

**INTEGRASI APLIKASI TELEGRAM UNTUK *MONITORING* DAN
KEAMANAN KENDARAAN BERMOTOR BERBASIS
ARDUINO DAN OPENWRT**

SKRIPSI



disusun oleh

Dhimas Roby Satrio Nugroho

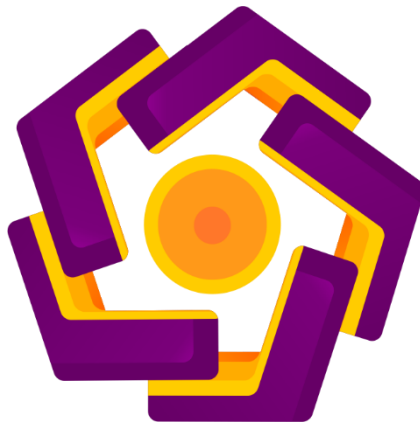
15.11.9238

**PROGRAM SARJANA
PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2018**

**INTEGRASI APLIKASI TELEGRAM UNTUK *MONITORING* DAN
KEAMANAN KENDARAAN BERMOTOR BERBASIS
ARDUINO DAN OPENWRT**

SKRIPSI

untuk memenuhi sebagian persyaratan
mencapai gelar Sarjana
pada Program Studi Informatika



disusun oleh

Dhimas Roby Satrio Nugroho

15.11.9238

**PROGRAM SARJANA
PROGRAM STUDI INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2018**

PERSETUJUAN

SKRIPSI

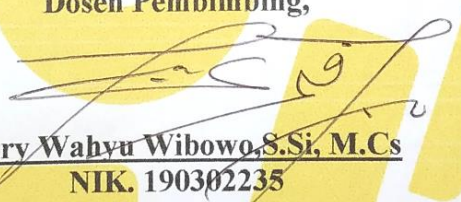
INTEGRASI APLIKASI TELEGRAM UNTUK *MONITORING* DAN KEAMANAN KENDARAAN BERMOTOR BERBASIS ARDUINO DAN OPENWRT

yang dipersiapkan dan disusun oleh

Dhimas Roby Satrio Nugroho
15.11.9238

telah disetujui oleh Dosen Pembimbing Skripsi
pada tanggal 28 Maret 2018

Dosen Pembimbing,



Ferry Wahyu Wibowo, S.Si, M.Cs
NIK. 190302235

PENGESAHAN
SKRIPSI
INTEGRASI APLIKASI TELEGRAM UNTUK *MONITORING* DAN
KEAMANAN KENDARAAN BERMOTOR BERBASIS
ARDUINO DAN OPENWRT

yang disusun oleh

Dhimas Roby Satrio Nugroho
15.11.9238

telah dipertahankan di depan Dewan Penguji
pada tanggal 21 September 2018

Susunan Dewan Penguji

Nama Penguji

Tanda Tangan

Hastari Utama, M.Cs.
NIK. 190302230

Ferry Wahyu Wibowo, S.Si, M.Cs.
NIK. 190302235

Rizqi Sukma Kharisma, M.Kom.
NIK. 190302215

Skripsi ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana Komputer
Tanggal 21 Oktober 2018

DEKAN FAKULTAS ILMU KOMPUTER



Krisnawati, S.Si, M.T.
NIK. 190302038

PERNYATAAN

Saya yang bertandatangan dibawah ini menyatakan bahwa, skripsi ini merupakan karya saya sendiri (ASLI), dan isi dalam skripsi ini tidak terdapat karya yang pernah diajukan oleh orang lain untuk memperoleh gelar akademis di suatu institusi pendidikan tinggi manapun, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis dan/atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Segala sesuatu yang terkait dengan naskah dan karya yang telah dibuat adalah menjadi tanggungjawab saya pribadi.

Yogyakarta, Oktober 2018



Dhimas Roby Satrio Nugroho

NIM. 15.11.9238

MOTTO

“Selesaikan apa yang sudah Anda mulai, hidup di dunia hanya berjalan sekali.
Apapun itu, Semua Akan dipertanggungjawabkan kelak di hari akhir”

Dhimas Roby Satrio Nugroho

PERSEMBAHAN



Dengan Rahmat Allah Yang Maha Pengasih Lagi Maha Penyayang, Berkat Rahmat dan Hidayah nya saya bisa menyelesaikan Tugas Akhir Skripsi ini.

Selain itu saya berterimakasih kepada :

1. Ibu Saya, yang selalu sabar dengan curahan hati saya ketika jenuh dengan kehidupan ini.
2. Ibu Saya, yang tak henti-henti mendukung saya untuk lekas menyelesaikan perkuliahan ini dengan lancar.
3. Ibu Saya, yang membangunkan saya dengan sulitnya demi masa depan cerah, kehabisan kalimat untuk menjelaskan betapa bersyukur saya.
4. Ayah Saya, yang selalu memberi dukungan dan semangat, bahkan segalanya untuk saya melewati segala proses kehidupan ini, ATM Pribadi saya tentunya.
5. Kedua Adik kandung saya yang menjadi pelipur lara dikala jenuh saat melewati proses kuliah dan carut marut tugasnya.
6. Dosen Pembimbing saya, Pak Ferry Wahyu Wibowo, yang senantiasa cepat tanggap ketika saya ingin sekedar berkonsultasi.
7. Teman-teman saya di keluarga besar 'Mac Cring' yaitu Zulfi yang susah bangun, Ipul yang tambah buncit, Giga yang bangun sore, Fajar yang senang *ngegame*, Isna yang ga keliatan, Ryan yang belanja mulu gak ajak-ajak, Faqih yang ngopi terus, Bagus yang kesiangan terus, Idea yang suka bawa jajan, Ferdi yang akhirnya skripsi juga, Arif yang pesta selalu, dan Hartsa yang harus tetap semangat menyelesaikan kuliah.
8. Teman sepermainan dari lama, Via, yang berisik mulu kalo soal skripsi harus cepet diselesain, udah selesai nih, cepat menyusul ya!
9. Lotek Tami, Kopi Paijo, dan Magelangan AA yang menjadi amunisi saya selama mengerjakan skripsi ini.
10. Perawat dan Dokter Rumah Sakit Hermina, yang sempat merawat saya ketika tumbang melawan arus revisi.
11. Semua teman-teman yang maaf sekali tak bisa saya sebutkan satu persatu karena keterbatasan tempat, karena banyak dari kalian sangat membantu dan mendukung saya dalam proses pengerjaan skripsi ini.

KATA PENGANTAR

Puji dan syukur penulis panjatkan ke hadirat Allah subhanahu wa ta'ala yang telah melimpahkan kasih dan sayang-Nya kepada kita, sehingga penulis bisa menyelesaikan skripsi dengan tepat waktu, yang kami beri Judul “Integrasi Aplikasi Telegram Untuk *Monitoring* dan Keamanan Kendaraan Bermotor Berbasis Arduino Dan Openwrt”.

Saya mengharapkan semoga skripsi saya ini dapat bermanfaat dan berguna bagi kemajuan ilmu pada umumnya dan kemajuan bidang pendidikan pada khususnya. Saya mengucapkan terima kasih kepada semua pihak yang sudah saya sebutkan sebelumnya maupun yang tidak bisa saya sebutkan satu persatu dalam proses pembuatan skripsi ini.

Saya menyadari kalau dalam menyusun skripsi ini masih jauh dari kata sempurna. Oleh sebab itu dengan hati yang terbuka, saya mengharapkan kritik serta saran yang membangun guna kesempurnaan skripsi ini. Semoga makalah ini bermanfaat bagi kita semua. Amin.

Yogyakarta, Oktober 2018

Penulis,

Dhimas Roby Satrio Nugroho

DAFTAR ISI

JUDUL	ii
PERSETUJUAN	
PENGESAHAN	iii
PERNYATAAN.....	
MOTTO	v
PERSEMBAHAN	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN	xiv
INTISARI.....	xv
ABSTRACT	xvi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Batasan Masalah.....	3
1.4. Maksud dan Tujuan.....	4
1.5. Metodologi Penelitian	4
1.1. Studi Literatur	5
1.2. Kepustakaan	5
1.3. Metode Perancangan	5
1.4. Metode Testing.....	5
1.6. Sistematika Laporan.....	5

BAB II LANDASAN TEORI	7
2.1. Tinjauan Pustaka	7
2.2. Dasar Teori.....	9
2.2.1. Pengertian Mikrokontroler	9
2.2.2. Pengertian Arduino Uno	11
2.2.3. Pengertian Software Arduino	15
2.2.4. Pengertian Router.....	16
2.2.5. Pengertian OpenWRT	17
2.2.6. Telegram	19
2.2.7. Modem	20
2.2.8. GPS Module GPS NEO6MV2.....	20
2.2.9. GL.iNet AR-150.....	21
2.2.10. Buzzer	21
2.2.11. Modul Relay.....	21
2.2.12. Flowchart	23
BAB III METODE PENELITIAN.....	26
3.1. Objek Penelitian	26
3.2. Jenis Penelitian.....	26
3.3. Alat dan Bahan Penelitian.....	26
3.3.1. Perangkat Keras	26
3.3.2. Perangkat Lunak.....	34
3.4. Alur Penelitian	36
3.5. Analisis Data	37
3.5.1. Rumusan Masalah	38
3.5.2. Studi Literatur dan Kepustakaan.....	38
3.5.3. Persiapan Alat	38
3.5.4. Perancangan Alat	38
3.5.5. Uji Fungsional Rangkaian Mikrokontroler	39
3.5.6. Uji Relay	39
3.5.7. Uji Buzzer	39

3.5.8. Uji Modul GPS.....	40
3.5.9. Uji Kinerja Mikrokontroler	40
3.5.10. Uji Fungsional Perangkat Pengendali Mikrokontroler	40
3.5.11. Uji Router.....	40
3.5.12. Uji Modem	41
3.5.13. Uji Kinerja Rangkaian Sistem.....	41
3.5.14. Kesimpulan	41
3.6. Rancangan Sistem	41
3.6.1. Flowchart Sistem.....	41
3.6.2. Perancangan Hardware.....	42
BAB IV HASIL DAN PEMBAHASAN	43
4.1. Alur Pembuatan Sistem Kendali Peralatan Elektronika.....	43
4.2. Pembuatan Produk	45
4.2.1. Pemasangan Komponen Elektronik	45
4.2.2. Pembuatan Program untuk Arduino.....	48
4.2.3. Pemasangan Komponen Pengendali Perangkat Elektronik	58
4.2.4. Rangkaian Komponen Elektronik	65
4.3. Pengujian Rangkaian Sistem.....	65
4.4. Pengemasan Produk	67
BAB V KESIMPULAN.....	69
5.1. Kesimpulan	69
5.2. Saran.....	69
DAFTAR PUSTAKA	71
LAMPIRAN.....	74

DAFTAR TABEL

Tabel 2.1 Spesifikasi Arduino.....	14
Tabel 2.2 Simbol <i>Flowchart</i>	24
Tabel 3.1 Spesifikasi <i>hardware (laptop)</i> untuk perancangan	27
Tabel 3.2 Spesifikasi Komponen Untuk Perancangan.....	27
Tabel 3.3 Spesifikasi <i>Smartphone</i> Untuk Implementasi	32
Tabel 3.4 Spesifikasi <i>Router</i> Untuk Implementasi	32
Tabel 4.1 Jalur Pin Modul GPS.....	45
Tabel 4.2 Jalur Pin Buzzer	46
Tabel 4.3 Jalur Pin Modul Relay.....	47
Tabel 4.4 Daftar Perintah Mikrokontroler	66

DAFTAR GAMBAR

Gambar 2.1 <i>Module Arduino Uno</i>	12
Gambar 2.2 Jendela Aplikasi Arduino IDE	16
Gambar 3.1 Arduino Uno R3	28
Gambar 3.2 Kabel Usb Tipe B	29
Gambar 3.3 Kabel Usb <i>Micro</i> Tipe B	29
Gambar 3.4 Perangkat Usb Hub.....	30
Gambar 3.5 Piezo Buzzer.....	30
Gambar 3.6 Modul Relay	31
Gambar 3.7 Modul GPS Neo6mv2	32
Gambar 3.8 Alur Penelitian.....	37
Gambar 3.9 Desain Perangkat Keras	42
Gambar 4.1 Alur Pembuatan Sistem Kendali Peralatan Elektronika.....	43
Gambar 4.2 Pemasangan Modul Gps Neo6mv2	46
Gambar 4.3 Pemasangan Buzzer.....	47
Gambar 4.4 Pemasangan Modul Relay 2 <i>Channel</i>	48
Gambar 4.5 Baris Variabel Pada Program	49
Gambar 4.6 Baris Program Inisialisasi Modul GPS	49
Gambar 4.7 Baris Program <i>Void Setup</i>	50
Gambar 4.8 Baris Perintah Menyalakan Stop Kontak Listrik.....	51
Gambar 4.9 Baris Perintah Mematikan Stop Kontak Listrik	52
Gambar 4.10 Baris Perintah Menyalakan Mesin Kendaraan	53
Gambar 4.11 Baris Perintah Mematikan Mesin Kendaraan.....	54

Gambar 4.12 Baris Perintah Mendapatkan Lokasi Kendaraan	55
Gambar 4.13 Baris Perintah Membunyikan Alarm.....	56
Gambar 4.14 Pengecekan Baris Kode Program	56
Gambar 4.15 Hasil <i>Upload</i> Program.....	57
Gambar 4.16 Tampilan Utama Openwrt.....	58
Gambar 4.17 Tampilan Submenu Network	59
Gambar 4.18 Registrasi Bot Chat Pada Telegram	60
Gambar 4.19 Tampilan Bot Chat Pada Telegram	64
Gambar 4.20 Rangkaian Perangkat Elektronik.....	65
Gambar 4.21 Jendela Tampilan Aplikasi Telegram.....	67
Gambar 4.22 Pengemasan Alat Tampak Dalam	68
Gambar 4.23 Pengemasan Alat Tampak Luar	68

DAFTAR LAMPIRAN

Lampiran A <i>Flowchart</i> Sistem.....	1
---	---

INTISARI

Kendaraan bermotor merupakan barang yang berharga dan rawan kejahatan seperti pencurian kendaraan bermotor. Penggunaan alat keamanan seperti remot untuk kendaraan yang menyatu dengan kunci motor menjadi solusi untuk pencegahan keamanan. Namun seringkali kehilangan kunci memunculkan masalah baru, sehingga diperlukan sistem yang lebih mudah dalam penggunaannya dimana pengguna tidak perlu membawa alat untuk sekedar menyalakan kendaraan.

Sistem Monitoring dan Keamanan kendaraan berbasis mikrokontroler Arduino dan router OpenWRT ini menyediakan fitur keamanan yang akan mengawasi dan melaporkan kendaraan kondisi kendaraan bermotor kepada pengguna melalui aplikasi Telegram. Sistem ini dapat melaporkan secara realtime tentang status mesin motor, status stop kontak kendaraan, dan melaporkan lokasi kendaraan berdasarkan GPS. Selain itu terdapat juga fitur Answer Back dimana fitur ini hampir sama seperti remot alarm. Status kendaraan sekaligus pengendalian bisa dilakukan melalui aplikasi chatting Telegram pada perangkat smartphone.

Akan menjadi sebuah kemudahan karena aplikasi Telegram dapat diinstal di berbagai perangkat seperti Android dan iOS. Cukup dengan mengirimkan perintah melalui chat, kemudian status kendaraan maupun pengontrolan dilakukan secara realtime dari sistem yang terdapat pada kendaraan

Kata Kunci : Arduino, Telegram, Openwrt, Keamanan, dan Kendaraan.

ABSTRACT

Motorized vehicles are valuable and prone to crime such as theft of motorized vehicles. The use of security devices such as remote control for vehicles that are integrated with the key of the motor becomes a solution for security prevention. But often losing the key raises new problems, so a system that is easier to use is needed where the user does not need to bring a tool to just turn on the vehicle.

The vehicle monitoring and security system based on the Arduino microcontroller and the OpenWRT router provides security features that will monitor and report the vehicle's condition to the user via the Telegram application. This system can report in realtime about the status of the motor engine, the status of the vehicle's socket, and report the location of the vehicle based on GPS. In addition there is also the Answer Back feature where this feature is almost the same as remote alarm. Vehicle status as well as control can be done through the Telegram chat application on a smartphone device.

