

**SISTEM KONTROL DAN MONITORING PUTARAN MOTOR MOBIL RC
(REMOTE CONTROL) BERBASIS SMARTPHONE ANDROID**



ARWAN FAISYAL

5215110327

Skripsi ini Ditulis untuk Memenuhi Sebagai Persyaratan dalam Memperoleh Gelar
Sarjana Pendidikan

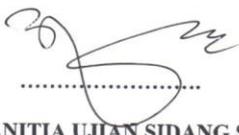
PROGRAM STUDI PENDIDIKAN TEKNIK ELEKTRONIKA

FAKULTAS TEKNIK

UNIVERSITAS NEGERI JAKARTA

2017

LEMBAR PENGESAHAN

Nama Dosen	Tanda Tangan	Tanggal
Drs. Pitoyo Yuliatmojo, MT (Dosen Pembimbing I)		8-2-2017
Efri Sandi, S.Pd, MT (Dosen Pembimbing II)		2 Feb 2017

PENGESAHAN PANITIA UJIAN SIDANG SKRIPSI

Nama Dosen	Tanda Tangan	Tanggal
<u>Dr. Muhammad Yusro, MT.</u> (Ketua Pengujian)		30 Jan 2017
<u>Aodah Diamah, Ph.D</u> (Sekretaris)		31 Januari 2017
<u>Drs. Jusuf Bintoro, MT.</u> (Dosen Ahli)		31 Januari 2017

Tanggal Lulus:

ABSTRAK

Arwan Faisyal, *Sistem Kontrol dan Monitoring Putaran Motor Mobil RC (Remote Control) Berbasis Smartphone Android*. Skripsi. Jakarta, Program Studi Pendidikan Teknik Elektronika, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Negeri Jakarta, 2017. Dosen Pembimbing, Drs. Pitoyo Yuliatmojo, MT dan Efri Sandi, S.pd, MT.

Tujuan pembuatan penelitian ini adalah merancang dan membuat alat yang dapat menentukan *level* putaran motor DC (*Direct Current*) serta memonitoring putaran motor DC (*Direct Current*). Alat ini akan mempermudah dalam pembacaan putaran motor pada mobil. Karena setiap hasil pembacaan akan dikirim ke *Android*.

Penelitian ini dilakukan menggunakan metode Penelitian dan Pengembangan (*Research and Development*) yang meliputi perencanaan, analisis kebutuhan perancangan, pengujian, dan implementasi sistem perangkat keras (*hardware*) dan perangkat lunak (*software*). Implementasi perangkat keras (*hardware*) yaitu pembuatan *prototype* mobil, sistem pengendali menggunakan mikrokontroler Arduino Mega2560 serta Arduino Nano328 dan implementasi perangkat lunak (*Software*) yaitu berupa *software* Arduino 1.6.8, *software* App Inventor yang digunakan untuk membuat Aplikasi pada *Smartphone Android*.

Kata Kunci: Mobil RC, Monitoring, Putaran, Mikrokontroler Arduino, *Android*.

ABSTRACT

Arwan Faisyal, *system control and monitoring of motor rotation RC car (Remote Control) Android Based Smartphone. Essay. Jakarta, Educational Program Electronic Engineering, Department of Electrical Engineering, Faculty of Engineering, University of Negeri Jakarta, 2015. Supervisor, Drs. Pitoyo Yuliatmojo, MT And Efri Sandi, S.Pd., MT.*

The purpose of this study was to design and create a tool that can determine the level of motor rotation DC (Direct Current), all the while monitoring motor rotation DC (Direct Current). This tool will facilitate the reading of the rotation of the motor car. Because each of the readings will be sent to Android.

This research was conducted using the method of Research and Development (Research and Development), the which includes planning, requirements analysis, design, testing, and implementation of the system hardware (hardware) and software (software). Implementation of the hardware (hardware) is the prototype cars, the system controller using a microcontroller Arduino Mega2560 and Arduino Nano328 and implementation of software (software) in the form of Arduino 1.6.8 and App Inventor is used to create applications on the Android Smartphone.

