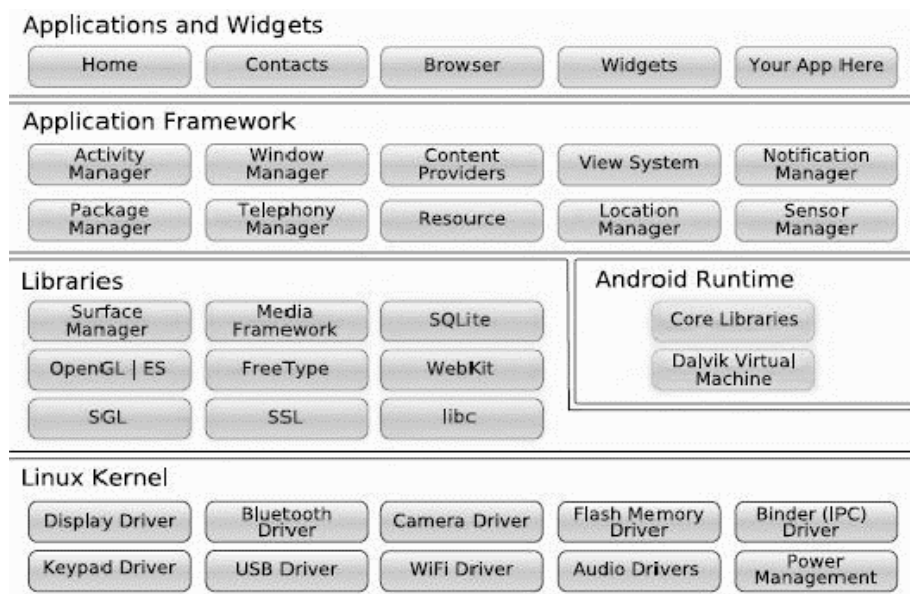
2.5 Android

Android adalah kumpulan perangkat lunak yang ditujukan bagi perangkat bergerak mencakup sistem operasi, middleware, dan aplikasi kunci. Android *Standart Development Kid* (SDK) menyediakan perlengkapan dan *Application*

Programming Interface (API) yang diperlukan untuk mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java.

Android dikembangkan oleh Google bersama *Open Handset Alliance* (OHA) yaitu aliansi perangkat selular terbuka yang terdiri dari 47 perusahaan *Hardware*, *Software* dan perusahaan telekomunikasi ditujukan untuk mengembangkan standar terbuka bagi perangkat selular.

Pada mulanya terdapat berbagai macam sistem operasi pada perangkat selular, diantaranya sistem operasi Symbian, Microsoft Windows Mobile, Mobile Linux, iPhone, dan sistem operasi lainnya. Namun diantara sistem operasi yang ada belum mendukung standar dan penerbitan API yang dapat dimanfaatkan secara keseluruhan dan dengan biaya yang murah. Kemudian Google ikut berkecimpung didalamnya dengan platform Android, yang menjanjikan keterbukaan, keterjangkauan, *open source*, dan *framework* berkualitas. Dalam paket sistem operasi Android terdiri dari beberapa unsur seperti tampak pada Gambar 2.10. Secara sederhana arsitektur Android merupakan sebuah kernel Linux dan sekumpulan pustaka C / C++ dalam suatu framework yang menyediakan dan mengatur alur proses aplikasi.



Gambar 2.10. Arsitektur Android(Suharyadi, 2013).

Di lapisan terbawah Arsitektur Android terdapat Linux Kernel, lapisan ini merupakan jantung dari seluruh sistem yang ada di Android karena lapisan ini memberikan fungsi menyediakan Driver Layar, Kamera, Keypad, WIFI, Flash Memory, Audio Dll. Diatas Linux Kernel ada Library yang berfungsi membawa sekumpulan instruksi untuk mengarahkan perangkat Android kita dalam menangani berbagai tipe data. Contohnya, perekam dari berbagai macam format Video dan Audio ditangani oleh Media Framework Library. Di lapisan ini terdapat Android Runtime yang memiliki *Core Libraries* yang menyediakan sebagian besar fungsi android. Inilah yang membedakan sistem operasi lain yang juga meng implementasikan Linux, karena terdapat sekumpulan Library Java yang dikhususkan untuk Android. Dalam lapisan Android *Runtime* juga terdapat *Dalvik VM (Virtual Machine)* yang merupakan Sejenis *Java Virtual Machine* yang didesain khusus dan Dioptimaskan untuk android. *Dalvik VM* menggunakan fitur linux untuk membuat setiap Aplikasi Android dapat berjalan dengan prosesnya sendiri. Lapisan berikutnya adalah *Application Framework* yang berhubungan langsung dengan aplikasi. program-program di atas memanajemen fungsi dasar dari perangkat seperti Manajemen *Resource*, Manajemen Panggilan, Manajemen Window dll. Sebagai seorang Developer, lapisan ini dapat dilihat sebagai alat dasar yang digunakan untuk mengembangkan aplikasi. *Application Layer* berada pada lapisan terluar dari Arsitektur Android, pengguna Android akan berinteraksi di Lapisan ini. Memiliki fungsi umum seperti menelepon, mengakses website, da

2.6 Router

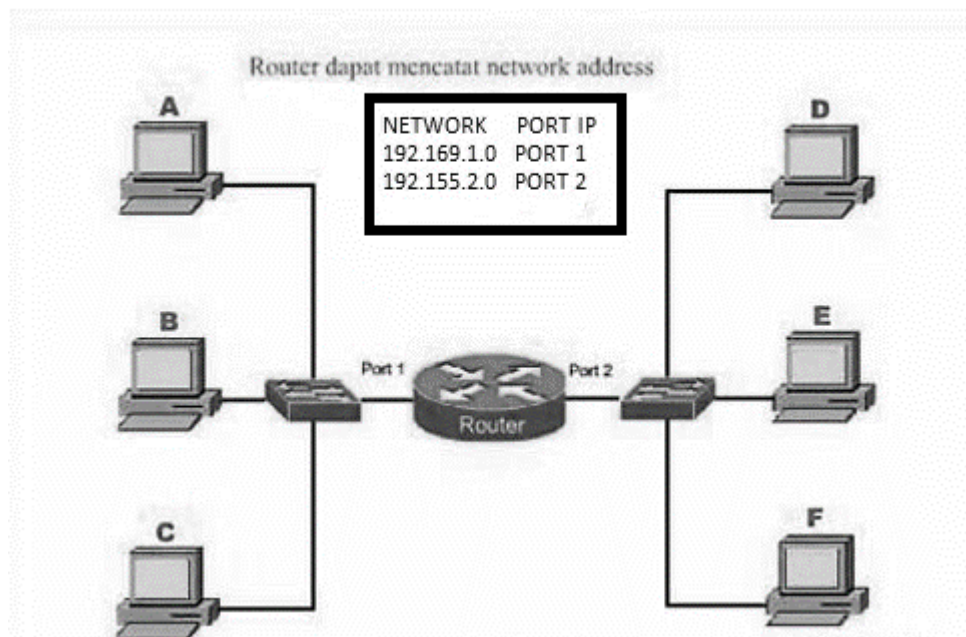
Router merupakan perangkat keras jaringan komputer yang dapat digunakan untuk menghubungkan beberapa jaringan yang sama atau berbeda. Router adalah sebuah alat untuk mengirimkan paket data melalui jaringan atau internet untuk dapat menuju tujuannya, proses tersebut dinamakan routing.



Gambar 2.11. Router

Fungsi utama dari Router adalah membagi atau mendistribusikan IP *address*, baik secara statis maupun [DHCP](#) atau Dynamic Host Configuration Protocol kepada semua komputer yang terhubung ke router tersebut. Dengan adanya IP address yang unik yang dibagikan router tersebut kepada setiap komputer dapat memungkinkan setiap komputer untuk saling terhubung serta melakukan komunikasi, baik itu pada LAN atau internet.

Router bekerja dengan cara merutekan paket atau data informasi yang disebut dengan *routing*. Dengan teknik routing tersebut, router dapat mengetahui arah rute perjalanan informasi tersebut akan dituju, apakah berada pada satu jaringan yang sama atau berbeda. Jika informasi yang dituju mengarah kepada jaringan yang berbeda, maka router akan meneruskannya kepada jaringan tersebut, sebaliknya apabila paket yang dituju adalah jaringan yang sama, maka router akan menghalangi paket keluar serta meneruskan paket tersebut dengan routing di jaringan yang sama sampai terkirim ke tujuan.



Gambar 2.12. Ilustrasi Cara Kerja Router (Helen, 2015)

Cara kerja router dapat dilihat pada Gambar 2.12 diatas. Pada gambar diatas terdapat dua buah network yang terhubung pada sebuah router. Network yang berada pada sebelah kiri yang terhubung ke port 1 router mempunyai alamat 192.168.1.0 serta pada network yang sebelah kanan yang terhubung ke port 2 router mempunyai alamat 192.155.2.0. Komputer A mengirim sebuah data kepada komputer C, router tidak akan meneruskan data tersebut kepada jaringan yang lainnya.

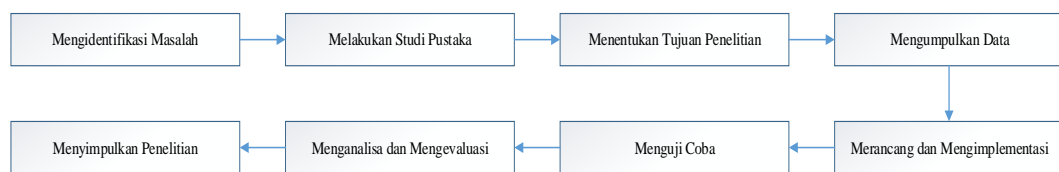
Begitu juga apabila ketika komputer F mengirim sebuah data kepada komputer E, router tidak akan meneruskan paket data tersebut kepada network yang lainnya barulah ketika komputer F mengirimkan sebuah data kepada komputer B, router akan meneruskan paket data tersebut ke komputer B.

BAB 3

METODOLOGI PENELITIAN

3.1. Alur Kerja Penelitian

Alur Kerja Penelitian ini di ilustrasikan pada gambar 3.1 :

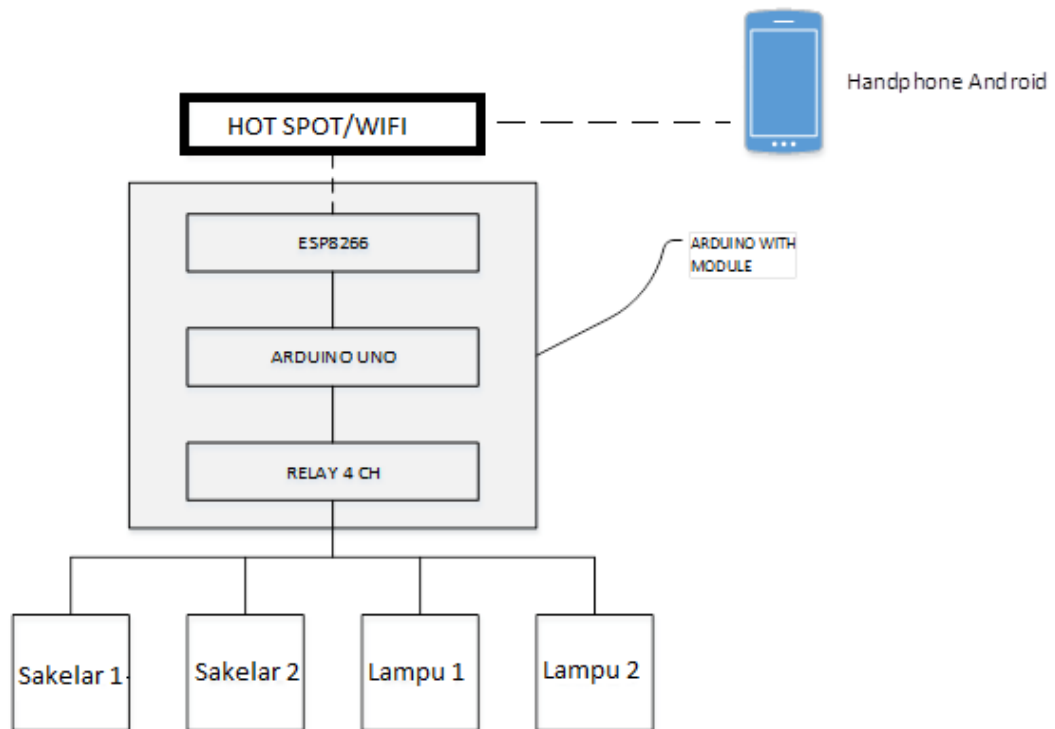


Gambar 3.1 Diagram Alur Kerja Penelitian

Berdasarkan gambar 3.1 dapat dijelaskan bahwa alur kerja penelitian ini dimulai dengan tahapan mengidentifikasi sebuah masalah yang akan diteliti, kemudian dilakukan studi pustaka yang berkaitan dengan masalah yang akan diteliti dilanjutkan dengan menentukan tujuan penelitian agar penelitian tidak menyebar ke ruang lingkup yang lain, selanjutnya dilakukan pengumpulan data atau sampel yang akan diteliti dilanjutkan dengan merancang dan mengimplementasi menggunakan sampel yang telah dikumpulkan dimana perancangan dan pengimplementasian sesuai dengan tujuan penelitian yang telah ditentukan. Selanjutnya dilakukan pengujian terhadap metode yang telah dirancang dan diimplementasikan dan pada tahapan akhir dilakukan analisa dan evaluasi metode sehingga dapat diambil kesimpulan terhadap penelitian.