It will be a convenience because the Telegram application can be installed on various devices such as Android and iOS. Enough to send commands via chat, then the status of the vehicle or control is carried out in real time from the system contained in the vehicle

Keywords: *Arduino, Telegram, Openwrt, Security, and Vehicle.*

BAB I

PENDAHULUAN

1.1. Latar Belakang

Kemajuan teknologi adalah sesuatu yang tidak bisa dihindari dalam kehidupan ini, karena kemajuan teknologi akan berjalan sesuai dengan kemajuan ilmu pengetahuan. Setiap inovasi diciptakan untuk memberikan manfaat positif bagi kehidupan manusia. Teknologi juga memberikan banyak kemudahan, serta sebagai cara baru dalam melakukan aktivitas manusia. Manusia juga sudah menikmati banyak manfaat yang dibawa oleh inovasi-inovasi teknologi yang telah dihasilkan dalam dekade terakhir ini.[1]

Menurut data yang dikutip dari Statistik Pencurian Kendaraan Bermotor oleh Badan Pusat Statistik Indonesia, peningkatan populasi kendaraan di Indonesia yang mengindikasikan bahwa kendaraan bermotor mulai menjadi kebutuhan primer bagi masyarakat, ternyata berbanding lurus dengan populasi kendaraan bermotor yang terus bertambah, angka kasus kriminalitas terhadap kendaraan bermotor juga mengalami peningkatan tiap tahunnya, pada tahun 2016 tercatat 41.816 kasus, pada tahun 2017 tercatat 42.506 kasus, dan pada 2018 triwulan pertama tercatat 11.008 kasus kriminalitas terhadap kendaraan bermotor terutama pencurian kendaraan bermotor. [2]

Berdasarkan dari permasalahan tersebut, maka perlu dibuatnya sistem pengamanan yang lebih efektif dan lebih aman guna mencegah terjadinya pencurian, selain dari alarm yang merupakan indikator dari pengeras suara yang

terdapat pada kendaraan, namun kendaraan juga perlu mengirimkan sebuah pesan yang memiliki jangkauan cukup luas guna memberikan informasi kepada pemilik kendaraan dan membuat kendaraan tidak bisa dinyalakan. Pada penelitian ini dibuatlah sebuah alat pengamanan kendaraan bermotor yang menggunakan teknik-teknik dasar pada sistem alarm yang sudah ada dengan media informasi jarak jauh melalui aplikasi *Chating* yang populer dan mudah digunakan yaitu *Telegram* sebagai pemberi informasi dan sistem keamanan sepeda motor menggunakan mikrokontroler *Arduino*. Adapun alat seperti *GPS tracker* yang dijual bebas di pasaran memiliki harga yang cukup mahal, namun perangkat *Arduino* memiliki fungsi yang bahkan lebih luas dan lebih banyak daripada *GPS tracker* namun dengan harga yang lebih murah.

Perangkat mikrokontroler *Arduino* yang digunakan sebagai perangkat keamanan kendaraan bermotor ini dapat dijadikan sebagai solusi keamanan kendaraan yang efektif, baik dari segi kemudahan penggunaan maupun dari sisi keamanan. Perangkat mikrokontroler yang digunakan terintegrasi langsung dengan perangkat *Router* dengan Sistem Operasi Linux *OpenWRT* yang berfungsi menjadi pengendali mikrokontroler agar dapat dikendalikan dari jarak jauh. Untuk membantu pemilik kendaraan mengendalikan kendaraanya melalui aplikasi *Telegram* yang dapat diinstal pada perangkat *Android* maupun *iOS* sehingga pengguna tidak perlu membawa alat khusus untuk pengendaliannya.

Sistem ini mengimplementasikan teknik *start-stop engine* yang akan dihubungkan dan dikendalikan melalui mikrokontroler *Arduino* yang dapat mengendalikan saklar kelistrikan kendaraan, saklar mesin kendaraan,

membunyikan alarm kendaraan serta mengirim lokasi kendaraan berdasarkan modul GPS (*Global Positioning System*) yang terintegrasi dengan sistem kepada pengguna melalui aplikasi *Telegram*. dimana data dikirim dari mikrokontroler melalui *Router* yang terhubung dengan internet melalui perangkat *modem* dan *API* (*Application Programming Interface*) dari *Telegram* berfungsi mengirimkan informasi kendaraan dan melakukan pengendalian modul *relay* agar sesuai dengan rancangan alat ini. GPS berfungsi untuk memonitoring sepeda motor sehingga dimana saja letak kendaraan bermotor dapat diketahui oleh pemiliknya. Dengan pembuatan alat ini diharapkan meningkatkan tingkat keamanan sepeda motor.

1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah dikemukakan sebelumnya, maka permasalahan yang dapat dirumuskan adalah bagaimana menghubungkan perangkat mikrokontroler yang berfungsi mengendalikan saklar dan menemukan lokasi kendaraan bermotor melalui perangkat *Smartphone* pengguna melalui aplikasi *Telegram* dengan memanfaatkan koneksi Internet dan *API Telegram*.

1.3. Batasan Masalah

Beberapa batasan masalah yang digunakan dalam penelitian ini adalah sebagai berikut.

1. Koneksi yang digunakan yaitu jaringan 2G, 3G ataupun 4G dan menggunakan modem GSM untuk mengkoneksikan *Router* dengan Internet.

2. Menggunakan *Router OpenWRT GL-iNet AR-150* untuk mengkoneksikan *arduino* ke jaringan luar.
3. Mikrokontroler yang digunakan menggunakan *Arduino* uno R3 sebagai *main process* dan pengolahan data.
4. Rangkaian dibangun untuk saklar on-off menggunakan *chat command* yang dikirim dari *Smartphone* melalui *API Telegram*.
5. Rangkaian digunakan untuk mengontrol saklar kontak kendaraan yang merupakan arus dari aki (*accu*) dan saklar mesin kendaraan.
6. GPS digunakan untuk melacak lokasi sepeda motor.

1.4. Maksud dan Tujuan

Maksud penelitian ini adalah :

1. Membuat rangkaian pengendali dan keamanan kendaraan bermotor jarak jauh dengan menggunakan mikrokontroler *Arduino* dan *Telegram* yang terintegrasi dengan *Router OpenWRT*.

Adapun tujuan dari perancangan alat ini yaitu :

1. Merancang alat kendali kendaraan jarak jauh menggunakan *Arduino*, *Router OpenWRT*, *Modem GSM*, GPS modul, Buzzer dan Relay.
2. Memanfaatkan perangkat *Arduino* sebagai pengendali kendaraan yang dapat digunakan dengan mudah oleh pengguna.

1.5. Metodologi Penelitian

Dalam pembuatan dan penyusunan tugas akhir ini, dilakukan langkah-langkah sebagai berikut :

1.1. Studi Literatur

Metode ini menggunakan *literatur* yang dapat dimanfaatkan seperti fasilitas internet yaitu dengan mengunjungi situs resmi maupun berbagai situs terpercaya yang berhubungan dengan mikrokontroler.

1.2. Kepustakaan

Metode membaca dan mempelajari yang akan digunakan sebagai bahan pedoman teknis penyelenggaraan kepustakaan serta mengumpulkan berbagai sumber resensi sebagai acuan dalam perancangan sistem serta penyusunan laporan.

1.3. Metode Perancangan

Metode ini merupakan perancangan dan pembuatan rangkaian yang akan digunakan.

1.4. Metode *Testing*

Metode *testing* yang dilakukan untuk mengetahui performa *hardware* dengan cara menjalankan dan melihat apa yang terjadi pada *hardware*.

1.6. Sistematika Laporan

BAB I PENDAHULUAN

Bab ini berisi tentang latar belakang permasalahan, perumusan masalah, batasan masalah, tujuan, manfaat, metode penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini berisi tentang landasan teori dan referensi penunjang dan penjelasan permasalahan yang dibahas dalam skripsi ini.

BAB III METODE PENELITIAN

Bab ini berisi runtunan pelaksanaan penelitian tentang permasalahan yang dikaji dalam skripsi ini.

BAB IV HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas mengenai perancangan sistem dan dilanjutkan dengan hasil sistem pada *hardware* yang telah dirancang.

BAB V PENUTUP

Berisi tentang kesimpulan dari pembahasan dan saran.

DAFTAR PUSTAKA

Daftar pustaka memuat keterangan buku-buku dan literatur yang menjadi acuan atau landasan dalam penulisan skripsi ini.

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

Literature Review dalam suatu penelitian adalah metode identifikasi, evaluasi dan penafsiran karya orang lain di bidang keilmuan tertentu yang menjadi masalah penelitian. Tujuan dari *Literature Review* adalah mengetahui apakah para peneliti lain telah menemukan jawaban untuk pertanyaan-pertanyaan penelitian yang kita rumuskan jika dapat menemukan jawaban pertanyaan penelitian tersebut dalam berbagai pustaka atau laporan hasil penelitian yang paling terbaru, maka kita tidak perlu melakukan penelitian yang sama [3]. Beberapa *Literature review* tersebut adalah sebagai berikut.

Penelitian ini dilakukan oleh Sumardi dalam jurnal yang berjudul “Perancangan Sistem Starter Sepeda Motor Menggunakan Aplikasi Android Berbasis Arduino Uno”, jurnal tersebut menjelaskan bagaimana cara kerja suatu rangkaian mikrokontroler terhadap keamanan sepeda motor yang dikendalikan melalui aplikasi *Android* [4]. Perbedaan dengan penelitian yang akan dibuat yaitu penggunaan *API Telegram*, Modem, Router dan tambahan GPS yang digunakan.

Penelitian ini dilakukan oleh Zul Amri Durrin Nafis dan Khafizh Khastuti dalam jurnal yang berjudul “Rancang Bangun Sistem Keamanan Kendaraan Bermotor dengan Metode Authentication dan Point Positioning menggunakan RFID berbasis Mikrokontroler”, jurnal tersebut menjelaskan bagaimana cara kerja suatu rangkaian mikrokontroler terhadap keamanan sepeda motor dengan

mengimplementasikan autentikasi menggunakan Radio Frequency Identification (RFID) untuk menghidupkan ataupun mematikan kendaraan bermotor serta metode Point Positioning akan menggunakan Modul GPS yang berfungsi untuk mengontrol keberadaan kendaraan bermotor [5]. Perbedaan dengan penelitian yang akan dibuat yaitu penggunaan *API Telegram*, Router serta autentikasi yang diimplementasikan melalui aplikasi pada *Smartphone* pemilik kendaraan.

Penelitian ini dilakukan oleh Mahfud Ichsan Adi P dalam jurnal yang berjudul “Rancangan Sistem Start Engine Dan Alarm Pada Sepeda Motor Menggunakan Arduino Uno Berbasis Android”, jurnal tersebut menjelaskan bagaimana cara kerja suatu rangkaian mikrokontroler terhadap keamanan sepeda motor yang dikendalikan melalui aplikasi *Android* [6]. Perbedaan dengan penelitian yang akan dibuat yaitu penggunaan *API Telegram* dan penggunaan Router dan Modem sebagai pengganti Bluetooth untuk sarana komunikasi data.

Penelitian ini dilakukan oleh Ari Nugroho dalam jurnal yang berjudul “Sistem Keamanan Sepeda Motor Menggunakan Arduino Berbasis Android”, jurnal tersebut menjelaskan bagaimana cara kerja suatu rangkaian mikrokontroler terhadap keamanan sepeda motor yang dikendalikan melalui aplikasi *Android* [7]. Perbedaan dengan penelitian yang akan dibuat yaitu penggunaan *API Telegram*, Router serta autentikasi yang diimplementasikan melalui aplikasi pada *Smartphone* pemilik kendaraan.

Penelitian ini dilakukan oleh Dias Utomo, Muchammad Sholeh dan Arry Avorizano dalam jurnal yang berjudul “Membangun Sistem Mobile Monitoring Keamanan Web Aplikasi Menggunakan Suricata dan Bot Telegram Channel”,

jurnal tersebut menjelaskan bagaimana cara kerja monitoring keamanan komputer server yang dikendalikan melalui aplikasi *Telegram* [8]. Perbedaan dengan penelitian yang akan dibuat yaitu mengimplementasikan konsep kerja keamanan sistem monitoring terhadap kendaraan bermotor.

2.2. Dasar Teori

2.2.1. Pengertian Mikrokontroler

Mikrokontroler adalah alat yang mengerjakan instruksi-instruksi yang diberikan, artinya bagian utama dari suatu sistem otomatis/terkomputerisasi adalah program di dalamnya yang dibuat oleh programmer. Program menginstruksikan mikrokontroler untuk melakukan jalinan yang panjang dari aksi-aksi sederhana untuk melakukan tugas yang lebih kompleks sesuai keinginan programmer [9].

Mikrokontroler adalah suatu IC dengan kepadatan yang sangat tinggi, dimana semua bagian yang diperlukan untuk suatu kontroler sudah dikemas dalam satu keping, biasanya terdiri dari CPU (*Central Processing Unit*), RAM (*Random Access Memory*), EEPROM/EPROM/PROM/ROM, I/O, Serial & Paralel, Timer, *Interrupt Controller*. Penjelasan bagian-bagian penyusun mikrokontoller standar adalah :

1. CPU (*Central Processing Unit*)

CPU adalah bagian yang paling penting dari suatu mikroprosesor, dan juga melakukan pemrosesan data.

2. RAM (*Random Access Memory*)

RAM digunakan sebagai penyimpanan data sementara yang berupa register-register. Register adalah tempat penyimpanan data yang berkaitan dengan banyak hal, misalnya variabel dalam program, keadaan *input/output*, serta pengaturan *timer/counter* dan komunikasi serial. Data pada RAM akan hilang apabila catu daya dicabut.

3. ROM (*Read Only Memory*)

ROM digunakan sebagai tempat penyimpanan program. ROM yang banyak dipakai pada mikrokontroler pada saat ini adalah flash PEROM(*Programmable Erasable ROM*), yang mirip seperti memori pada *flashdisk*, namun bedanya adalah *flash PEROM* hanya dapat dihapus dan ditulis secara sekaligus. EEPROM biasanya digunakan untuk menyimpan data yang tidak boleh hilang meski catu daya dihapus. Meski fungsinya mirip, EEPROM biasanya lebih sedikit digunakan dibanding RAM karena kecepatan akses EEPROM yang lebih lambat. Contoh penggunaannya adalah penyimpanan data password atau *setting* suatu sistem.

4. I/O (*Input/Output*) – Serial Paralel

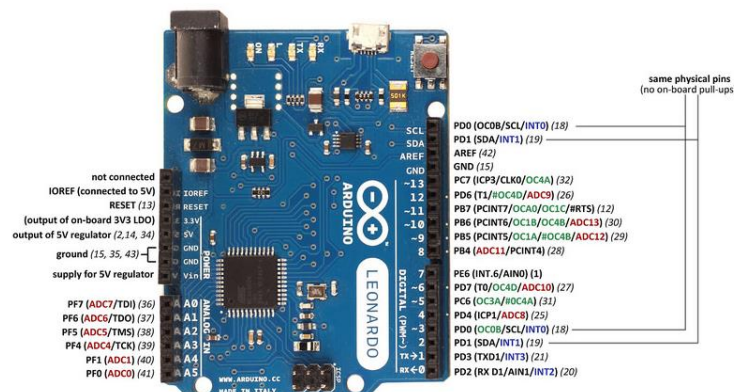
Unit ini berfungsi agar mikrokontroler dapat berkomunikasi dalam format serial atau paralel, sehingga dapat dengan mudah berkomunikasi dengan PC dan peralatan standar digital lainnya.

5. Timer

Timer berguna untuk mengatur perwaktuan pada sistem berbasis mikrokontroler, misal untuk delay. [10]

2.2.2. Pengertian Arduino Uno

Arduino Uno adalah salah satu produk berlabel Arduino yang sebenarnya adalah suatu papan elektronik yang mengandung mikrokontroler ATmega328. Arduino Uno mengandung mikroprosesor (berupa Atmel AVR) dan dilengkapi dengan *oscillator* 16 MHz, dan regulator 5 volt. Sejumlah pin tersedia di papan. Pin 0 hingga 13 digunakan untuk isyarat digital, yang hanya bernilai 0 atau 1. Pin A0-A5 digunakan untuk isyarat analog. Arduino Uno dilengkapi dengan *static random-access memory* (SRAM) berukuran 2KB untuk memegang data, *flash memory* berukuran 32KB, dan *erasable programmable read-only memory* (EEPROM) untuk menyimpan program. Sebagaimana digambarkan dalam gambar 2.1 merupakan modul dari Arduino Uno.



Gambar 2.1 Module Arduino Uno

Arduino uno sendiri memiliki beberapa kelebihan, diantaranya :

1. Murah

Papan Arduino biasanya dijual relatif murah, kisaran 125 ribu sampai 400 ribu rupiah saja, dibanding dengan *platform* mikrokontroler pro lainnya. Jika ingin lebih murah lagi, tentu bisa membuatnya sendiri dan itu sangat mungkin sekali, karena *source code* dari *software* maupun *hardware* untuk membuat sendiri perangkat Arduino tersedia lengkap di situs web Arduino.