Keywords: *RC Car, Monitroing, Round, Microcontroller Arduino, Android.*

KATA PENGANTAR

Assalamu'alaykum Warahmatullahi Wabarakaatuh

Segala puji bagi ALLAH Subhanahu Wata'ala Tuhan semesta alam. Penulis ucapkan syukur Alhamdulillah yang tiada terkira. Atas rahmat dan karunia-NYA, sehingga skripsi penelitian yang berjudul " SISTEM KONTROL DAN MONITORING PUTARAN MOTOR MOBIL RC (*REMOTE CONTROL*) BERBASIS *SMARTPHONE ANDROID*" ini dapat diselesaikan.

Skripsi ini ditulis dalam rangka menyelesaikan tugas akhir guna memperoleh gelar Sarjana Pendidikan pada Program Studi S1 Pendidikan Teknik Elektronika, Fakultas Teknik, Universitas Negeri Jakarta.

Penulis sampaikan terimakasih kepada semua pihak yang telah membantu hingga terselesaikannya skripsi ini. Penulis menyadari bahwa skripsi ini tidak serta merta hadir tanpa bantuan dan dukungan dari semua pihak. Oleh sebab itu, pada kesempatan ini, kepada pihak yang telah membantu penulis atas kelancaran dan dorongan semangat yang telah diberikan selama ini sehingga skripsi ini dapat terselesaikan, penulis haturkan terima kasih kepada:

1. Kedua orang tua dan keluarga saya yang telah memberikan kasih sayang yang tidak ternilai harganya dan juga atas doa yang tidak pernah henti untuk di panjatkan.
2. Bapak Drs. Pitoyo Yuliatmojo, MT. selaku ketua Program Studi Pendidikan Teknik Elektronika. dan selaku dosen pembimbing I.
3. Bapak Efri Sandi, S.Pd., MT. selaku dosen pembimbing II
4. Rekan - rekan penulis yang selalu memberikan motivasi dan bantuan kepada penulis.
5. Dan semua orang yang telah membantu dan memberikan inspirasi kepada penulis dalam menyelesaikan penulisan skripsi ini yang tidak dapat penulis sebutkan satu per satu.

Mudah-mudahan segala sesuatu yang telah diberikan menjadi manfaat dan bernilai ibadah di hadapan Allah Subhanahu Wata'ala.

Penulis memahami sepenuhnya bahwa skripsi ini tidak luput dari kesalahan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan dimasa mendatang. Semoga skripsi ini dapat memberikan inspirasi bagi para pembaca untuk melakukan hal yang lebih baik lagi dan semoga skripsi ini bermanfaat khususnya bagi penulis, dan pembaca pada umumnya. Akhirnya kepada Allah SWT jugalah semuanya kita kembali.

Penulis

Arwan Faisyal
5215110327

DAFTAR ISI

ABSTRAK.....	i
ABSTRACT.....	ii
KATA PENGANTAR.....	iii
DAFTAR ISI.....	iv
DAFTAR GAMBAR.....	vii
DAFTAR TABEL.....	ix
DAFTAR LAMPIRAN.....	x
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang Masalah.....	1
1.2 Identifikasi Masalah.....	2
1.3 Perumusan Masalah.....	3
1.4 Pembatasan Masalah.....	3
1.5 Tujuan Penelitian.....	4
1.6 Kegunaan Penelitian.....	4
BAB II TINJAUAN PUSTAKA.....	5
2.1 Kerangka Teoritik.....	5
2.1.1 Definisi Mobil RC (<i>Remote Control</i>).....	5
2.1.2 Perangkat Kendali Mobil RC (<i>Remote Control</i>).....	8
2.1.2.1 Smartphone Android.....	8
2.1.2.2 MIT App Inventor.....	10
2.1.2.3 Mikrokontroler Arduino Mega2560.....	12
2.1.2.4 Mikrokontroler Arduino Nano328.....	15
2.1.2.5 Software Arduino IDE.....	18
2.1.3 Komponen Sensor.....	20
2.1.3.1 Sensor Optocoupler.....	20
2.1.4 Komponen Penggerak.....	26
2.1.4.1 Motor.....	26
2.1.4.2 Motor Servo.....	27
2.1.4.3 Motor DC.....	30
2.1.4.3.1 Putaran Motor DC.....	32

