

**SISTEM KEAMANAN SEPEDA MOTOR BERBASIS
MICROCONTROLLER MENGGUNAKAN
*SENSOR FINGERPRINT***

DISUSUN OLEH

ANGKI AHMAD

T3115242

SKRIPSI

Untuk memenuhi salah satu syarat ujian

Guna memperoleh gelar sarjana



**PROGRAM SARJANA
TEKNIK INFORMATIKA
UNIVERSITAS ICHSAN GORONTALO
GORONTALO
2019**

PENGESAHAN SKRIPSI

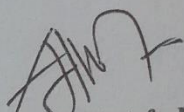
**SISTEM KEAMANAN SEPEDA MOTOR BERBASIS
MICROCONTROLLER MENGUNAKA
SENSOR FINGERPRINT**

Oleh
Angki Ahmad
T3115035

SKRIPSI

Untuk memenuhi salah satu syarat ujian
guna memperoleh gelar sarjana
dan telah disetujui oleh tim pembimbing pada tanggal
Desember 2019

Pembimbing I


Yasin Aril Mustofa, M.Kom
NIDN :0926088503

Pembimbing II

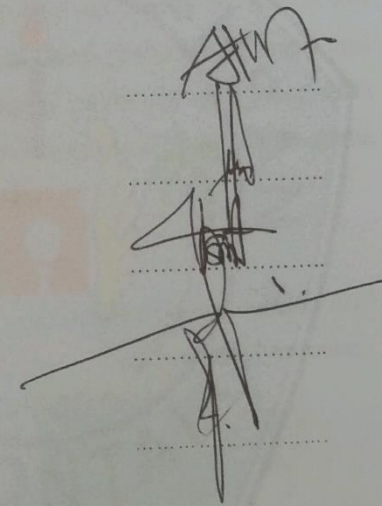

Warid Yunus, M.Kom
NIDN:0914059001

PERSETUJUAN SKRIPSI
SISTEM KEAMANAN SEPEDA MOTOR BERBASIS
***MICROCONTROLLER* MENGGUNAKAN**
SENSOR FINGERPRINT

Oleh
Angki Ahmad
T3115242

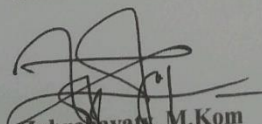
Diperiksa oleh Panitia Ujian Strata Satu (S1)
Universitas Ichsan Gorontalo
Gorontalo, Desember 2019

1. Pembimbing I
Yasin Aril Mustofa, M.Kom,
2. Pembimbing II
Warid Yunus, M.Kom
3. Penguji I
Asmaul Husna, M.Kom
4. Penguji II
Budy Santoso, S.Kom, M.Eng
5. Penguji III
Serwin, M.Kom



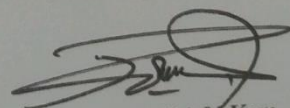
Mengetahui :

Dekan Fakultas Ilmu Komputer



Zohrahavati, M.Kom
NIDN : 0912117702

Ketua Program Studi



Irvan A. Salihi, M.Kom
NIDN : 0928028101

PERNYATAAN SKRIPSI

Dengan ini saya menyatakan bahwa :

1. Karya tulis (Skripsi) saya ini adalah asli dan belum pernah diajukan untuk mendapat gelar akademik (Sarjana) baik di Universitas Ichsan Gorontalo maupun di perguruan tinggi lainnya.
2. Karya tulis (Skripsi) saya ini adalah murni gagasan, rumusan, dan penelitian saya sendiri, tanpa bantuan pihak lain, kecuali arahan dari tim pembimbing.
3. Dalam karya tulis (Skripsi) saya ini tidak terdapat karya atau pendapat yang telah dipublikasikan orang lain, kecuali secara tertulis dicantumkan sebagai acuan/sitasi dalam naskah dan dicantumkan pula dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini, serta sanksi lainnya sesuai dengan norma-norma yang berlaku di Universitas Ichsan Gorontalo.

Gorontalo, Desember 2019

Yang Membuat Pernyataan,



Angki Ahmad

KATA PENGANTAR

Assalamu Alaikum Warahmatullahi Wabarakatuh.

Alhamdulillah, puji syukur kita ucapkan kehadiran *Allah Subhana huwataallah* karena dengan ijin dan kuasanya akhirnya penulis dapat menyelesaikan penelitian ini untuk memenuhi syarat mendapatkan gelar Sarjana (S1) Program Studi Teknik Informatika, Fakultas Ilmu Komputer Universitas Ichsan Gorontalo.

Penulis menyadari bahwa skripsi ini tidak akan terwujud / terselesaikan jika tanpa bantuan dan support dari berbagai pihak, seperti dari Dosen-dosen Pembimbing saya, teman-teman Mahasiswa dan juga dari seluruh Dosen yang ada di Fakultas Ilmu Komputer.

Penulis menyadari bahwa setiap usaha pasti akan membuahkan hasil yang kita inginkan meskipun banyak tantangan yang dihadapi saat penyusunan Skripsi ini, namun setiap tantangan itu bukan merupakan batu sandungan melainkan sebuah proses yang harus dilewati, pada akhirnya Skripsi ini dapat terselesaikan meskipun masih terdapat kekurangan baik itu dalam pengumpulan data maupun dalam penyusunan.

Oleh karena itu dengan segala keikhlasan dan kerendahan hati, penulis mengucapkan banyak terima kasih dan penghargaan yang setinggi-tingginya kepada:

1. Bapak Muhammad Ichsan Gafar, SE, M.Ak selaku ketua Yayasan Pengembangan Ilmu Pengetahuan dan Teknologi (YPIPT) Ichsan Gorontalo
2. Bapak. Dr. Abdul Gafar Latjoke, selaku Rektor Universitas Ichsan Gorontalo
3. Ibu. Zohrahayati, M.Kom, selaku Dekan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo
4. Bapak Sudirman S. Pana, M.Kom, selaku Wakil Dekan I Bidang Akademik Fakultas Ilmu Komputer.
5. Ibu Irma Surya Kumala Idris, M.Kom, selaku Wakil Dekan II Bidang Administrasi Umum dan Keuangan Fakultas Ilmu Komputer.

6. Bapak Sudirman Melangi, M.Kom, selaku Wakil Dekan III Bidang Kemahasiswaan Fakultas Ilmu Komputer.
7. Bapak. Irvan Abraham Salihi, M.Kom, selaku Ketua Jurusan Fakultas Ilmu Komputer Universitas Ichsan Gorontalo
8. Bapak, Yasin Aril Mustofa, M.Kom, selaku Pembimbing Utama
9. Bpk. Warid Yunus, M.Kom, selaku Pembimbing Pendamping
10. Bapak dan Ibu dosen fakultas ilmu komputer yang sudah mendidik dan mengajar selama saya berada di kampus ini
11. Bapak dan Ibu saya yang sudah mendukung sepenuh hati baik secara materi maupun moril.
12. Seluruh teman-teman mahasiswa fakultas ilmu komputer, khususnya teman-teman satu kelas saya yang tak hentinya-hentinya memberikan semangat kepada saya.

Semoga *Allah Subhanahuwatallah* melimpahkan rahmat dan balasan atas jasa-jasa mereka kepada kami. Penulis menyadari bahwa apa yang telah dicapai ini masih jauh dari kesempurnaan dan masih banyak terdapat kekurangan. Oleh karena itu, penulis sangat adanya kritik dan saran yang konstruktif. Akhirnya penulis berharap semoga hasil yang telah dicapai ini dapat bermanfaat untuk kita semua, Amin.

Wassalamu Alaikum Warahmatullahi Wabarakatuh...

Gorontalo, Desember 2019

Penulis

ABSTRACT

Safety system for wheeled vehicles is needed to protect vehicles from crime. One of the security systems that has been provided by the manufacturer for two-wheeled vehicles is a double lock. Double keys have disadvantages. The solution is to overcome the shortcomings that is easily broken using a special key so that a solution is needed to overcome the shortcomings, namely by making a security system by utilizing a fingerprint as a vehicle system controller. In this study the design of a vehicle safety system that can regulate the ignition system and starter system on the vehicle. Alarms on vehicles can be known via SMS (Short Message Service) by utilizing Arduino Uno 328P, GSM SIM900A, JQC Relay 3F. The test results show that the system is running well, and can provide SMS notifications to the vehicle owner when the vehicle is burglarized.

Keywords: Security System, Microcontroller, Fingerprint

ABSTRAK

Sistem keamanan untuk kendaraan roda sangat dibutuhkan untuk melindungi kendaraan dari tindak kejahatan. Salah satu sistem keamanan yang telah disediakan oleh pabrikan untuk kendaraan roda dua adalah kunci ganda. Kunci ganda memiliki kekurangan, Solusi yang dilakukan untuk mengatasi kekurangan yaitu mudah dibobol menggunakan kunci khusus sehingga diperlukan solusi untuk mengatasi kekurangan yaitu dengan pembuatan sistem keamanan dengan memanfaatkan *fingerprint* sebagai alat pengendali sistem pada kendaraan. Pada penelitian ini dirancang sistem keamanan kendaraan yang dapat mengatur sistem kunci kontak dan sistem starter pada kendaraan. Alarm pada kendaraan dapat diketahui melalui SMS (*Short Message Service*) dengan memanfaatkan *Arduino Uno 328P*, *GSM SIM900A*, *Relay JQC 3F*. Hasil pengujian menunjukkan bahwa sistem berjalan dengan baik, dan dapat memberikan notifikasi berupa SMS kepada pemilik kendaraan saat kendaraan dibobol pencuri.

Kata Kunci : Sistem Keamanan, Microcontroller, Fingerprint.

DAFTAR ISI

PENGESAHAN SKRIPSI	Error! Bookmark not defined.
PERSETUJUAN SKRIPSI	Error! Bookmark not defined.
PERNYATAAN SKRIPSI	Error! Bookmark not defined.
KATA PENGANTAR	iv
ABSTRACT	vi
DAFTAR TABEL	xi
BAB I PENDAHULUAN	1
1.1 LATAR BELAKANG	1
1.2 Identifikasi Masalah.....	2
1.3 Rumusan Masalah.....	2
1.4 Tujuan Penelitian	Error! Bookmark not defined.
1.5 Manfaat Penelitian	3
1.5.1 Manfaat Teoritis	3
1.5.2 Manfaat Praktis	3
BAB II LANDASAN TEORI	4
2.1 Tinjauan Studi.....	4
2.2 Sepeda Motor (Motor Cycle)	4
2.3 Fingerprint.....	5
2.4 Arduino	5
2.4.1 Bagian-bagian Dan Fungsi Arduino Uno.....	7
2.4.2 IDE Arduino.....	9
2.4.3 Bahasa Pemrograman Arduino	11
2.5 Modul GSM SIM900A	13
2.6 Relay JQC 3F	15
2.7 Buzzer	16
2.8 Inverter (Adaptor)	17
2.9 Kerangka Pikir	18
BAB III METODE PENELITIAN	19
3.1 Jenis, Metode, Subjek, Objek, Waktu, dan Lokasi Penelitian	19
3.2 Alat dan Bahan Penelitian	19

3.3 Metode Penelitian	20
3.3.1 Sumber Data.....	20
3.3.2 Pengumpulan Data	20
3.3.3 Teknik Pengolahan dan Analisis Data	21
3.3.4 Teknik Pengujian	22
3.3.5 Pembuatan Laporan.....	22
BAB IV PERANCANGAN SISTEM.....	23
4.1 Perancangan Alat dan Sistem.....	23
4.1.1 Perancangan Blok Diagram	24
4.1.2 Perancangan Kerja Sistem	25
4.2 Perancangan Perangkat Keras.....	27
4.3 Perancangan Perangkat Lunak	30
4.3.1 <i>Flowchart</i> (Alur Program)	31
4.3.2 Kode Program Sensor	32
4.3.3 Kode Program GSM SIM900A.....	33
4.4 Perancangan sistem Keseluruhan.....	34
BAB V IMPLEMENTASI DAN PENGUJIAN SISTEM.....	35
5.1 Implementasi.....	35
5.1.1 Hasil Perancangan Perangkat Keras	35
5.1.2 Pemasangan Alat Pada Sepeda Motor.....	36
5.2 Pengujian Sistem.....	36
5.2.1 Pengujian Sensor Fingerprint.....	38
5.2.2 Pengujian Modul GSM SIM 900A	41
5.2.3 Pengujian Sistem Secara Keseluruhan	42
BAB VI PENUTUP.....	44
6.1 Kesimpulan	44
6.2 Saran	44
DAFTAR PUSTAKA	46

DAFTAR GAMBAR

Gambar 2.1 : Fingerprint	5
Gambar 2.2 : Arduino Uno.....	7
Gambar 2.3 : Bagian-Bagian Arduino Uno	7
Gambar 2.4 : Ide Arduino	10
Gambar 2.5 : Modul Gsm Sim 900a	13
Gambar 2.6 : Relay JQC 3F	15
Gambar 2.7 : Buzzer	16
Gambar 2.8 : Inverter	17
Gambar 4.1 : Diagram Perancangan Alat dan Sistem.....	23
Gambar 4.2 : Blok Diagram Kerja Sistem	24
Gambar 4.3 : Diagram Kerja Sistem	26
Gambar 4.4 : Perancangan inverter.....	27
Gambar 4.5 : Perancangan Fingerprint Sensor	28
Gambar 4.6 : Perancangan Modul GSM SIM.....	28
Gambar 4.7 : Perancangan Modul Relay	29
Gambar 4.8 : Perancangan Buzzer	29
Gambar 4.9 : Tampilan Perangkat Lunak Arduino	30
Gambar 5.0 : Flowcart Kerja Sistem.....	31
Gambar 5.1 : Testing Kode Program Sensor Fingerprint	32
Gambar 5.2 : Tampilan Program Modul GSM	33
Gambar 5.3 : Perancangan Sistem Keseluruhan	34
Gambar 5.4 : Hasil Rangkaian Perangkat Keras.....	35
Gambar 5.5 : Pemasangan Alat pada Sepeda Motor.....	36
Gambar 5.6 : Langkah-Langkah Pengujian Sistem.....	37
Gambar 5.7 : Data di Temukan	38
Gambar 5.8 : Data Tidak di Temukan.....	39
Gambar 5.9 : Pengiriman SMS saat Data tidak Ditemukan	39

DAFTAR TABEL

Tabel 2.1 : Spesifikasi Arduino Uno.....	6
Tabel 2.2 : Bagian-bagian Arduino.....	8
Tabel 2.3 : Perintah - perintah AT-Command	14
Tabel 2.4 : Daftar Alat dan Bahan	19
Tabel 5.1 : Hasil Pengujian Sensor fingerprint	40
Tabel 5.2 : Hasil Pengamatan Sensor fingerprint.....	40
Tabel 5.3 : Tampilan Hasil Pengiriman SMS	42
Tabel 5.4 :Hasil Pengujian Keseluruhan Sistem	43

BAB I

PENDAHULUAN

1.1 LATAR BELAKANG

Sepeda motor adalah alat transportasi yang paling efektif bagi masyarakat Indonesia. Pilihan sepeda motor sebagai moda transportasi paling populer bergantung pada keunggulan sepeda motor itu sendiri dalam hal biaya perawatan, biaya bahan bakar, efektivitas waktu tempuh, kenyamanan dan kemampuannya untuk mengatasi kemacetan di jalan raya. [1].