3.2.Prancangan Sistem

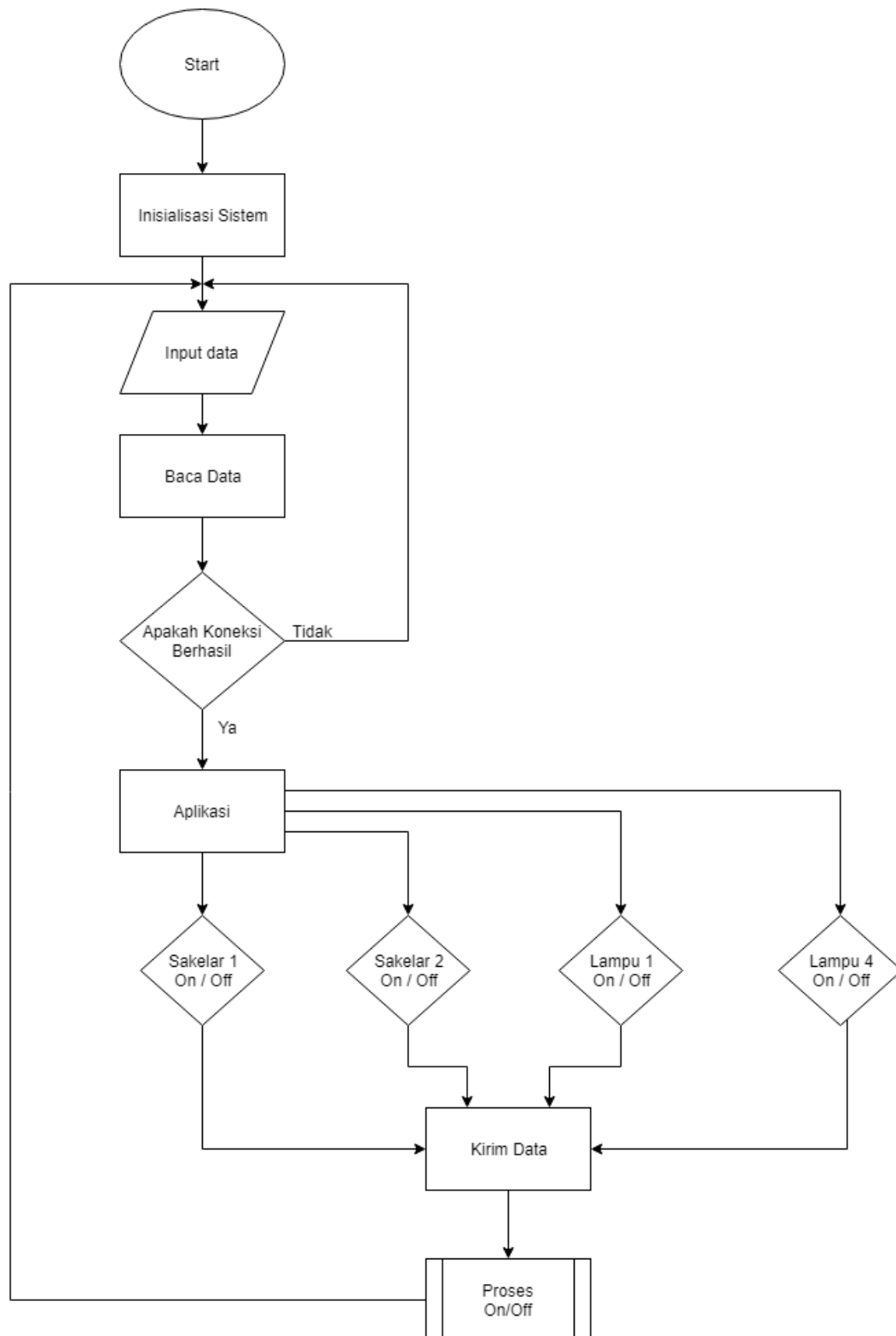
3.2.1 Skema Diagram Rancangan kontroler lampu



Gambar 3.2 Skema Diagram Rancangan Kontroler Lampu.

Penelitian ini merancang kontroler lampu dengan android dan memakai modul wifi esp 8266 serta menghubungkan lampu pada relay 4 channel seperti pada gambar 3.2. Handphone android memanfaatkan modul wifi 8266 untuk masuk ke dalam jaringan hotspot android tersebut. Dengan menggunakan perintah AT Command pada modul wifi esp8266 dapat mengetahui ip address yang didapatkan. Selanjutnya membuka aplikasi android yang telah dirancang pada handphone android untuk mengontrol relay lampu berdasarkan ip address yang telah didapatkan.

3.2.2 Flowchart Sistem

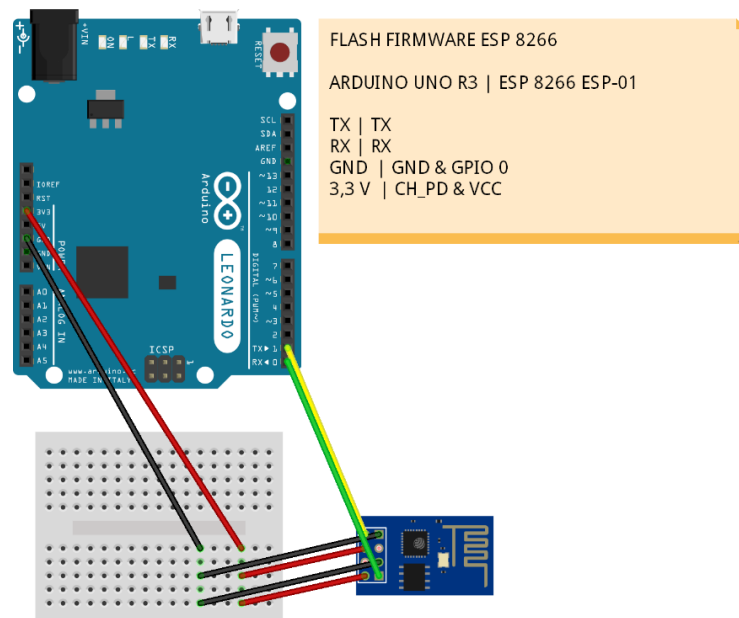


Gamabar 3.3 Flowchart Sistem

Gambar 3.3 Merupakan *Flowchart* sistem yang dimulai dari inisialisasi sistem kemudian dilanjutkan ke input data dan baca data yang terjadi pada ESP8266 dan Arduino apabila data tidak diterima berarti koneksi tidak berhasil, bila koneksi telah berhasil aplikasi pada Android dapat mengirim data yang yang dipilih kemudian akan dikirim kepada arduino dan diproses sesuai pilihan.

3.3. Flashing Firmware ESP 8266

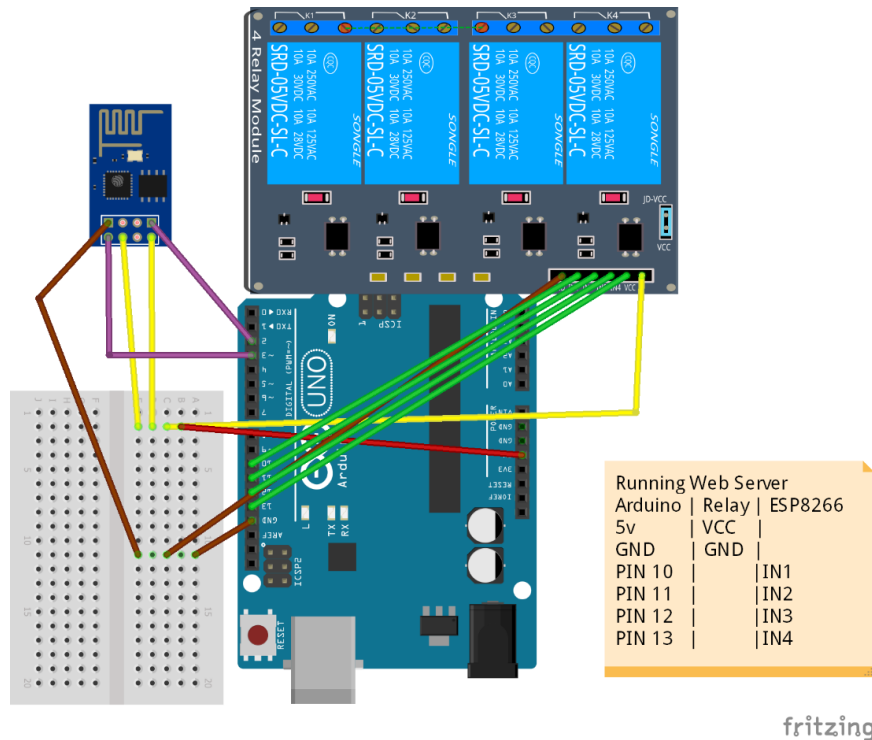
Sebelum dapat menggunakan modul esp 8266 pada arduino dan relay, maka modul wifi esp 8266 harus dilakukan flashing terlebih dahulu. Flashing ini dilakukan untuk memasukkan AT command dan fungsionalitas wifi pada esp 8266. Berikut adalah susunan rangkaiannya.



Gambar 3.4 Susunan Rangkaian Flashing Firmware esp8266.

3.4. Rangkaian Kontroler Lampu

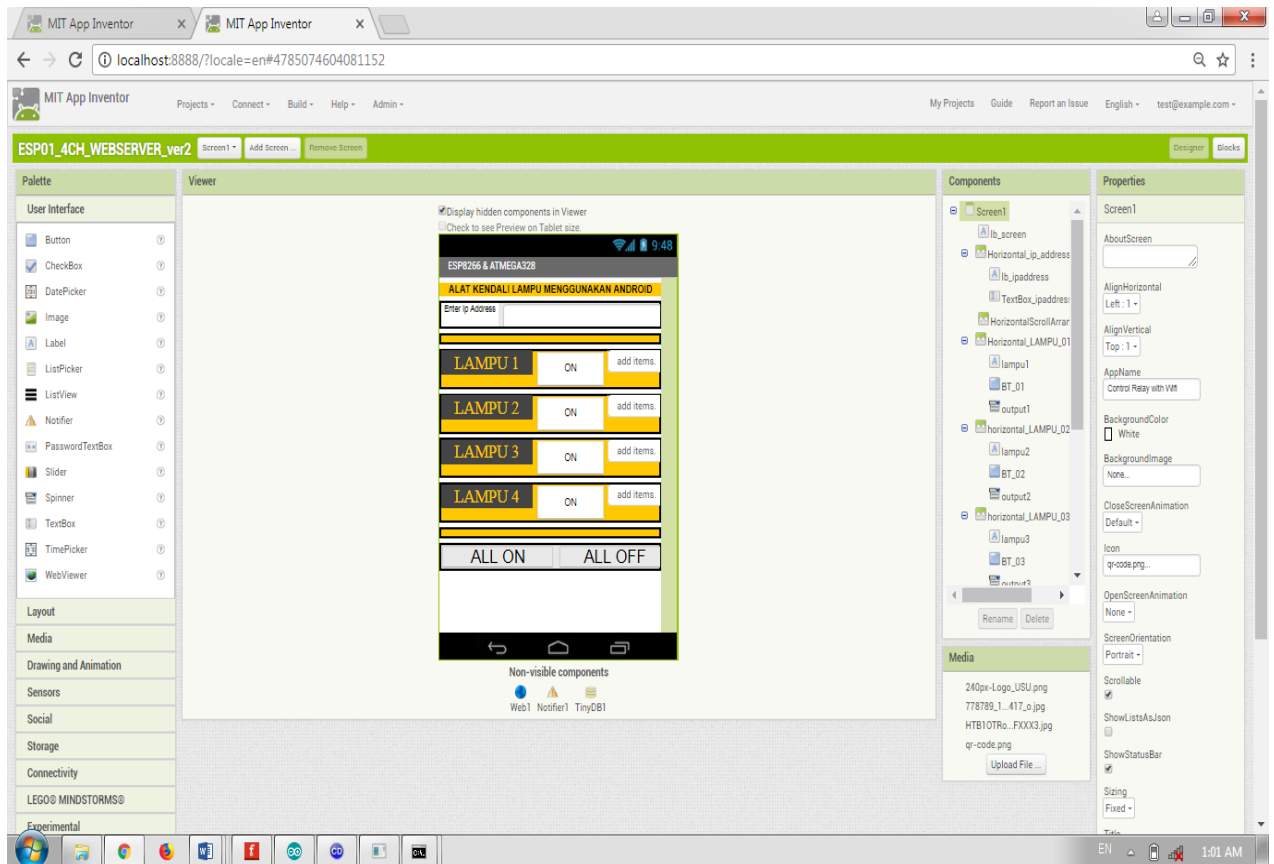
Setelah melakukan flashing firmware pada modul esp 8266 selanjutnya menyusun rangkaian arduino, relay dan esp 8266 untuk dapat terhubung. Berikut adalah susunan rangkaiannya.