2.1.4.3.2	Driver Motor DC H-Bridge.....	33
2.1.5	Perangkat Komunikasi Data	37
2.1.5.1	Definisi Bluetooth	37
2.1.1.1	Modul Bluetooth HC-05.....	39
2.1.6	Kerangka Berpikir.....	42
2.1.6.1	Blok Diagram	42
2.1.6.2	Flowchart.....	44
2.1.7	Kontrol dan Monitoring	47
2.1.7.1	Kontrol	47
2.1.7.2	Monitoring.....	48
BAB III METODOLOGI PENELITIAN		50
3.1	Tempat dan Waktu Penelitian	50
3.2	Alat dan Bahan Penelitian.....	50
3.3	Diagram Alir Penelitian	52
3.3.1	Tahap Penelitian dan Pengumpulan Informasi	53
3.3.2	Tahap Perancangan	54
3.3.3	Tahap Pengembangan Produk	54
3.4	Tahap Uji Coba	54
3.4.1	Tahap Perbaikan Produk.....	55
3.5	Perancangan Alat	55
3.5.1	Perancangan Perangkat Keras.....	55
3.5.1.1	Desain Perakitan Mekanik	55
3.5.1.2	Desain Elektronik.....	57
3.5.2	Perancangan Perangkat Lunak.....	58
3.5.2.1	Perancangan Program Arduino	58
3.5.2.2	Perancangan Aplikasi Remote Control Pada Smartphone Android	60
3.6	Teknik dan Prosedur Pengumpulan Data.....	61
3.6	Teknik Analisis Data.....	62
3.6.1.	Pengujian <i>Driver Motor</i>	63
3.6.2.	Pengujian Motor Servo	63
3.6.3.	Pengujian Pembacaan Sensor Optocoupler	64
3.6.4.	Pengujian Konektivitas <i>Smartphone</i> Dengan Mobil RC	65

3.6.5.	Prngujian pembacaan data komunikasi serial aplikasi <i>remote control</i> mobil RC.....	65
3.6.6.	Pengujian kendali <i>remote control</i> terhadap mobil RC.....	66
3.6.7.	Pengujian jarak konektivitas <i>Bluetooth HC-05</i> dengan <i>Bluetooth smartphone</i>	67
BAB IV HASIL PENELITIAN		68
4.1.	Deskripsi Hasil Penelitian.....	68
4.1.1.	Hasil Pembuatan dan Pengujian Perangkat Keras (<i>Hardware</i>).....	68
4.1.1.1.	Hasil Pengujian Driver Motor	70
4.1.1.2.	Hasil Pengujian Motor Servo	71
4.1.1.3.	Hasil Pengujian Sensor Optocoupler.....	73
4.1.2.	Hasil pembuatan dan pengujian perangkat lunak (<i>software</i>).....	75
4.1.2.1.	Hasil Pengujian Konektivitas Smartphone dengan Mobil RC	77
4.1.2.2.	Hasil Pengujian Pembacaan Data Komunikasi Serial Aplikasi Remote Control.....	79
4.1.3.	Hasil Pengujian <i>Remote Control</i> dalam Mengendalikan Gerak Mobil RC	80
4.1.3.1.	Hasil Pengujian Kendali Remote Control Terhadap Mobil RC	81
4.1.3.2.	Pengujian Jarak Konektivitas Bluetooth HC-05 dengan Bluetooth Smartphone	83
4.2.	Pembahasan.....	85
4.3.	Kekurangan Alat	86
BAB V KESIMPULAN DAN SARAN.....		88
5.1	Kesimpulan	88
5.2	Saran	89
DAFTAR PUSTAKA.....		90
LAMPIRAN.....		91