Sistem keamanan sepeda motor saat ini belum cukup efektif untuk menangkal kejahatan sepeda motor. Tidak adanya sistem keselamatan pada kendaraan bermotor, hanya menggunakan kunci kontak dan penutup kunci, dan sistem keselamatan tersebut dipahami oleh para praktisi sebagai ukuran. Beberapa mode pencurian yang sering digunakan oleh pencuri termasuk penggunaan palang (T-key) yang dimodifikasi untuk merusak kunci stang. [2].

Pertumbuhan teknologi berbasis komputer dalam beberapa tahun terakhir juga telah memengaruhi kebutuhan akan keamanan canggih, termasuk kepemilikan, keamanan, atau kerahasiaan lainnya. Sehubungan dengan pengembangan dan kemajuan teknologi, sumber daya manusia yang kompeten dan siap untuk menggunakannya, sehingga orang tidak ketinggalan atau dengan kata lain, dapat menggunakan teknologi yang ada. Salah satu perkembangan teknologi terlihat di bidang teknologi berbasis komputer [3].

Microcontroller adalah terobosan dalam teknologi mikroprosesor dan mikrokomputer untuk memenuhi tuntutan teknologi baru. Sebagai teknologi baru, yaitu teknologi semikonduktor dengan kandungan transistor yang lebih banyak, namun hanya membutuhkan sedikit ruang dan dapat diproduksi secara massal [4].

Fingerprint merupakan pengembangan teknologi yang memiliki tingkat keamanan yang cukup tinggi yang hanya dapat diakses oleh orang yang sidik jarinya sudah di input atau disimpan. Seiring perkembangan teknologi, *fingerprint* merupakan bentuk biometri yang menggunakan karakter fisik untuk identifikasi.

Oleh karena itu, perlunya sistem keamanan sidik jari untuk kendaraan bermotor harus *dibuat* untuk menghindari aktivitas yang tidak diinginkan [5].

Berdasarkan masalah di atas, peneliti ingin merancang sistem keamanan sepeda motor menggunakan sidik jari, dengan *fingerprint* sebagai inputan data, modul *SIM GSM* sebagai pengirim pesan menggunakan *SMS (Short Message Service)* dan *Arduino Uno* sebagai *CPU* dari sistem yang dibuat. nantinya akan menggunakan modul sidik jari R307, *Arduino Uno*, *GSM SIM 900A*. Oleh karena itu, dipandang perlu untuk membangun sebuah sistem dengan judul penelitian **“Sistem Keamanan Sepeda Motor Berbasis *Microcontroller* Menggunakan Sensor *Fingerprint*”**.

1.2 Identifikasi Masalah

Berdasarkan Latar Belakang, dapat diidentifikasi masalah dari dua (2) yakni dari sisi permasalahan umum dan dari sisi ilmu komputer.

1. Bagaimana caranya mengurangi pencurian kendaraan bermotor
2. Bagaimana caranya mengirimkan informasi kepada pemilik kendaraan bahwa kendaraannya akan di curi.

1.3 Rumusan Masalah

1. Bagaimana merancang dan mendesain Sistem Keamanan Sepeda Motor Berbasis *Microcontroller* Menggunakan Sensor *Fingerprint* sehingga Alat tersebut dapat berkerja dan berfungsi dengan baik.
2. bagaimana Sistem tersebut memberikan informasi kepada *user* bahwa tempat sampah tersebut sudah penuh atau belum penuh.

1.4 Batasan Masalah

1. Sistem keamanan sepeda motor ini hanya penambahan keamanan pada sepeda motor yang ada sekarang ini
2. Sistem ini hanya bisa mendeteksi objek dengan cara mendeteksi ID atau sidik jari yang terdaftar.

1.5 Tujuan Penelitian

Adapun tujuan penelitian berdasarkan masalah diatas yaitu:

- 1) Menguji coba secara langsung sistem keamanan *fingerprint* untuk mendapatkan hasil yang sesuai, akurat dan efisien serta bisa melihat kelemahan.
- 2) Menghasilkan sistem keamanan yang mampu bekerja dengan baik, serta bisa diandalkan sehingga bisa diterapkan dilingkungan masyarakat.

1.6 Manfaat Penelitian

1.5.1 Manfaat Teoritis

Sistem keamanan *fingerprint* pada kendaraan bisa dikembangkan dan dibuat secara masal, yang penting harus dirancang dengan matang agar menghasilkan Sistem yang berkualitas dan siap pakai.

1.5.2 Manfaat Praktis

Sistem keamanan *fingerprint* pada kendaraan dibuat untuk kebutuhan masyarakat, terutama masyarakat yang memiliki kendaraan yang perlu dilindungi dari pencurian. Masyarakat akan sangat terbantu dengan lahirnya sistem keamanan ini, karena dengan adanya sistem ini masyarakat tidak merasa khawatir untuk memarkir kendaraan di tempat umum.

BAB II

LANDASAN TEORI

Berdasarkan Penelitian sebelumnya yang menjadi tinjauan studi pada penelitian ini adalah sebagai berikut.

2.1 Tinjauan Studi

Dwiyana Indra Prasetya dan Mushlihudin, 2018 “**Sistem keamanan sepeda motor dengan Menggunakan kata sandi berbasis Arduino Nano**” menggunakan metode perancangan sistem. Tombol 3x4 digunakan untuk menginput kata sandi. Jika kata sandi yang dimasukkan benar, *relay coil* pengapian aktif. Jika kata sandi salah sebanyak tiga kali, *relay* alarm aktif dan *relay coil* tetap dalam keadaan mati. Sistem keamanan sepeda motor ini menggunakan 2 buah *relay* sebagai konduktor coil dan alarm. *Relay coil* berfungsi sebagai pemutus dan penghubung tegangan *CDI* ke *coil* [6].

Kemudian penelitian oleh Arycca Septian Mulyana Dkk., tahun 2017 “**Perancangan Sistem Keamanan Sepeda Motor Dengan Menggunakan State Machine**” Menggunakan hasil uji untuk waktu mengirim dan menerima data lokasi ditemukan 3 waktu rata-rata, berdasarkan 3 *state* menggunakan Status data untuk mengirim dan menerima data lokasi yaitu. Status B2 6,64 detik, status B1 7,829 detik dan status B3 7,273 detik. Dan tingkat keberhasilan yang ditemukan sistem dengan menerapkan *State Machine* berhasil dan berjalan dengan baik [7].

2.2 Sepeda Motor (Motor Cycle)

Sepeda motor adalah kendaraan roda dua yang ditenagai mesin. Posisi kedua roda sebaris lurus dan pada kecepatan tinggi sepeda motor tetap stabil karena gaya sentrifugal. Pada kecepatan rendah, stabilitas atau keseimbangan sepeda motor tergantung pada pengaturan stang pengemudi [1].

Menurut peneliti perlu adanya pembahasan mengenai Sepeda Motor Injeksi, Sepeda Motor Injeksi juga membutuhkan pembangkit listrik yang lebih

besar di bandingkan dengan Sepeda Motor Karburator, Baterai pada Sepeda Motor menjadi penyuplai tenaga ke *Engine Control Unit (ECU)*, *Injektor* dan *Fuel Pump*. Dengan kata lain “Kalau tidak ada Baterai, *ECU*, *Fuel Pump* dan komponen kelistrikan lain pada Sepeda Motor tidak akan berfungsi,”

2.3 Fingerprint

Sebagai Kunci kontak dari desain ini, dipilihnya *fingerprint* karena alat ini memiliki teknologi yang bagus untuk menghindari manipulasi data yang ada. *Fingerprint* merupakan salah satu metode teraman sejauh ini karena data sidik jari tidak dapat dimanipulasi. Sidik jari terbukti sangat akurat, aman, mudah dan nyaman untuk identifikasi data.

- Perennial nature, merupakan garis-garis pada sidik jari yang melekat pada kulit manusia seumur hidup.
- Immutability, merupakan sidik jari seseorang tidak pernah berubah, kecuali telah mengalami kecelakaan serius.
- Individuality, yaitu merupakan pola sidik jari, unik dan berbeda untuk setiap orang[8].



Gambar 2.1 : *fingerprint r307*

2.4 Arduino

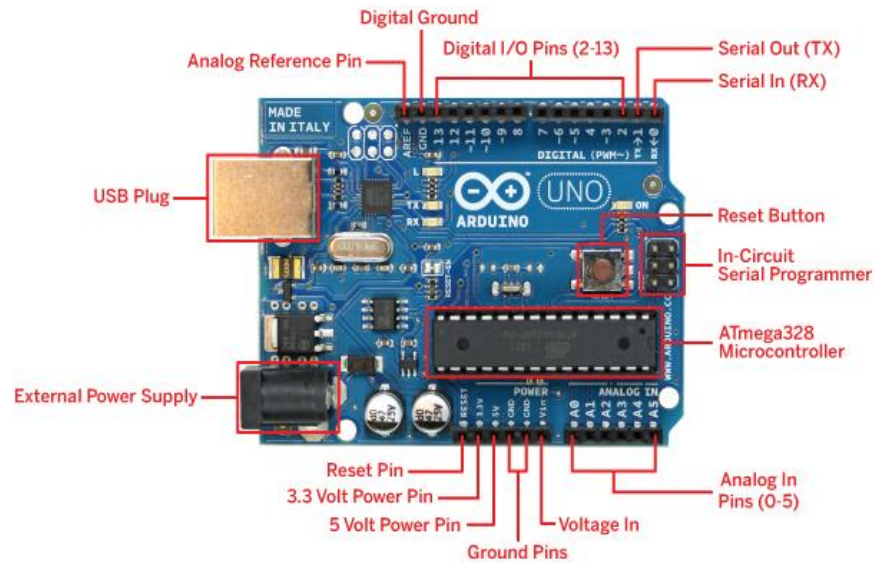
Arduino adalah perangkat keras berdesain sederhana menggunakan prosesor Atmel AVR yang didukung oleh modul I/O[7]. Arduino adalah komputer kecil yang dapat diprogram sebagai proses input dan output. Arduino pertama kali ditemukan pada 2005 oleh Massimo Banzi dan David Cuartielles, yang mencoba melakukan proyek pembuatan perangkat kontrol. Proyek ini menghasilkan

perangkat dengan harga yang lebih murah daripada harga perangkat yang tersedia saat itu [9]

Arduino yang terbaru adalah seri *UNO*. *Arduino Uno* berbeda dari semua board *arduino* sebelumnya, *Arduino Uno* terbaru sekarang adalah revisi ketiga atau biasa disebut *Arduino Uno R3*. Spesifikasi *Arduino Uno R3* dapat dilihat pada tabel 2.1 dan *Arduino uno R3* dapat dilihat pada gambar 2.3 berikut

Tabel 2.1: Spesifikasi *Arduino Uno R3*[10]

Spesifikasi	Keterangan
Mikrokontroler	ATmega328
Operasi Tegangan	5 Volt
Input Tegangan	7-12 Volt
Pin I/O Digital	14
Pin Analog	6
Arus DC tiap pin I/O	50 mA
Arus DC ketika 3.3V	50 mA
Memori flash	32 KB
SRAM	2 KB
EEPROM	1 KB

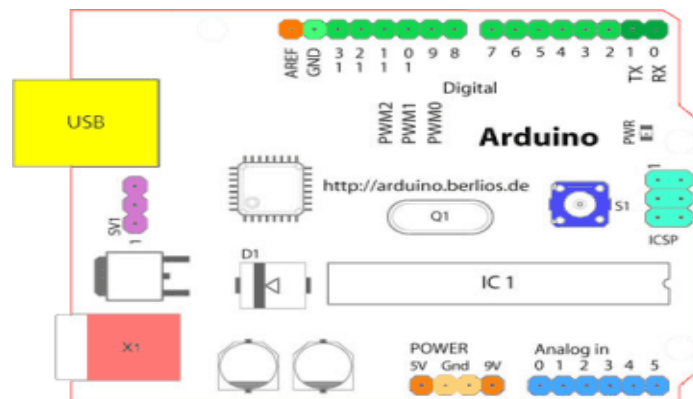


Gambar 2.2 :Arduino Uno [11]

Arduino Uno berukuran sebesar kartu kredit. Walaupun berukuran kecil seperti itu, papan tersebut mengandung mikrokontroler dan sejumlah input/output (I/O) yang memudahkan pemakai untuk menciptakan berbagai proyek elektronika yang di khususkan untuk menangani tujuan tertentu [12].

2.4.1 Bagian-bagian Dan Fungsi Arduino Uno

Arduino uno R3 memiliki beberapa bagian serta fungsi yang berfungsi untuk menjalankan alat tersebut untuk lebih jelasnya bisa dilihat pada gambar dan tabel berikut ini



Gambar 2.3 :Bagian-bagian Arduino Uno [13]

Gambar 2.3 di atas contoh sebuah papan Arduino tipe *USB*, bagian-bagiannya dapat dijelaskan di tabel 2.4 sebagai berikut.

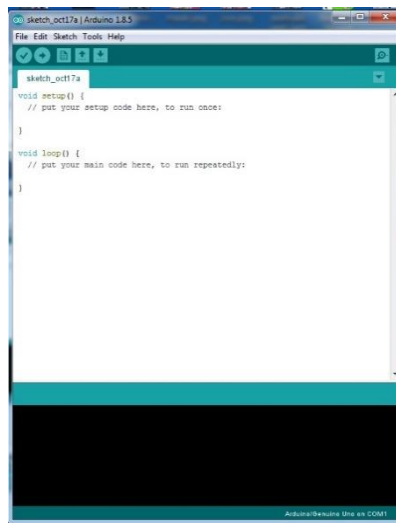
Tabel 2.2 : Bagian-bagian Arduino R3 [13]

Bagian	Fungsi
14 pin input/output digital (0-13)	Berfungsi sebagai input atau output, dapat diatur oleh program. Khusus untuk 6 buah pin 3, 5, 6, 9, 10 dan 11, dapat juga berfungsi sebagai pin analog output dimana tegangan output-nya dapat diatur. Nilai sebuah pin <i>output analog</i> dapat diprogram antara 0 – 255, dimana hal itu mewakili nilai tegangan 0 – 5V.
USB	Berfungsi untuk: • Memuat program dari komputer ke dalam papan • Komunikasi serial antara papan dan komputer • Memberi daya listrik kepada papan
Sambungan SV1	Sambungan atau <i>jumper</i> untuk memilih sumber daya papan, apakah dari sumber eksternal atau menggunakan <i>USB</i> . Sambungan ini tidak diperlukan lagi pada papan Arduino versi terakhir karena pemilihan sumber daya eksternal atau <i>USB</i> dilakukan secara otomatis.
Q1 – Kristal (quartz crystal oscillator)	Jika microcontroller dianggap sebagai sebuah otak, maka kristal adalah jantung-nya karena komponen ini menghasilkan detak-detak yang dikirim kepada <i>microcontroller</i> agar melakukan sebuah operasi untuk setiap detak-nya. Kristal ini dipilih yang berdetak 16 juta kali per detik (16MHz).
Tombol Reset S1	Untuk me-reset papan sehingga program akan mulai lagi dari awal. Perhatikan bahwa tombol reset ini bukan untuk menghapus program atau mengosongkan <i>microcontroller</i> .