Gambar 3.5 Susunan Rangkaian Kontroler Lampu

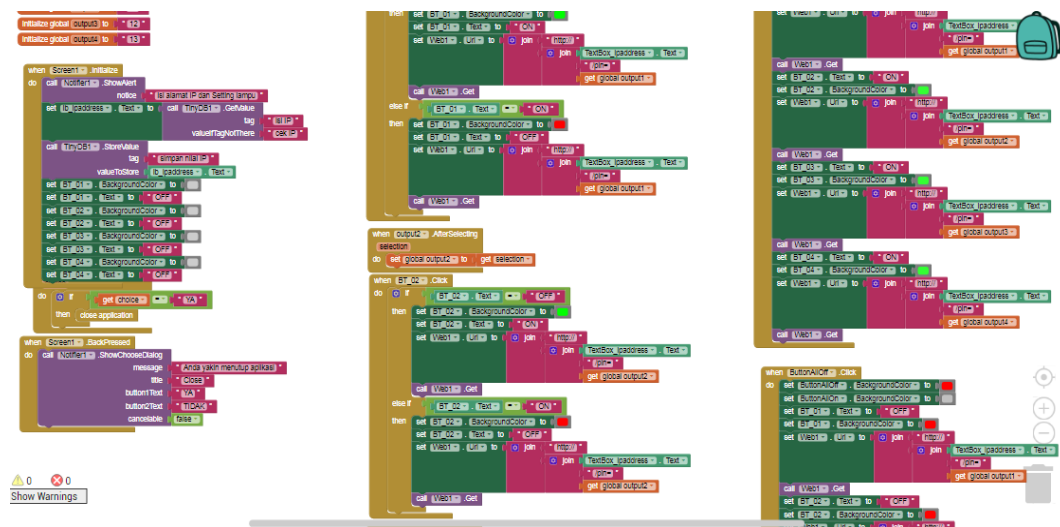
3.5. Perancangan Antar Muka Sistem Android

Untuk dapat mengontrol lampu melalui handphone android, maka diperlukan perancangan aplikasi pada handphone android tersebut. Berikut adalah rancangan aplikasi androidnya.



Gambar 3.6 Rancangan Aplikasi Android.

Pada gambar 3.6 merupakan rancangan tampilan utama aplikasi android untuk kontroler lampu dengan menggunakan app inventor 2. Terlihat pada gambar 3.6 tombol on dan off yang akan digunakan untuk mengontrol lampu pada rangkaian arduino uno, relay dan modul wifi esp8266.



Gambar 3.7 Code Block Aplikasi Android

Pada gambar 3.7 merupakan code block pada rancangan aplikasi android gambar 3.7. code block ini menjelaskan operasional tombol-tombol pada aplikasi android yang dirancang untuk lebih jelasnya dapat dilihat di lampiran.

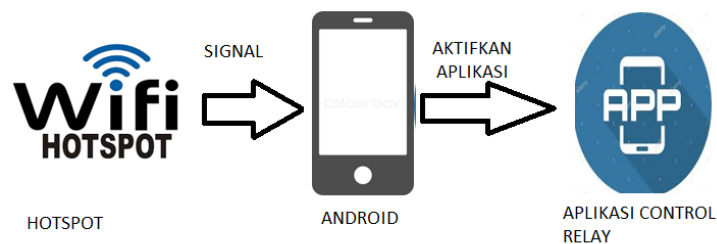
BAB 4

PENGUJIAN DAN IMPLEMENTASI SISTEM KENDALI LAMPU MELALUI ANDROID DAN WIFI

4.1 Rancangan Sistem Kendali Lampu Melalui Android dan Wifi

4.1.1. Mengaktifkan Aplikasi control relay pada android

Tahapan implementasi rancangan pada pembuatan sistem kendali lampu yang berbasis android dan wifi ini dimulai dari penggunaan handphone android yang menggunakan *hotspot* dan mengaktifkan aplikasi control relay dengan wifi.



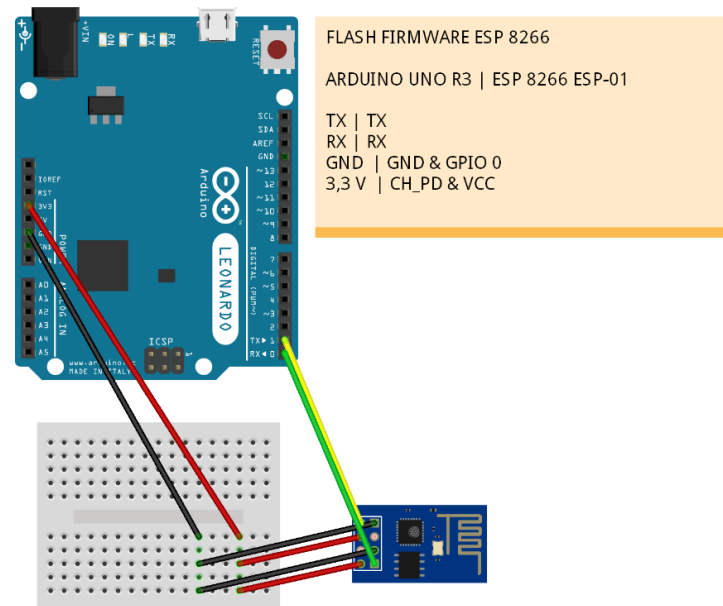
Gambar 4.1 Aplikasi Relay pada Android

4.1.2. Perancangan Sistem Kendali Lampu Melalui Android Dan Wifi

Proses pemograman dan perancangan ini dimulai dari:

A. Flashing esp8266

Pada tahapan ini dilakukan flashing modul wifi esp8266, untuk mengetahui apakah modul wifi esp8266 berjalan dengan baik atau tidak. Langkah pertama untuk melakukan flashing modul wifi esp8266 adalah dengan menyusun rangkaian firmware-nya.

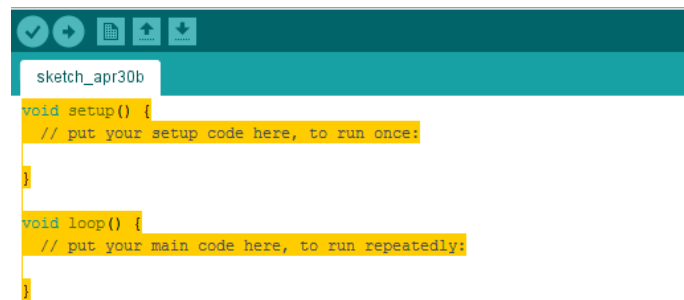


Gambar 4.2 Susunan Rangkaian Flashing Firmware esp8266

No	ARDUINO R3	RELAY	keterangan
1	TX	TX	Hubungan pin dengan kabel jumper
2	RX	RX	Hubungan pin dengan kabel jumper
3	GND	GND + GPIO 0	Hubungan pin dengan kabel jumper
4	3,3 V	CH-PD + VCC	Hubungan pin dengan kabel jumper

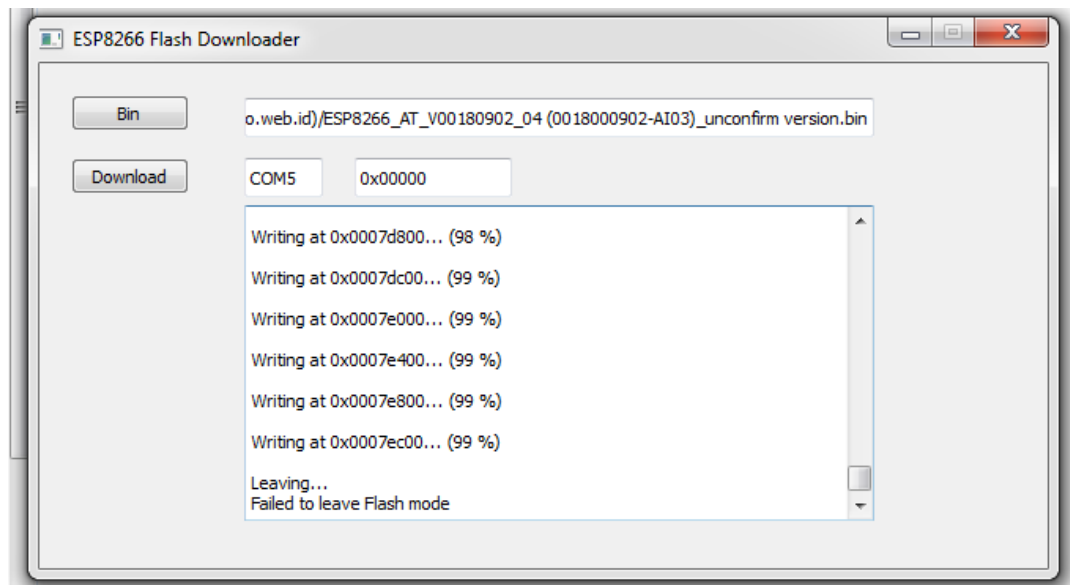
Tabel 4.1 Susunan Rangkaian Flashing Firmware Esp8266

Setelah rangkaian tersusun, maka dilakukan upload perintah sketch pada board arduino melalui software arduino 1.8.5 seperti pada gambar 4. 2.



Gambar 4.3 Upload sketch pada board arduino

Selanjutnya menghubungkan rangkaian ke PC melalui kabel USB, dan lalu membuka aplikasi ESP tool Firmware.

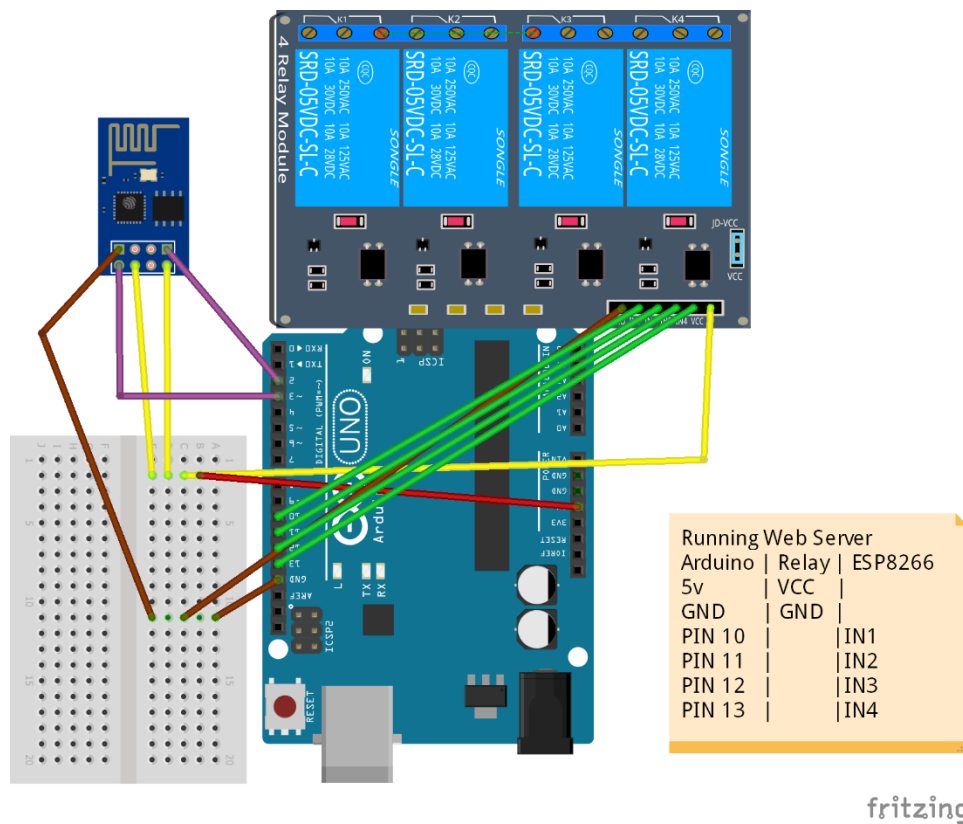


Gambar 4.4 Proses Flashing Firmware Esp8266

Gambar 4.4 menunjukkan flashing firmware esp8266 yang berjalan lancar.

B.Perangkaian modul hardware

Langkah selanjutnya dengan mengubah susunan rangkaian dengan menghubungkan relay dan modul wifi esp8266 seperti pada gambar 4.5.