2. Sederhana dan mudah dalam pengembangan programnya

Penggunaan Arduino tergolong mudah bagi pemula yang bahkan belum pernah menggunakan Arduino sebelumnya, namun juga cukup flexibel bagi mereka yang sudah tingkat lanjut. Pengembangan program untuk perangkat Arduino juga tidak ada kesulitan yang berarti, karena dalam program *compiler* Arduino menggunakan bahasa yang serupa dengan C yang mana merupakan bahasa pemrograman yang sudah familiar.

3. *Software* yang *Open Source*

Perangkat lunak Arduino IDE dipublikasikan sebagai *Open Source*, tersedia bagi para *programmer* yang berpengalaman guna pengembangan lebih lanjut. Arduino IDE juga menyediakan *plug-in* dan *library* yang tersedia cukup lengkap untuk berbagai perangkat maupun contoh program sederhana hingga yang rumit sekalipun,

bahkan pengguna dapat menambahkan sendiri *library* ke dalam program Arduino IDE.

4. *Hardware yang Open Source*

Papan sirkuit Arduino yang berbasis mikrokontroler *ATMEGA8*, *ATMEGA168*, *ATMEGA328*, *ATMEGA1280* dan *ATMEGA2560*, menjadikan Arduino sebuah perangkat yang bisa dikembangkan, bahkan dibuat sendiri oleh penggunanya. Semua komponen pada Arduino tersedia di pasaran, bahkan untuk *bootloader* dari Arduino itu sendiri, tersedia di dalam program Arduino IDE.

Arduino Uno berbeda dari semua papan Arduino lainnya, karena tidak menggunakan FTDI *chip driver* USB-to-Serial. Sebaliknya, fitur *ATMega16U2* hingga versi R2 diprogram sebagai konverter USB-to-serial. Pada tabel 2.1 menjelaskan spesifikasi dan fitur-fitur dari Arduino Uno R3.

Tabel 2.1 Spesifikasi Arduino Uno

Tegangan Pengoperasian	5V
Tegangan <i>input</i> yang disarankan	7-12V
Batas tegangan input	6-20V
Jumlah pin I/O digital & analog	14 & 6
Arus DC tiap pin I/O	40 mA
Arus DC untuk pin 3.3V	50 mA
Memori <i>Flash</i>	32KB
SRAM	2KB
EEPROM	1KB
<i>Clock Speed</i>	16MHz

Arduino menyediakan 20 pin I/O, yang terdiri dari 6 pin input analog dan 14 pin digital input/output. Untuk 6 pin analog sendiri bisa juga difungsikan sebagai *output* digital jika diperlukan output digital tambahan selain 14 pin yang sudah tersedia. Untuk mengubah pin analog menjadi digital cukup mengubah konfigurasi pin pada program. Dalam *board* kita bisa lihat pin digital diberi keterangan 0-13, jadi untuk menggunakan pin analog menjadi output digital, pin analog yang pada keterangan board 0-5 kita ubah menjadi pin 14-19. Dengan kata lain pin analog 0-5 berfungsi juga sebagai pin *output* digital 14-16. [11]

ATMega328 adalah mikrokontroler keluaran dari Atmel yang mempunyai arsitektur RISC (*Reduce Instruction Set Computer*) yang dimana setiap proses eksekusi data lebih cepat dari pada arsitektur CISC (*Completed Instruction Set Computer*). Mikrokontroler ini memiliki beberapa fitur antara lain:

- a. 130 macam instruksi yang hampir semuanya dieksekusi dalam satu siklus *clock*.
- b. 32 x 8-bit *register* serba guna.
- c. Kecepatan mencapai 16MIPS dengan *clock* 16MHz.
- d. 32KB *flash memory* dan pada arduino memiliki *bootloader* yang menggunakan 2KB dari *flash memory* sebagai *bootloader*
- e. Memiliki EEPROM (*Electrically Erasable Programmable Read Only Memory*) sebesar 1KB sebagai tempat penyimpanan data semi permanent karena EEPROM tetap dapat menyimpan data meskipun daya dimatikan.

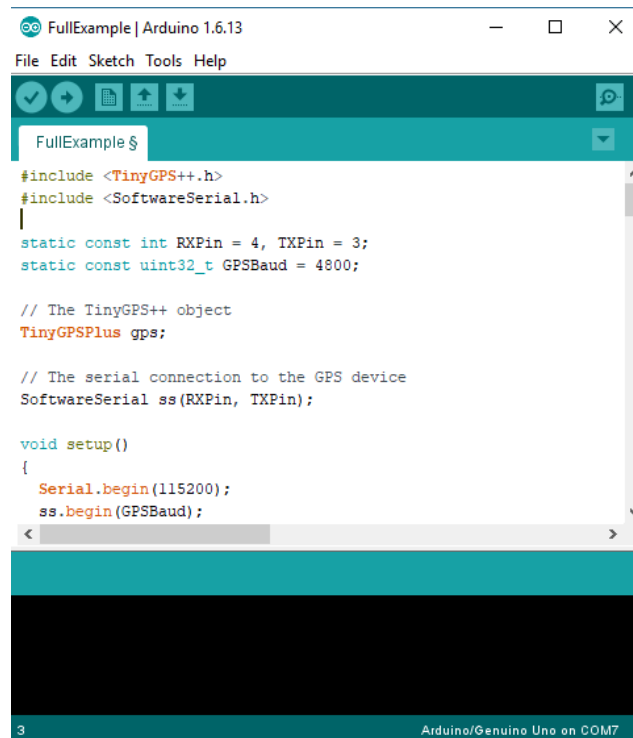
Memiliki SRAM (*Static Random Access Memory*) sebesar 2KB g. Memiliki pin I/O digital sebanyak 14 pin. Mikrokontroller ATmega 328 memiliki arsitektur Harvard, yaitu memisahkan memori untuk kode program dan memori untuk data sehingga dapat memaksimalkan kerja dan *parallelism*. [12]

2.2.3. Pengertian Software Arduino

IDE itu merupakan kependekan dari *Integrated Development Environment*, atau secara bahasa mudahnya merupakan lingkungan terintegrasi yang digunakan untuk melakukan pengembangan. Disebut sebagai lingkungan karena melalui software inilah Arduino dilakukan pemrograman untuk melakukan fungsi-fungsi yang dibenamkan melalui sintaks pemrograman.

Arduino menggunakan bahasa pemrograman sendiri yang menyerupai bahasa C. Bahasa pemrograman Arduino (Sketch) sudah dilakukan perubahan untuk memudahkan pemula dalam melakukan pemrograman dari bahasa aslinya. Sebelum dijual ke pasaran, IC mikrokontroler Arduino telah ditanamkan suatu program bernama Bootlader yang berfungsi sebagai penengah antara compiler Arduino dengan mikrokontroler. [13]

Pada gambar 2.2 merupakan tampilan jendela aplikasi Arduino IDE.



Gambar 2.2 Jendela Aplikasi Arduino IDE

2.2.4. Pengertian *Router*

Router adalah alat yang dapat menghubungkan dua atau lebih jaringan komputer yang berbeda. Beberapa vendor router yang ada di pasaran adalah : *Cisco*, *3com* dan lain-lain. Selain router yang dibuat khusus oleh vendor-vendor tersebut terdapat juga PC router yaitu sebuah PC (*Personal Computer*) yang berfungsi sebagai router dengan menjalankan sistem operasi yang memiliki kemampuan meneruskan paket dari jaringan yang satu ke jaringan yang lain. Pada dasarnya router adalah sebuah alat pada jaringan komputer yang bekerja di network layer pada OSI (*Open System Interconnection*) .

Pada Router terdapat routing table yaitu tabel yang berisi alamat-alamat jaringan yang dibutuhkan untuk menentukan tujuan dari paket-paket data yang akan dilewatkan. Beberapa hal yang dikerjakan oleh router adalah :

1. Mengirim paket data antar dua tipe jaringan fisik yang berbeda.
2. Membatasi lalu lintas paket data dari/ke beberapa alamat jaringan.
3. Mengubah alamat asal sehingga seakan-akan paket data tersebut berasal dari sebuah alamat yang berbeda dari pengirim aslinya.
4. Berfungsi sebagai BOOTP atau DHCP *server* yang memberikan alamat IP dan konfigurasi lainnya kepada seluruh komputer pada jaringan. [14]

2.2.5. Pengertian OpenWRT

OpenWRT adalah sebuah *firmware open source* untuk router. OpenWRT adalah firmware berbasis Linux, sehingga memiliki keunggulan - keunggulan yang ditawarkan Linux. OpenWRT juga memungkinkan untuk memodifikasi router secara leluasa dibanding *firmware* bawaan. Keuntungan yang didapatkan antara lain konfigurasi port secara leluasa, jangkauan transmisi bisa lebih luas, dan bisa difungsikan sebagai server. Berikut ini adalah *Firmware* OpenWrt yang telah dirilis:

1. *White Russian* (2007)

Codename dari OpenWrt yang pertama dirilis adalah *White Russian*. Versi terakhir dari *White Russian* adalah 0.9, yang dirilis pada tahun 2007.

2. *Kamikaze and Buildroot-NG* (2006-2010)

OpenWrt *Kamikaze* secara resmi dirilis pada pertengahan Oktober 2006. Dalam pengembangannya terdapat beberapa versi dari *Kamikaze*, yaitu *Kamikaze* 7.06, 7.07, 7.09, 8.09, 8.09.1, 8.09.2.

3. *Backfire* (2010-2011)

OpenWrt *Backfire* pertama rilis adalah OpenWrt 10.03 di rilis pada April 2010. *Backfire* 10.03.1 yang merupakan hasil *maintenance* di rilis pada Desember 2011.

4. *Attitude Adjustment* (2013)

Attitude Adjustment 12.09 dirilis pada 25 April 2013. Versi ini berbasis r36088. *Attitude Adjustment* merupakan versi OpenWrt yang direkomendasikan untuk perangkat dengan ROM sebesar 4MB.

5. *Barrier Breaker* (2014)

OpenWrt *Barrier Breaker* 14.07 di rilis pada 2 Oktober 2014.

6. *Chaos Calmer* (2015)

Chaos Calmer yang terakhir dirilis merupakan versi *stable release*. *Chaos Calmer* 15.06 menggunakan kernel linux 3.18. OpenWrt *Chaos Calmer* resmi dirilis pada 11 September 2015.

7. *LEDE* (2017)

Pada awal tahun 2018, OpenWrt dan LEDE *Projects* melakukan *merger* dan mengumumkan proyek bersama dibawah nama OpenWrt. OpenWrt LEDE memiliki kode rilis 17.01 yang masih menggunakan versi kernel yang sama dengan versi sebelumnya, namun mendapatkan dukungan dan peningkatan

keamanan yang lebih baik karena bergabungnya LEDE dengan OpenWrt.[15]

2.2.6. Telegram

Instant Messaging (IM) Telegram yang diluncurkan pada Agustus tahun 2013 menjadi salah satu aplikasi IM yang banyak digunakan oleh masyarakat di seluruh dunia. Kelebihan IM Telegram salah satunya adalah adanya landasan untuk menggunakan *Application Programming Interface* (API) untuk masyarakat luas. Salah satu API yang disediakan adalah fitur Bot. Implementasi Bot sudah mulai banyak digunakan, salah satu keunggulan Bot adalah keandalan untuk menyediakan data ke pengguna yang tidak terbatas oleh waktu.

Telegram menyediakan 2 bentuk API, API yang pertama adalah klien IM Telegram, yang berarti semua orang dapat menjadi pengembang klien IM Telegram jika diinginkan. Ini berarti jika seseorang ingin mengembangkan Telegram versi mereka sendiri mereka tidak harus memulai semua dari awal lagi. Telegram menyediakan *source code* yang mereka gunakan saat ini. Tipe API yang kedua adalah Telegram Bot API.

API jenis kedua ini memungkinkan siapa saja untuk membuat bot yang akan membalas semua penggunaannya jika mengirimkan pesan perintah yang dapat diterima oleh Bot tersebut. Layanan ini masih hanya tersedia bagi pengguna yang menggunakan aplikasi Telegram saja. Sehingga pengguna yang ingin menggunakan Bot harus terlebih dahulu memiliki akun Telegram. [16]

2.2.7. Modem

Modem GSM adalah jenis khusus dari modem yang menerima kartu SIM, dan mengoperasikan lebih dari berlangganan ke operator selular, seperti ponsel. Ketika modem GSM terhubung dengan komputer, ini memungkinkan komputer untuk menggunakan modem GSM untuk berkomunikasi melalui jaringan seluler, sementara ini modem GSM yang sering digunakan untuk menyediakan konektivitas *mobile internet*, banyak dari mereka juga dapat digunakan untuk mengirim dan menerima SMS. Modem GSM juga dapat menjadi perangkat modem yang berdedikasi dengan, serial USB atau sambungan *Bluetooth*, atau bisa menjadi ponsel yang menyediakan kemampuan GSM modem [17].

2.2.8. GPS Module GPS NEO6MV2

Global Positioning System (GPS) merupakan sistem yang digunakan untuk menentukan letak di permukaan Bumi dengan bantuan penyalarsan sinyal satelit. Sistem ini menggunakan 24 satelit dan 3 satelit cadangan yang mengirimkan sinyal gelombang mikro ke Bumi [18].

Modul GPS uBlox GY-NEO6MV2 berfungsi sebagai penerima GPS yang dapat mendeteksi lokasi dengan menangkap dan memproses sinyal dari satelit navigasi. Modul ini melingkupi sistem navigasi, sistem keamanan pada perangkat bergerak, akuisisi data pada sistem pemetaan medan, *location tracking*, dan sebagainya [19].

2.2.9. GL.iNet AR-150

GL.iNet merupakan perusahaan pembuat alat-alat jaringan seperti *Router*, *Switch*, *Modem*. Salah satu produk andalan GL.iNet adalah *router* jaringannya yang populer di pasaran berkat fiturnya yang lengkap, penggunaanya yang praktis dan mudah, serta harga yang bersaing. AR-150 merupakan salah satu produk GL.iNet yang berupa sebuah *wireless router*. Diperkenalkan pada publik pada bulan September 2015, AR-150 merupakan suksesor dari GL.iNet 6416 dengan pembaruan papan PCB, *firmware* yang baru dan beberapa fitur yang lebih baru dari pendahulunya [20].

2.2.10. Buzzer

Buzzer adalah sebuah komponen elektronika yang dapat dipakai untuk mengeluarkan suara, sekaligus bisa dimanfaatkan sebagai alat masukan pendeteksi ketukan. Sebagai pembangkit suara, suara yang dihasilkan oleh komponen ini tentu saja tidak sekeras yang dihasilkan oleh *loudspeaker*. Kutub positif ada di bagian tengah dan kutub negatif ada di bagian tepi. Harga komponen ini relatif murah, hanya sekitar tiga ribuan [21].

2.2.11. Modul Relay

Relay adalah komponen elektronika yang berfungsi untuk memutuskan atau menghubungkan suatu rangkaian elektronik yang satu dengan rangkaian elektronika lainnya. Pada dasarnya relay adalah saklar yang bekerja berdasarkan prinsip electromagnet yang akan bekerja apabila arus mengalir melalui kumparan,

inti besi akan menjadi magnet dan akan menarik kontak yang ada di dalam relay. Kontak dapat ditarik apabila gaya magnet dapat mengalahkan gaya pegas yang melawannya. Besarnya gaya magnet ditetapkan oleh medan magnet yang ada pada celah udara, jangkar, inti magnet, banyaknya lilitan kumparan, kuat arus yang mengalir (impedansi lilitan) dan Palawan magnet yang berada pada sirkuit magnet.

Untuk memperbesar kuat medan magnet dibentuk suatu sirkuit. Kontak atau kutub relay pada umumnya memiliki tiga jenis konstruksi dasar yaitu:

1. Bila kumparan dialiri arus listrik maka kontakannya akan menutup dan disebut sebagai kontak *Normally Open* (NO).
2. Bila kumparan dialiri arus listrik maka kontakannya akan membuka dan disebut dengan kontak *Normally Close* (NC).
3. Tukar-sambung (*Change Over/CO*), relay jenis ini mempunyai kontak tengah yang normalnya tertutup tetapi melepaskan diri dari posisi ini dan membuat kontak dengan yang lain bila relay dialiri listrik.

Relay sering digunakan dalam peralatan-peralatan elektronika dan mempunyai fungsi untuk menghubungkan dan memutuskan aliran arus listrik yang dikontrol dengan memberikan tegangan dan arus tertentu pada koilnya sehingga dengan relay dapat menghubungkan arus dan tegangan yang besar dengan arus dan tegangan yang kecil. Misal dengan tegangan label TTL dapat menghidupkan motor dengan arus dan tegangan yang besar (misal 220V). tegangan yang dibutuhkan relay bermacam-macam dari DC 6V hingga 220 VAC. Sifat – sifat relay :

- a. Impedansi kumparan, biasanya impedansi ditentukan oleh tebal kawat yang digunakan serta banyaknya lilitan.