Bagian	Fungsi
In-Circuit Serial Programming (ICSP)	Port ICSP memungkinkan pengguna untuk memprogram microcontroller secara langsung, tanpa melalui bootloader. Umumnya pengguna Arduino tidak melakukan ini sehingga ICSP tidak terlalu dipakai walaupun disediakan.
IC 1 – Microcontroller Atmega	Komponen utama dari papan Arduino, di dalamnya terdapat <i>CPU</i> , <i>ROM</i> dan <i>RAM</i> .
X1 – sumber daya eksternal	Jika hendak disuplai dengan sumber daya eksternal, papan <i>Arduino</i> dapat diberikan tegangan DC antara 9-12V.
6 pin input analog (0-5)	Pin ini sangat berguna untuk membaca tegangan yang dihasilkan oleh sensor analog, seperti sensor suhu. Program dapat membaca nilai sebuah pin input antara 0 – 1023, dimana hal itu mewakili nilai tegangan 0 – 5V.

2.4.2 IDE Arduino

IDE Arduino merupakan singkatan dari *Integrated Development Environment*, atau dalam bahasa sederhana adalah lingkungan terintegrasi yang digunakan untuk pengembangan. Lingkungan dinamai karena *Arduino* diprogram oleh perangkat lunak ini untuk melakukan fungsi yang tertanam oleh sintaks pemrograman [14]. *Arduino* menggunakan bahasa pemrogramannya sendiri, yang mirip dengan bahasa C. Bahasa pemrograman *Arduino* (Sketsa) telah diubah untuk memudahkan pemula untuk memprogram dari bahasa asli. Sebelum meluncurkan IC mikrokontroler *Arduino*, ia telah menyematkan program yang disebut boot loader yang bertindak sebagai perantara antara *compiler Arduino* dan mikrokontroler. [15] *Arduino Uno* ditunjukkan pada Gambar 2.4 di bawah ini.



Gambar 2.4. : IDE Arduino



Icon menu **verify** berfungsi untuk melakukan checking apakah kode yang dimasukan sudah sesuai dengan aturan pemrograman, program yang ditulis apakah ada yang salah atau error.



Icon menu **upload** berfungsi mengkompilasi program atau kode yang anda buat menjadi bahasa yang dapat dipahami oleh *Arduino*.



Icon menu **New** berfungsi untuk membuat *Sketch* baru



Icon menu **Open** Berfungsi untuk membuka *sketch* yang pernah kamu buat dan membuka kembali untuk dilakukan editing atau sekedar upload ulang ke *Arduino*.



Icon menu **Save** berfungsi untuk menyimpan *Sketch* yang telah anda buat.



Icon menu **serial monitor** Berfungsi untuk membuka serial monitor. Serial monitor disini merupakan jendela yang menampilkan data apa saja yang dikirimkan atau dipertukarkan antara arduino dengan sketch pada port serialnya. Serial Monitor ini sangat berguna sekali ketika kamu ingin membuat program atau melakukan *debugging* tanpa menggunakan LCD pada Arduino. Serial monitor ini dapat digunakan untuk menampilkan nilai proses, nilai pembacaan, bahkan pesan error[14].

2.4.3 Bahasa Pemrograman Arduino

Bahasa pemrograman *arduino* terdiri dari tiga bagian utama, yaitu struktur, variable dan fungsi;

2.4.3.1 Struktur Program Arduino

Struktur dasar bahasa pemrograman arduino cukup sederhana. Agar program dapat berjalan dengan baik, diperlukan setidaknya dua bagian atau fungsi yaitu *setup()* yang dipanggil hanya satu kali, biasanya untuk inisialisasi program (set input, dll) dan *loop ()* tempat untuk mengeksekusi program berulang kali, biasanya untuk membaca input atau memicu *output*. Berikut ini bentuk penulisannya:

Void setup()

```
{
//Statement;
```

```
}
```

Void loop()

```
{
//Statement;
}
```

Setup()

setup() hanya dipanggil satu kali saja saat program mulai berjalan. Fungsi *setup ()* berguna untuk melakukan inisialisasi mode pin atau memulai komunikasi serial. *Setup ()* ini harus ada meskipun tidak ada program yang akan dieksekusi. Berikut ini bentuk penulisannya:

Void **setup()**

```
{
pinMode(led, OUTPUT); //set led sebagai keluaran
}
```

Void loop()

```
{
//statement;
```

```
}
```

Loop()

Fungsi *loop()* ini mengulang program yang ada secara terus-menerus, sehingga program akan berubah dan merespon sesuai masukan. Berikut contoh penulisan fungsi *loop()*:

Void setup()

```
{
pinMode(led, OUTPUT); //set led sebagai keluaran
}
```

Void loop()

```
{
digitalWrite(led, HIGH); //set led on
delay (400); //tunda untuk ½ detik
digitalWrite(led, LOW); //set led off
delay (400); //tunda untuk ½ detik
}
```

2.4.3.2 Variabel

Variabel berfungsi untuk menyimpan nilai numerik dan memberi nama sesuai dengan program yang diberikan. Dengan menggunakan variabel, nilainya dapat diubah secara bebas. Suatu variabel perlu dideklarasikan terlebih dahulu, dan akan digunakan sebagai *input* pembaca untuk digunakan atau diberi nilai awal.

2.4.3.3 Fungsi

Function terdiri dari :

Fungsi Digital I/O

Fungsi untuk digital I/O ada tiga buah yaitu *pinMode(pin, mode)*, *digitalWrite(pin, value)*, dan *int digitalRead(pin)*.

Fungsi Analog I/O

Fungsi untuk analog I/O ada tiga buah yaitu *analogReference(type), int analogRead(pin)*, dan *analogWrite(pin, value)-PWM*.

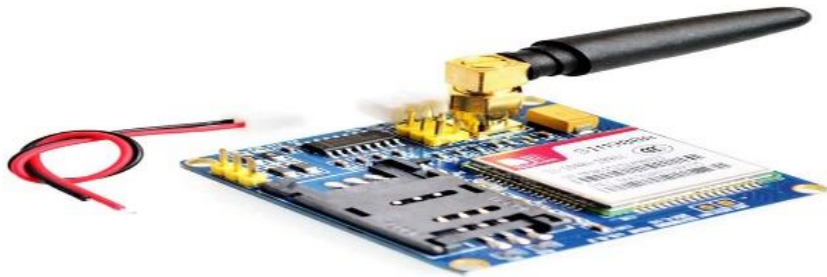
Fungsi Waktu

Fungsi waktu terdiri dari *unsigned long millis ()*, *delay(ms)* Dan *delayMicroseconds(us)*.

Fungsi Matematika

Fungsi matematika terdiri dari *min(x,y)*, *max(x,y)*, *abs(x)*, *sqrt(x)* dan *pow(base, exponent)* [16]

2.5 Modul GSM SIM900A



Gambar 2.5 : Modul GSM SIM900A v.2 [17]

Berikut ini spesifikasi Module GSM SIM900A :

- Quad-band 850/900/1800/1900MHz
- Terhubung dengan jaringan GSM global menggunakan 3G dan 4GSIM (Telkomsel, Indosat, Three)
- Voice call dengan external 8 speaker dan electret microphone.
- Kirim dan terima SMS.
- Kirim dan terima GPRS data (TCP/IP, HTTP, etc.)
- GPIO ports, misalnya untuk buzzer dan vibrational motor.
- AT command interface dengan deteksi "auto baud".

Modul SIM yang digunakan pada penelitian ini, yaitu Modul SIM900A. GSM/GPRS adalah bagian yang berfungsi untuk berkomunikasi antara pemantau utama dengan Handphone. AT-Command adalah perintah yang dapat diberikan modem GSM/CDMA seperti untuk mengirim dan menerima data berbasis GSM/GPRS, atau mengirim dan menerima SMS. SIM800L GSM/GPRS dikendalikan melalui perintah AT. AT-Command adalah sebuah kumpulan perintah yang digabungkan dengan karakter lain setelah karakter "AT" yang biasanya digunakan pada komunikasi serial. Perintah AT-Command dimulai dengan karakter "AT" atau "at" dan diakhiri dengan kode (0x0d) [18]. Perintah-perintah AT-Command dapat di lihat pada tabel 2.3 berikut.

Tabel 2.3 : Perintah - perintah AT-Command [18]

Perintah	Fungsi
AT+CPBF	Mencari nomor telephone yang tersimpan
AT+CPBR	Membaca buku telephone
AT+CPBW	Menulis nomor telephone di buku telephone
AT+CMGF	Menyeting mode SMS teks atau PDU
AT+CMGF=0	Menyeting mode PDU
AT+CMGF=1	Menyeting mode SMS teks
AT+CMGS	Mengirim sebuah perintah SMS
AT+CMGR	Membaca sebuah pesan
AT+CMGR=1	Membaca sebuah pesan di alamat 1
AT+CMG	Melihat semua daftar sms yang ada
AT+CMGD	Menghapus sebuah SMS
AT+CMNS	Menyeting sebuah lokasi penyimpanan SMS
AT+COPS?	Untuk mengetahui sebuah nama provider kartu GSM
AT+CSCA	Untuk mengetahui alamat SMS Center
AT+CGMI	Untuk mengetahui nama dan jenis ponsel
AT+CGMM	Untuk mengetahui jenis ponsel
AT+CBC	Untuk mengetahui level baterai

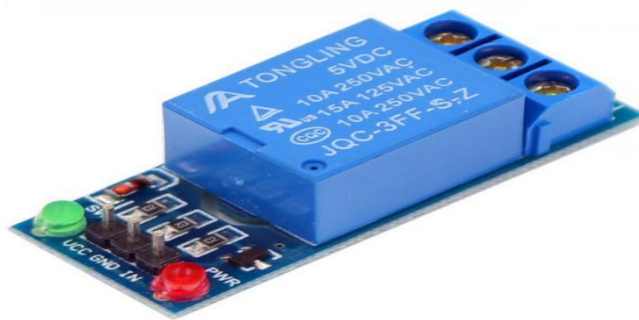
2.6 Relay JQC 3F

Relay adalah sakelar yang dioperasikan secara elektrik dan merupakan komponen elektromekanis yang terdiri dari dua bagian utama, yaitu solenoid (koil) dan mekanisme (satu set sakelar kontak / sakelar). *Relay* menggunakan prinsip elektromagnetik untuk menggerakkan saklar kontak sehingga dapat menghantarkan listrik pada tegangan yang lebih tinggi dengan arus listrik kecil (daya rendah) [19].

Pada dasarnya, *Relay* terdiri dari beberapa komponen dasar yaitu :

- *Electromagnet (Coil)*
- *Armature*
- *Switch Contact Point (Saklar)*
- *Spring*

Berikut ini merupakan gambar *Relay*JQC 3F :



Gambar 2.6 :Relay JQC 3F

2.7 Buzzer

Buzzer adalah komponen elektronik yang mengubah getaran listrik menjadi getaran suara. Prinsip pengoperasian *buzzer* pada dasarnya sesuai dengan loudspeaker. Oleh karena itu, bel juga terdiri dari koil yang dipasang pada diafragma, dan kemudian koil dialiri arus sehingga menjadi elektromagnet. Koil ditarik masuk atau keluar tergantung pada arah arus dan polaritas magnetik, karena koil dipasang pada membran. Dengan setiap gerakan kumparan, membran bergerak maju dan mundur, sehingga udara bergetar dan menghasilkan suara. *Buzzer* biasanya digunakan sebagai indikator bahwa proses telah selesai atau kesalahan telah terjadi pada perangkat (alarm) [20].

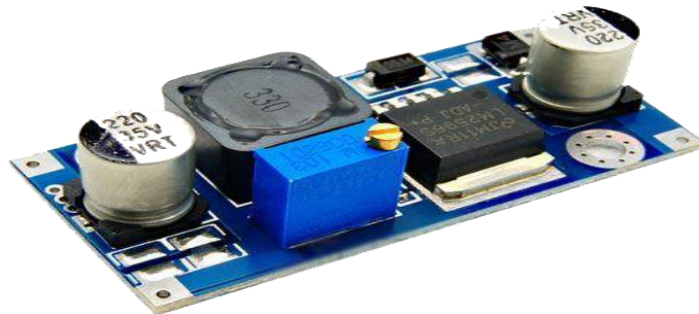


Gambar 2.7 :*Buzzer*

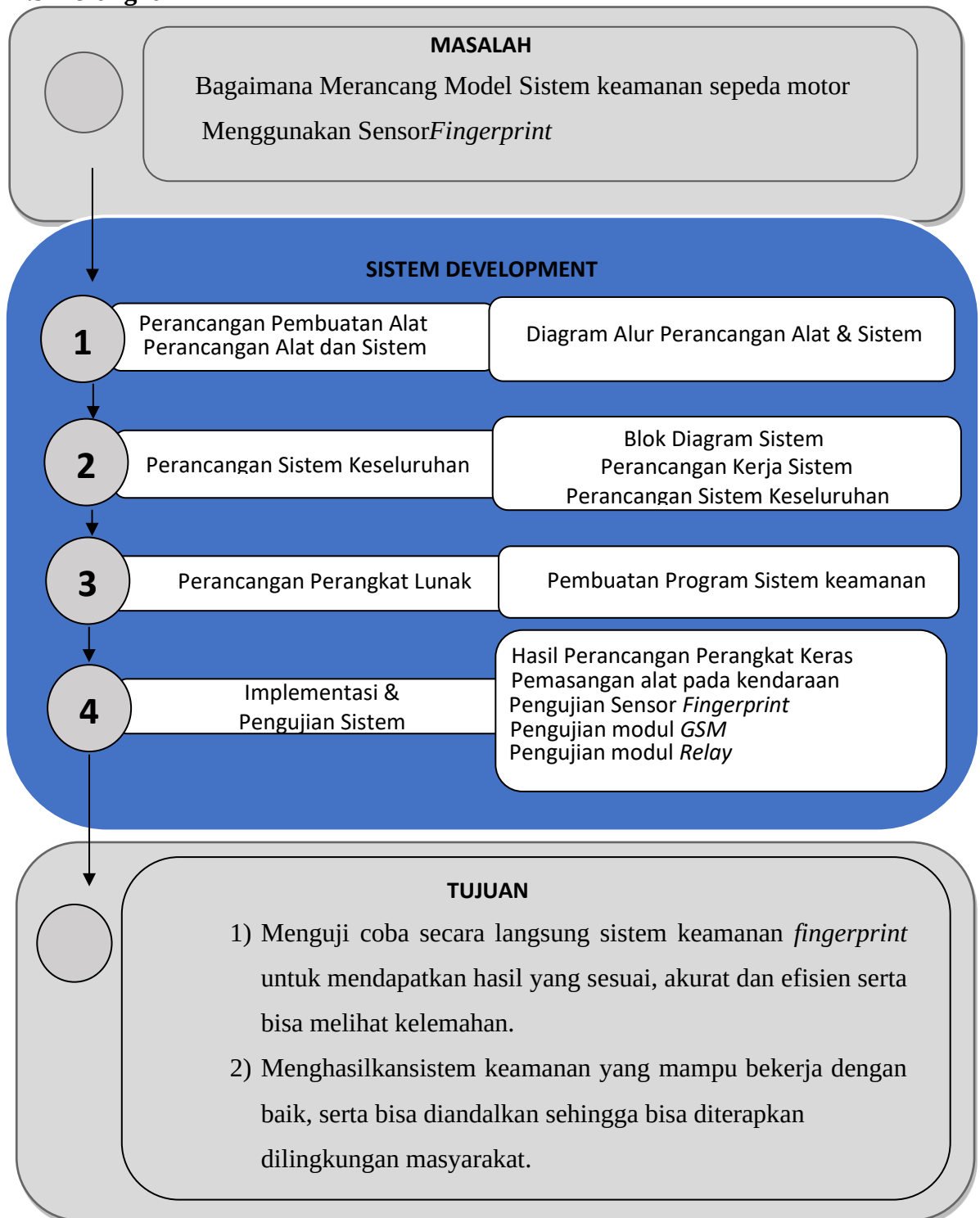
2.8 Inverter (Adaptor)

Inverter adalah salah satu perangkat elektronik yang digunakan untuk mengubah tegangan DC (arus searah) menjadi tegangan AC (arus bolak-balik). Output dari inverter dapat berupa tegangan bolak-balik dengan gelombang sinus, gelombang persegi dan gelombang sinus termodifikasi. Inverter ini bertindak sebagai pemasok daya darurat baik di kendaraan maupun di rumah ketika listrik padam. Selain itu, inverter DC di masa depan akan memainkan peran penting dalam konversi DC dari sumber energi terbarukan seperti sel surya menjadi AC yang digunakan setiap hari [21].