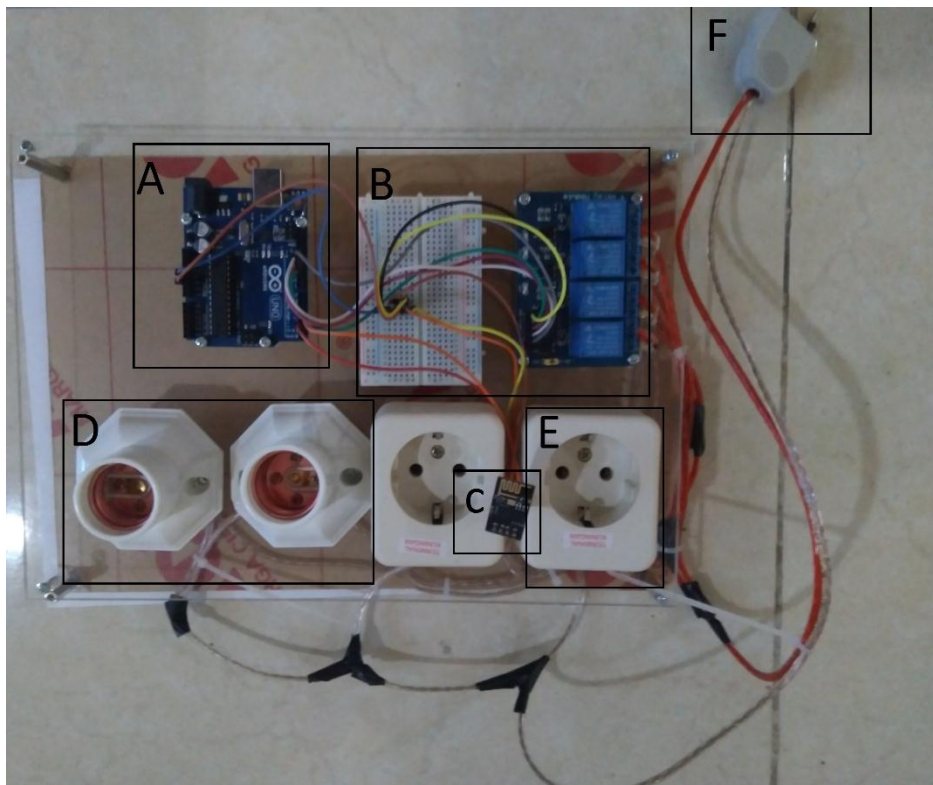


Gambar 4.5 Rangkaian Arduino Relay dan ESP 8266

No	Arduiono r3	Relay	ESP8266
1	5v	VCC	Hubungan pin-pin dengan kabel jumper
2	GND	GND	
3	PIN 10		IN 1
4	PIN 11		IN 1
5	PIN 12		IN 1
6	PIN 13		IN 1

Tabel 4.2 Susunan Rangkaian Arduino Relay dan ESP8266

Setelah menyusun rangkaian seperti pada gambar 4.4 dan tabel 4.2, maka hardwarenya akan berbentuk seperti Gambar 4.5



Gambar 4.6 Hasil Perancangan Hardware

Pada gambar 4.6 adalah hasil dari perancangan modul yang dimulai dengan penjelasan rangkaian sebagai berikut:

1. Dimulai dengan menghubungkan alat pada pln.(F)
2. Kemudian ESP8266 (C) akan menerima sinyal dari hotspot Handphone.
3. Sinyal diteruskan ke Arduino Atmega328 (A) kemudian di proses.
4. Hasil proses dari Arduino Atmega328 (A) akan dikirim ke Relay (B).
5. Hasil dari Relay (B) akan dilanjutkan ke lampu (D) atau ke Stop kontak sesuai dengan pilihan.

langkah selanjutnya mengupload sketch pada software arduino 1.8.5. Sintaks sketch yang diupload adalah sebagai berikut:

```

#include <SoftwareSerial.h>

#define DEBUG true

SoftwareSerial wifi(2,3); // RX, TX


void setup() {
  Serial.begin(9600);
  wifi.begin(9600);


  pinMode(10,OUTPUT);
  digitalWrite(10,HIGH);
  pinMode(11,OUTPUT);
  digitalWrite(11,HIGH);
  pinMode(12,OUTPUT);
  digitalWrite(12,HIGH);
  pinMode(13,OUTPUT);
  digitalWrite(13,HIGH);


  // Reset modul
  sendCommand("AT+RST\r\n",2000,DEBUG);

  // Konfigurasi sebagai Akses poin
  sendCommand("AT+CWMODE=3\r\n",1000,DEBUG);

  // Sesuaikan dengan SSID dan Password
  sendCommand("AT+CWLAP=\"VIBE Shot\", \"12345678\"\r\n",10000,DEBUG);

  // Mendapatkan IP address nama hotspot dan password disesuaikan
  sendCommand("AT+CIFSR\r\n",1000,DEBUG);

  // Konfigurasi untuk multiple connection
  sendCommand("AT+CIPMUX=1\r\n",1000,DEBUG);

  // Aktifkan server pada port 80
  sendCommand("AT+CIPSERVER=1,80\r\n",1000,DEBUG);
  Serial.println("Server sudah siap!");
}

```

```

void loop() {
    // Cek jika ESP mengirimkan pesan
    if(wifi.available()) {

        if(wifi.find("+IPD,") {
            delay(1000);
            int connectionId = wifi.read()-48;
            wifi.find("pin=");
            int pinNumber = (wifi.read()-48)*10;
            pinNumber += (wifi.read()-48);

            int secondNumber = (wifi.read()-48);
            if(secondNumber>=0 && secondNumber<=9)
            {
                pinNumber*=10;
                pinNumber +=secondNumber;
            }
            pinMode(pinNumber,OUTPUT);
            digitalWrite(pinNumber, !digitalRead(pinNumber));

            String closeCommand = "AT+CIPCLOSE=";
            closeCommand+=connectionId;
            closeCommand+="\r\n";
            sendCommand(closeCommand,1000,DEBUG);
        }
    }
}

```

```
String sendCommand(String command, const int timeout, boolean debug)
```

```
{  
    String response = "";  
    wifi.print(command);  
    long int time = millis();  
  
    while( (time+timeout) > millis())  
    {  
        while(wifi.available())  
        {  
            char c = wifi.read();  
            response+=c;  
        }  
    }  
  
    if(debug) {  
        Serial.print(response);  
    }  
  
    Return response;  
}
```

4.2 Pemograman pada sistem

4.2.1 Pemograman Pada Arduino Uno

Penjelasan pada pemograman ini dimulai dengan *mengimport library* ini

```
#include <SoftwareSerial.h>
#define DEBUG true
SoftwareSerial wifi(2,3); // RX, TX

void setup() {
  Serial.begin(9600);
  wifi.begin(9600);
}
```

Gambar 4.7 Library Yang Digunakan

Pada Gambar 4.7 adalah pemograman dasar yang dilakukan untuk berkomunikasi dengan komputer dengan Baudrate, semakin tinggi Baudrate maka dapat mengirim transmisi lebih cepat disini penulis menggunakan 9600 Baudrate.

```
pinMode(10,OUTPUT);|
digitalWrite(10,HIGH);
pinMode(11,OUTPUT);
digitalWrite(11,HIGH);
pinMode(12,OUTPUT);
digitalWrite(12,HIGH);
pinMode(13,OUTPUT);
digitalWrite(13,HIGH);
```

Gambar 4.8 Kode Program Untuk Menghidupkan Sakelar

Pada gambar 4.8 menjelaskan pin-pin yang digunakan untuk berkomunikasi pada arduino ATMEGA328.

```
// Reset modul
sendCommand("AT+RST\r\n",2000,DEBUG);|
// Konfigurasikan sebagai Akses poin
sendCommand("AT+CWMODE=3\r\n",1000,DEBUG);
// Sesuaikan dengan SSID dan Password
sendCommand("AT+CWJAP=\"badsfille\", \"badsfille\"\r\n",10000,DEBUG);
// Mendapatkan IP adress nama hotspot danpassword disesuaikan
sendCommand("AT+CIFSR\r\n",1000,DEBUG);
// Konfigurasikan untuk multiple connection
sendCommand("AT+CIPMUX=1\r\n",1000,DEBUG);
// Aktifkan server pada port 80
sendCommand("AT+CIPSERVER=1,80\r\n",1000,DEBUG);
Serial.println("Server sudah siap!");
}
```

Gambar 4.9 Kode Pemrograman Untuk Menampilkan Status Koneksi ke Serial Monitor

Pada gambar 4.9 adalah penggabungan antara Esp8266 dengan android agar bisa dilakukan melalui wifi. Esp8266 memiliki 3 mode operasi yaitu *Station (STA)*, *Acces Point (AP)* dan Gabungan Mode STA dan AP disini penulis menggunakan mode gabungan *Station* dan *Acess Point*, sehingga fungsi yang digunakan adalah AT+CWMODE=3. koneksi dengan android dimulai dengan mengkonfigurasi Akses point wifi dengan perintah AT+CIFSR dan AT+CWMODE, setelah itu penulis harus menyesuaikan *username* dan *password* pada wifi yang ingin dituju yang, lalu server diaktifkan pada port 80, kemudian akan memunculkan *output* ke monitor.


```

void loop() {
  // Cek jika ESP mengirimkan pesan
  if(wifi.available()) {

    if(wifi.find("+IPD,") {
      delay(1000);
      int connectionId = wifi.read()-48;
      wifi.find("pin=");
      int pinNumber = (wifi.read()-48)*10;
      pinNumber += (wifi.read()-48);

      int secondNumber = (wifi.read()-48);
      if(secondNumber>=0 && secondNumber<=9)
      {
        pinNumber*=10;
        pinNumber +=secondNumber;
      }
      pinMode(pinNumber,OUTPUT);
      digitalWrite(pinNumber, !digitalRead(pinNumber));

      String closeCommand = "AT+CIPCLOSE=";
      closeCommand+=connectionId;
      closeCommand+="\r\n";
      sendCommand(closeCommand,1000,DEBUG);
    }

    String sendCommand(String command, const int timeout, boolean debug)
    {
      String response = "";
      wifi.print(command);
      long int time = millis();

      while( (time+timeout) > millis())
      {
        while(wifi.available())
        {
          char c = wifi.read();
          response+=c;
        }
      }

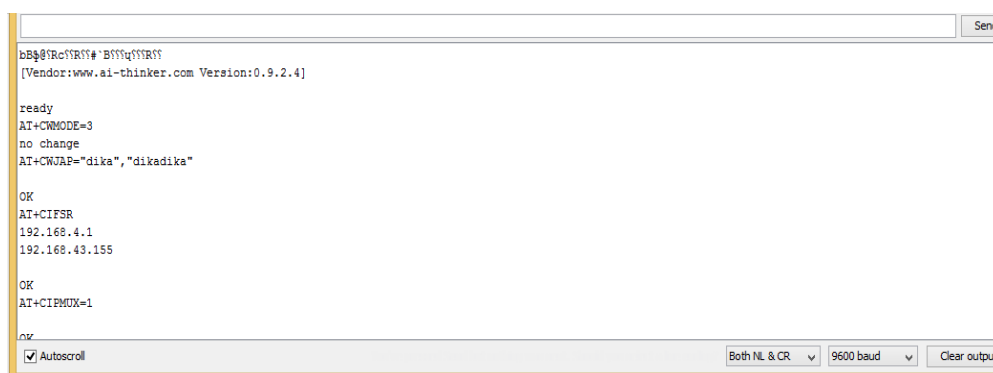
      if(debug) {
        Serial.print(response);
      }

      return response;
    }
  }
}

```

Gambar 4.10 Kode Untuk Mencek Jika Esp8266 Mengirim Pesan

Pada gambar 4.10 kode pemograman akan menerima perintah dari android yang dikirim kemudian akan menampilkan perintah yang diterima ke serial monitor. Setelah mengupload sketch pada board arduino melalui software arduino 1.8.5 selanjutnya dilakukn pengujian alat ya ng telah dirancang, akan menampilkan ip address dan mode modul wifi esp8266 dengan perintah AT+CIFSR dan AT+CWMODE seperti pada gambar 4.9.



```

bBq@IRc$R$S$B$R$S$R$S$
[Vendor:www.ai-thinker.com Version:0.9.2.4]

ready
AT+CWMODE=3
no change
AT+CWMJAP="dika","dikadika"

OK
AT+CIFSR
192.168.4.1
192.168.43.155

OK
AT+CIPMUX=1

OK

```

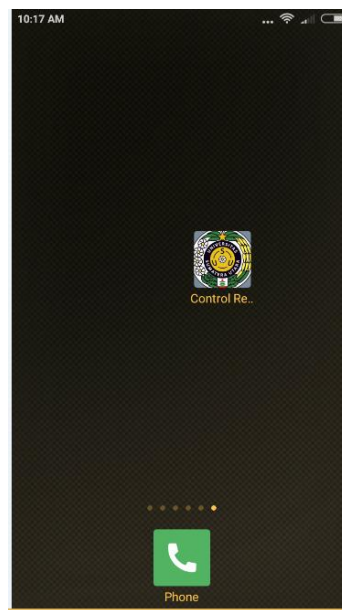
Gambar 4.11 Pengujian Pada Software Arduino 1.8.5

Pada gambar 4.11 terlihat ip address dan SSID hotspot mobile phone android yang dituju melalui modul wifi esp8266. Dengan mengetahui ip adress

pada software arduino 1.8.5 maka langkah selanjutnya adalah mengontrol lampu pada relay melalui mobile phone android.