- b. Kuat arus yang digunakan untuk menggerakkan relay, biasanya arus ini diberikan oleh pabrik. Relay dengan perlawanan kecil memerlukan arus besar sedangkan relay dengan perlawanan besar memerlukan arus yang kecil.
- c. Membutuhkan tegangan untuk menggerakkan relay.
- d. Daya yang diperlukan untuk mengoperasikan relay besarnya sama dengan nilai tegangan dikalikan arus.
- e. Banyaknya kontak-kontak jangkar dapat membuka dan menutup lebih dari satu kontak sekaligus tergantung pada kontak dan jenis relay-nya. Jarak antara kontak-kontak menentukan besarnya tegangan maksimum yang diizinkan antara kontak tersebut [22].

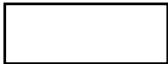
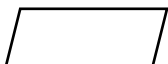
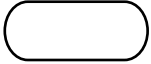
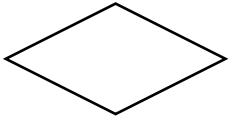

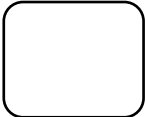
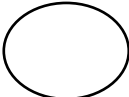
2.2.12. Flowchart

Flowchart atau diagram alir merupakan diagram yang digunakan untuk menggambarkan proses-proses yang terjadi pada sistem yang akan dikembangkan. Dengan model ini, data-data yang terlibat pada masing-masing proses dapat diidentifikasi. *Flowchart* merupakan sebuah diagram dengan simbol-simbol grafis yang menyatakan aliran algoritma atau proses yang menampilkan langkah-langkah yang disimbolkan dalam bentuk kotak, beserta urutannya dengan menghubungkan masing-masing langkah tersebut menggunakan tanda panah [23].

Tujuan utama penggunaan *flowchart* adalah untuk menyederhanakan rangkaian proses atau prosedur untuk memudahkan pemahaman pengguna terhadap

informasi tersebut. Untuk itu desain sebuah *flowchart* harus ringkas, jelas, dan logis. Tabel 2.2 berikut adalah simbol *flowchart* yang umum digunakan.

Tabel 2.2 Simbol Flowchart

Gambar	Simbol Untuk	Keterangan
	Proses / Langkah	Menyatakan kegiatan yang akan ditampilkan dalam diagram alir.
	Masukan / Keluaran Data	Digunakan untuk mewakili data masuk, atau data keluar.
	Terminasi	Menunjukkan awal atau akhir sebuah proses.
	Titik Keputusan	Proses/Langkah dimana perlu adanya keputusan atau adanya kondisi tertentu. Di titik ini selalu ada dua keluaran untuk melanjutkan aliran kondisi yang berbeda.
	Garis Alir	Menunjukkan arah aliran proses atau algoritma.
	Kontrol / Inspeksi	Menunjukkan proses / langkah dimana ada inspeksi atau pengontrolan.
	Penghubung	Keluar atau masuk dari bagian lain <i>flowchart</i> khususnya halaman yang sama.

Sterneckret menyarankan untuk membuat model diagram alir yang berbeda sesuai dengan perspektif pemakai (*managers, system analysts and clerks*) sehingga dikenal ada 4 jenis diagram alir secara umum :

- a. *Diagram Alir Dokumen*, menunjukkan kontrol dari sebuah sistem aliran dokumen.
- b. *Diagram Alir Data*, menunjukkan kontrol dari sebuah sistem aliran data.
- c. *Diagram Alir Sistem*, menunjukkan kontrol dari sebuah sistem aliran secara fisik.
- d. *Diagram Alir Program*, menunjukkan kontrol dari sebuah program dalam sebuah sistem. [24]

BAB III

METODE PENELITIAN

3.1. Objek Penelitian

Penelitian ini memanfaatkan sepeda motor sebagai simulasi dari sistem kendali kendaraan bermotor, dimana perangkat dari sistem akan dihubungkan dengan sistem kelistrikan dari sepeda motor.

3.2. Jenis Penelitian

Jenis Penelitian yang digunakan dalam pembuatan penelitian ini adalah Metode Eksperimen, yaitu dengan melakukan penelitian berupa perancangan dan pembuatan sistem secara keseluruhan serta menguji alat secara keseluruhan dengan sistem yang sudah dibuat.

3.3. Alat dan Bahan Penelitian

Pada penelitian ini, dalam pembuatan sepeda motor pintar menggunakan Arduino Uno dan Router berbasis OpenWRT yang terintegrasi dengan *API* Telegram membutuhkan alat dan bahan yaitu :

3.3.1. Perangkat Keras

Perangkat keras yang digunakan dalam penelitian pembuatan sistem skendali motor pintar ini meliputi :

1. Perangkat keras untuk perancangan dapat dilihat pada tabel 3.1 berikut

Tabel 3.1 Spesifikasi *hardware (Laptop)* untuk perancangan

No	Spesifikasi	
1.	Merk dan Seri Laptop	Acer Aspire 4750
2.	Processor	Intel® Core™ i3-2330 @2.2 GHz
3.	RAM (<i>Random Access Memory</i>)	8,00 GB
4.	Memory Type	DDR3
5.	Harddisk	500 GB
6.	VGA (<i>Video Graphic Adapter</i>)	Intel® HD Graphics 3000
7.	Network Card	Atheros

2. Perangkat keras (komponen elektronika) untuk rangkaian sistem dapat dilihat pada tabel 3.2 berikut.

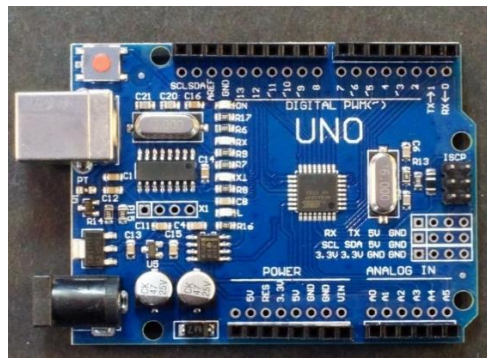
Tabel 3.2 Spesifikasi komponen untuk perancangan

No	Komponen	Spesifikasi
1.	Mikrokontroler	Arduino Uno R3 (5 Volt)
2.	Relay	Modul relay 2 <i>channel</i> (5 Volt)
3.	Kabel jumper	-+ 30 Kabel
4.	Adaptor	9 Volt
5.	Modul GPS	NEO 6mV2 + <i>Receiver Antenna</i>
6.	Penyearah Tegangan	12 Volt to 9 Volt
7.	Kabel USB	USB A to B
8.	Kabel USB	Micro USB type B
9.	USB Hub	4 Slot with USB 3.0
10.	Buzzer	Piezo Buzzer 3,5 Volt

Adapun penjelasan dari fungsi dan kegunaan dari alat-alat di atas yaitu sebagai berikut :

a. Arduino Uno R3

Papan Arduino Uno berfungsi sebagai pengendali utama dari keseluruhan sistem atau dapat disebut sebagai otak dari sistem. Rangkaian ini dilengkapi dengan beberapa port dimana port tersebut berfungsi untuk menghubungkan perangkat elektronik lainnya, selain itu papan Arduino Uno mempunyai beberapa jenis memori dengan fungsi yang berbeda-beda, salah satunya adalah untuk menyimpan kode program yang akan dijalankan. Bentuk perangkat Arduino Uno R3 dapat dilihat pada gambar 3.1.



Gambar 3.1 Arduino Uno R3

b. Kabel USB A to B

USB *Connection type B* berfungsi sebagai jembatan penghubung antara sistem minimum Arduino Uno dengan PC dan antara Sistem Arduino dengan perangkat Router, agar program dapat diisikan dari PC/laptop ke dalam rangkaian mikrokontroler, serta sebagai sarana komunikasi serial antara perangkat

mikrokontroler dengan perangkat Router. Sebagaimana digambarkan dalam gambar 3.2 merupakan bentuk kabel USB tipe B.



Gambar 3.2 Kabel USB tipe B

c. Kabel USB *Micro USB type B*

USB *Micro type B* berfungsi sebagai sumber catu daya perangkat router agar terhubung dengan sumber daya listrik, yang mana arus listrik dari baterai disalurkan menggunakan kabel melalui *port micro usb* pada router. Sebagaimana digambarkan dalam gambar 3.3 merupakan bentuk kabel USB *Micro type B*.



Gambar 3.3 Kabel USB *micro* tipe B

d. USB Hub

USB Hub berfungsi untuk membuat sebuah cabang untuk *port* USB dari Router, karena *port* yang tersedia pada Router hanya 1 buah, sedangkan sistem membutuhkan 2 port yaitu untuk modem dan untuk mikrokontroler. Sebagaimana digambarkan dalam gambar 3.4 merupakan bentuk USB Hub.



Gambar 3.4 Perangkat USB Hub

e. Piezo Buzzer

Piezo Buzzer berfungsi sebagai perangkat yang mengubah aliran listrik menjadi suara, dimana buzzer ini berperan memberikan notifikasi suara atau isyarat sebagai alarm maupun isyarat setiap perintah pada sistem. Sebagaimana digambarkan dalam gambar 3.5 merupakan bentuk Piezo Buzzer.



Gambar 3.5 Piezo Buzzer

f. Modul Relay

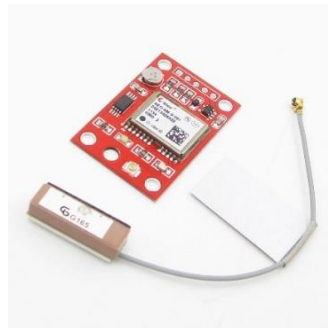
Modul Relay digunakan untuk menyambungkan Arduino ke perangkat elektronika. modul ini berfungsi sebagai saklar untuk menghidupkan dan mematikan perangkat elektronika, dalam hal ini yaitu saklar listrik dan saklar mesin pada kendaraan bermotor. Bentuk dari modul relay dapat dilihat pada gambar 3.6.



Gambar 3.6 Modul Relay

g. Modul GPS NEO6mV2

Modul GPS NEO6mV2 ini berfungsi sebagai penerima dan pengirim sinyal GPS (*Global Positioning System Receiver*) yang dapat mendeteksi navigasi. Aplikasi dari modul ini melingkupi sistem navigasi, akuisisi data pada sistem pemetaan medan dan penjejak lokasi. Gambar modul GPS dapat dilihat pada gambar 3.7.



Gambar 3.7 Modul GPS NEO6mV2

3. Perangkat keras (*smartphone*) untuk implementasi dapat dilihat pada tabel 3.3

Tabel 3.3 Spesifikasi *Smartphone* untuk Implementasi

No	Spesifikasi	
1.	Merk dan Tipe <i>Smartphone</i>	Xiaomi Redmi 5A
2.	<i>Processor</i>	Quad-core 1.4 GHz Cortex-A53
3.	Layar	720 x 1280 pixels (~296 ppi density)
4.	RAM (<i>Random Access Memory</i>)	2 GB
5.	Penyimpanan	16 GB
6.	GPU (<i>Graphic Processor Unit</i>)	Adreno 308
7.	WLAN	Wi-Fi 802.11 a/b/g/n
8.	Jaringan	GSM / EDGE / HSPA / LTE
9.	Sistem Operasi	MIUI 9 based Android 7.1.2 Nougat

4. Sedangkan Perangkat keras (*router*) untuk implementasi dapat dilihat pada tabel 3.4.

Tabel 3.4 Spesifikasi Router untuk Implementasi

No	Spesifikasi	
1.	Merk dan Type Router	GL-iNet GL-AR150
2.	<i>Processor</i>	Atheros 9331 SoC, 400MHz CPU
3.	RAM (<i>Random Access Memory</i>)	64 MB
4.	Penyimpanan	16 MB <i>flash</i>
5.	WLAN	Wi-Fi 802.11 a/b/g/n
6.	Network Interfaces Speed	150 Mbps

7.	Catu Daya	<i>Micro USB type B 5-9 Volt</i>
8.	Sistem Operasi	OpenWRT 15.05 Chaos Chalmer
9.	Lain-lain	WAN & LAN Ethernet RJ4
10.	Modem	Modem USB

Dalam menjalankan sistem yang dirancang, sebuah perangkat keras (*hardware*) memiliki kebutuhan minimal yang mencukupi, yaitu :

- a. PC (*Personal Computer*) atau *Laptop* dengan minimal processor 1 GHz (*Giga Hertz*).
- b. Minimal RAM (*Random Access Memory*) dengan kapasitas 1 GB (*Giga Byte*).
- c. Minimal Harddisk dengan ruang kosong 5 GB (*Giga Byte*).
- d. VGA (*Video Graphic Adapter*) minimal 512 MB (*Mega Byte*).

Kebutuhan minimal *hardware (smartphone device)* untuk implementasi dalam menjalankan sistem kontrol ini adalah sebagai berikut :

- a. Semua jenis handphone yang dapat diinstal aplikasi *Telegram*.
- b. Minimal RAM (*Random Access Memory*) 1 GB (*Giga Byte*).
- c. Ketersediaan ruang penyimpanan ponsel 100 MB (*Mega Byte*).
- d. Perangkat *smartphone* yang terhubung dengan koneksi internet.

Sedangkan untuk kebutuhan minimal *hardware (router device)* untuk implementasi dalam menjalankan sistem kontrol ini adalah sebagai berikut :

- a. Minimal RAM (*Random Access Memory*) 32 MB (*Mega Byte*).
- b. Ketersediaan ruang penyimpanan *flash* 8 MB (*Mega Byte*).

- c. Memiliki Port USB untuk komunikasi data dengan mikrokontroler dan modem.
- d. Dapat terkoneksi dengan internet maupun jaringan lokal.

3.3.2. Perangkat Lunak

Perangkat lunak yang digunakan dalam penelitian ini adalah sebagai berikut :

1. Perangkat lunak untuk perancangan

Perangkat lunak untuk perancangan disini merupakan perangkat lunak yang digunakan oleh PC/Laptop yang menggunakan sistem operasi *Windows 10 Pro*.

2. Perangkat lunak untuk implementasi

Perangkat lunak untuk implementasi dalam penelitian ini yaitu :

a. Sistem Operasi *Smartphone*

Sistem operasi pada ponsel yang dapat mendukung aplikasi *Telegram* sebagaimana kebutuhan minimal perangkat untuk menjalankannya adalah menggunakan sistem operasi iOS 7 atau *Android 4.4 KitKat*.

b. Sistem Operasi Router

Sistem operasi pada router yang digunakan merupakan sistem operasi berbasis linux yaitu *OpenWRT* dengan menggunakan *distro* “Pulpstone” yang mana merupakan turunan dari versi aslinya yaitu *OpenWRT 17.01 LEDE*.

c. Aplikasi Telegram

Aplikasi Telegram yang terinstal pada perangkat *smartphone* yang berfungsi sebagai aplikasi pengendali sistem yang akan digunakan oleh pemilik kendaraan, dimana setiap perintah dan notifikasi akan dilakukan melalui aplikasi ini menggunakan fitur *Chatbot* yang terhubung dengan *API* yang telah diatur pada sistem kendaraan.

d. *API* Telegram

API (Application Programming Interface) berfungsi agar perangkat router dapat berkomunikasi dengan pengguna menggunakan layanan Telegram. *API* telah dikonfigurasi dan berperan sebagai robot yang dapat melakukan komunikasi dengan pengguna. *API* juga berfungsi membuat router memiliki sebuah akun bot *chat* yang berisi daftar perintah dan respon yang didapatkan dari sistem apabila berinteraksi dengan pengguna.

e. *Software* Arduino IDE

Software Arduino IDE berfungsi sebagai *text editor* dalam menulis baris perintah sekaligus sebagai *compiler* menjadi kode biner yang akan di *upload* ke dalam memori mikrokontroler. Selain itu, *software* arduino juga berfungsi untuk menghubungkan PC/Laptop dan board arduino melalui koneksi USB.

f. *Software* Proteus

Software Proteus digunakan untuk membuat rancangan dari perangkat keras ke dalam bentuk visual, sekaligus membuat simulasi sistem sebelum di bangun dan di rakit menjadi sebuah alat.