Gambar 2.8 :*Inverter*



2.9 Kerangka Pikir



BAB III

METODE PENELITIAN

3.1 Jenis, Metode, Subjek, Objek, Waktu, dan Lokasi Penelitian

Dari perspektif aplikasi, penelitian ini adalah penelitian terapan.

Metode pengembangan sistem yang dipakai adalah model *prototype* digunakan karena penyajian aspek perangkat keras yang akan dibuat tampaknya cepat bagi pengguna. Selanjutnya, *prototype* dievaluasi oleh kedua belah pihak, sehingga penyaringan persyaratan pengembangan perangkat keras dapat dilakukan dengan cepat sesuai dengan keinginan dan persyaratan penelitian ini dari beberapa fase, mulai dengan akuisisi data, alat desain dan sistem, desain alat-alat manufaktur, desain perangkat lunak, Implementasi dan pengujian sistem, persiapan laporan.

Subjek penelitian ini adalah Penerapan Keamanan Sepeda Motor. Penelitian ini akan dilaksanakan selama lima bulan yang berlokasi pada laboratorium Universitas Icshan Gorontalo.

3.2 Alat dan Bahan Penelitian

Peralatan dan bahan – bahan yang digunakan dalam melakukan penelitian Tugas akhir ini dapat dilihat pada tabel 3.2

Tabel 3.1 :Daftar Alat dan Bahan

No	Nama Alat dan Bahan	Fungsi
1.	Laptop dan Software Arduino	Sebagai Pembuat Program Arduino
2.	<i>fingerprint</i> R307	Sebagai Penginputan data/Sidik Jari
3.	Arduino Uno R3	Sebagai mikrokontroler pengolah data
4.	Modul GSM SIM900A	Sebagai komunikator antara alat dengan ponsel
5.	Relay JQC 3F	saklar atau <i>switch</i> elektrik
6.	YTX7L BS-12V, 6A	Sebagai <i>power supply system</i>

No	Nama Alat dan Bahan	Fungsi
7.	Kabel <i>Jumper</i>	Sebagai penghubung antar komponen
8.	<i>Smartphone</i>	Alat Pemantau Kendaraan
9.	<i>Buzzer</i>	Sebagai Notifikasi Sistem
10.	<i>Inverter</i>	Sebagai pengubah tegangan pada <i>power supply</i>

3.3 Metode Penelitian

Penelitian dapat diselesaikan dengan melalui beberapa tahapan-tahapan pelaksanaan, yaitu:

3.3.1 Sumber Data

Sumber data pada penelitian ini adalah menggunakan Library Research yang merupakan cara mengumpulkan data dari beberapa buku, jurnal, skripsi, tesis maupun literatur lainnya yang dapat dijadikan acuan pembahasan dalam masalah ini. Penelitian ini terkait pada sumber-sumber data online atau internet ataupun hasil dari penelitian sebelumnya sebagai bahan referensi bagi peneliti selanjutnya.

3.3.2 Pengumpulan Data

3.3.2.1 Observasi

Studi lapangan (observasi) merupakan teknik pengumpulan data dengan langsung terjun ke lapangan untuk mengamati permasalahan yang terjadi secara langsung di tempat kejadian secara sistematis kejadian-kejadian, perilaku, objek-objek yang dilihat dan hal-hal lain yang diperlukan dalam mendukung penelitian yang sedang berlangsung. Dalam penelitian ini, peneliti melakukan pengamatan terhadap kasus-kasus pencurian kendaraan yang terjadi selama penelitian berlangsung dan tempat-tempat yang lain yang dianggap penting yang berhubungan dengan penelitian ini.

3.3.2.2 Wawancara

Wawancara adalah teknik pengumpulan data atau tatap muka yang mencakup pertanyaan dan jawaban antara pengumpul data terhadap narasumber/sumber data. Sumber data penelitian adalah para ahli yang sudah lama berkecimpung di bidang elektronik.

3.3.2.3 Studi Literatur

Pengumpulan data dengan cara mengumpulkan literatur, jurnal, paper dan bacaan-bacaan yang ada kaitannya dengan judul penelitian.

3.3.3 Teknik Pengolahan dan Analisis Data

3.3.3.1 Pengolahan Data

Pengolahan data diartikan sebagai proses mengartikan data-data lapangan yang sesuai dengan tujuan, rancangan, dan sifat penelitian. Metode pengolahan data dalam penelitian ini yaitu:

1. Reduksi Data adalah mengurangi atau memilah-milah data yang sesuai dengan topik dimana data tersebut dihasilkan dari penelitian.
2. Koding data adalah penyesuaian data diperoleh dalam melakukan penelitian kepustakaan maupun penelitian lapangan dengan pokok pada permasalahan dengan cara memberi kode-kode tertentu pada setiap data tersebut.

3.3.3.2 Analisis Data

Teknik analisis data bertujuan menguraikan dan memecahkan masalah yang berdasarkan data yang diperoleh. Analisis yang digunakan adalah analisis data kualitatif. Analisis data kualitatif adalah upaya yang dilakukan dengan jalan mengumpulkan, memilah-milah, mengklasifikasikan, dan mencatat yang dihasilkan catatan lapangan serta memberikan kode agar sumber datanya tetap dapat ditelusuri.

3.3.4 Teknik Pengujian

Metode pengujian yang digunakan pada penelitian ini adalah metode pengujian langsung yaitu dengan menggunakan pengujian Black Box. Digunakan untuk menguji fungsi-fungsi khusus dari perangkat yang dirancang. Kebenaran perangkat lunak yang diuji hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut. Dari keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai dapat diukur sekaligus dapat diketahui kesalahan-kesalahannya.

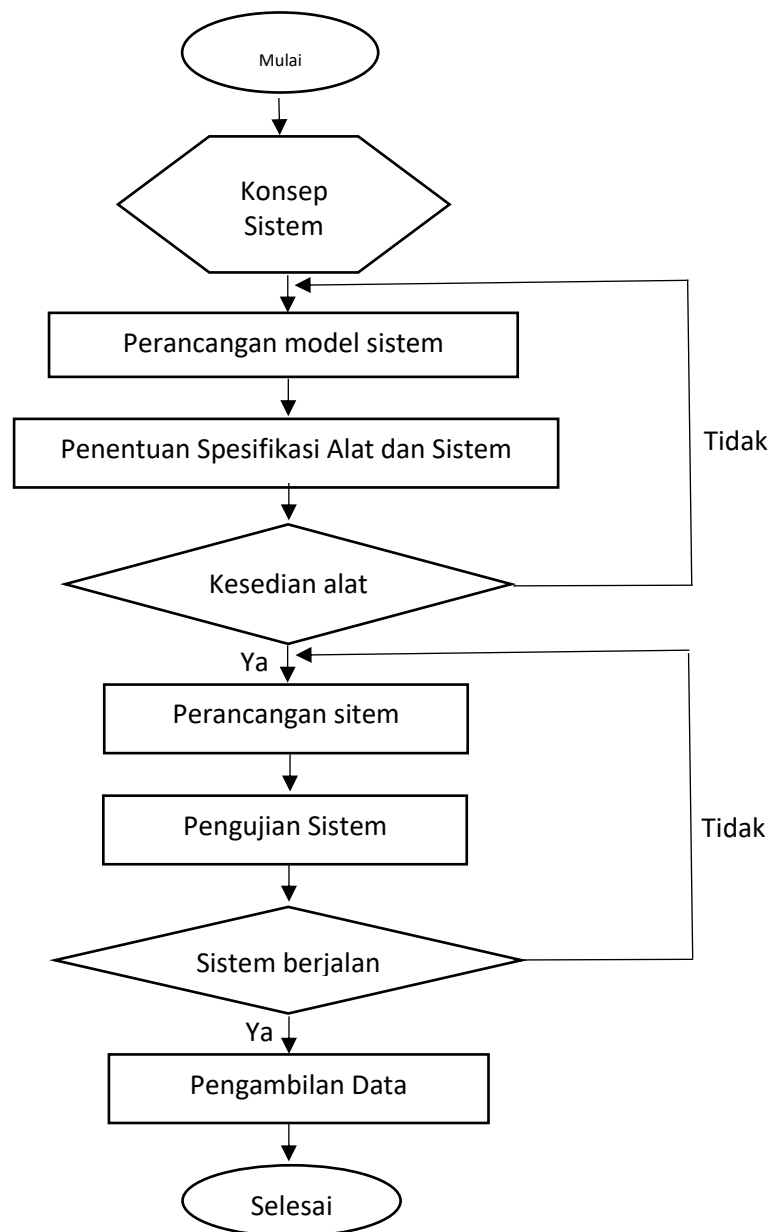
3.3.5 Pembuatan Laporan

Setelah melakukan pengujian dan evaluasi sistem, maka langkah selanjutnya melakukan penyusunan laporan akhir sesuai dengan standar dan format yang ditentukan, yang berguna untuk pengembangan sistem selanjutnya.

BAB IV

PERANCANGAN SISTEM

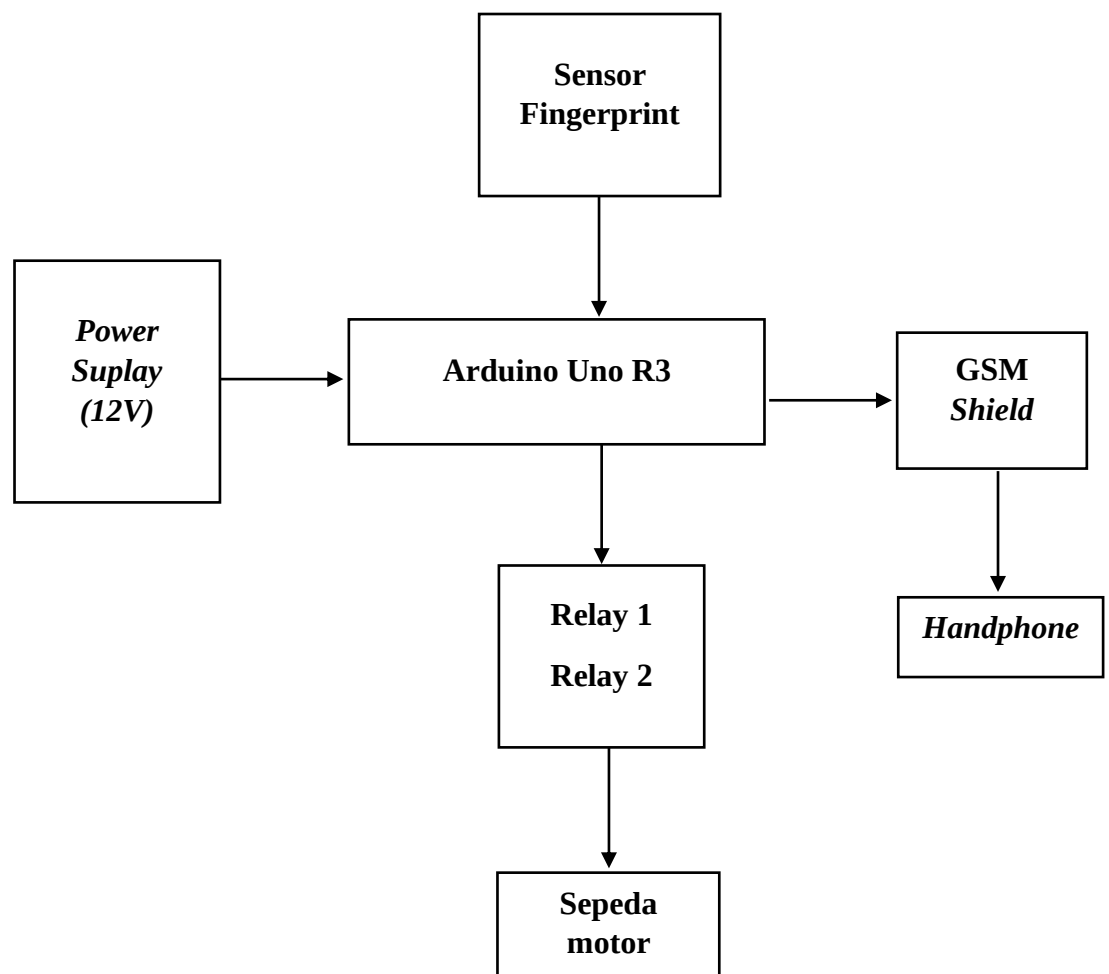
4.1 Perancangan Alat dan Sistem



Gambar 4.1 : *Diagram Alir Perancangan Alat dan Sistem*

4.1.1 Perancangan Blok Diagram

Untuk menjelaskan perancangan sistem yang dilakukan dalam mewujudkan penelitian tentang Rancang Bangun Sistem Monitoring Sampah berbasis Arduino Uno dengan keluaran informasi kepenuhan sampah, pertama dengan membuat blok diagram kerja sistem.



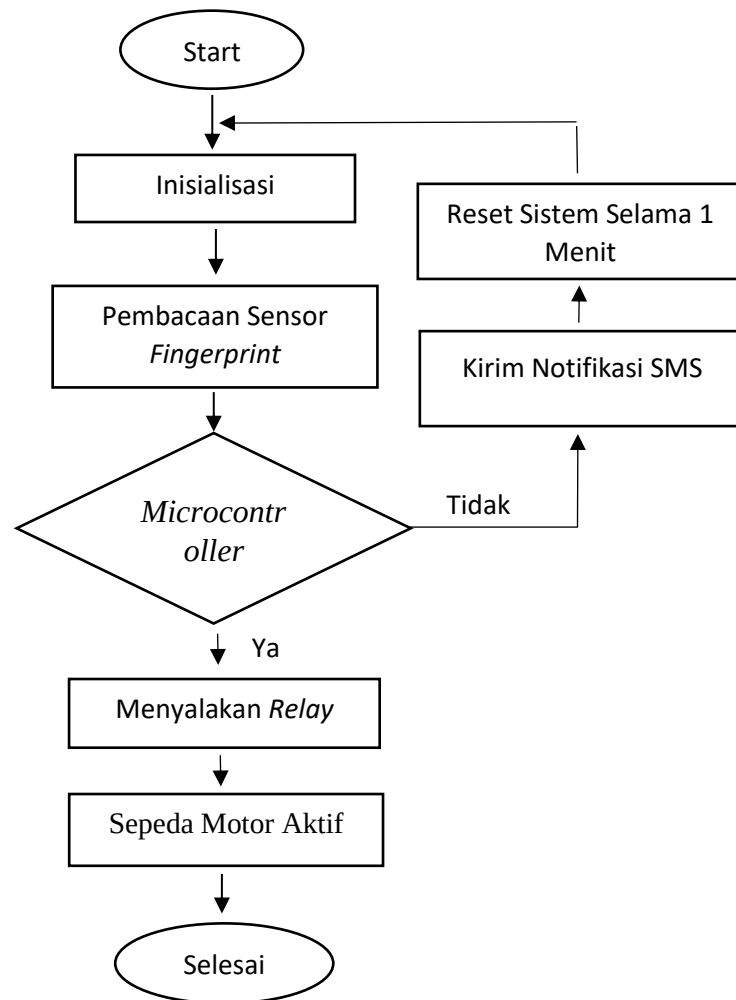
Gambar 4.1: Blok Diagram Kerja Sistem

Arduino Uno sebagai komponen utama berfungsi sebagai pengolah data masukan dan keluaran dengan di suplai sumber daya sebagai aliran listrik untuk menghidupkan arduino uno dan komponen lainnya. Sensor *fingerprint* berfungsi sebagai kunci kontak dan *switch* starter pada kendaraan. Kegunaan *Relay* berfungsi sebagai Pemutus sirkuit koneksi arus dan *GSM Shield* sebagai komunikator penghubung antara sistem dan pengguna dengan megirimankan pesan melalui *Handphone*.