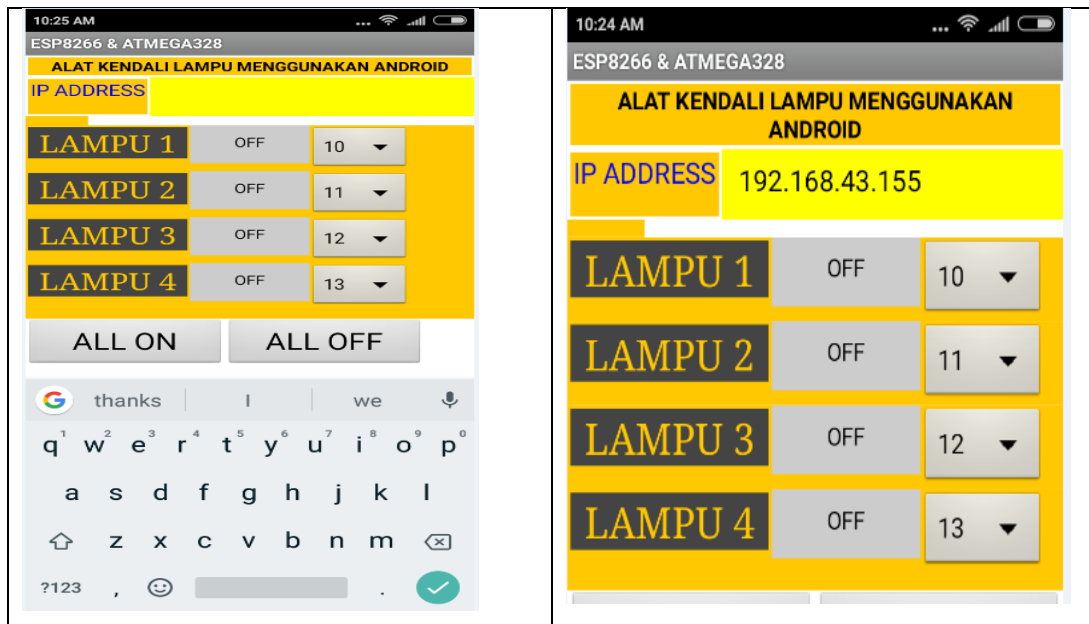
4.2.2 Pemograman pada Android dan Hasil

Selanjutnya dilakukan pengujian pada mobile phone android sebagai kontroler lampu. Pengujian ini dilakukan dengan menghidupkan dan mematikan ke empat lampu pada relay. Berikut adalah hasil pengujiannya :



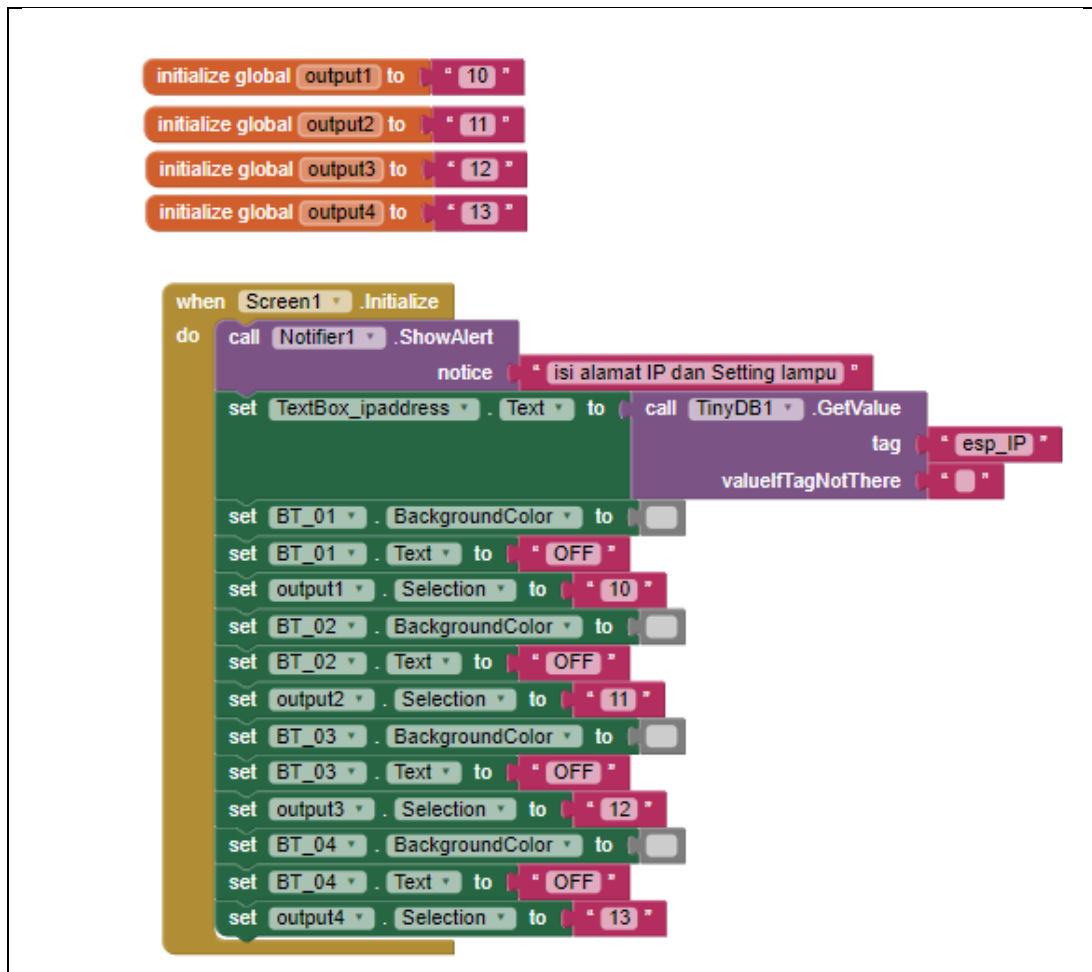
Gambar 4.12 Tampilan App Pada Smartphone

Pada gambar 4.12 adalah tampilan pada smartphone ketika dipilih dan pada gambar 4.6 akan terlihat modul hardwarenya. Setelah itu ketika aplikasi ditekan akan terlihat seperti pada gambar 4.13.



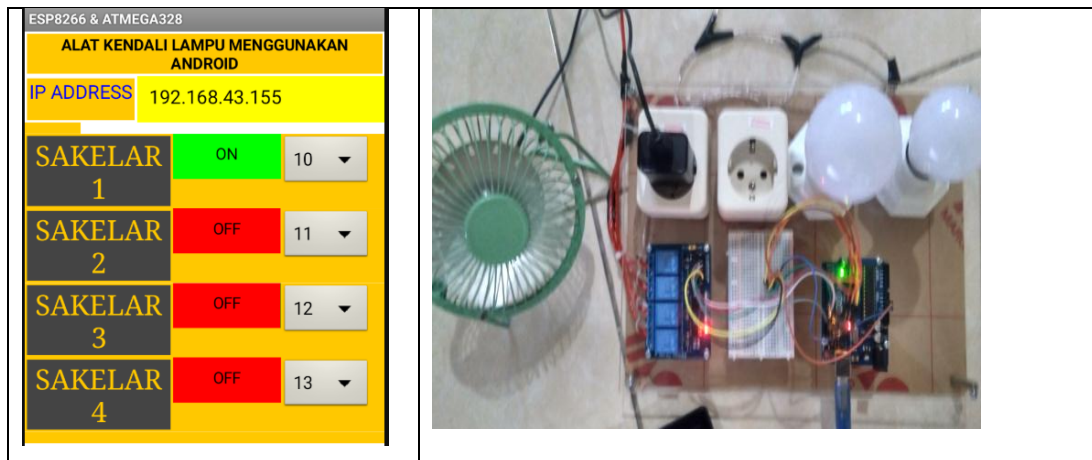
Gambar 4.13 Tampilan Awal Aplikasi

Pada Gambar 4.13 Terlihat ip address yang didapatkan pada pengujian dengan software arduino gambar 4.9. Ip address tersebut digunakan untuk mengendalikan lampu 1, lampu 2, lampu 3 dan lampu 4 pada relay. Pemograman dapat dilihat di gambar 4.14.



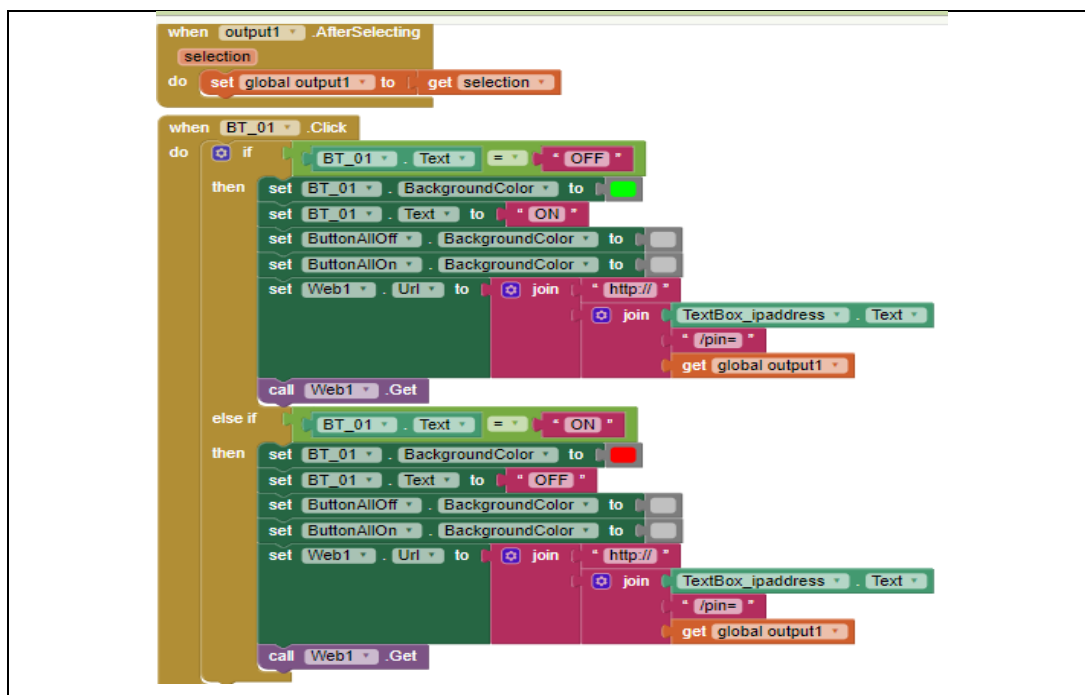
Gambar 4.14 Pemrograman Android Pada Tampilan Awal

Pada gambar 4.13 ketika akan memulai aplikasi maka akan muncul permintaan untuk mengisi alamat ip yang dilakukan oleh Notifier1. Lalu pemrograman tombol satu sampai dengan empat dimulai dengan BT_01 sampai BT_04.



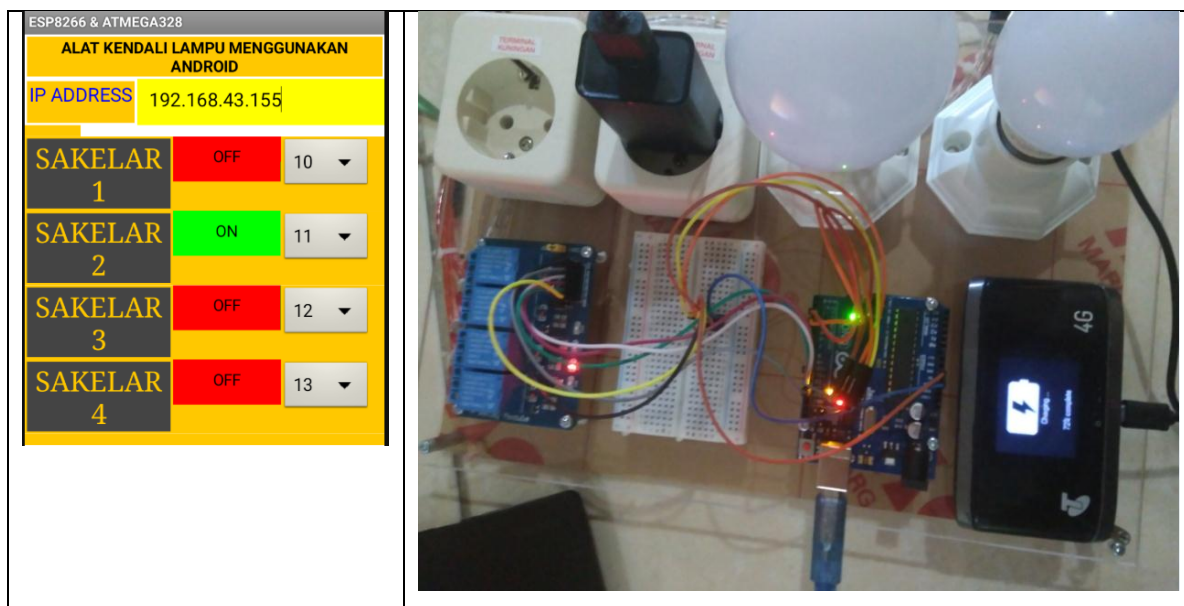
Gambar 4.15 Tampilan Pada Software dan Hardware ketika Sakelar 1 Ditekan

Pada gambar 4.15 ketika sakelar 1 ditekan maka akan menghidupkan kipas angin mini. Program yang digunakan bisa dilihat pada gambar 4.16.