g. *Software PuTTY*

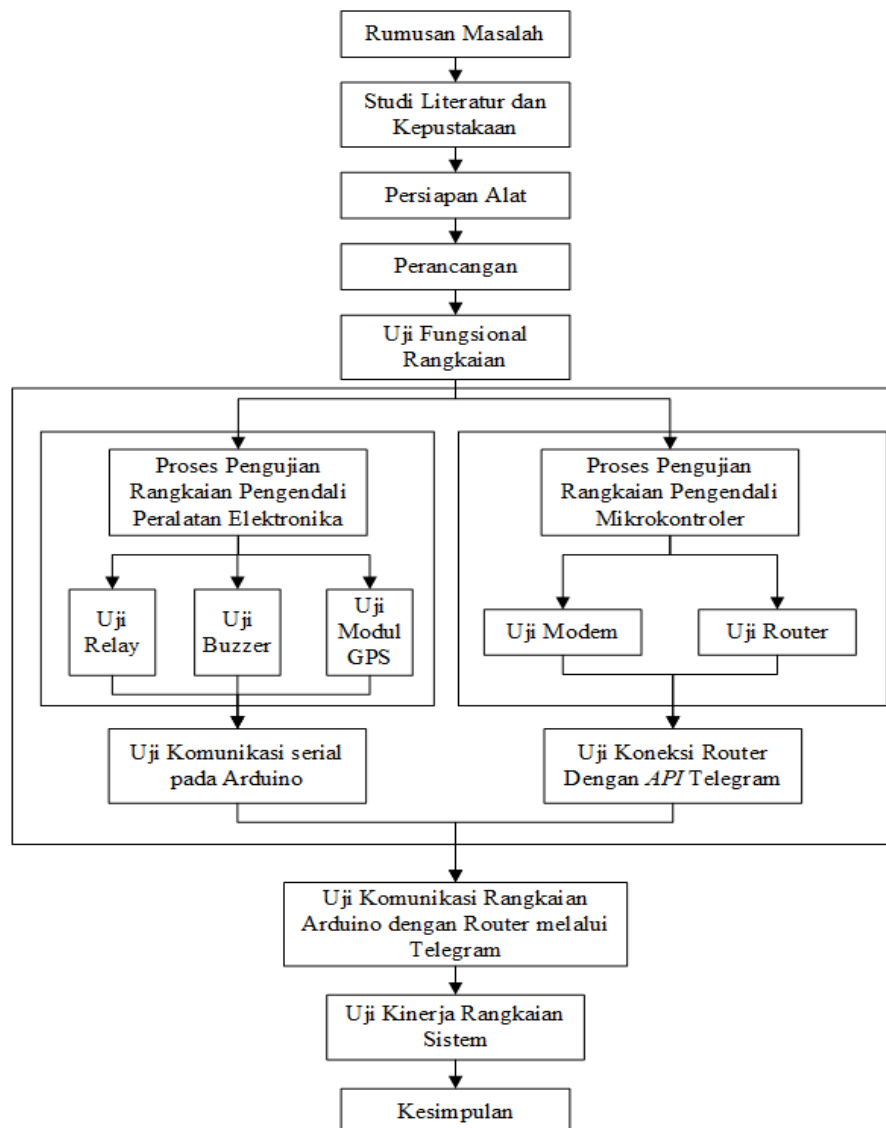
Software PuTTY berfungsi sebagai aplikasi penghubung antara perangkat Router dengan perangkat Laptop/PC agar bisa menulis baris perintah ke dalam perangkat router melalui protokol SSH (*Secure Shell*) dimana PuTTY melakukan remot koneksi agar bisa berkomunikasi serta mengkonfigurasi perangkat router melalui kabel LAN maupun melalui WiFi dari perangkat router itu sendiri.

h. *Software WinSCP*

Software WinSCP berfungsi sebagai aplikasi penghubung antara perangkat Router dengan perangkat Laptop/PC agar bisa meng-*upload* file yang dibutuhkan untuk konfigurasi ke dalam perangkat router dimana WinSCP menggunakan protokol SCP (*Secure Copy*) yaitu protokol yang sama dengan SSH yang melalui *port* 22.

3.4. Alur Penelitian

Alur penelitian dalam penulisan skripsi ini menjelaskan mengenai tahapan atau prosedur penelitian untuk merancang alat kendaraan bermotor pintar. Adapun *Flowchart* alur penelitian dapat dilihat pada gambar 3.8



Gambar 3.8 Alur Penelitian

3.5. Analisis Data

Analisis data adalah proses mengatur urutan data, mengorganisasikannya ke dalam suatu pola, kategori dan uraian dasar. Fungsinya memberikan gambaran tentang beberapa pentingnya kedudukan analisis data, dilihat dari segi tujuan penelitian. Teknik analisis data yang digunakan dalam penelitian ini menggunakan langkah-langkah sebagai berikut.

3.5.1. Rumusan Masalah

Pada Alur ini dijelaskan Bagaimana cara membuat sistem keamanan kendaraan pintar menggunakan Arduino dan OpenWRT menggunakan Telegram.

3.5.2. Studi Literatur dan Kepustakaan

Mencari sumber informasi dan referensi untuk penggunaan mikrokontroler Arduino Uno, modul GPS NEO6Mv2, Buzzer, Relay, Router OpenWRT, dan modem melalui internet serta buku atau jurnal ilmiah mengenai topik yang berkaitan dengan hal tersebut.

3.5.3. Persiapan Alat

Mempersiapkan alat- alat yang diperlukan untuk membuat sistem keamanan kendaraan pintar seperti Arduino Uno, Router OpenWRT, Modem, Modul GPS NEO6Mv2, *relay*, *buzzer*, kabel *jumper*, USB *Hub*, kabel *Power Supply* dan lain sebagainya.

3.5.4. Perancangan Alat

Sebelum menerapkan rangkaian pada mikrokontroler, maka sebaiknya dilakukan perancangan terlebih dahulu. Pada alur penelitian ini, penulis merancang rangkaian mikrokontroler menggunakan *software* bernama *Proteus ISIS*, dimana perangkat lunak dibuat khusus untuk merancang dan mensimulasikan rangkaian mikrokontroler.

3.5.5. Uji Fungsional Rangkaian Mikrokontroler

Setelah melakukan penerapan rangkaian mikrokontroler, maka langkah selanjutnya yang harus dilakukan adalah menguji rangkaian mikrokontroler tersebut untuk memastikan apakah rangkaian mikrokontroler tersebut berjalan sesuai dengan apa yang diharapkan oleh penulis.

3.5.6. Uji Relay

Relay merupakan komponen terpenting dalam penelitian ini, karena relay berfungsi untuk mengontrol peralatan elektronika pada kendaraan bermotor, seperti memutus arus listrik dari aki ke komponen kendaraan lain. Relay menggunakan prinsip elektromagnetik untuk menggenrakan kontak saklar, sehingga dengan arus listrik yang kecil (*low power*) dapat menghantarkan listrik yang bertegangan tinggi.

3.5.7. Uji Buzzer

Buzzer merupakan komponen yang berperan sebagai indikator dan alarm dari sistem, dimana setiap perintah dari sistem akan mengirimkan sinyal berupa suara yang dikeluarkan oleh buzzer ini. Buzzer mengubah arus listrik menjadi gelombang suara yang dapat diatur melalui perangkat mikrokontroler.

3.5.8. Uji Modul GPS

Fungsi utama dari modul GPS NEO6Mv2 adalah sebagai penerima sekaligus pengirim sinyal GPS (*Global Positioning System*) yang dapat mendeteksi lokasi dengan menangkap dan memproses sinyal dari satelit.

3.5.9. Uji Kinerja Mikrokontroler

Pengujian ini dilakukan untuk mengetahui kondisi pada mikrokontroler sehingga dapat dipastikan bahwa rangkaian mikrokontroler dapat berfungsi sesuai dengan program yang telah dibuat dan tidak ada kesalahan pada rangkaian sistem mikrokontroler.

3.5.10. Uji Fungsional Perangkat Pengendali Mikrokontroler

Pengujian fungsi perangkat pengendali mikrokontroler diperlukan untuk memastikan apakah perangkat mikrokontroler dapat dikendalikan sesuai perintah yang telah dikonfigurasi.

3.5.11. Uji Router

Router berperan sangat penting sebagai pengendali utama mikrokontroler, dimana router akan berkomunikasi dengan perangkat mikrokontroler melalui komunikasi serial menggunakan kabel yang menghubungkan router dan mikrokontroler.

3.5.12. Uji Modem

Perangkat modem berfungsi menghubungkan perangkat router dengan jaringan internet agar bisa berkomunikasi dengan API (*Application Programming Interface*) dari Telegram.

3.5.13. Uji Kinerja Rangkaian Sistem

Pengujian router diperlukan untuk menguji apakah perintah yang dikirimkan dari pengguna melalui aplikasi Telegram dapat diterima oleh perangkat router dan diteruskan menuju perangkat mikrokontroler dapat berfungsi sebagaimana mestinya.

3.5.14. Kesimpulan

Memberikan hasil kesimpulan alur penelitian mengenai sistem keamanan kendaraan bermotor pintar berbasis Arduino dan OpenWRT melalui aplikasi Telegram. Kesimpulan ini ditinjau mmulai dari perancangan alat hingga uji kinerja rangkaian sistem.

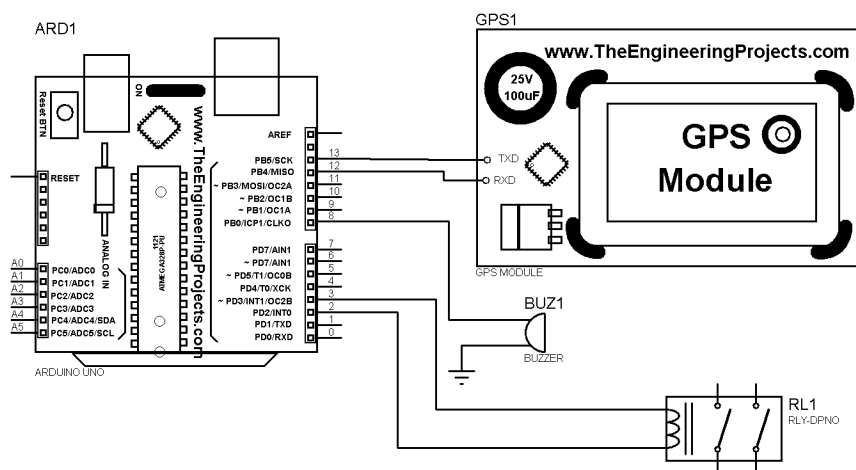
3.6. Rancangan Sistem

3.6.1. Flowchart Sistem

Flowchart sistem memberikan gambaran bagaimana sistem bekerja, sebagaimana digambarkan dalam lampiran A yang merupakan gambar *flowchart* sistem terlampir pada lembar lampiran.

3.6.2. Perancangan *Hardware*

Konsep awal produk adalah melalui perancangan sistem yang akan diimplementasikan terhadap mikrokontroler Arduino Uno, Relay GPS NEO6Mv2 dan Buzzer berupa rancangan dalam bentuk desain menggunakan Proteus *Isis*. Selanjutnya mikrokontroler akan terhubung dengan router melalui kabel USB. Desain produk memberikan pandangan secara visual sehingga mampu mewakili konsep awal dari rancangan sistem yang telah dibuat. Sebagaimana digambarkan dalam gambar 3.9 yang menjelaskan desain perangkat keras.



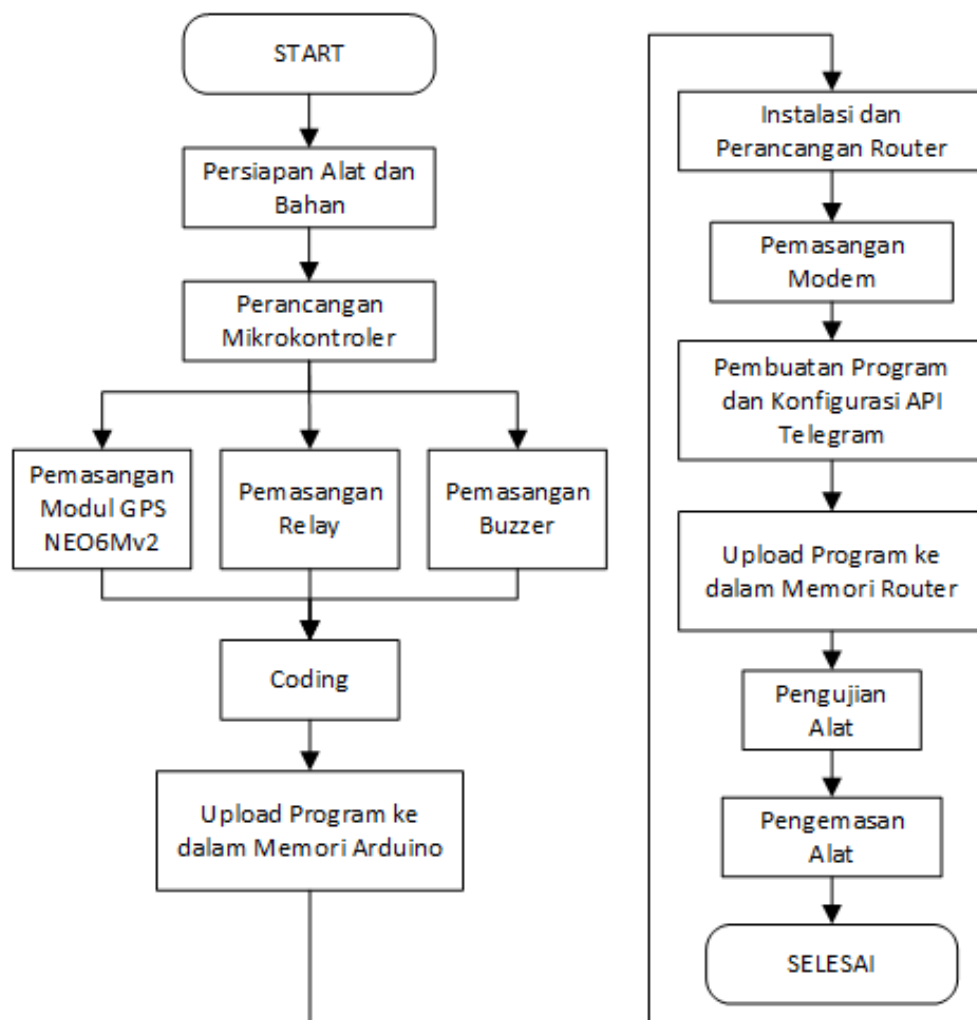
Gambar 3.9 Desain Perangkat Keras

BAB IV

HASIL DAN PEMBAHASAN

5.1. Alur Pembuatan Sistem Kendali Peralatan Elektronika

Flowchart pembuatan rangkaian mulai dari proses awal perancangan sampai proses pengujian akhir dapat dilihat pada gambar 4.1.



Gambar 4.1 Alur Pembuatan Sistem Kendali Peralatan Elektronika

Alur yang digunakan dalam pembuatan alat adalah sebagai berikut :

1. Memulai.
2. Mempersiapkan alat dan bahan yang diperlukan
3. Merancang komponen untuk mikrokontroler yang akan dirakit satu persatu secara keseluruhan, mulai dari pemasangan modul GPS, pemasangan relay, serta pemasangan buzzer.
4. Membuat program untuk Arduino dengan menggunakan *software* yang telah dipersiapkan.
5. Mengupload program jika dirasa telah sesuai dan berjalan seperti yang direncanakan serta tidak ada error, ditandai dengan berhasilnya *upload* program ke dalam Arduino.
6. Menginstalasi sistem operasi OpenWRT ke dalam perangkat router dan dilanjutkan dengan mengkonfigurasi jaringan dengan menghubungkan perangkat modem agar dapat terkoneksi dengan jaringan internet.
7. Membuat program agar perangkat router dapat terkoneksi dengan perangkat mikrokontroler melalui jalur komunikasi serial. Dilanjutkan dengan dengan membuat sebuah program agar perangkat router dan perangkat Arduino dapat dikendalikan melalui API Telegram.
8. Pengujian sistem baik dari *hardware* maupun *software* dari kedua bagian sistem, yaitu perangkat mikrokontroler dan perangkat router.
9. Pengemasan alat yang telah dibuat agar tampilan lebih menarik dan mudah digunakan saat instalasi.
10. Selesai.

5.2. Pembuatan Produk

Proses pembuatan Sistem kendali peralatan elektronika dengan cara merancang komponen elektronik menjadi sebuah rangkaian.

4.2.1. Pemasangan Komponen Elektronik

Komponen yang dirangkai antara lain adalah Arduino Uno, Modul GPS NEO6Mv2, modul relay, dan buzzer sesuai dengan rancangan yang sudah dibuat, komponen tersebut dihubungkan dengan menggunakan kabel jumper.