4.1.2 Perancangan Kerja Sistem

Secara umum sistem monitoring sampah ini terbagi dua yaitu mendeteksi tingkat kepenuhan sampah oleh modul sensor, pengiriman data oleh modul transmitter dan tampilan pemantauan pada ponsel. Tahapan perancangan adalah sebagai berikut :

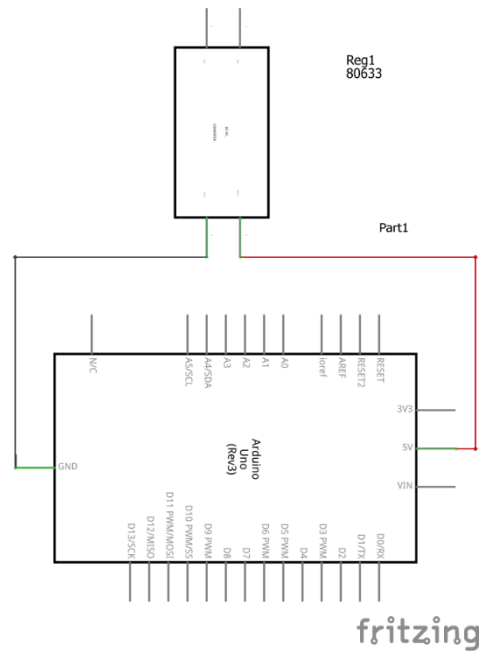
1. Modul sensor *fingerprint* akan mendeteksi data sidik jari yang terdaftar, kemudian data diolah oleh *Arduino* Uno R3.
2. Modul Relay akan menyala dan pada posisi standby sepeda motor siap di nyalakan mesinnya (starter).
3. Modul GSM SIM900A akan mengirimkan data yang didapat dari pengolahan arduino ke dalam ponsel pemantau. Setelah itu, modul SIM900A akan melakukan pengiriman sms jika terjadi pengimputan data yang salah. Gambar 4.3 menunjukkan diagram alir kerja sistem yang digunakan untuk alat ini.



Gambar 4.3 : Diagram Alir Kerja Sistem

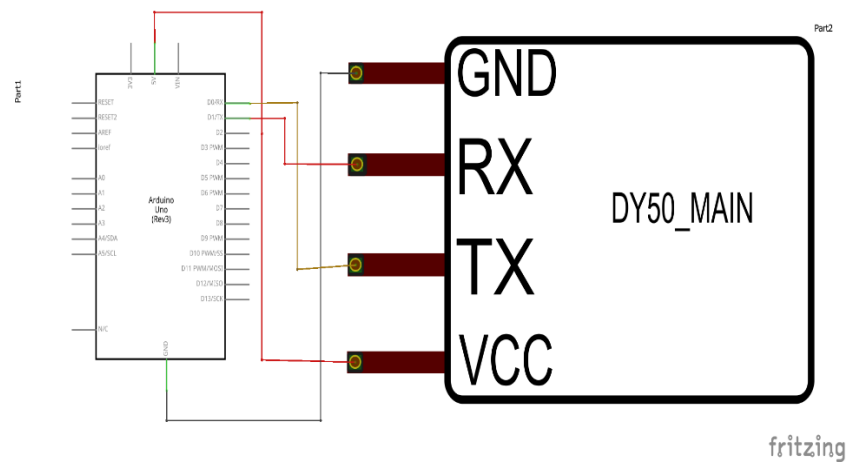
4.2 Perancangan Perangkat Keras

1. Perancangan Rangkaian Sumber tegangan diambil dari baterai sepeda motor 12v dan kemudian masuk ke dalam *Inverter*. *Inverter* disini berfungsi sebagai pengaman untuk menghindari kesalahan supply tegangan yang terbalik.



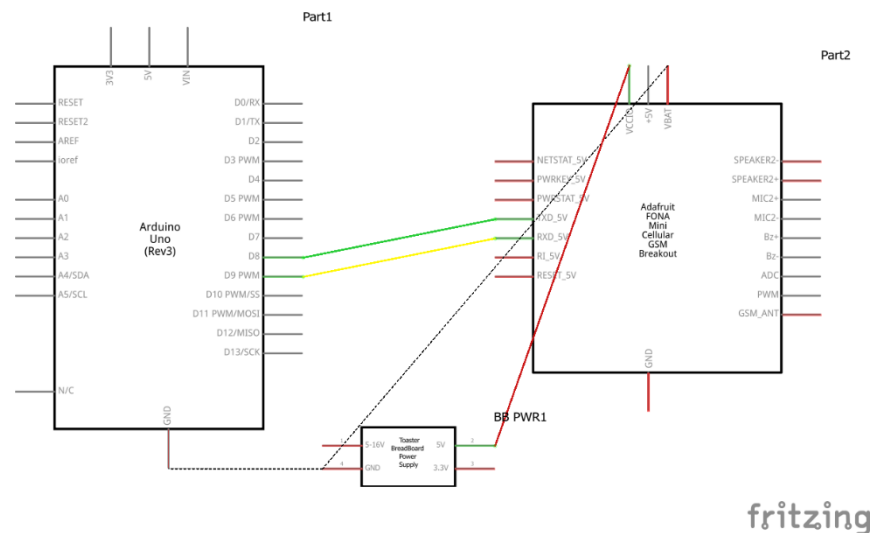
Gambar 4.2 : Perancangan *Inverter*

2. Perancangan Blok input Fingerprint Sensor Pada perancangan sistem ini, sensor yang digunakan untuk mendeteksi sidik jari adalah Fingerprint dimana dalam perancangan sistem ini, sensor akan mendeteksi sidik jari yang akan digunakan untuk menghidupkan kelistrikan motor, dan menghidupkan mesin motor, serta mematikan kelistrikan motor.



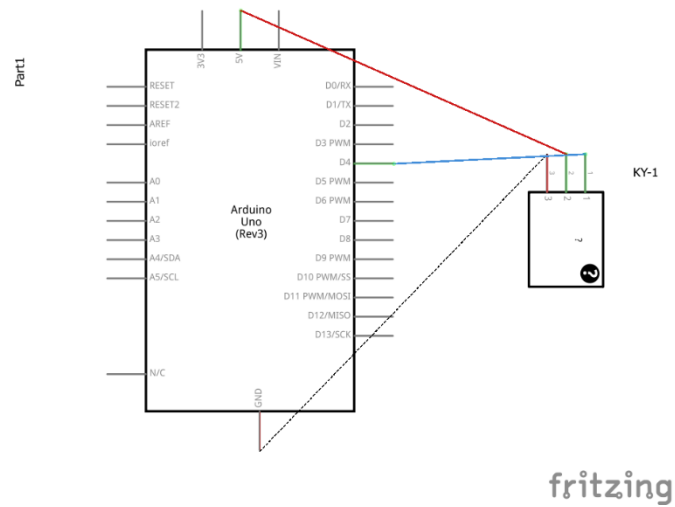
Gambar 4.3 :Perancangan Fingerprint Sensor

3. Perancangan modul *GSM SIM900A*. Pada langkah ini juga ditentukan nomor ponsel yang akan mendapatkan peringatan dari modul *GSMSIM900A*. Kemudian dari modul *GSM SIM900A* akan melakukan pengiriman informasi berupa SMS yang berisikan pemberitahuan ke nomor ponsel yang telah ditentukan sebelumnya. Sehingga pada saat sensor sidik jari mendeteksi data yang tidak cocok maka modul *GSMSIM900A* akan melakukan pengiriman pesan ke ponsel yang telah ditentukan sebelumnya



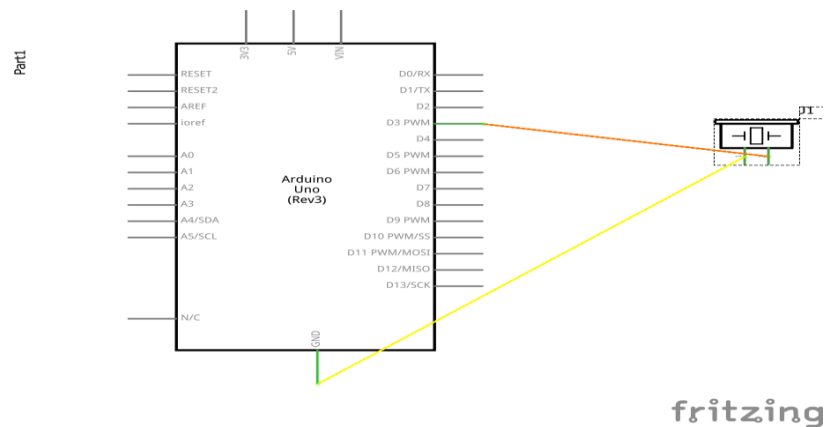
Gambar 4.4 : Perancangan Modul GSM SIM

4. Perancangan Rangkaian *Relay*. *Relay* berfungsi sebagai saklar untuk kelistrikan pada sepeda motor dan juga sebagai starter untuk sepeda motor.



Gambar 4.5 : Perancangan Modul Relay

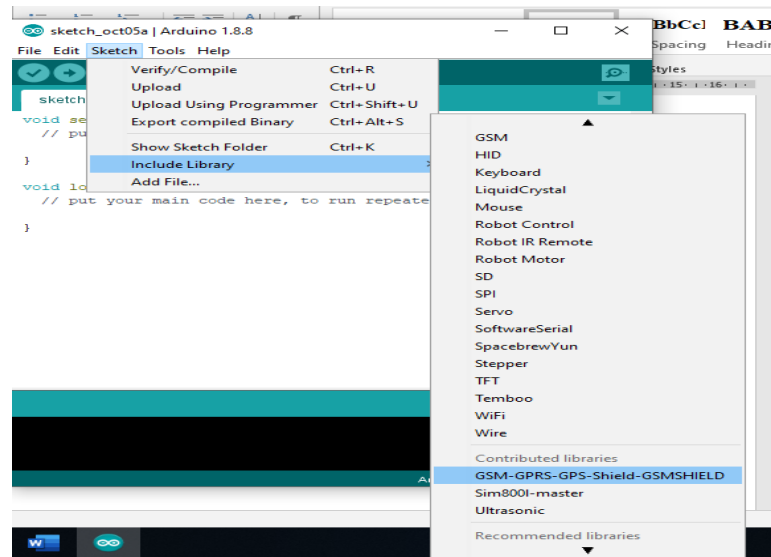
5. Perancangan rangkaian *buzzer*. *Buzzer* berfungsi sebagai notifikasi sistem pada sepeda motor



Gambar 4.6 : Perancangan Buzzer

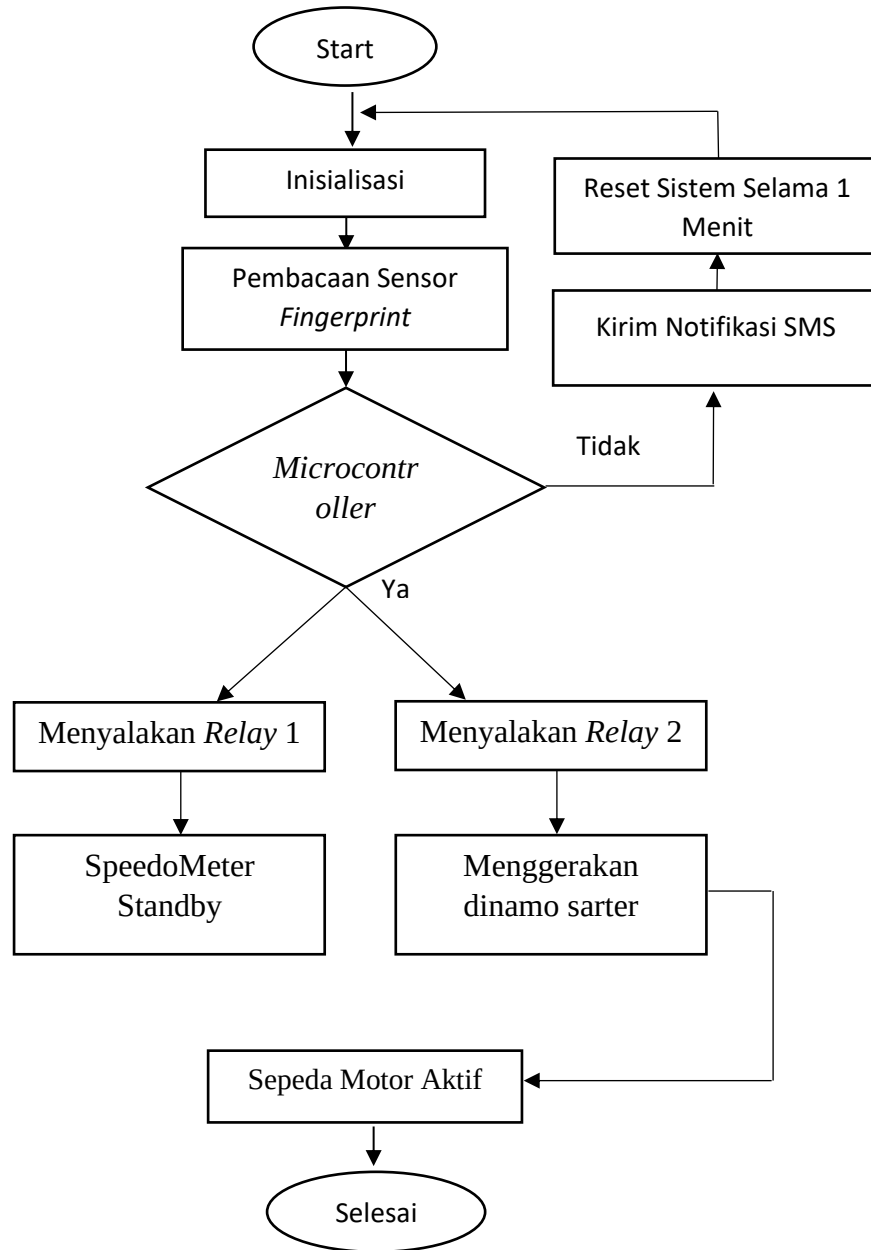
4.3 Perancangan Perangkat Lunak

Untuk melakukan perancangan perangkat lunak, Arduino memiliki perangkat lunak tersendiri, yang tersedia di website resmi milik Arduino. Untuk Bahasa pemrograman Arduino yaitu Bahasa C/C++ dan tambahan beberapa Library Sistem keamanan Sepeda motor yang berbasis Arduino.