DAFTAR GAMBAR

Gambar 2.1. Mobil RC (<i>Remote Control</i>)	5
Gambar 2.2. Logo MIT App Inventor	10
Gambar 2.3. Tampilan App Inventor <i>Designer</i>	11
Gambar 2.4. Tampilan App Inventor <i>Block Editor</i>	12
Gambar 2.5. Arduino Mega2560	13
Gambar 2.6. Blok Diagram ATmega 2560 (Corporation, 2012)	14
Gambar 2.7 Konfigurasi pin Arduino Mega2560.....	15
Gambar 2.8. Bentuk Fisik Arduino Nano328.....	16
Gambar 2.9. Form <i>Splash</i> Arduino Versi 1.6.8	19
Gambar 2.10. Tampilan Arduino IDE versi 1.6.8	19
Gambar 2.11. Sensor Optocoupler	23
Gambar 2.12. Bentuk Fisik Sensor Optocoupler	24
Gambar 2.13. Bentuk Fisik Beam Photoelectric Sensor	24
Gambar 2.14. Bentuk Fisik Motor Servo	27
Gambar 2.15. Sistem Mekanis Servo	28
Gambar 2.16. Lebar Pulsa dan Posisi Servo.....	29
Gambar 2.17. Motor DC.....	31
Gambar 2.18. Putaran Motor DC	33
Gambar 2.19. Rangkaian <i>Driver</i> Motor DC <i>H-Bridge</i> Transistor.....	34
Gambar 2.20. Bentuk Fisik Modul Bluetooth HC-05	39
Gambar 2.21. Diagram Blok Sistem Kontrol Dan Monitoring Putaran Motor Mobil RC (<i>Remote Control</i>) Berbasis Smartphone Android	43
Gambar 3.1. Diagram Alir Penelitian.....	53
Gambar 3.2. Desain Perakitan Mekanik Mobil RC.....	56
Gambar 3.3. <i>Steering</i> Mobil RC.....	56
Gambar 3.4 Bagian Dasar Mobil RC	57
Gambar 3.5 Skematik Rangkaian <i>Control</i> Dan Monitoring Putaran Motor Mobil RC.....	58

Gambar 4.1. (a) Bentuk Mobil RC (b) Mobil RC Tampak Samping Kanan (c) Bentuk Mobil RC Tampak Samping Kiri.....	69
Gambar 4.2. Cara Pengujian <i>Driver</i> Motor DC	70
Gambar 4.3. Tampilan Awal Aplikasi <i>Remote Control</i>	75
Gambar 4.4. Tampilan <i>Screen</i> Kontrol Dan Monitoring.....	75
Gambar 4.5. Tampilan <i>Screen</i> Informasi Aplikasi	76
Gambar 4.6. Tampilan <i>Screen</i> Profil Pembuat Aplikasi	76
Gambar 4.7. Tampilan <i>Screen</i> Jika Ingin Keluar Aplikasi.....	76
Gambar 4.8. Tampilan pencarian konektivitas <i>Bluetooth</i>	78
Gambar 4.9. Pengujian Pembacaan Data Bit Serial <i>Remote Control</i>.....	79
Gambar 4.10. Kondisi Awal Mobil Sebelum Dikendalikan.....	81
Gambar 4.11. Pengukuran Jarak Tempuh Mobil RC Menggunakan Meteran	84

DAFTAR TABEL

Tabel 2.1. Spesifikasi Arduino Mega 2560 (Arduino.cc, 2016)	14
Tabel 2.2. Simbol-simbol flowchart.....	45
Tabel 3.1 Penggunaan Pin Input Pada Arduino Mega2560	59
Tabel 3.2 Penggunaan Pin Output Pada Arduino Mega 2560.....	59
Tabel 3.3. Penggunaan Pin Input Pada Arduino Nano328.....	59
Tabel 3.4. Penggunaan Pin Output Pada Arduino Nano328	59
Tabel 3.5. Pengujian Driver Motor DC	63
Tabel 3.6. Pengujian Motor Servo.....	63
Tabel 3.7. Pengujian Pembacaan Sensor Optocoupler	64
Tabel 3.8. Pengukuran Tegangan Pada Sensor Optocoupler	64
Tabel 3.9. Kriteria Pengujian Konektivitas Smartphone Dengan Mobil RC	65
Tabel 3.10. Pengujian Pembacaan Data Komunikasi Serial Remote Control.....	65
Tabel 3.11. Pengujian Kendali Remote Control Terhadap Mobil RC	66
Tabel 3.12. Pengujian Konektivitas Bluetooth HC-05 dengan Bluetooth Smartphone.....	67
Tabel 4.1. Hasil Pengujian Driver Motor DC.....	71
Tabel 4.2. Hasil Pengujian Motor Servo	71
Tabel 4.3. Hasil Pengujian Sensor Optocoupler.....	73
Tabel 4.4. Pengukuran Tegangan Pada Sensor Optocoupler	74
Tabel 4.5. Pengujian Konektivitas Smartphone Dengan Mobil RC.....	78
Tabel 4.6. Hasil Pengujian Pembacaan Bit Serial Data Remote Control	80
Tabel 4.7. Hasil Pengujian Kendali Remote Control Terhadap Mobil RC	82
Tabel 4.8. Hasil Pengujian Konektivitas Bluetooth HC-05 Dengan Bluetooth Smartphone.....	84