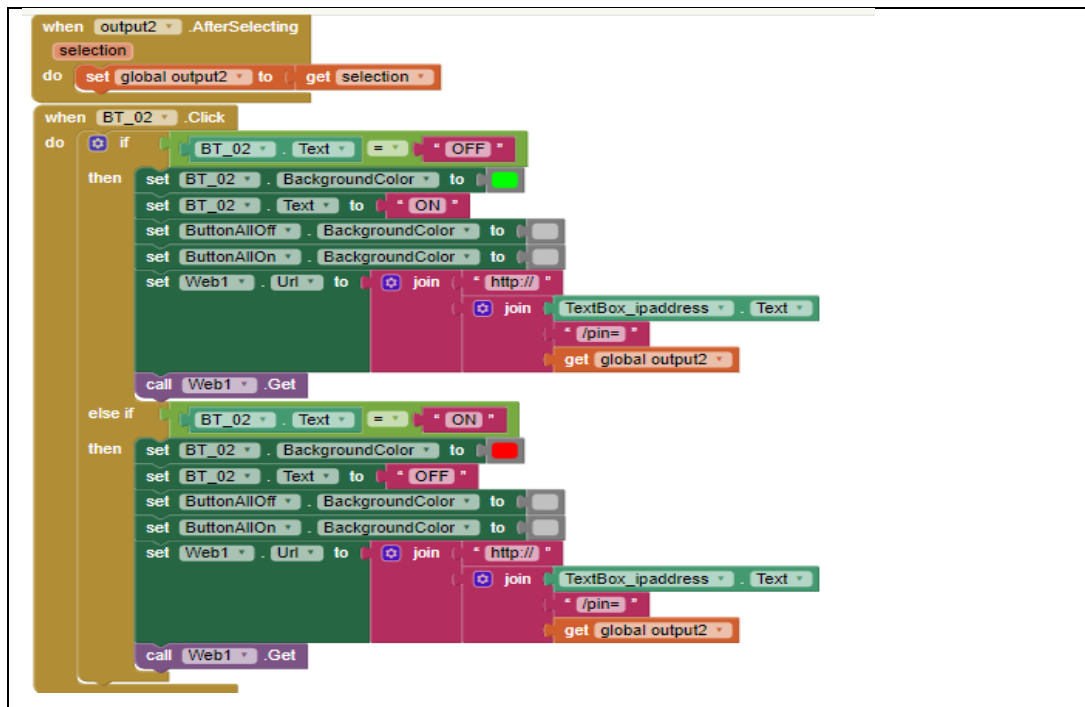
Gambar 4.16 Pemograman Pada Sakelar 1

Ketika bt_01 di tekan maka tombol off dengan Backgorundcolour yang awalnya abu-abu maka akan menset text menjadi ON dengan warna hijau, lalu program akan melanjutkan ke web1 dimana perintah ini akan mengirimkan perintah ke Pin yang dipilih, pada button 1 Pin yang dipilih adalah 10, ketika perintah telah dikirim oleh web1 maka akan dilanjutkan ke esp8266 dengan tampilan pada serial monitor seperti Gambar Ketika BT_01 yang berteks ON ditekan maka backgroundcolour hijau akan menjadi merah beserta teks akan menjadi OFF.



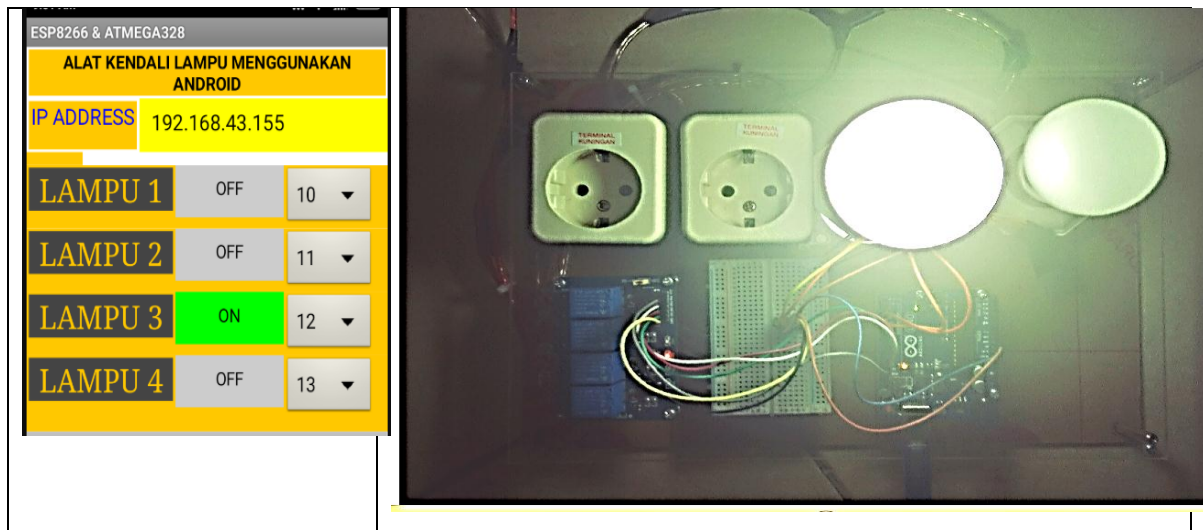
Gambar 4.17 ketika Sakelar Dua di Tekan Maka Modem Diisi Baterainya

Pada gambar 4.17 ketika sakelar kedua dihidupkan dan sakelar pertama dimatikan terlihat modem diisi baterainya. pemogramannya dapat dilihat pada gambar 4.18



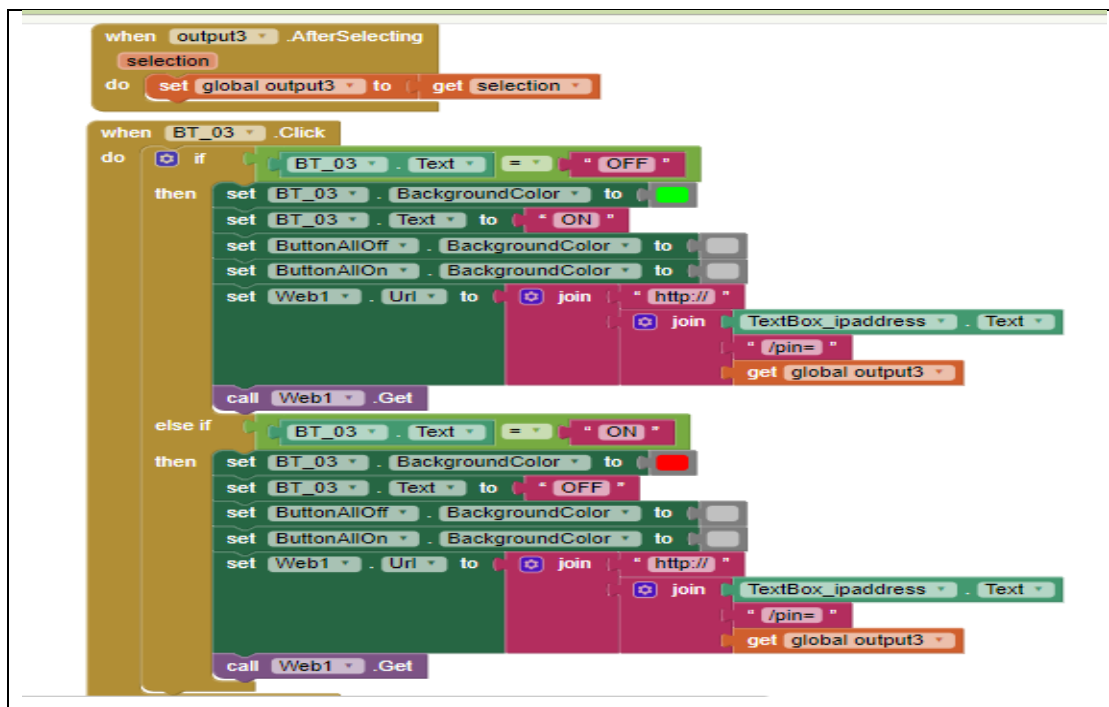
Gambar 4.18 Pemograman Pada Sakelar 2

Ketika bt_02 di tekan maka tombol off dengan Backgorundcolour yang awalnya abu-abu maka akan menset text menjadi ON dengan warna hijau, lalu program akan melanjutkan ke web1 dimana perintah ini akan mengirimkan perintah ke Pin yang dipilih, pada button 2 Pin yang dipilih adalah 11, ketika perintah telah dikirim oleh web1 maka akan dilanjutkan ke esp8266 dengan tampilan pada serial monitor seperti Gambar Ketika BT_02 yang berteks ON ditekan maka backgroundcolour hijau akan menjadi merah beserta teks akan menjadi OFF.



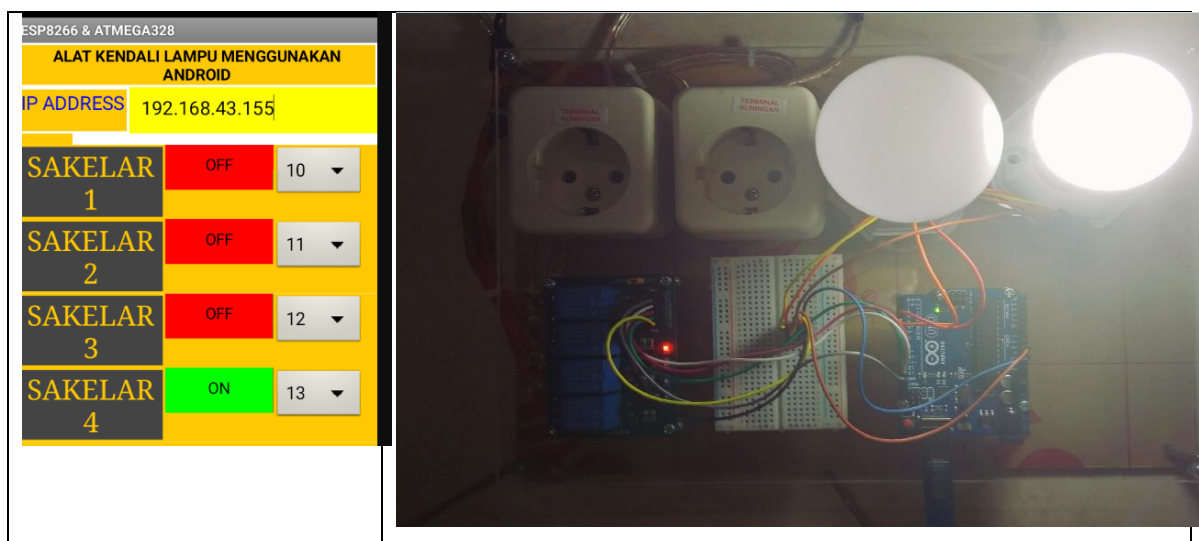
Gambar 4.19 ketika Sakelar Tiga di Tekan Maka Lampu Hidup

Pada gambar 4.19 ketika sakelar ketiga dihidupkan dan sakelar pertama dan kedua dimatikan terlihat modem diisi baterainya. pemogramannya dapat dilihat pada gambar 4.20



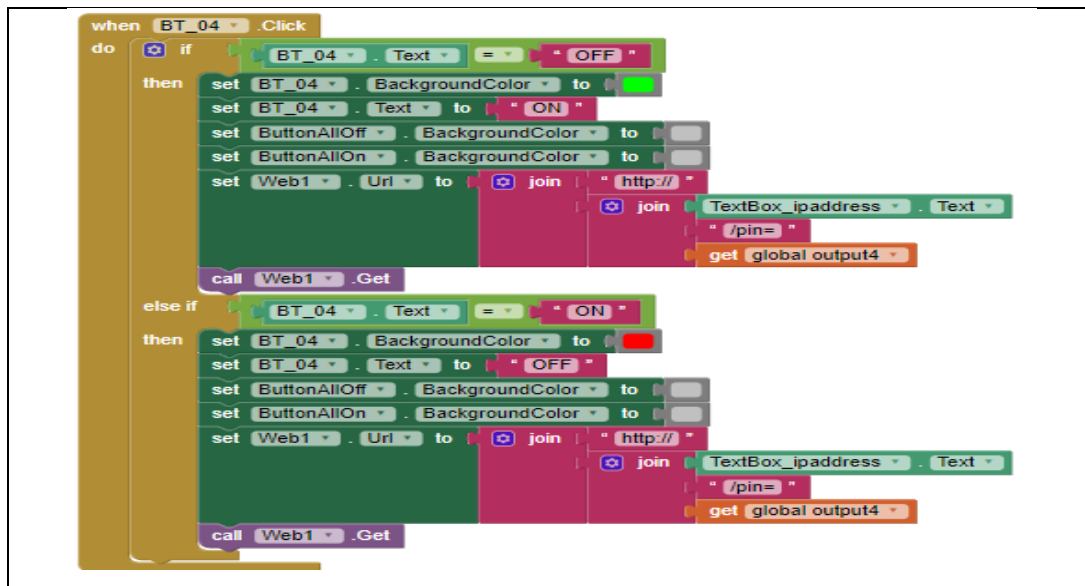
Gambar 4.20 Pemograman Pada Sakelar 3

Ketika bt_03 di tekan maka tombol off dengan Backgorundcolour yang awalnya abu-abu maka akan menset text menjadi ON dengan warna hijau, lalu program akan melanjutkan ke web1 dimana perintah ini akan mengirimkan perintah ke Pin yang dipilih, pada button 3 Pin yang dipilih adalah 12, ketika perintah telah dikirim oleh web1 maka akan dilanjutkan ke esp8266 dengan tampilan pada serial monitor seperti Gambar Ketika BT_03 yang berteks ON ditekan maka backgroundcolour hijau akan menjadi merah beserta teks akan menjadi OFF.