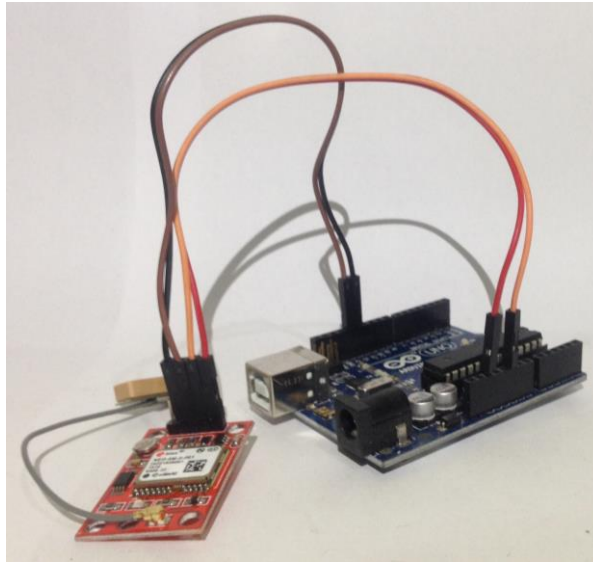
4.2.1.1. Pemasangan Modul GPS NEO6Mv2

Pemasangan modul GPS menggunakan 4 kabel jumper yang menghubungkan langsung antara papan arduino dengan modul GPS. Modul GPS ini sebenarnya memiliki 5 pin yaitu GND, TXD, RXD, PPS dan VCC, namun dalam sistem ini hanya dibutuhkan 4 pin tanpa pin PPS. Konfigurasi pin modul gps dengan arduino dapat dilihat pada tabel 4.1.

Tabel 4.1 Jalur Pin Modul GPS

Pin Modul GPS	Pin Arduino Uno
GND	Pin GND
TXD	Pin D13
RXD	Pin D12
VCC	Pin 3.3V
PPS	-

Setelah menentukan pin yang digunakan, maka langkah selanjutnya adalah memasang modul ke dalam rangkaian langsung seperti pada gambar 4.2



Gambar 4.2 Pemasangan Modul GPS NEO6Mv2

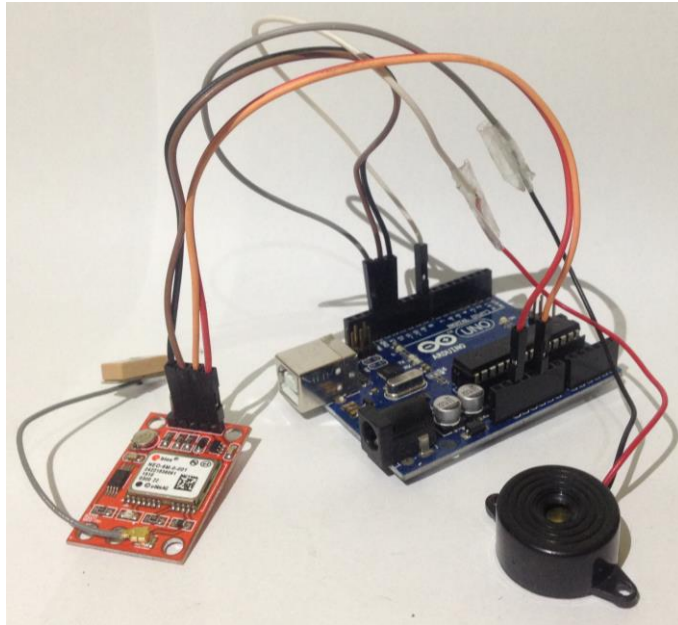
4.2.1.2. Pemasangan Buzzer

Pemasangan pin Buzzer pada sistem minimum Arduino Uno juga menggunakan kabel jumper sebagai penghubung. Buzzer memiliki 2 pin yaitu pin IN dan pin GND. Konfigurasi pin buzzer dengan arduino dapat dilihat pada tabel 4.2.

Tabel 4.2 Jalur Pin Buzzer

Pin Buzzer	Pin Arduino Uno
Pin IN	Pin D8
GND	Pin GND

Setelah menentukan pin yang digunakan, maka langkah selanjutnya adalah memasang buzzer ke dalam rangkaian langsung seperti pada gambar 4.3



Gambar 4.3 Pemasangan Buzzer

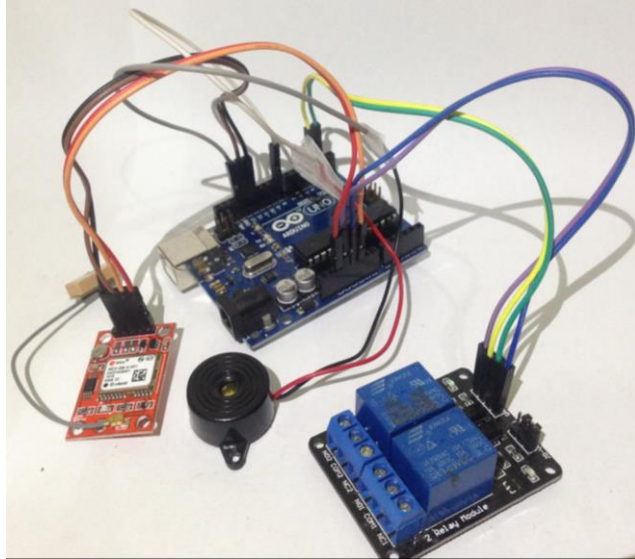
4.2.1.3. Pemasangan Modul Relay

Pemasangan modul relay menggunakan 4 kabel jumper yang menghubungkan langsung antara papan arduino dengan modul relay 2 *channel*. Terdapat 4 pin pada modul yaitu GND, IN1, IN2, VCC. Untuk Pin IN1 akan digunakan sebagai saklar kelistrikan kendaraan, sedangkan untuk IN2 akan digunakan sebagai saklar mesin kendaraan. Konfigurasi pin modul relay dengan arduino dapat dilihat pada tabel 4.3.

Tabel 4.3 Jalur Pin Modul Relay

Pin Modul Relay	Pin Arduino Uno
GND	Pin GND
IN1	Pin D3
IN2	Pin D4
VCC	Pin 5V

Setelah menentukan pin yang digunakan, maka langkah selanjutnya adalah memasang modul ke dalam rangkaian langsung seperti pada gambar 4.4



Gambar 4.4 Pemasangan Modul Relay 2 Channel

4.2.2. Pembuatan Program untuk Arduino

4.2.3.1. Coding untuk program Arduino

1) *Library* dan Variabel

Variabel dalam baris program memiliki fungsi mendeklarasikan pin pada port arduino dan menentukan *library* apa saja yang akan dipakai. Gambar 4.5 merupakan deklarasi *library* dan variabel yang digunakan.

```

#include <TinyGPS++.h>
#include <SoftwareSerial.h>

TinyGPSPlus gps;

bool statusKontak;
bool statusMesin;

double latitude, longitude;
char cekPerintah = 0;

//Deklarasi Relay
static const int relay1 = 4, relay2 = 3, alarm = 8;

//Deklarasi GPS
static const int RXPin = 13, TXPin = 12;
SoftwareSerial ss(RXPin, TXPin);
static const uint32_t GPSBaud = 9600;

```

Gambar 4.5 Baris Variabel pada Program

2) Inisialisasi modul GPS NEO6Mv2

Pada baris program ini berisikan tentang data dari modul GPS itu sendiri. Baris program untuk inisialisasi modul GPS digambarkan pada gambar 4.6.

```

while (ss.available() > 0)
{
    if (gps.encode(ss.read())) {
        if (gps.location.isValid()) {
            latitude = gps.location.lat(), 6;
            longitude = gps.location.lng(), 6;
        }
    }
}

```

Gambar 4.6 Baris Program Inisialisasi Modul GPS

3) Inisialisasi Sistem

Pada baris program *function void setup* berfungsi untuk melakukan inisialisasi setiap pin yang akan digunakan pada papan arduino sebagai input maupun output. Selain itu dalam fungsi ini terdapat perintah untuk *reset* kondisi relay kembali ke posisi mati. Gambar 4.7 merupakan baris kode program pada fungsi *void setup*.

```
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);
  ss.begin(GPSBaud);
  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);
  pinMode(alarm, OUTPUT);
  digitalWrite(relay1, 1);
  digitalWrite(relay2, 1);
}
```

Gambar 4.7 Baris Program Void Setup

4) Core System

Pada baris program inti sistem yaitu fungsi saklar yang berfungsi untuk mengirimkan perintah kepada relay sesuai dengan perintah yang diterima oleh Arduino dari pengguna melalui komunikasi serial. Perintah dibagi menjadi 9, dimana semua perintah berdasarkan angka numerik dari 1 sampai 9 yang dikirimkan dari perangkat router.

1. Perintah 1 menyalakan stop kontak

Berikut ini pada gambar 4.8 adalah baris program untuk menyalakan saklar listrik dari kendaraan bermotor.

```

void cekCmd (char perintahMasuk) {
  // check if a number was received
  if ((perintahMasuk >= '0')) {
    if ((perintahMasuk == '1')) {
      if ((statusKontak == true)){
        delay(100);
        Serial.println("Stop Kontak sudah Menyala");
        tone(alarm,3000);
        delay(100);
        noTone(alarm);
        delay(100);
        tone(alarm,3000);
        delay(100);
        noTone(alarm);
      }

      else {
        digitalWrite(relay1, 0);
        statusKontak = true;
        delay(70);
        Serial.println("Menyalakan Stop Kontak");
        tone(alarm,2000);
        delay(100);
        noTone(alarm);
        delay(100);
        tone(alarm,4000);
        delay(100);
        noTone(alarm);
        delay(100);
        tone(alarm,6000);
        delay(100);
        noTone(alarm);
      }
    }
  }
}

```

Gambar 4.8 Baris Perintah Menyalakan Stop Kontak Listrik

2. Perintah 2 mematikan stop kontak

Berikut ini pada gambar 4.9 adalah baris program untuk mematikan saklar listrik dari kendaraan bermotor.


```

else if ((perintahMasuk == '2')) {
  if ((statusKontak == true)){
    digitalWrite(relay1, 1);
    statusKontak = false;
    statusMesin = false;
    delay(70);
    Serial.println("Mematikan Stop Kontak");
    tone(alarm, 6000);
    delay(100);
    noTone(alarm);
    delay(100);
    tone(alarm, 4000);
    delay(100);
    noTone(alarm);
    delay(100);
    tone(alarm, 2000);
    delay(100);
    noTone(alarm);
  }
else {
  delay(100);
  Serial.println("Stop Kontak tidak Menyala");
  tone(alarm, 3000);
  delay(100);
  noTone(alarm);
  delay(100);
  tone(alarm, 3000);
  delay(100);
  noTone(alarm);
}
}

```

Gambar 4.9 Baris Perintah Mematikan Stop Kontak Listrik

3. Perintah 3 menyalakan mesin kendaraan

Berikut ini pada gambar 4.10 adalah baris program untuk menyalakan mesin kendaraan bermotor.

```

else if ((perintahMasuk == '3')) {
  if ((statusKontak == true)) {
    if ((statusMesin == false)) {
      digitalWrite(relay2, 0);
      delay(70);
      Serial.println("Menyalakan Mesin");
      tone(alarm, 2000);
      delay(100);
      noTone(alarm);
      delay(100);
      tone(alarm, 4000);
      delay(100);
      noTone(alarm);
      delay(100);
      tone(alarm, 6000);
      delay(100);
      noTone(alarm);
      delay(3000);
      digitalWrite(relay2, 1);
      statusMesin = true;
    }

    else {
      delay(100);
      Serial.println("Mesin sudah Menyala");
      tone(alarm, 3000);
      delay(100);
      noTone(alarm);
      delay(100);
      tone(alarm, 3000);
      delay(100);
      noTone(alarm);
    }
  }
  else {
    delay(100);
    Serial.println("Stop Kontak belum Menyala");
    tone(alarm, 3000);
    delay(100);
    noTone(alarm);
    delay(100);
    tone(alarm, 3000);
    delay(100);
    noTone(alarm);
  }
}

```

Gambar 4.10 Baris Perintah Menyalakan Mesin Kendaraan

4. Perintah 4 mematikan mesin kendaraan

Berikut ini pada gambar 4.11 adalah baris program untuk mematikan mesin kendaraan bermotor.

```

else if ((perintahMasuk == '4')) {
    if ((statusKontak == true) && (statusMesin == true)){
        digitalWrite(relay1, 1);
        statusMesin = false;
        delay(70);
        Serial.println("Mematikan Mesin");
        tone(alarm, 6000);
        delay(100);
        noTone(alarm);
        delay(100);
        tone(alarm, 4000);
        delay(100);
        noTone(alarm);
        delay(100);
        tone(alarm, 2000);
        delay(100);
        noTone(alarm);
        delay(2000);
        digitalWrite(relay1, 0);
    }
else {
    delay(100);
    Serial.println("Mesin tidak Menyala");
    tone(alarm, 3000);
    delay(100);
    noTone(alarm);
    delay(100);
    tone(alarm, 3000);
    delay(100);
    noTone(alarm);
}
}

```

Gambar 4.11 Baris Perintah Mematikan Mesin Kendaraan

5. Perintah 5 mendapatkan lokasi kendaraan

Lokasi kendaraan didapatkan berdasarkan lokasi yang diterima oleh modul GPS berupa koordinat. Berikut ini pada gambar 4.12 adalah baris program untuk mendapatkan lokasi kendaraan dimana perintah dibagi menjadi koordinat *longitude* dan *latitude*.

```

else if ((perintahMasuk == '5')) {
    printLat();

}

else if ((perintahMasuk == '7')) {
    printLong();

}

void printLat () {
    Serial.println(gps.location.lat(), 6);
}

void printLong () {
    Serial.println(gps.location.lng(), 6);
}

```

Gambar 4.12 Baris Perintah Mendapatkan Lokasi Kendaraan

6. Perintah 6 menyalakan ‘*answer back alarm*’ pada kendaraan

Buzzer akan berbunyi berdasarkan baris perintah yang didapatkan. Berikut ini pada gambar 4.13 adalah baris program untuk membunyikan alarm kendaraan.

```

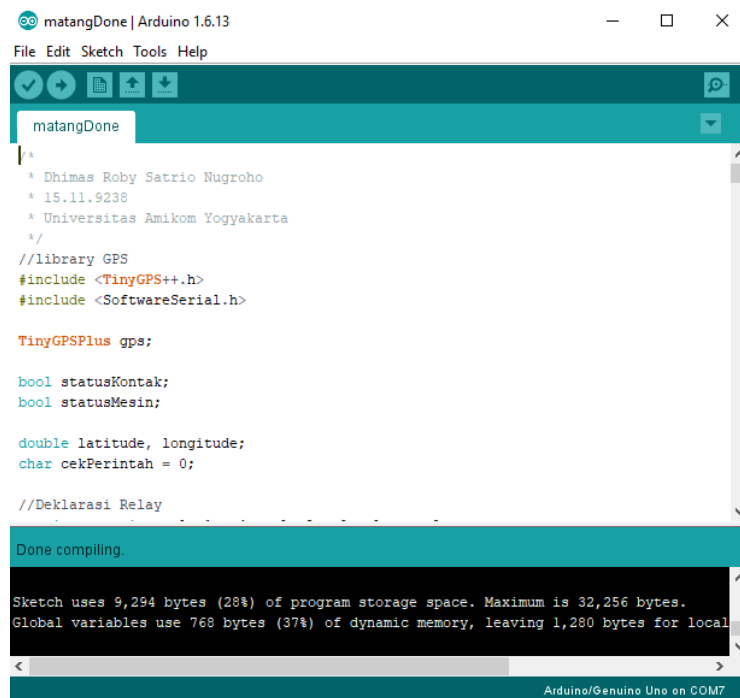
void findMyVehicle() {
    tone(alarm, 7000);
    delay(200);
    noTone(alarm);
    delay(200);
    tone(alarm, 7000);
    delay(200);
    noTone(alarm);
    delay(200);
    tone(alarm, 7000);
    delay(200);
    noTone(alarm);
    delay(200);
    Serial.println("Alarm Sudah Berbunyi");
}

```

Gambar 4.13 Baris Perintah Membunyikan Alarm

4.2.3.2. Pengecekan Program

Pengecekan dilakukan apakah terdapat baris yang error atau tidak. Langkah ini dengan menggunakan fitur verify dari program Arduino IDE. Langkah verifikasi ditunjukkan pada gambar 4.14 untuk memastikan tidak adanya *error* pada program.