DAFTAR LAMPIRAN

1.	Lampiran 1 List Program Arduino Mega2560.....	90
2.	Lampiran 2 List Program Arduino Nano328.....	92
3.	Lampiran 3 Program Android App Inventor.....	93
	A. Program Screen 4 (About).....	93
	B. Program Screen 3 (Info).....	93
	C. Program Screen 2 (Navigasi).....	94
	D. Program Screen 1 (Home).....	94

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Pada saat ini perkembangan *smartphone* sangat pesat. Namun seiring berjalannya waktu pada *smartphone* tersebut tertanam sistem operasi canggih diantaranya *Android, Ios, Symbian, Java, BlackBerry, Windows Mobile* Sistem operasi yang paling mendominasi pada pasar *smartphone* Indonesia adalah *Android*, dimana sistem operasi ini bersifat *open source* atau aplikasi dapat dikembangkan oleh siapapun, tanpa harus meminta izin dari pihak pembuat. *Smartphone Android* dapat memungkinkan penggunaanya untuk men-*download* berbagai aplikasi gratis.

Android merupakan salah satu dari berbagai macam sistem operasi pada telepon seluler. Telepon seluler berbasis *Android* dijuluki sebagai telepon pintar atau lebih sering disebut sebagai *smartphone*. Dikatakan sebagai telepon pintar karena kepintaran dan kelengkapan fitur-fitur yang dimilikinya membuat telepon seluler ini memiliki fungsi ganda atau multifungsi. Tidak hanya sekedar berfungsi mengirim pesan menerima telepon, tapi dapat mengambil gambar, merekam, menemukan lokasi, dapat terkoneksi dengan internet 24 jam, bahkan dapat memiliki fungsi sebagai *remote control*.

Karena kepintaran yang dimilikinya inilah yang membuat *smartphone Android* layak untuk dipelajari, diteliti dan dikembangkan. Fungsi yang akan diuji dan dipelajari dalam tugas akhir ini adalah fungsinya sebagai *remote control* yang dapat mengontrol dan monitoring suatu perangkat atau piranti.

Pada umumnya sistem sebuah robot diatur oleh mikrokontroler. Dan untuk melakukan suatu sistem tertentu atau dalam kondisi tertentu sebuah robot memerlukan sebuah prosesor untuk mengatur kinerja sebuah mikrokontroler. Pada tugas akhir ini *smartphone android* akan digunakan sebagai otak pemrosesannya dan robot mobil dipilih sebagai disain robotnya.

Dalam tugas akhir ini pengujian dan pembelajaran mengenai fungsi *smartphone* Android sebagai *remote control* bukan hanya membuat aplikasi yang dapat mengontrol sebuah robot mobil saja, tapi melakukan studi komparasi atas fitur-fitur yang dimilikinya dalam mengontrol robot mobil, untuk mengatasi kesulitan dalam memilih media komunikasi dan inputan perintah yang paling cocok digunakan dalam mengontrol sebuah robot mobil. Fitur-fitur yang akan dikomparasi yaitu, media komunikasi *Bluetooth* yang umum ada pada *smartphone*. Satu lagi yang akan dikomparasi yaitu inputan perintah dengan menggunakan *screen button*. Oleh karena itu, tugas akhir ini diberi judul “Sistem Kontrol dan Monitoring Putaran Motor Mobil RC (*Remote Control*) Berbasis *Smartphone Android*”.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang telah diuraikan, maka dapat diidentifikasi beberapa masalah yaitu sebagai berikut.

1. Perlu diteliti fungsi *smartphone android* sebagai *remote control*, karena pada umumnya aplikasi *android* digunakan untuk mengontrol perangkat yang ada pada *smartphone* bukan mengontrol perangkat

yang berada diluar *handphone android* seperti mengontrol sekaligus memonitoring putaran motor mobil RC (*Remote Control*).