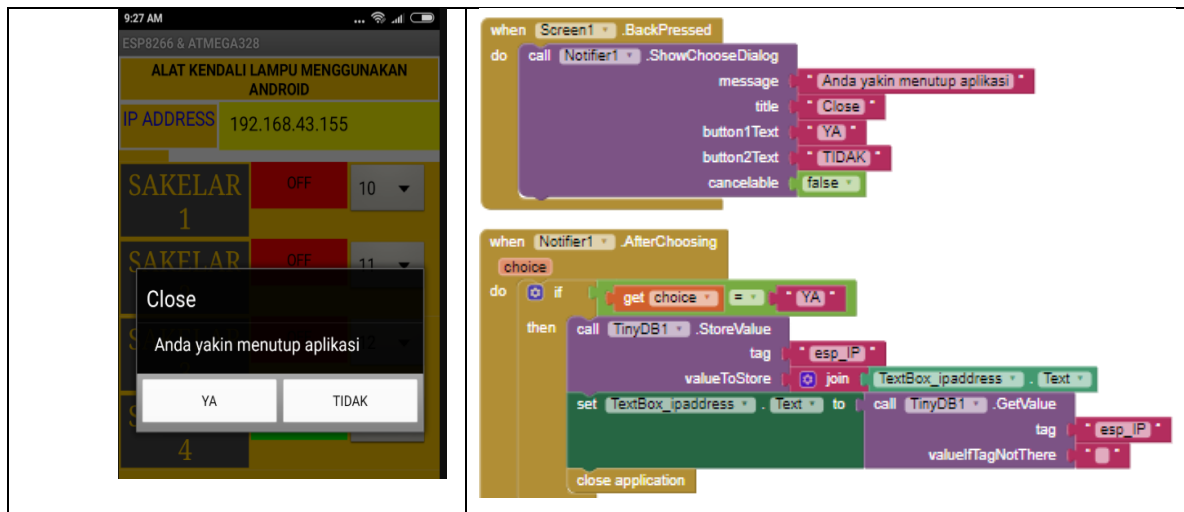
Gambar 4.21 Ketika Sakelar Ke empat Ditekan Maka Lampu Dua Hidup

Pada gambar 4.21 ketika sakelar ketiga dihidupkan dan sakelar pertama dan kedua dimatikan terlihat modem diisi baterainya. pemogramannya dapat dilihat pada gambar 4.22



Gambar 4.22 Pemograman Pada Sakelar 4

Ketika bt_04 di tekan maka tombol off dengan Backgorundcolour yang awalnya abu-abu maka akan menset text menjadi ON dengan warna hijau, lalu program akan melanjutkan ke web1 dimana perintah ini akan mengirimkan perintah ke Pin yang dipilih, pada button 4 Pin yang dipilih adalah 13, ketika perintah telah dikirim oleh web1 maka akan dilanjutkan ke esp8266 dengan tampilan pada serial monitor seperti Gambar Ketika BT_03 yang berteks ON ditekan maka backgroundcolour hijau akan menjadi merah beserta teks akan menjadi OFF.



Gambar 4.23 Pemograman Ketika Hendak Mengclose Aplikasi

Ketika tombol back ditekan pada handphone maka akan muncul notifier1 yang berisi “Anda yakin menutup aplikasi” dengan pilihan pada button1Text “ya” atau button2text “Tidak”. Jika pengguna memilih ya maka tinydb akan menyimpan value pada ip yang diterima kemudian akan menutup aplikasi. Setelah dilakukan pengujian, Sakelar satu sampai empat telah berhasil maka hardware dan software telah berhasil dikendalikan oleh handphone melalui wifi.

BAB 5

KESIMPULAN DAN SARAN

5.1 kesimpulan

Modul ESP8266 merupakan platform yang sangat murah tetapi benar-benar efektif untuk digunakan berkomunikasi atau kontrol melalui internet baik digunakan secara *standalone* (berdiri sendiri) maupun dengan menggunakan mikrokontroler tambahan dalam hal ini Arduino sebagai pengendalinya. Modul ini sangat tergantung dengan frekuensi wifi yang diterima pada sehingga menyebabkan ketergantungan pada frekuensi wifi yang stabil. Untuk menambah jumlah sakelar yang digunakan dapat dilakukan dengan menambah modul relay.

5.2 Saran

Dengan kemajuan jaman telah muncul esp8266 dengan tipe yang berbeda-beda sesuai dengan kebutuhan penggunaan dalam skripsi ini misalnya terdapat tipe yang sudah ada didalamnya arduino, juga terdapat alat yang mampu mempermudah proses flashing dengan nodeMCU flasher. Karena esp8266 memiliki *sleepmode* maka sumber dayanya kecil sehingga ketika menggunakan sumber daya seperti baterai dapat bertahan lama. Modul ini juga dapat dilanjutkan dengan menggunakan wifi yang dapat terhubung ke internet dengan membuat website dan server yang digunakan sebagai penghubung ke *hardware*.

DAFTAR PUSTAKA

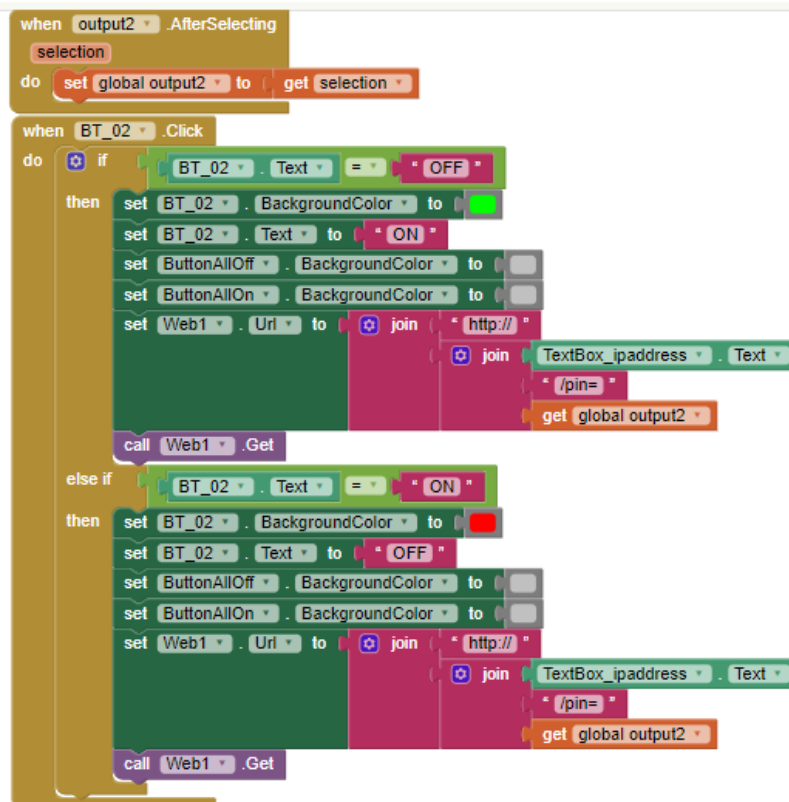
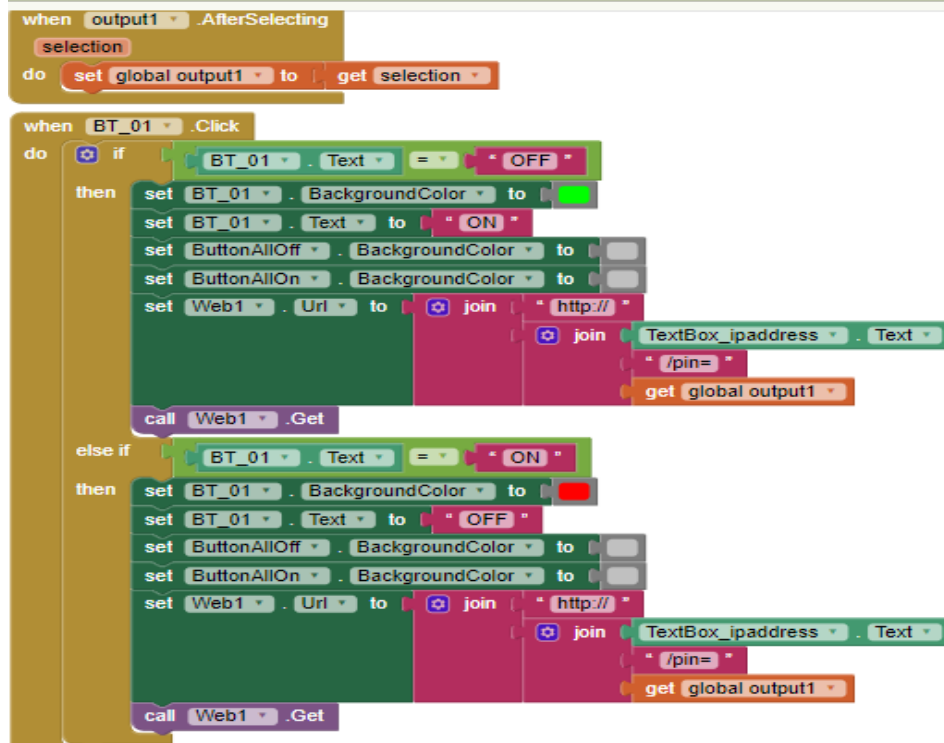
- Andrianto, H. & Darmawan, A. 2016. Arduino Belajar Cepat dan Pemrograman. INFORMATIKA : Bandung.
- Arduino,[online]. <https://www.arduino.cc/en/Main/ArduinoBoardUno>. [Diakses Agustus 2017]
- Ichwan,M., Husada, M.G. & Iqbal,M. 2013. Pembangunan Prototipe Sistem Pengendalian Perlatan Listrik Pada Platform Android. Jurnal informatika 4(1):13.(Online)<http://lib.itenas.ac.id/kti/wp-content/uploads/2013/10/Jurnal-No1Vol4-2.pdf> (Diakses 20 november 2015).
- Rofiq,M., Yusron. 2014. Perancangan sistem kontrol dan monitoring lampu dengan memanfaatkan teknologi bluetooth pada smartphone. Jurnal Ilmiah Teknologi dan Informasi Asia 8(1):16-22. (Online) <http://lp3m.asia.ac.id/wp-content/uploads/2015/05/6.-Jurnal-Rofiq-14-22.pdf>(Diakses 29 April 2016).
- Prasetyo, M. A. 2015. Konfigurasi ESP8266 Sebagai Client dan Access Point. (Online)<http://www.boarduino.web.id/2015/08/konfigurasi-esp8266-sebagai-client-dan.html> (Diakses 16 Februari 2016).
- Purbakawaca, R. 2015. Yuk! Kita Kenali Arduino Uno. (Online) <http://www.erekutoro.com/2015/12/yuk-kita-kenali-arduino-uno.html> (Diakses 29 April 2016).
- Purnomo, E. 2015. Dasar Mikrokontroler. (Online) <http://www.nulis-ilmu.com/2015/09/dasar-mikrokontroler.html> (Diakses 9 September 2015)
- Sinuardino. 2016. Modul Wifi ESP8266. (Online) <http://www.sinuardino.com/artikel/esp8266/> (Diakses 9 April 2016)
- Santoso, H. 2015. Panduan Praktis Arduino untuk Pemula. Elang Sakti: Trenggalek.
- Warangkiran, I., Kaunang.MT, I., Arie S.M Lumenta, S., & Arthur, M. R. (2014). Perancangan Kendali Lampu Berbasis Android. Jurnal Teknik Elektro dan

Komputer.7(3):4-8 (online) <https://ejournal.unsrat.ac.id/index.php/elekdankom/article/view/3827/3345>