Gambar 4.7: Tampilan Perangkat Lunak Arduino

4.3.1 Flowchart (Alur Program)



Gambar 4.8 : Flowcart kerja sistem

Pada saat sistem dalam keadaan menyala, sistem akan melakukan proses inisialisasi bagian-bagian pada rangkaian sistem mulai dari inisialisasi header, deklarasi variable, port yang digunakan , serta fungsi-fungsi lainnya. Ketika alat mulai berkerja maka secara otomatis sensor akan bekerja , dan secara bersamaan sensor siap mendeteksi sidik jari dari sensor *fingerprint*, kemudian akan memicu Arduino UNO untuk mengambil data berupa value yang selanjutnya akan dilanjutkan ke *relay* dan diteruskan pengguna melalui modul *GSM Shield*.

4.3.2 Kode Program Sensor

Pembuatan kode program sensor ditujukan untuk memastikan apakah sensor akan membaca objek yang ada di dalam tempat sampah atau tidak. Untuk itu perlu dilakukan testing pada program sensor sehingga menghasilkan nilai yang diinginkan. Berikut tampilan kode program sensor:



```

fingerprint | Arduino 1.8.10
File Edit Sketch Tools Help

fingerprint$

#include <Adafruit_Fingerprint.h>
#include <SoftwareSerial.h>

uint8_t id;
const int buzzer = 3;
int alamat=0;
int relay1=4;
int relay2=5;
SoftwareSerial mySerial(1, 2);
Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);

const int enroll = 8;
const int deleteall = 9;
const int ground1 = 12;
const int vcc1 = 7;

void setup() {
  Serial.begin(57600);
  pinMode(buzzer, OUTPUT);
  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);
  digitalWrite(ground1, LOW);
  digitalWrite(vcc1, HIGH);
  finger.begin(57600);

  pinMode(enroll, INPUT);
  pinMode(deleteall, INPUT);

  pinMode(ground1, OUTPUT);
  pinMode(vcc1, OUTPUT);

```

Gambar 4.9 : Testing Kode Program Sensor fingerprint

4.3.3 Kode Program GSM SIM900A

Pembuatan kode program *GSM Shield* dilakukan agar modul bisa berkomunikasi, pada tahap ini juga nomor ponsel pengguna akan dimasukan serta nama pesan apa yang akan di masukan kedalam program.



```

GSM_GPRS_Library_Call | Arduino 1.8.10
File Edit Sketch Tools Help

GSM_GPRS_Library_Call$

#include "SIM900.h"
#include <SoftwareSerial.h>
#include "sms.h"
#include "call.h"

CallGSM call;
SMSSMS sms;

char number[20];
byte stat=0;
int value=0;
int pin=1;
char value_str[5];

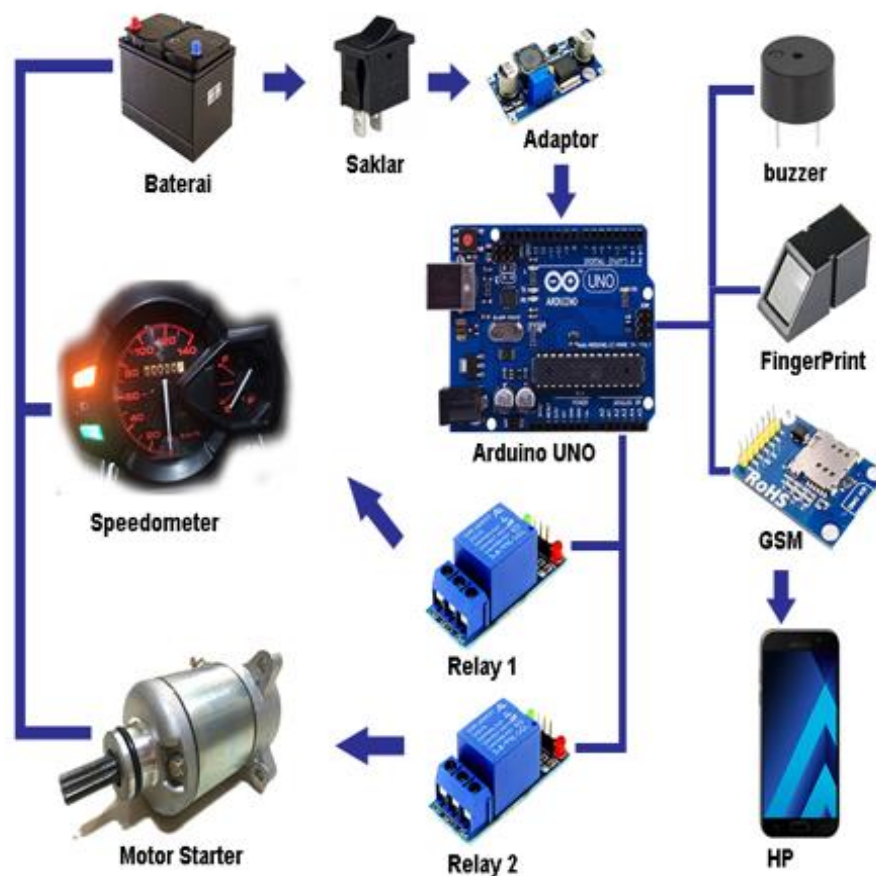
void setup()
{
    pinMode(pin, INPUT);
    //Serial connection.
    Serial.begin(9600);
    Serial.println("GSM Shield testing.");
    //Start configuration of shield with baudrate.
    //For http uses is recommended to use 4800 or slower.
    if (gsm.begin(2400))
        Serial.println("\nstatus=READY");
    else Serial.println("\nstatus=IDLE");
};

void loop()
{
    //Chekcs status of call
    stat=call.CallStatusWithAuth(number,1,3);
  
```

Gambar 5.0: Tampilan Program Modul GSM

4.4 Perancangan sistem Keseluruhan

Perancangan sistem keseluruhan merupakan gabungan dari rangkaian modul dengan Rangkaian *Arduino Uno R3* untuk memproses data, sensor *fingerprint* untuk mendeteksi sidik jari, *relay* sebagai penghubung dan pemutus arus, *GSM Shield* sebagai komunikator penghubung antara sistem dan *handphone*, *batteray* sebagai sumber daya sistem serta alat pendukung lainnya. Perancangan sistem keseluruhan juga menentukan tata letak dari rangkaian modul yang akan dipasang.



Gambar 5.1: Perancangan sistem keseluruhan

Setelah perancangan keseluruhan sistem selesai maka proses selanjutnya melakukan proses uji coba sistem, pada tahapan ini seluruh komponen sistem akan di uji. Baik itu perangkat keras maupun perangkat lunak.

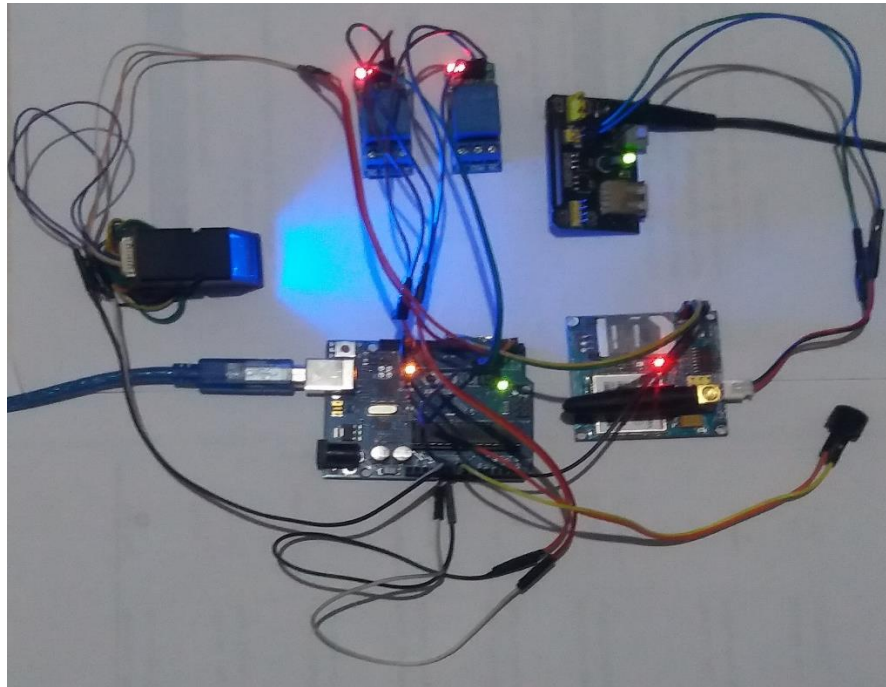
BAB V

IMPLEMENTASI DAN PENGUJIAN SISTEM

5.1 Implementasi

5.1.1 Hasil Perancangan Perangkat Keras

Perancangan perangkat keras adalah penggabungan keseluruhan alat menjadi sebuah sistem yang saling terhubung. Berikut gambar hasil perancangan alat keseluruhan.

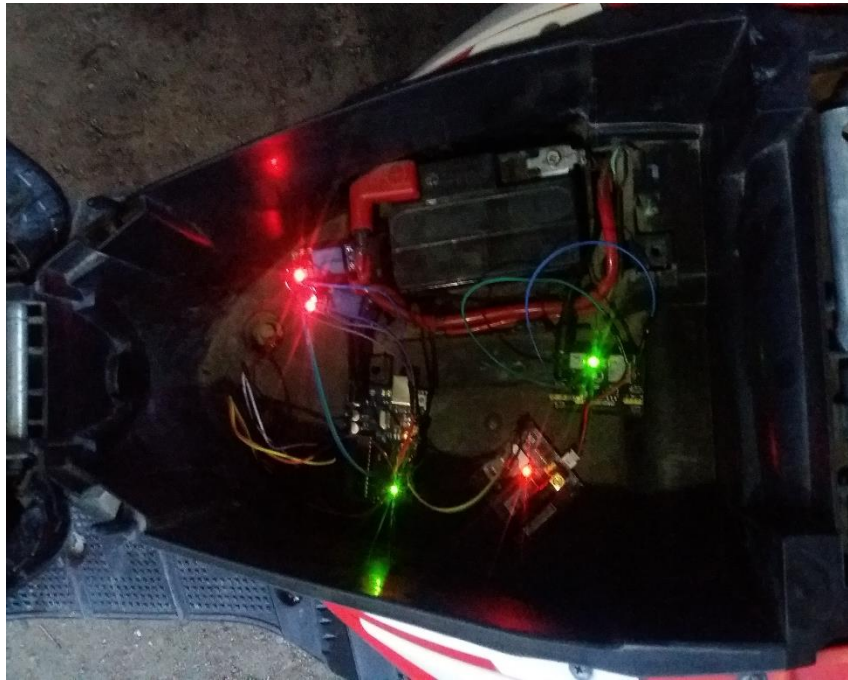


Gambar 5.2 : Hasil Rangkaian Perangkat Keras

Dari gambar 5.1 dapat dilihat perancangan alat keseluruhan yaitu berupa bentuk fisik dari sebuah sistem yang terhubung antara satu dengan yang lainnya. Yang terdiri dari. 1 buah *Arduino Uno R3*, 1 buah Sensor *Fingerprint FPM10A*, 2 buah *Relay JQC3FF*, 1 buah Modul *GSM SIM 900A*. Rancangan alat ini nantinya akan di tempatkan pada sebuah sepeda motor.

5.1.2 Pemasangan Alat Pada Sepeda Motor

Pada tahapan ini adalah penggabungan rangkaian alat dengan sepeda motor sebagai objek utama dari sistem ini. Sepeda motor yang digunakan merupakan jenis sepeda motor terbaru (injeksi).



Gambar 5.1: Pemasangan Alat pada Kendaraan

Pada gambar diatas menerangkan bahwa seluruh rangkaian alat dari Sistem Keamanan Sepeda Motor Besrbasis *Microconroller* Menggunakan *Fingerprint* dipasang pada sebuah sepeda motor.

5.2 Pengujian Sistem

Pada tahapan ini adalah, tahapan dimana sebuah sistem yang sudah dibuat akan diuji, melalui proses eksekusi perangkat keras dan perangkat lunak untuk melihat apakah sistem berjalan sesuai yang diinginkan oleh peneliti atau sistem mengalami sebuah masalah.

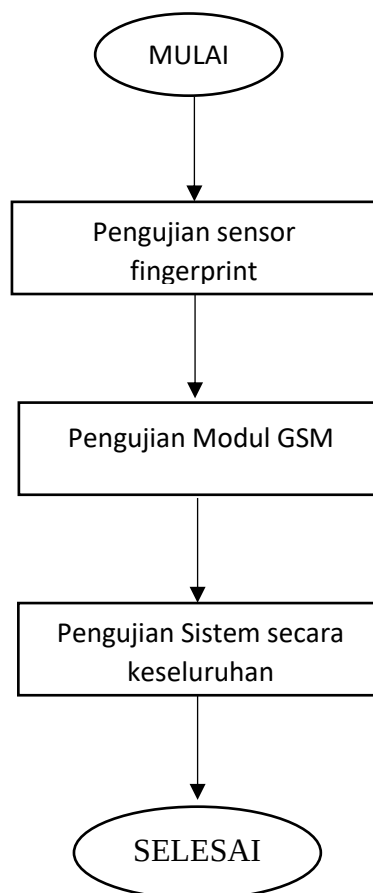
Adapun pengujian sistem yang digunakan adalah Black Box. Pengujian Black Box yaitu menguji perangkat dari segi spesifikasi fungsional tanpa menguji desain dan kode program. Pengujian dimaksudkan untuk mengetahui apakah fungsi fungsi dan keluaran sudah berjalan sesuai keinginan peneliti.

Dalam melakukan pengujian, tahapan yang dilakukan pertama adalah menguji perangkat-perangkat inputan, yaitu pengujian terhadap sensor fingerprint dan modul GSM, kemudian pengujian terhadap sistem secara keseluruhan.

Adapun tahapan-tahapan dalam melaksanakan pengujian sistem secara keseluruhan adalah sebagai berikut:

1. Menyiapkan sebuah Sepeda motor
2. Menyiapkan sebuah *HandPhone*
3. Melakukan proses pengujian terhadap sistem
4. Mencatat hasil pengujian.

Berikut ini adalah langkah-langkah dalam melakukan proses pengujian sistem secara keseluruhan.