2. Apakah pengendalian yang digunakan dalam mengontrol sekaligus monitoring putaran motor pada mobil RC (*Remote Control*) dengan *screen button* dan optocoupler sebagai inputan perintah?

1.3 Perumusan Masalah

Berdasarkan identifikasi maslaah diatas maka rumusan masalah dalam penelitian ini adalah:

1. Bagaimana membuat aplikasi yang dapat mengontrol sekaligus memonitoring putaran motor pada sebuah robot mobil sederhana?
2. Apakah pengendalian yang digunakan dalam mengontrol sekaligus monitoring putaran motor pada mobil RC (*Remote Control*) dengan *screen button* dan optocoupler sebagai inputan perintah?

1.4 Pembatasan Maslah

Berdasarkan rumusan masalah diatas maka batasan masalah dalam penelitian ini adalah sebagai berikut:

1. Pembuatan aplikasi untuk *smartphone android* menggunakan *software App Inventor*.
2. Komunikasi yang digunakan untuk dapat mengontrol dan memonitoring putaran mobil RC (*Remote Control*) yakni menggunakan *module* Bluetooth HC-05 dan jarak yang dapat di tempuh maksimal kurang lebih 10 meter.

1.5 Tujuan Penelitian

Berdasarkan batasan masalah di atas maka tujuan dalam penelitian ini adalah sebagai berikut:

1. Membuat sebuah mobil RC (*Remote Control*) yang terintegrasi dengan *smartphone android*.
2. Menjadikan *handphone* sebagai kontroler dan monitoring sebuah perangkat yang berada diluar *device*.
3. Membuat putaran motor pada mobil RC (*Remote Control*) yang dapat diatur dan sekaligus dimonitori oleh pengontrol berupa *smartphone Android*.

1.6 Kegunaan Penelitian

Adapun kegunaan dari penelitian Perancangan Sistem Kontrol dan Monitoring Putaran Motor Mobil RC (*Remote Control*) berbasis *Smartphone android* adalah:

1. Dengan sistem pengontrolan secara *push button* putaran level motor pada mobil akan lebih mudah dibandingkan dengan putaran motor yang diatur dengan cara menaikkan atau menurunkan *slider* secara manual dan perlahan.
2. Memudahkan pengguna untuk dapat mengontrol putaran motor pada mobil RC (*Remote Control*).
3. Dengan menggunakan sistem adanya *feedback* nilai putaran, maka memudahkan pengguna untuk mengontrol level putaran sesuai yang diinginkan.

BAB II

TINJAUAN PUSTAKA

2.1 Kerangka Teoritik

2.1.1 Definisi Mobil RC (*Remote Control*)

Di dalam Kamus Besar Bahasa Indonesia (KBBI) Mobil merupakan jenis kendaraan darat yang bergerak menggunakan tenaga mesin.

Mobil *remote control* adalah sejenis mobil mainan yang di kendalikan dengan menggunakan modul transmitter. Ukurannya pun tak sebesar mobil aslinya yang manusia pakai untuk beraktifitas, mobil *remote* hanya berupa mainan yang biasa di mainkan oleh anak - anak.



Gambar 2.1. Mobil RC (*Remote Control*)

(Sumber: <http://www.mobilremotecontrol.net/mobil-remote-control-audi-r8-gtr-lemans-24-ghz.html>)

Kemajuan jaman juga mengiringi perkembangan dunia permainan anak-anak. Jika dahulu kita mengenal permainan sederhana seperti mobil-mobilan yang ditarik dengan tali atau dengan menggunakan pegas, lain lagi dengan mainan saat kini. Kemajuan jaman telah menghadirkan pengganti kedudukan mainan mobil

remote control versi lama tersebut. Kehadiran permainan terbaru ini, tentu saja diterima dengan senang hati oleh anak-anak serta menjadi hal paling dicari dan paling diinginkan oleh mereka.

Kehadiran mobil *remote control*, juga dikenal dengan nama lain, mobil RC, selain dikenal hingga disukai oleh anak-anak juga digandrungi oleh orang dewasa. Bahkan dengan munculnya berbagai jenis, merk atau modelnya di pasaran, menjadikannya sebagai salah satu ladang bisnis bagi mereka yang jeli memanfaatkan kehadiran dan ketenaran dari mainan tersebut.