LISTING PROGRAM

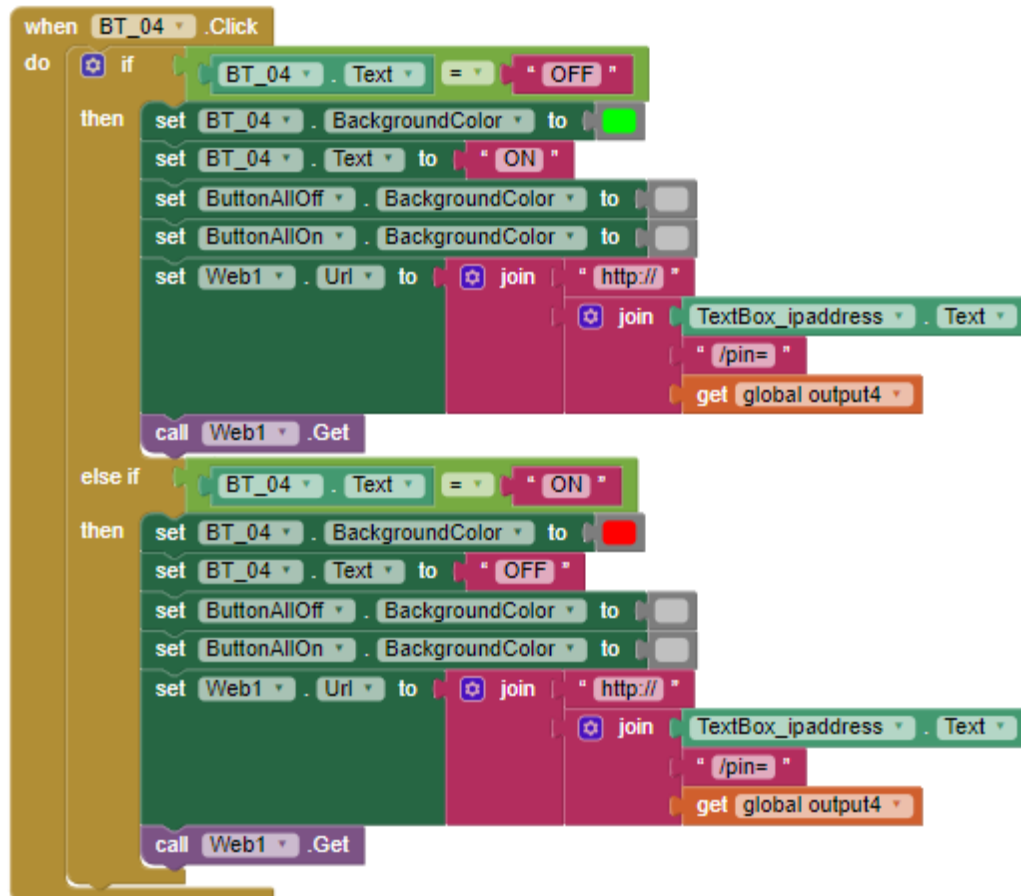
```
initialize global output1 to " 10 "
initialize global output2 to " 11 "
initialize global output3 to " 12 "
initialize global output4 to " 13 "

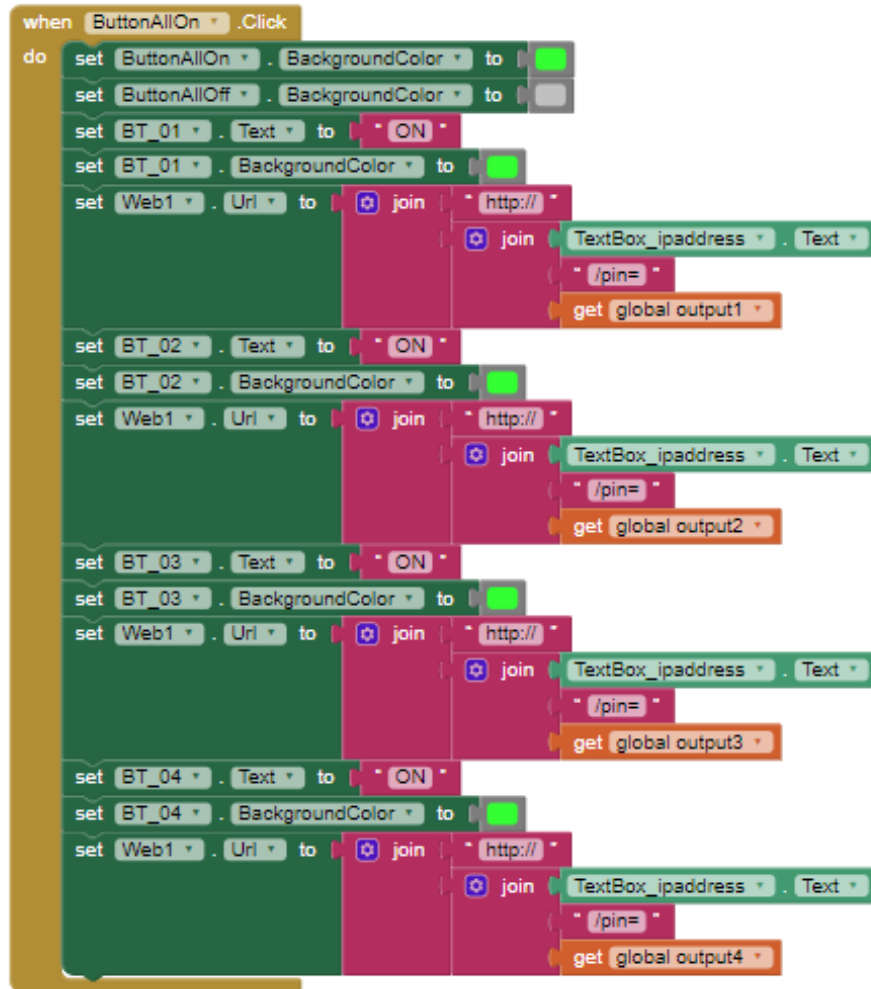
when Screen1.Initialize
do
  call Notifier1.ShowAlert
  notice "isi alamat IP dan Setting lampu "
  set TextBox_ipaddress.Text to call TinyDB1.GetValue
  tag "esp_IP"
  valueIfTagNotThere " "
  set BT_01.BackgroundColor to 
  set BT_01.Text to " OFF "
  set output1.Selection to " 10 "
  set BT_02.BackgroundColor to 
  set BT_02.Text to " OFF "
  set output2.Selection to " 11 "
  set BT_03.BackgroundColor to 
  set BT_03.Text to " OFF "
  set output3.Selection to " 12 "
  set BT_04.BackgroundColor to 
  set BT_04.Text to " OFF "
  set output4.Selection to " 13 "
```









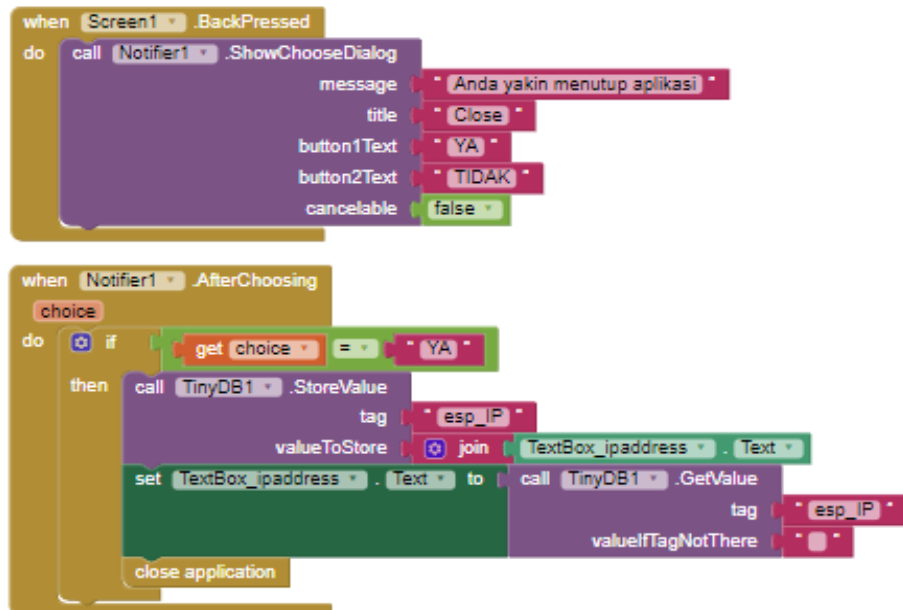

```
when output3 .AfterSelecting
  selection
do set global output3 to get selection

when BT_03 .Click
do if BT_03 .Text = " OFF "
  then
    set BT_03 . BackgroundColor to 
    set BT_03 . Text to " ON "
    set ButtonAllOff . BackgroundColor to 
    set ButtonAllOn . BackgroundColor to 
    set Web1 . Uri to join " http://"
    join TextBox_ipaddress . Text
    " /pin="
    get global output3
    call Web1 .Get
  else if BT_03 .Text = " ON "
  then
    set BT_03 . BackgroundColor to 
    set BT_03 . Text to " OFF "
    set ButtonAllOff . BackgroundColor to 
    set ButtonAllOn . BackgroundColor to 
    set Web1 . Uri to join " http://"
    join TextBox_ipaddress . Text
    " /pin="
    get global output3
    call Web1 .Get
```





```
when ButtonAllOff .Click
do
  set ButtonAllOff . BackgroundColor to 
  set ButtonAllOn . BackgroundColor to 
  set BT_01 . Text to " OFF "
  set BT_01 . BackgroundColor to 
  set Web1 . Uri to join ( " http://"
                        join (
                          join (
                            TextBox_ipaddress . Text
                            "/pin="
                            get global output1
                          )
                        )
  )
  call Web1 .Get
  set BT_02 . Text to " OFF "
  set BT_02 . BackgroundColor to 
  set Web1 . Uri to join ( " http://"
                        join (
                          join (
                            TextBox_ipaddress . Text
                            "/pin="
                            get global output2
                          )
                        )
  )
  call Web1 .Get
  set BT_03 . Text to " OFF "
  set BT_03 . BackgroundColor to 
  set Web1 . Uri to join ( " http://"
                        join (
                          join (
                            TextBox_ipaddress . Text
                            "/pin="
                            get global output3
                          )
                        )
  )
  call Web1 .Get
  set BT_04 . Text to " OFF "
  set BT_04 . BackgroundColor to 
  set Web1 . Uri to join ( " http://"
                        join (
                          join (
                            TextBox_ipaddress . Text
                            "/pin="
                            get global output4
                          )
                        )
  )
  call Web1 .Get
```



DAFTAR RIWAYAT HIDUP

CURRICULUM VITAE

<p>Personal Information</p> <p>Name : Morris Egmon Sibagariang</p> <p>Email : morris.egmon.sibagariang@gmail.com</p> <p>Place/Date of Birth : Kabanjahe, August 30th, 1991</p> <p>Contact Address : Jl. Bungawijaya kusuma Medan</p> <p>Contact Number : +6285276318164</p> <p>Marital Status : Single</p> <p>Gender : male</p>	
---	---

Formal Education

Level	School/Address	Period	Graduate Year	GPA
Bachelor (computer Science)	Universitas Sumatera Utara	2013-Present		-
Diploma-3 (Elektrical engineering)	Politehnik Negeri Medan	2009-2012	2012	3.01
Senior High School	SMU Negeri 1, Berastagi	2006-2009	2009	-
Junior High School	SMP Negeri 1, Berastagi	2003-2006	2006	-

B-2

Elmentary School	SD Letjen Jamin Ginting, Berastagi	1998-2003	2003	-
------------------	---------------------------------------	-----------	------	---

Foreign Language

Language	Level
English	Intermediate

Technical Skills

Technical Specification	Skills Specification
Programming Language	C,C++, Visual Basic,assembly
Development Tool	Microsoft Visual Basic, turbo C++,edsim51
Operating System	Windows
Simulator Tools	Multisim, PLC simulator,Eagle
Miscellaneous	Adobe Photoshop, Microsoft office

Projects Undertaken During Study:

Project Name	Design and Development Walking Tool for the Blind based MICROCONTROLLER AVR
Project Owner	Politeknik Negeri Medan, Indonesia
Duration	August 2012 – october 2012
Platform & Skills	Code vision AVR, bahasa C
Description	The purpose of this project is make a walking tool for the blind
Role	Design and Development
Responsibilities	make the project from begining/desaign until finish/development

Project Name	Sistem Hardwere kwh Meter
Project Owner	Politeknik Negeri Medan, Indonesia
Duration	March 2012 – april 2012
Platform & Skills	Mesin Tera/ Mesin pengukur
Description	The purpose of this project is testing the KWH meter before distributed
Role	Testing
Responsibilities	check the accuracy of kwh meter so that consumers and producers are not suffer a financial loss

Project Name	Make a digital watch using mikrokontroler AT89S51/89S52
Project Owner	Politeknik Negeri Medan, Indonesia
Duration	December 2011
Platform &	Eagle simulator,mide51,C

Skills	
Description	This project goal is make digital watch using mikrokontroler
Role	Project maker
Responsibilities	Make the project from begining/simulation until finish/development

Project Name	Make push button switch using seven segment as Display
Project Owner	Politeknik Negeri Medan, Indonesia
Duration	November 2011
Platform & Skills	Eagle simulator, edsim51, C
Description	The purpose of the project is to learn using bahasa C and implementation it to a project board
Role	Project maker
Responsibilities	Make the project from begining/simulation until finish/development

Project Name	Make running led using IC555
Project Owner	Politeknik Negeri Medan, Indonesia
Duration	April 2010
Platform & Skills	Eagle simulator
Description	The purpose of this project is make a running led
Role	Project maker
Responsibilities	Make the project from begining/simulation until finish/development

Statement of Purpose:

All the above activities have helped me in many ways so I

- Am comfortable working both in teams and individually
- Accept challenge and responsibility
- Am eager to learn new things
- Dedicated toward my work

I hereby certify that the information provided is true.

Medan , July, 2018

Morris Egmon Sibagariang