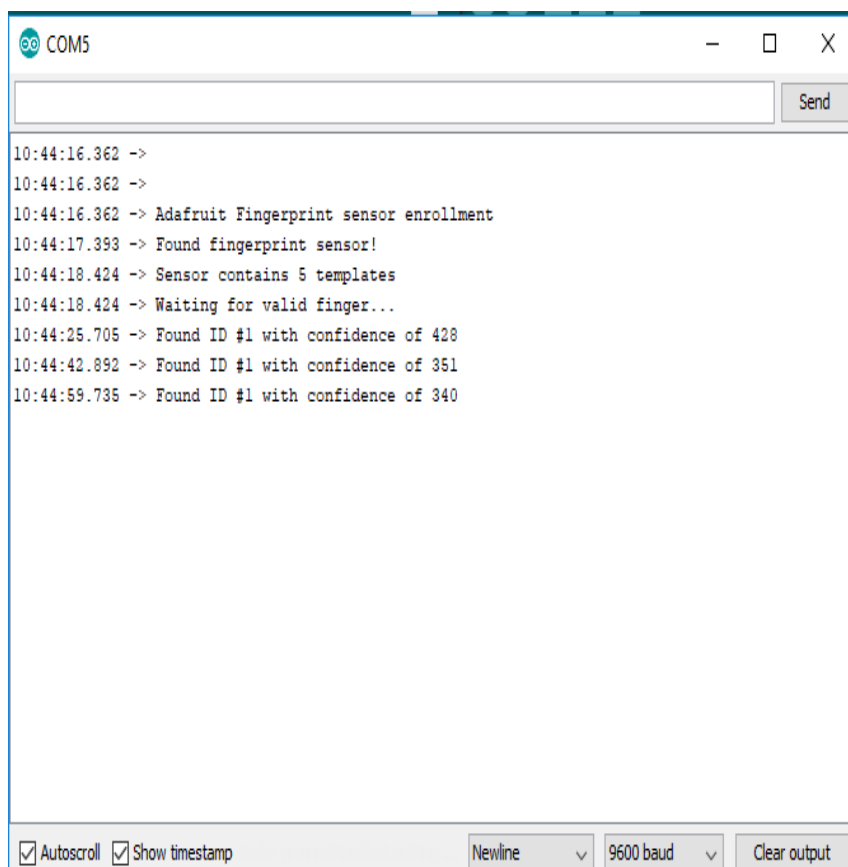


Gambar 5.4 : Langkah-Langkah Pengujian Sistem

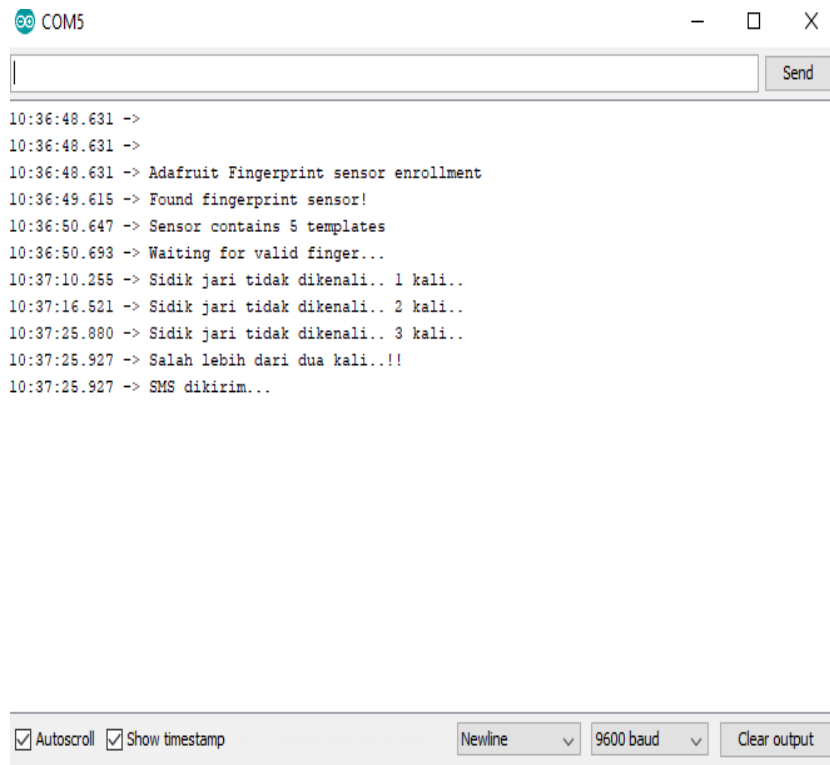
5.2.1 Pengujian Sensor Fingerprint

Pada pengujian sensor *fingerprint* dilakukan untuk melihat apakah sensor bekerja atau tidak, jika iya maka sensor akan mendeteksi data yang tersimpan pada sensor. Untuk mengetahui alat mengirim sms atau tidak mengirim sms, perlu dilakukan pengujian terhadap *fingerprint*, apakah objek yang di deteksi sesuai dengan *ID* yang sudah di tentukan sebelumnya.

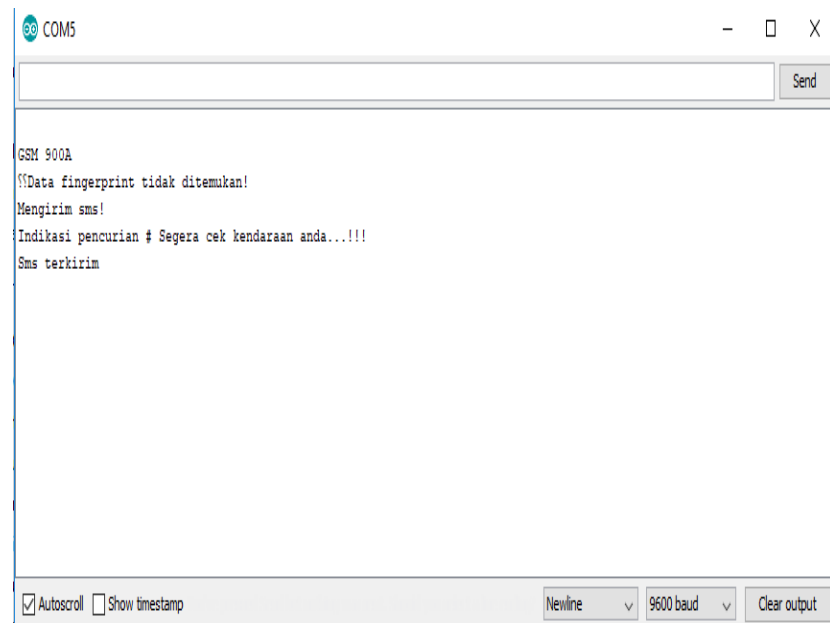
Untuk pengujian sensor akan dilakukan beberapa tahap pengujian, mulai dari pengenalan *ID* yang terdaftar pada *fingerprint* sampai pengenalan *ID* yang tidak terdaftar. Untuk melihat nilai tersebut kita bisa melihat pada terminal IDE Arduino uno. Untuk lebih jelasnya bisa dilihat pada gambar berikut ini :



Gambar 5 5 : Data di temukan



Gambar 5.6 : Data tidak di temukan



Gambar 5.7 : Pengiriman SMS saat Data tidak Ditemukan

Pada gambar 5.5 dan 5.6 sensor *fingerprint* akan membaca data yang sudah terdaftar dan tidak terdaftar di dalam sensor *fingerprint*, jika data tersebut tidak terdaftar pada sensor maka otomatis sistem akan mengirimkan notifikasi berupa pesan SMS kepada pengguna sistem ini.

Untuk melihat hasil pengujian terhadap sensor fingerprint ini maka dapat dilihat pada table 3.2 berikut ini :

Tabel 5.1 : Hasil Pengujian Sensor fingerprint

Sensivitas Sensor	Keterangan
Posisi awal	Sepeda motor tidak aktif
Fingeran 1	Sensor Mendeteksi Objek. Mengaktifkan sistem kelistrikan pada kendaraan
Fingeran 2	Sensor Mendeteksi Objek. Menstarter kendaraan
Fingeran 3	Menonaktifkan kelistrikan dan mesin pada kendaraan

Tabel 5.2 : Hasil Pengamatan Sensor fingerprint

Massalah dan Hasil Pengujian (Data tidak Benar)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan/Hasil
Data dari sensor finger	Sensor akan membaca setiap data yang mencoba mengakses sistem, alat akan mengirimkan SMS ke pengguna sistem ini.	Sensor membaca ID pada finger, pada posisi ID salah/ tidak benar maka sistem ini akan mengirimkan SMS Ke penggunka sistem ini.	✓ Diterima

Dari hasil pengujian sistem keseluruhan dapat disimpulkan bahwa Sistem Keamanan Sepeda Motor Berbasis *Microcontroller* Menggunakan Sensor *Fingerprint* dapat berjalan dengan baik, sistem secara otomatis langsung mengirimkan pesan berupa SMS ke pengguna di saat sepeda motor di akses secara paksa.

5.2.2 Pengujian Modul GSM SIM 900A

Pengujian pada Modul GSM dilakukan untuk mengirimkan SMS bahwa kendaraan telah di akses secara paksa melalui sensor *Fingerprint*, kepada pengguna sistem keamanan sepeda motor ini yang telah ditentukan sebelumnya. Untuk pengujian dilakukan sebanyak 3 kali dengan waktu yang berbeda-beda. Adapun tampilan sms yang dikirimkan ke *smartphone* pengguna dapat dilihat pada Gambar 5.8 berikut ini :



Gambar 5.8 : Tampilan Handphone saat menerima SMS

Untuk melihat hasil pengujian dari Modul GSM dapat dilihat pada tabel berikut ini :

Tabel 5.1: Tampilan Hasil Pengiriman Dan Penerimaan SMS

Waktu Pengiriman	Waktu Penerimaan	Keterangan
	<i>Smartphone</i> (Pengguna)	
11:49	11:50	Terkirim
11:51	11:52	Terkirim
11:52	11:53	Terkirim

5.2.3 Pengujian Sistem Secara Keseluruhan

Pengujian sistem keamanan sepeda motor berbasis *microcontroller* menggunakan sensor *fingerprint* dilakukan dengan cara melihat proses dan fungsi dari keseluruhan sistem ini, mulai dari pembacaan sensor terhadap data sampai pengiriman sms ke pengguna sistem ini.

Sistem keamanan ini diletakan pada sebuah sepeda motor yang kemudian akan memonitor kendaraan tersebut. Jika ada yang ingin mengakses kendaraan dengan data yang tidak terdaftar maka sistem ini secara otomatis akan mengirimkan pesan sms kepada pemilik kendaraan bahwa sepeda motor di akses secara paksa atau adanya indikasi pencurian.



Gambar 5.9 : Sistem dalam Keadaan Hidup

Tabel 5.2: Hasil Pengujian Keseluruhan Sistem

Kasus dan hasil Uji (Data tidak benar)			
Data Masukan	Yang Diharapkan	Pengamatan	Kesimpulan
Data dari sensor	Sistem berjalan sesuai dengan perintah yang diharapkan, sistem secara otomatis akan mengirimkan pesan SMS ke pengguna sistem ini karena sensor telah membaca data yang tidak terdaftar	Alat mengirimkan pesan sms ke pengguna karena adanya indikasi pencurian	✓ Diterima

Dari hasil pengujian sistem keseluruhan dapat disimpulkan bahwa Sistem Keamanan Sepeda Motor Berebasis *Microcontroler* Menggunakan Sensor *Fingerprint* dapat berjalan dengan baik, sistem secara otomatis langsung mengirimkan pesan berupa SMS kepengguna di saat sistem mendeteksi data ID yang salah.

BAB VI

PENUTUP

6.1 Kesimpulan

Berdasarkan hasil penelitian serta pengujian yang sudah dilaksanakan bahwa Sistem Keamanan Sepeda Motor Berbasis *Microcontroller* Menggunakan Sensor *Fingerprint* dapat disimpulkan sebagai berikut :

1. Sistem Keamanan Sepeda Motor Berbasis *Microcontroller* Menggunakan Sensor *Fingerprint* telah berhasil dirancang dengan menggunakan *Arduino Uno R3*, Modul *GSM SIM 900A*, dan perangkat pendukung lainnya, serta sebuah sepeda motor.
2. Berdasarkan dari hasil pengujian maka dapat disimpulkan sebagai berikut :
 - a. Jika sensor fingerprint mendeteksi data yang benar maka sistem akan mengeksekusi perintah ke sepeda motor.
 - b. Jika data sensor *fingerprint* mendeteksi data yang tidak benar maka Modul GSM akan mengirim pesan kepada pemilik sepeda motor.

6.2 Saran

Sistem Keamanan Sepeda Motor Berbasis *Microcontroller* Menggunakan Sensor *Fingerprint* ini masih jauh dari kesempurnaan. Untuk membangun sebuah sistem yang baik dan sempurna tentu perlu dilakukan pengembangan yang lebih lanjut, baik dari sisi manfaat maupun dari sisi kerja sistem. Berikut beberapa saran yang dapat disampaikan peneliti :

1. Untuk mendapatkan hasil yang lebih baik, sebaiknya perlu ditambahkan modul gps Ublox Neo yang berfungsi mengetahui titik lokasi
2. Untuk penambahan sistem keamanan perlu penambahan module OV7670 untuk pengenalan wajah

3. Untuk perakitan alat perlu ditambahkan *hard case* yang mampu melindungi rangkaian alat dari panas mesin serta basah terhadap air.

DAFTAR PUSTAKA

- [1]. Beni Setya Nugraha S.Pd.T, “TEKNOLOGI SEPEDA MOTOR” Teknik Otomotif Fakultas Teknik Universitas Negeri Yogyakarta. 2007.
- [2]. A.Muhammad Syafar, “Perancangan Sekuriti Sistem Kendaraan Motor Dengan Teknologi *Near Field Communication* (NFC)”, Fakultas Sains & Teknologi UIN Alauddin Makassar, 2017
- [3]. Suharjo Beman, “*Sistem Biometrika Konsep Dasar*,”Faculty of Engineering, Binus University”. Jakarta Barat, 2011.
- [4]. Triyastutik Tutut, “Teknologi *Mikrokontroler*”, Fakultas Teknik Elektro, Universitas Jember, 2012.
- [5]. Eko Nugroho, “Mengenal Sistem Biometrika fingerprint Identifikasi Masa depan Universitas Gadjah Mada, 2014.
- [6]. Haris Isyanto, Teknik Elektro Fakultas Teknik Universitas Muhammadiyah Jakarta, 2016
- [7]. Justin, L, “Taking Open_source Approach to Hardware”, 2009.
- [8]. Sigit Nugroho Wicaksono “Aplikasi Kran Otomatis Berbasis Arduino”. Sekolah Tinggi Manajemen Informatika Dan Komputer, Akakom Yogyakarta, 2017.
- [9]. ArduinoArts" <http://arduinoarts.com//the-arduino-uno-anatomy/> ", agustus 2011.
- [10]. Elga Aris Prastyo “https://www.arduinoindonesia.id//_arduino.html”, 2014.
- [11]. Feri Djuandi, "Pengenal Arduino" <http://tobuku.com/docs/Arduino-Pengenalan.pdf>", 2011.
- [12]. Sinarduino, "Mengenal Arduino Software (IDE)" <https://www.sinarduino.com/artikel/mengenal-arduino-software-ide/>. Yogyakarta, 2016.