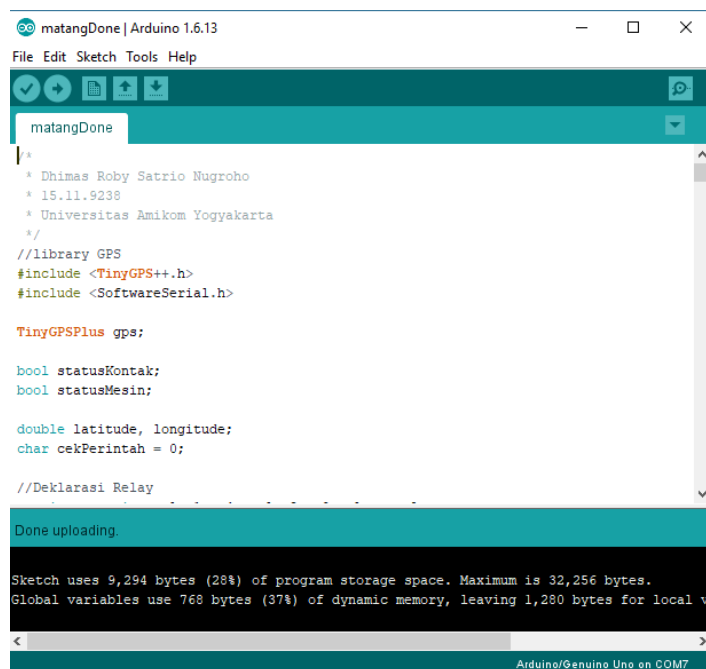


Gambar 4.14 Pengecekan Baris Kode Program

Pada gambar 4.14 menunjukkan program yang telah di verifikasi tidak terjadi *error* dengan indikator munculnya peringatan *Done Compiling*.

4.2.3.3. Upload Program

Pada tahap ini akan dilakukan *upload code* yang telah dibuat menggunakan software Arduino IDE kedalam *memory* Arduino Uno, agar alat dapat bekerja sesuai dengan yang telah direncanakan. Untuk melakukan *upload* program dapat dilakukan dengan menggunakan ikon upload, sebelumnya perangkat arduino harus sudah terhubung ke perangkat PC dengan menggunakan kabel USB tipe B. Sebagaimana ditunjukkan pada gambar 4.15 yang menunjukkan hasil dari program yang telah di upload.



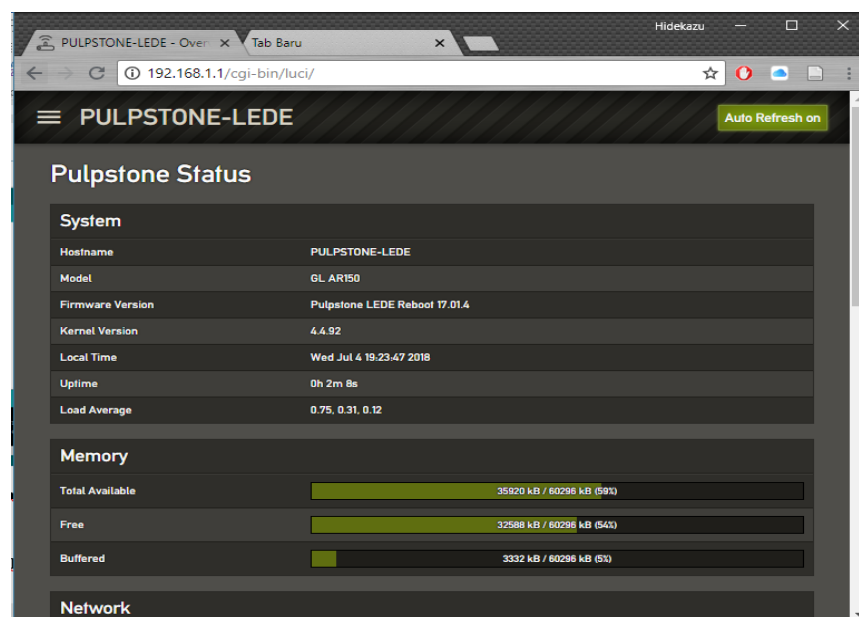
Gambar 4.15 Hasil Upload Program

Pada gambar 4.15 menunjukkan program yang telah di *upload* tidak terjadi *error* dengan indikator munculnya peringatan *Done Uploading*.

4.2.3. Pemasangan Komponen Pengendali Perangkat Elektronik

4.2.3.1. Instalasi dan Konfigurasi OpenWRT untuk router

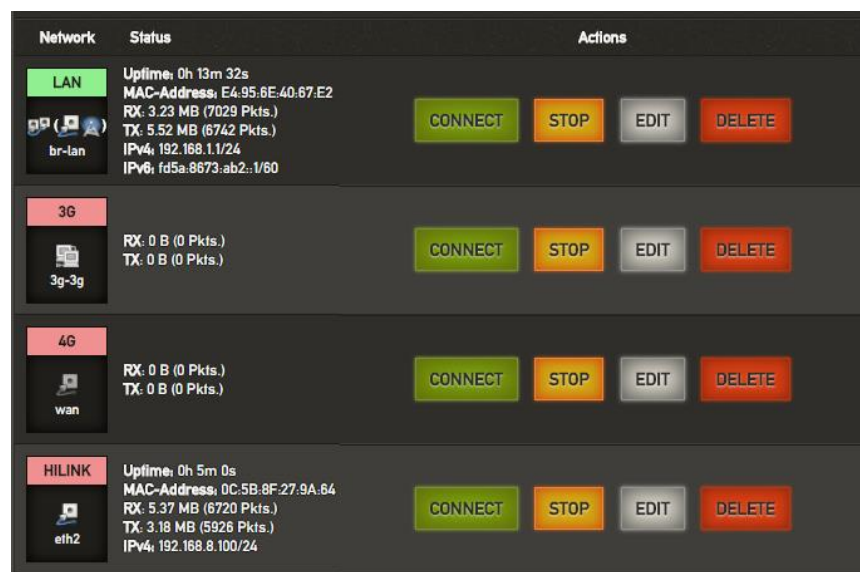
Untuk melakukan instalasi sistem operasi ke dalam perangkat router, maka dibutuhkan file OpenWRT yang didapatkan melalui *website* resmi dari OpenWRT. Pada kasus ini, router sudah *built-in* dengan sistem operasi OpenWRT sehingga tidak memerlukan instalasi, hanya konfigurasi dasar sistem saja. Alamat IP *default* dari OpenWRT adalah 192.168.1.1 yang dapat diakses melalui browser maupun melalui SSH (*Secure Shell*) menggunakan aplikasi PuTTY. Pada gambar 4.16 menunjukkan tampilan utama setelah konfigurasi router yang diakses melalui web browser.



Gambar 4.16 Tampilan Utama OpenWRT

4.2.3.2. Pemasangan Modem dan Konfigurasi Jaringan

Untuk pemasangan modem USB langsung ditancapkan ke router melalui *port* USB yang terdapat pada router, pada kasus ini karena *port* juga akan digunakan oleh Arduino, maka sistem menggunakan USB Hub untuk menambah *port* dari USB. setelah modem USB beserta kartu SIM yang terdapat akses koneksi ke jaringan internet terhubung dengan router, maka langkah selanjutnya adalah cukup memastikan melalui menu *Network* dan masuk ke menu *Interfaces*. maka akan tertera apakah router sudah terkoneksi dengan internet. Pada gambar 4.17 adalah tampilan dari submenu Interface yang menunjukkan router telah terhubung ke internet.

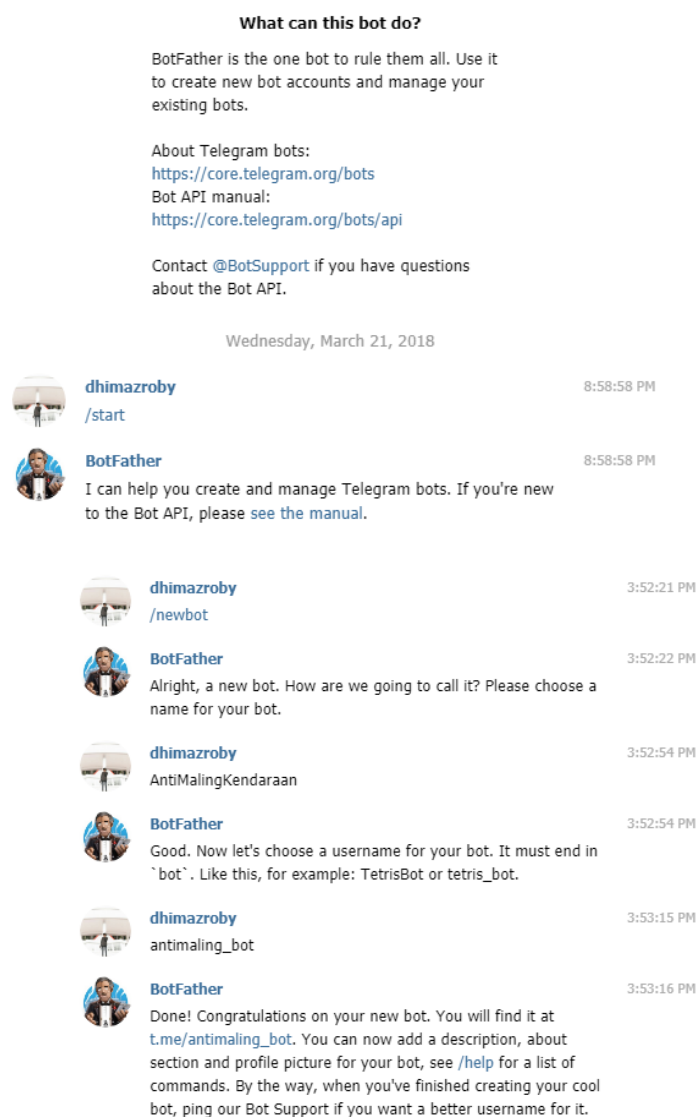


Gambar 4.17 Tampilan Submenu Network

Pada gambar 4.17 terlihat bahwa *interface* HiLink yang merupakan interface dari Modem USB telah terkoneksi dengan jaringan internet dan sekaligus menghubungkan router dengan jaringan internet.

4.2.3.4. *Coding untuk program API Telegram*

Langkah selanjutnya adalah melakukan *coding* untuk API Telegram. Sebelum membuat program untuk Telegram, yang dibutuhkan pertama adalah membuat *BotChat* pada telegram. Langkah pertama adalah menggunakan fitur *Bot Father* yang tersedia pada telegram dimana bot tersebut berfungsi membuat akun robot untuk penggunaanya. Gambar 4.18 adalah langkah membuat *Bot Chat*.



Gambar 4.18 Registrasi Bot Chat pada Telegram

Setelah mendaftarkan program, maka didapatkan *access token* untuk digunakan pada program yang akan berfungsi menjadi jembatan antara router dengan rangkaian elektronik. Untuk melakukan *coding* program menggunakan aplikasi WinSCP dengan menghubungkan langsung ke router melalui alamat ip router.

Berikut ini adalah baris kode program yang berfungsi untuk menghubungkan perangkat router dengan mikrokontroler melalui API Telegram :

1. Baris perintah untuk komunikasi *router* dengan mikrokontroler :

```
"kontak_on")
kontakon=$(echo 1 > /dev/ttyACM0 && head -
1/dev/ttyACM0)
echo -e "$kontakon"
;;
"kontak_off")
kontakoff=$(echo 2 > /dev/ttyACM0 && head -
1/dev/ttyACM0)
echo -e "$kontakoff"
;;
"mesin_on")
mesinon=$(echo 3 > /dev/ttyACM0 && head -
1/dev/ttyACM0)
echo -e "$mesinon"
;;
"mesin_off")
mesinoff=$(echo 4 > /dev/ttyACM0 && head -1
/dev/ttyACM0)
echo -e "$mesinoff"
;;
"getlokasi")
lat=$(echo 5 > /dev/ttyACM0 && head -
1/dev/ttyACM0)
long=$(echo 7 > /dev/ttyACM0 && head -1
/dev/ttyACM0)
cal=$(echo Mengirim Lokasi Kendaraan)
echo -e "$cal"
```

```

    curl -k -s -X POST $api/sendlocation -d
chat_id=$my_chat_id -d latitude=$lat -d
longitude=$long &> $telegram_log_file
;;
"answer_back")
    abs=$(echo 6 > /dev/ttyACM0 && head -1
/dev/ttyACM0)
    echo -e "$abs"
    ;;
"kalibrasi")
    kalibrasi=$(echo 1 > /dev/ttyACM0 && head -1
/dev/ttyACM0)
    echo -e "$kalibrasi"
    ;;
"list")
    echo -e "Vehicle Control:\nKontak On :
/kontak_on\nKontak Off : /kontak_off\nMesin On :
/mesin_on \nMesin Off : /mesin_off \nAnswer Back :
/answer_back \nLokasi Kendaraan : /getlokasi
\nKalibrasi Sensor : /kalibrasi\n\nRouter Control
: /router\n\nBot Anti Maling Kendaraan\nby Dhimas
Roby Satrio\nUniversitas Amikom Yogyakarta\nAbout
Me : /about"
    ;;
*)
    echo -e "List of Commands \n/list"

```

2. Baris perintah untuk menghubungkan perangkat *router* dengan API

Telegram :

```

("/kontak_on")
echo "[ $DATE ] kontak on!" >> $cmd_log_file
clients=$(/root/pulpstone_bot/functions/pulpstone_
general.sh kontak_on)
RETURN_TEXT="$(printf "$clients")"
reply_to_msg $msg_id $origin "\${RETURN_TEXT}"
;;
("/kontak_off")
echo "[ $DATE ] kontak off!" >> $cmd_log_file
clients=$(/root/pulpstone_bot/functions/pulpstone_
general.sh kontak_off)
RETURN_TEXT="$(printf "$clients")"
reply_to_msg $msg_id $origin "\${RETURN_TEXT}"

```

```

;;
    ("/mesin_on")
echo "[ $DATE ] mesin on!" >> $cmd_log_file
clients=$(/root/pulpstone_bot/functions/pulpstone_
general.sh mesin_on)
    RETURN_TEXT="$(printf "$clients")"
reply_to_msg $msg_id $origin "\${RETURN_TEXT}"
;;
    ("/mesin_off")
echo "[ $DATE ] mesin off!" >> $cmd_log_file
clients=$(/root/pulpstone_bot/functions/pulpstone_
general.sh mesin_off)
    RETURN_TEXT="$(printf "$clients")"
reply_to_msg $msg_id $origin "\${RETURN_TEXT}"
;;
    ("/answer_back")
echo "[ $DATE ] answer back!" >> $cmd_log_file
clients=$(/root/pulpstone_bot/functions/pulpstone_
general.sh answer_back)
    RETURN_TEXT="$(printf "$clients")"
reply_to_msg $msg_id $origin "\${RETURN_TEXT}"
;;
    ("/getlokasi")
echo "[ $DATE ] get lokasi!" >> $cmd_log_file
clients=$(/root/pulpstone_bot/functions/pulpstone_
general.sh getlokasi)
    RETURN_TEXT="$(printf "$clients")"
reply_to_msg $msg_id $origin "\${RETURN_TEXT}"
;;
    ("/kalibrasi")
echo "[ $DATE ] kalibrasi!" >> $cmd_log_file
clients=$(/root/pulpstone_bot/functions/pulpstone_
general.sh kalibrasi)
    RETURN_TEXT="$(printf "$clients")"
reply_to_msg $msg_id $origin "\${RETURN_TEXT}"
;;
    ("/list")
echo "[ $DATE ] list all!" >> $cmd_log_file
clients=$(/root/pulpstone_bot/functions/pulpstone_
general.sh list)
    RETURN_TEXT="$(printf "$clients")"
reply_to_msg $msg_id $origin "\${RETURN_TEXT}"
;;
    (*)
echo "[ $DATE ] wrong command!" >> $cmd_log_file
clients=$(/root/pulpstone_bot/functions/pulpstone_
general.sh)

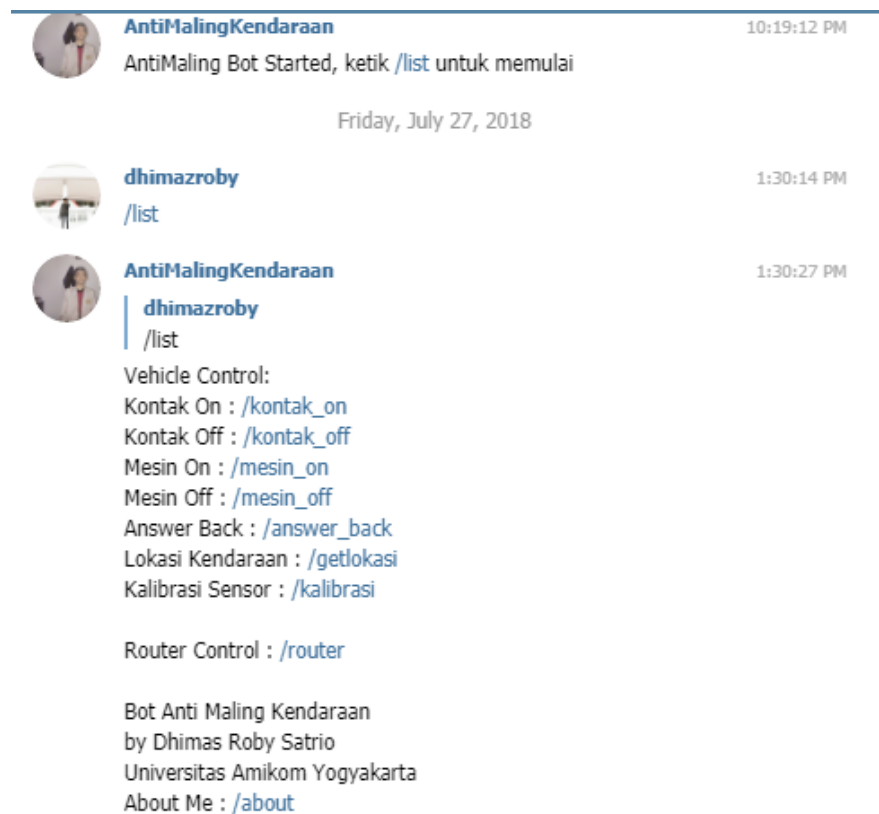
```

```

RETURN_TEXT="$(printf "$clients")"
reply_to_msg $msg_id $origin "\${RETURN_TEXT}"
;;

```

Langkah selanjutnya adalah memastikan router terkoneksi dengan API Telegram dengan melakukan uji *chat* dengan bot yang telah dibuat sebelumnya. sebagaimana terlampir pada gambar 4.19.