- [13]. Febby Muhammad Ilham "Mengenal Arduino Uno R3". Stmik Raharja, Tangerang, 2017
- [14]. Artanto, "Bahasa Pemrograman Arduino", Politeknik Sriwijaya, Palembang, 2012.
- [15]. Ona Anisa "Rancang Bangun Pengukur Kadar Gas Metana Pada Lahan Gambut Menggunakan Sms Gateway Dan Sensor Mq4 Berbasis Mikrokontroler". Palembang : Politeknik Negeri Sriwijaya Palembang, 2017.
- [16]. Dwi Yulianto "Rancang Bangun Pengendali Lampu Dengan Menggunakan Sms Berbasis Arduino Uno R3". Yogyakarta : Sekolah Tinggi Manajemen Informatika Dan Komputer AKAKOM Yogyakarta, 27 Juli 2018
- [17]. Muhamad Saleh, "RANCANG BANGUN SISTEM KEAMANAN RUMAH MENGGUNAKAN RELAY", Teknik Elektro Universitas Suryadarma, Jakarta September, 2017.
- [18]. Joko Christian, "Prototipe Sistem Pendeteksi Kebocoran Gas LPG Menggunakan Sensor Gas MQ2, Board Arduino Duemilanove, Buzzer, dan Arduino GSM Shield pada PT. Alfa Retailindo (Carrefour Pasar Minggu)". Teknik Informatika, Fakultas Teknologi Informasi, Universitas Budi Luhur, Jakarta, 2013.
- [19]. Supriyanto Anang, "RANCANG BANGUN INVERTER DC KE AC SATU FASA MODE PUSH PULL BERBASIS ARDUINO", Fakultas Teknik, Jurusan teknik elektro Universitas Muhammadiyah Surakarta, 2016
- [20]. Sigit Purnomo, "Perancangan sistem keamanan rumah berbasis sms gateway menggunakan mikrokontroller", Tanjung Pinang, 2016

LAMPIRAN

```

1. #include <Adafruit_Fingerprint.h>
2. SoftwareSerial mySerial(2, 3);
3. Adafruit_Fingerprint finger = Adafruit_Fingerprint(&mySerial);
4.
5. int id;
6. char msg;
7. int relay_speedo = 4;
8. int relay_motor = 5;
9. boolean indikator_speedo = false;
10. boolean indikator_mesin = false;
11.
12.
13. void setup()
14. {
15.   Serial.begin(9600);
16.   while (!Serial); // For Yun/Leo/Micro/Zero/...
17.   delay(100);
18.   Serial.println("\n\nAdafruit Fingerprint sensor enrollment");
19.
20.   // set the data rate for the sensor serial port
21.   finger.begin(57600);
22.
23.   if (finger.verifyPassword()) {
24.     Serial.println("Found fingerprint sensor!");
25.   } else {
26.     Serial.println("Did not find fingerprint sensor :(");
27.     while (1) { delay(1); }
28.   }
29.   pinMode(relay_speedo, OUTPUT);
30.   digitalWrite(relay_speedo, HIGH);
31.   pinMode(relay_motor, OUTPUT);
32.   digitalWrite(relay_motor, HIGH);
33. }
34.
35. uint8_t readnumber(void) {
36.   uint8_t num = 0;
37.
38.   while (num == 0) {
39.     while (! Serial.available());
40.     num = Serial.parseInt();
41.   }
42.   return num;
43. }

```

```

44.
45. int enroll = 0;
46.
47. int finger_detect = 1;
48. int finger_detect_setup = 0;
49.
50. int delete_finger = 0;
51. int sidik;
52.
53. void loop()                                // run over and over again
54. {
55.   if (enroll) {
56.     start_enroll();
57.     while (! getFingerprintEnroll() );
58.   }
59.
60.   if (finger_detect) {
61.     if (! finger_detect_setup) {
62.       warming_detect_fingerprint();
63.       finger_detect_setup = 1;
64.     }
65.     sidik = getFingerprintIDez();
66.     if (sidik > 0){
67.       if (!(indikator_speedo and indikator_mesin)){
68.         indikator_speedo = true;
69.         digitalWrite(relay_speedo, LOW);
70.         delay(5000);
71.       }
72. if (indikator_speedo and indikator_mesin){
73.     indikator_speedo = false;
74.     indikator_mesin = false;
75.     digitalWrite(relay_speedo, HIGH);
76.     digitalWrite(relay_motor, HIGH);
77.     delay(3000);
78.   }
79.
80.   if (indikator_speedo){
81.     indikator_mesin = true;
82.     digitalWrite(relay_motor, LOW);
83.     delay(2000);
84.     digitalWrite(relay_motor, HIGH);
85.     delay(2000);
86.   }
87. }
88. }

```

```

89. if (delete_finger) {
90.     deleteNowFingerprint();
91. }
92.
93.
94. }
95.
96. void read_sms(){
97.     // will return sms content
98. }
99.
100. void start_enroll(){
101.     Serial.println("Ready to enroll a fingerprint!");
102.     Serial.println("Please type in the ID # (from 1 to 127) you
    want to save this finger as...");
103.     id = readnumber();
104.     if (id == 0) { // ID #0 not allowed, try again!
105.         return;
106.     }
107.     Serial.print("Enrolling ID #");
108.     Serial.println(id);
109. }
110.
111. uint8_t getFingerprintEnroll() {
112.
113.     int p = -1;
114.     Serial.print("Waiting for valid finger to enroll as #");
    Serial.println(id);
115.     while (p != FINGERPRINT_OK) {
116.         p = finger.getImage();
117.         switch (p) {
118.             case FINGERPRINT_OK:
119.                 Serial.println("Image taken");
120.                 break;
121.             case FINGERPRINT_NOFINGER:
122.                 Serial.println(".");
123.                 break;
124.             case FINGERPRINT_PACKETRECEIVEERR:
125.                 Serial.println("Communication error");
126.                 break;
127.             case FINGERPRINT_IMAGEFAIL:
128.                 Serial.println("Imaging error");
129.                 break;
130.             default:
131.                 Serial.println("Unknown error");

```

```

135.         if (delete_finger) {
136.             deleteNowFingerprint();
137.         }
138.
139.
140.     }
141.
142.     void read_sms(){
143.         // will return sms content
144.     }
145.
146.     void start_enroll(){
147.         Serial.println("Ready to enroll a fingerprint!");
148.         Serial.println("Please type in the ID # (from 1 to 127) you
want to save this finger as...");
149.         id = readnumber();
150.         if (id == 0) { // ID #0 not allowed, try again!
151.             return;
152.         }
153.         Serial.print("Enrolling ID #");
154.         Serial.println(id);
155.     }
156.
157.     uint8_t getFingerprintEnroll() {
158.
159.         int p = -1;
160.         Serial.print("Waiting for valid finger to enroll as #");
Serial.println(id);
161.         while (p != FINGERPRINT_OK) {
162.             p = finger.getImage();
163.             switch (p) {
164.                 case FINGERPRINT_OK:
165.                     Serial.println("Image taken");
166.                     break;
167.                 case FINGERPRINT_NOFINGER:
168.                     Serial.println(".");
169.                     break;
170.                 case FINGERPRINT_PACKETRECEIVEERR:
171.                     Serial.println("Communication error");
172.                     break;
173.                 case FINGERPRINT_IMAGEFAIL:
174.                     Serial.println("Imaging error");
175.                     break;
176.                 default:
177.                     Serial.println("Unknown error");
178.                     break;
179.             }
180.         }

```

```

181.      // OK success!
182.
183.      p = finger.image2Tz(1);
184.      switch (p) {
185.          case FINGERPRINT_OK:
186.              Serial.println("Image converted");
187.              break;
188.          case FINGERPRINT_IMAGEMESS:
189.              Serial.println("Image too messy");
190.              return p;
191.          case FINGERPRINT_PACKETRECEIVEERR:
192.              Serial.println("Communication error");
193.              return p;
194.          case FINGERPRINT_FEATUREFAIL:
195.              Serial.println("Could not find fingerprint features");
196.              return p;
197.          case FINGERPRINT_INVALIDIMAGE:
198.              Serial.println("Could not find fingerprint features");
199.              return p;
200.          default:
201.              Serial.println("Unknown error");
202.              return p;
203.      }
204.      Serial.println("Remove finger");
205.      delay(2000);
206.      p = 0;
207.      while (p != FINGERPRINT_NOFINGER) {
208.          p = finger.getImage();
209.      }
210.      Serial.print("ID "); Serial.println(id);
211.      p = -1;
212.      Serial.println("Place same finger again");
213.      while (p != FINGERPRINT_OK) {
214.          p = finger.getImage();
215.          switch (p) {
216.              case FINGERPRINT_OK:
217.                  Serial.println("Image taken");
218.                  break;
219.              case FINGERPRINT_NOFINGER:
220.                  Serial.print(".");
221.                  break;
222.              case FINGERPRINT_PACKETRECEIVEERR:
223.                  Serial.println("Communication error");
224.                  break;
225.              case FINGERPRINT_IMAGEFAIL:
226.                  Serial.println("Imaging error");
227.                  break;
228.              default:
229.                  Serial.println("Unknown error");
230.                  break;
231.          }
232.      }

```



```

233. // OK success!
234.
235. p = finger.image2Tz(2);
236. switch (p) {
237.     case FINGERPRINT_OK:
238.         Serial.println("Image converted");
239.         break;
240.     case FINGERPRINT_IMAGEMESS:
241.         Serial.println("Image too messy");
242.         return p;
243.     case FINGERPRINT_PACKETRECEIVEERR:
244.         Serial.println("Communication error");
245.         return p;
246.     case FINGERPRINT_FEATUREFAIL:
247.         Serial.println("Could not find fingerprint features");
248.         return p;
249.     case FINGERPRINT_INVALIDIMAGE:
250.         Serial.println("Could not find fingerprint features");
251.         return p;
252.     default:
253.         Serial.println("Unknown error");
254.         return p; }
255.
256. // OK converted!
257. Serial.print("Creating model for #"); Serial.println(id);
258.
259. p = finger.createModel();
260. if (p == FINGERPRINT_OK) {
261.     Serial.println("Prints matched!");
262. } else if (p == FINGERPRINT_PACKETRECEIVEERR) {
263.     Serial.println("Communication error");
264.     return p;
265. } else if (p == FINGERPRINT_ENROLLMISMATCH) {
266.     Serial.println("Fingerprints did not match");
267.     return p;
268. } else {
269.     Serial.println("Unknown error");
270.     return p; }
271.
272. Serial.print("ID "); Serial.println(id);
273. p = finger.storeModel(id);
274. if (p == FINGERPRINT_OK) {
275.     Serial.println("Stored!");
276. } else if (p == FINGERPRINT_PACKETRECEIVEERR) {
277.     Serial.println("Communication error");
278.     return p;
279. } else if (p == FINGERPRINT_BADLOCATION) {
280.     Serial.println("Could not store in that location");
281.     return p;
282. } else if (p == FINGERPRINT_FLASHERR) {
283.     Serial.println("Error writing to flash");
284.     return p;
285. } else {
286.     Serial.println("Unknown error");
287.     return p;
288. }
289. }

```

```

290.     void warming_detect_fingerprint(){
291.         finger.getTemplateCount();
292.         Serial.print("Sensor contains ");
293.         Serial.print(finger.templateCount); Serial.println(" templates");
294.         Serial.println("Waiting for valid finger...");
295.     }
296.     // Detect fingerprint
297.     // returns -1 if failed, otherwise returns ID #
298.     int getFingerprintIDez() {
299.         uint8_t p = finger.getImage();
300.         if (p != FINGERPRINT_OK) return -1;
301.
302.         p = finger.image2Tz();
303.         if (p != FINGERPRINT_OK) return -1;
304.
305.         p = finger.fingerFastSearch();
306.         if (p != FINGERPRINT_OK) return -1;
307.
308.         // found a match!
309.         // digitalWrite(relay_speedo, LOW);
310.         // digitalWrite(relay_motor, LOW);
311.         Serial.print("Found ID #"); Serial.print(finger.fingerID);
312.         Serial.print(" with confidence of ");
313.         Serial.println(finger.confidence);
314.         // delay(3000);
315.         // digitalWrite(relay_motor, HIGH);
316.         return finger.fingerID;
317.     }
318.
319.     void deleteNowFingerprint(){
320.         Serial.println("Please type in the ID # (from 1 to 127) you
321. want to delete...");
322.         uint8_t id = readnumber();
323.         if (id == 0) { // ID #0 not allowed, try again!
324.             return;
325.         }
326.         Serial.print("Deleting ID #");
327.         Serial.println(id);
328.
329.         deleteFingerprint(id);
330.     }

```

```

331.// fungsi menghapus fingerprint berdasarkan id
332.uint8_t deleteFingerprint(uint8_t id) {
333.  uint8_t p = -1;
334.
335.  p = finger.deleteModel(id);
336.
337.  if (p == FINGERPRINT_OK) {
338.    Serial.println("Deleted!");
339.  } else if (p == FINGERPRINT_PACKETRECEIVEERR) {
340.    Serial.println("Communication error");
341.    return p;
342.  } else if (p == FINGERPRINT_BADLOCATION) {
343.    Serial.println("Could not delete in that location");
344.    return p;
345.  } else if (p == FINGERPRINT_FLASHERR) {
346.    Serial.println("Error writing to flash");
347.    return p;
348.  } else {
349.    Serial.print("Unknown error: 0x"); Serial.println(p, HEX);
350.    return p; }}
351.
352.void SendMessage()
353.{
354.  mySerial.println("AT+CMGF=1"); //Sets the GSM Module in Text Mode
355.  delay(1000); // Delay of 1000 milli seconds or 1 second
356.  mySerial.println("AT+CMGS=\"+6285241068818\"\\r"); // Replace x with mobile
    number
357.  delay(1000);
358.  mySerial.println("sim900a sms");// The SMS text you want to send
359.  delay(100);
360.  mySerial.println((char)26);// ASCII code of CTRL+Z
361.  delay(1000);}
362.
363.void ReceiveMessage()
364.{
365.  mySerial.println("AT+CNMI=2,2,0,0,0"); // AT Command to receive a live SMS
366.  delay(1000);
367.  if (mySerial.available()>0)
368.  {
369.    msg=mySerial.read();
370.    Serial.print(msg);
371.    Serial.print("Akhir pesan");
372.  }}
373.
374.void MakeCall()
375.{
376.  mySerial.println("ATD+6285241068818;"); // ATDxxxxxxxxxx; -- watch out here
    for semicolon at the end!!
377.  Serial.println("Calling "); // print response over serial port
378.  delay(1000);
379.
380.}

```

SISTEM KEAMANAN SEPEDA MOTOR BERBASIS MICROCONTROLLER MENGGUNAKAN SENSOR FINGERPRINT

ORIGINALITY REPORT



PRIMARY SOURCES

repositori.uin-alauddin.ac.id

1 Internet Source 6%

2 docplayer.info

3 Internet Source 5%

Submitted to Academic Library
Consortium

Student Paper 4%

4 eprints.uny.ac.id

Internet Source 3%

5 www.scribd.com

Internet Source 2%

6 es.scribd.com

7 Internet Source 2%

eprints.akakom.ac.id

Internet Source 2%

8 digilib.unila.ac.id

Internet Source 2%

9 Submitted to Universitas Putera Batam

Student Paper 1%

10 Submitted to Universitas Islam Syekh-Yusuf
Tangerang 1%

Student Paper

11 Submitted to Universitas 17 Agustus 1945
Surabaya 1%

Student Paper

12 Submitted to Universitas Muria Kudus 1%

Student Paper

repository.untag-
sby.ac.id 13 1%

Internet Source

eprints.um
s.ac.id 14 1%

Internet Source

widuri.rahar ja.info 15 Internet Source	1%
eprints.ump o.ac.id 16 Internet Source	1%
www.edukasielektronik a.com 17 Internet Source	<1%
pt.slidesha re.net 18 Internet Source	<1%
Submitted to Udayana University 19 Student Paper	<1%
repository.its.ac.id 20 Internet Source <1%	
jurnal.fikom.umi.ac .id 21 Internet Source	<1%

22

Submitted to Universitas Brawijaya

Student Paper

<1%

Exclude quotes

On

Exclude matches

< 25 words

Exclude bibliography On



DAFTAR RIWAYAT HIDUP

Angki Ahmad, Lahir di Bitung, Kota. Sulawesi Utara pada tanggal 03 Januari 1993, anak Pertama dari pasangan Bapak Jony Ahmad dan Ibu Nelly Rauf

Riwayat Pendidikan :

1. Tahun 2005, menyelesaikan Pendidikan di Madrasah Ibtidaiyah Bitung
2. Tahun 2008, menyelesaikan Pendidikan di Sekolah Menengah Pertama, SMP Negeri 8 Gorontalo.
3. Tahun 2011, menyelesaikan Pendidikan Sekolah Menengah Kejuruan, SMK Negeri 3 Gorontalo.
4. Tahun 2015, Mendaftar dan diterima menjadi Mahasiswa di Fakultas Ilmu Komputer Universitas Ichsan Gorontalo