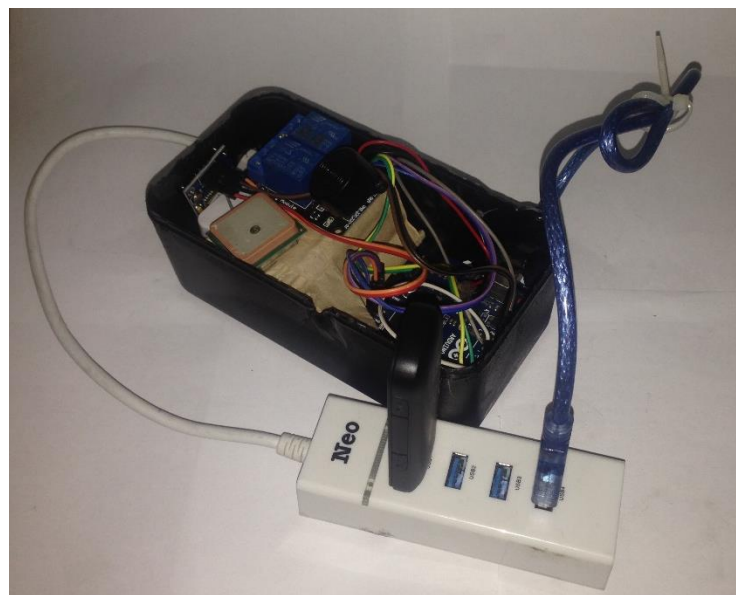


Gambar 4.19 Tampilan Bot Chat pada Telegram

Pada gambar 4.19 terlihat bot chat telah dapat merespon perintah dari user dan diterima oleh router dengan indikator router mengirim balasan kepada user dengan balasan menu dari fitur sistem.

4.2.4. Rangkaian Komponen Elektronik

Setelah melakukan perakitan pada perangkat Arduino dan perangkat *Router*, maka langkah selanjutnya adalah menghubungkan perangkat Arduino dengan perangkat *router* melalui koneksi USB yang dihubungkan dengan router. Sebagaimana ditunjukkan pada gambar 4.20.



Gambar 4.20 Rangkaian Perangkat Elektronik

5.3. Pengujian Rangkaian Sistem

Pengujian rangkaian sistem dapat dilakukan apabila seluruh rangkaian elektronik sudah terpasang dan file program telah di *upload* ke dalam memori dari Arduino. Setelah rangkaian elektronik dipastikan terhubung, langkah selanjutnya adalah memastikan perangkat router sudah terkonfigurasi dan *API* dari aplikasi Telegram sudah dikonfigurasi dan berfungsi dengan benar.

Berikut langkah-langkah pengujian rangkaian sistem:

1. Hubungkan sistem minimum Arduino Uno pada PC menggunakan USB *connection type B*.
2. Dengan menggunakan aplikasi Arduino IDE, kirimkan perintah ke perangkat mikrokontroler melalui komunikasi serial.
3. Memastikan apakah arduino dapat menerima perintah, dan skenario fungsi sudah sesuai dengan program yang dibuat sebelumnya.
4. Datar perintah yang dikirimkan ke perangkat arduino terlampir pada tabel 4.4.

Tabel 4.4 Daftar Perintah Mikrokontroler

Perintah	Fungsi Perintah
1	Menyalakan stop kontak
2	Mematikan stop kontak
3	Menyalakan mesin kendaraan
4	Mematikan mesin kendaraan
5	Mengirimkan koordinat <i>Latitude</i>
6	Membunyikan alarm
7	Mengirimkan koordinat <i>Longitude</i>
9	Kalibrasi Relay

Setelah perangkat arduino dipastikan dapat berfungsi, selanjutnya hubungkan perangkat arduino dengan perangkat *router* kemudian uji melalui aplikasi Telegram pada perangkat *smartphone*. Berikut merupakan hasil tampilan jendela *chat* pada aplikasi telegram sebagaimana digambarkan pada gambar 4.21.



Gambar 4.21 Jendela Tampilan Aplikasi Telegram

Gambar 4.21 merupakan jendela tampilan pada aplikasi Telegram dari *smartphone* pengguna yang menampilkan informasi berupa status saklar dan mesin dari kendaraan, serta notifikasi alarm dan lokasi terkini yang dikirimkan oleh perangkat GPS.

5.4. Pengemasan Produk

Pengemasan produk merupakan proses dimana seluruh rangkaian mikrokontroler dan perangkat *router* dikemas lebih rapi, sehingga rangkaian tersebut dapat berbentuk menjadi simulasi perangkat pengendali peralatan elektronika. Wadah berbahan plastik yang digunakan sebagai tempat peletakan rangkaian digambarkan pada gambar 4.22 dan 4.23.



Gambar 4.22 Pengemasan Alat Tampak Dalam



Gambar 4.23 Pengemasan Alat Tampak Luar

Pengemasan menggunakan wadah yang tidak besar guna memudahkan dalam peletakan alat pada kendaraan bermotor, selain itu lebih menghemat tempat dan memudahkan pemasangan pada kendaraan bermotor.

BAB V

KESIMPULAN

5.1. Kesimpulan

Dari keseluruhan proses perancangan “Integrasi Telegram untuk monitoring dan keamanan kendaraan berbasis Arduino dan Openwrt” dapat disimpulkan :

1. Sistem minimum Arduino Uno dan router OpenWRT dapat berfungsi sebagai pengendali saklar pada sepeda motor berbasis arduino.
2. Relay dapat menghidupkan maupun mematikan sepeda motor dengan benar sesuai fungsi yang diinginkan.
3. GPS dapat mengirimkan koordinat lokasi kendaraan bermotor dengan tepat dan jeda yang tidak terlalu lama dengan lokasi kendaraan sekarang.
4. *Router* dapat menghubungkan perangkat Arduino dengan API Telegram sehingga sistem dapat dikendalikan melalui aplikasi Telegram dan direspon oleh sistem dengan baik dan sesuai fungsi yang diinginkan.

5.2. Saran

Dari perancangan sistem yang telah direalisasikan pada skripsi ini, maka penulis memberikan saran, diantaranya :

1. Kedepannya agar alat ini dapat dikembangkan seperti *real time tracking* untuk melacak lokasi kendaraan tanpa harus menunggu jeda waktu mendapatkan lokasi.
2. Penambahan fitur seperti autentikasi login agar keamanan kendaraan menjadi berlapis.

3. Menambahkan fitur akses melalui WiFi dari perangkat router untuk mengontrol kendaraan jarak dekat.
4. Penambahan sensor juga diperlukan untuk menambah fungsionalitas dari sistem.
5. Menambahkan notifikasi apabila kendaraan dinyalakan tanpa melalui sistem, agar tujuan *monitoring* keamanan dapat terpenuhi.

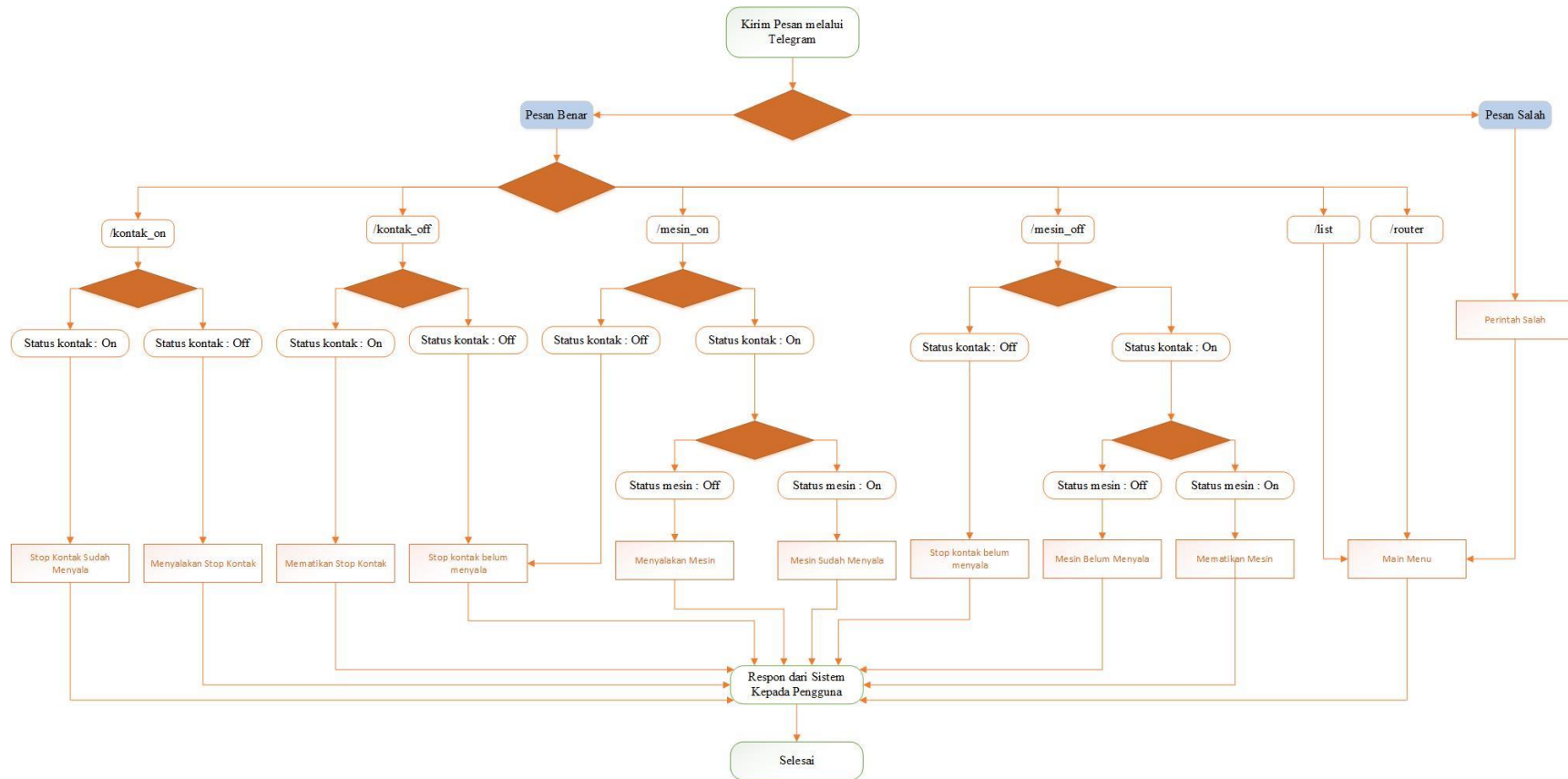
DAFTAR PUSTAKA

- [1] Muhamad Ngafifi. 2016, *Kemajuan Teknologi dan Pola Hidup Manusia*. Jurnal Pembangunan Pendidikan: Fondasi dan Aplikasi.
- [2] Badan Pusat Statistik, Statistik Kriminal. 2017, *Sub Direktorat Statistik Politik dan Keamanan*, Ed.: Badan Pusat Statistik, Jakarta.
- [3] M. A. Ari Samadhi. 2016, *Literature Review and State of the Art*. Bimbingan Teknik Penulisan Artikel pada Jurnal Ilmiah bagi Dosen di Lingkungan Kopertis Wilayah VI.
- [4] Sumardi. 2017, *Perancangan Sistem Starter Sepeda Motor Menggunakan Aplikasi Android Berbasis Arduino Uno*. Prosiding Seminar Ilmu Komputer dan Teknologi Informasi.
- [5] Zul Amri Durrin Nafis, Khafizh Khastuti. 2015, *Rancang Bangun Sistem Keamanan Kendaraan Bermotor dengan Metode Authentication dan Point Positioning menggunakan RFID berbasis Mikrokontroller*. Skripsi Fakultas Ilmu Komputer UDINUS Semarang.
- [6] Mahfud Ichsan Adi P. 2017, *Rancangan Sistem Start Engine Dan Alarm Pada Sepeda Motor Menggunakan Arduino Uno Berbasis Android*. Skripsi Fakultas Komunikasi dan Informatika Universitas Muhammadiyah Surakarta.
- [7] Ari Nugroho. 2017, *Sistem Keamanan Sepeda Motor Menggunakan Arduino Berbasis Android*. Skripsi Fakultas Teknik Universitas Widyatama Bandung.
- [8] Dias Utomo, Muchammad Sholeh, Arry Avorizano. 2017, *Membangun Sistem Mobile Monitoring Keamanan Web Aplikasi Menggunakan Suricata dan Bot Telegram Channel*. Jurnal ISSN Seminar Nasional TEKNOKA.
- [9] Barnett, R.H., Cox, S.A., O'Cull, L.D. 2003, *Embedded C Programming and the Atmel AVR*. New York: Thomson Delmar Learning.

- [10] Mada Sanjaya W,S, Ph.D, 2016, *Panduan Praktis Membuat Robot Cerdas Menggunakan Arduino dan MATLAB*. Penerbit ANDI, Yogyakarta, hal.23.
- [11] Abdul Kadir. 2013, *Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemrogramannya Menggunakan Arduino*. Penerbit ANDI, Yogyakarta, hal.16.
- [12] Setiawan, Arie. 2013, *20 Aplikasi Mikrokontroler ATmega 8535 & ATmega 16 Menggunakan Bascom-AVR*. Penerbit ANDI, Yogyakarta, hal 1-3.
- [13] Abdul Kadir. 2013, *Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemrogramannya Menggunakan Arduino*. Penerbit ANDI, Yogyakarta, hal.25.
- [14] Purbo. W. Onno.2013, *Jaringan Mesh – Solusi Jitu Membangun Jaringan Wireless Gotong Royong Tanpa Access Point*. Penerbit ANDI : Yogyakarta, hal.10.
- [15] OpenWRT, Pulpstone LEDE Team. 2018, *OpenWRT LEDE Official Documentation Vol 1*. IEEE Journal Cloud & Firewall.
- [16] Telegram, *Telegram F.A.Q.* [Online], <https://telegram.org/faq#q-what-is-telegram-what-do-i-do-here>. [Diakses 26 09 2018]
- [17] I Putu Agus Eka Pratama, Sinung Suakanto. 2015, *Wireless Sensor Network*. Penertbit Informatika, Bandung, hal.165.
- [18] El-Rabbany Ahmed, 2002. *Introduction to GPS*, Artech House, Boston.
- [19] I Putu Agus Eka Pratama, Sinung Suakanto. 2015, *Wireless Sensor Network*. Penertbit Informatika, Bandung, hal.364.
- [20] GL-iNet, *GL-AR150 – GL.iNet*, <https://www.gl-inet.com/products/gl-ar150/>. [Diakses 26 09 2018]
- [21] Abdul Kadir. 2013, *Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemrogramannya Menggunakan Arduino*. Penerbit ANDI, Yogyakarta, hal.266.

- [22] Mada Sanjaya W,S, Ph.D, 2016, *Panduan Praktis Membuat Robot Cerdas Menggunakan Arduino dan MATLAB*. Penerbit ANDI, Yogyakarta, hal.99-101.
- [23] Hanif Al Fatta. 2007, *Analisis dan Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan dan Organisasi Modern*. Penerbit ANDI, Yogyakarta, hal.105.
- [24] Boennie Soeherman, Marion Pinontoan. 2013. *Design Information System* . Elex Media Komputindo, Jakarta, hal.133.

LAMPIRAN



Lampiran A *Flowchart* Sistem