Mobil RC yang beredar di pasaran tergolong kepada jenis yang dapat difungsikan sebagai mainan, hobi ataupun sekaligus kedua-duanya. Fungsi mobil RC sebagai mainan, rata-rata memiliki harga cukup terjangkau dan memiliki kualitas standar. *Remote control* yang digunakan juga difungsikan dengan sumber tenaga berupa baterai sekali pakai, begitu pula sumber tenaga yang digunakan pada kendaraan tersebut. Ukuran baterai yang digunakan pun tergantung dari ukuran serta jenis mobil RC tersebut, bisa baterai berukuran kecil hingga besar. Pembekalan pada mainan ini kembali lagi kepada fungsi awalnya sebagai alat permainan saja, dengan waktu penggunaan terbatas dan tidak terlalu lama.

Sedangkan mobil RC dengan fungsi sebagai hobi, tentu saja lagi memiliki harga, kualitas yang tinggi pula, serta dibekali fitur-fitur ekstra dimana tidak dimiliki oleh mobil RC jenis biasa. Penggunaan jenis sumber tenaga biasanya berasal dari berbagai sumber energi, seperti baterai jenis isi ulang, bahan bakar minyak hingga gas. Di luar negeri, kendaraan RC dengan bahan bakar

bersumber dari minyak dan gas bukanlah hal aneh lagi, karena memang sumber daya akan bahan bakar tersebut mudah di dapatkan.

Dikarenakan penggunaan jenis sumber energi atau bahan bakar, mobil RC jenis ini biasanya memiliki ukuran cukup besar karena memiliki tanki penyimpan sumber energi atau bahan bakarnya. Dengan penggunaan sumber energi, tentu saja jangka waktu penggunaannya cukup lama sehingga cocok digunakan untuk mereka dengan hobi besar terhadap mainan ini, karena mampu menempuh waktu dan jarak perjalanan panjang. Kekurangan dari jenis kendaraan RC untuk hobi ini adalah terdapat beberapa jenisnya yang menghasilkan gas buangan serta menghasilkan suara mesin cukup berisik. Sedangkan untuk penggunaan sumber energi dengan baterai isi ulang, rata-rata menghasilkan suara mesin tidak terlalu berisik dan tidak menghasilkan gas buangan apapun, sehingga ramah lingkungan.

Model dan ukuran kendaraan remote control beredar di pasaran ada yang dapat dipergunakan pada jalanan biasa dan adapula untuk fungsi *off-road*. Perbedaannya terdapat pada penggunaan kualitas ban karena disesuaikan dengan medan penggunaannya. Untuk mobil pada medan jalanan biasa, rata-rata menggunakan jenis mobil sedan ataupun SUV mini. Sedangkan untuk medan *off-road*, mobil *remote control* yang digunakan rata-rata adalah jenis SUV besar, truk, jeep hingga hummer, dimana pada dunia nyata jenis kendaraan ini memang kebanyakan digunakan untuk fasilitas *off-road*.

Kehadiran kendaraan RC ini juga turut menghadirkan kompetisi balapan untuk permainan ini dan menjadikannya sebagai ajang adu ketangkasan dan

keahlian, dimana dapat disalurkan lewat alat pengendali kendaraan tersebut, yaitu *remote control*. Kompetisi balapan ini memang masih jarang ditemukan di Indonesia, tapi jika diluar negeri hal ini sudah menjadi hal biasa dan sering dilakukan pada *event-event* yang diadakan untuk merayakan hari-hari tertentu.

2.1.2 Perangkat Kendali Mobil RC (*Remote Control*)

2.1.2.1 *Smartphone Android*

Smartphone secara harfiah artinya telepon pintar, yakni telepon seluler yang memiliki kemampuan seperti PC walaupun terbatas. Selain itu, *smartphone* juga mendukung email dan organizer (Zaki, 1999). Fitur lainnya adalah kemampuannya untuk ditambah aplikasi aplikasi baru. Aplikasi yang dapat diinstal kedalam *smartphone* tidak hanya yang dibuat oleh produsen pembuat piranti tersebut, namu juga bisa dibuat oleh pihak ketiga.

Smartphone atau bisa disebut dengan telepon pintar/cerdas sudah menjadi sebuah kebutuhan bagi sekian orang di dunia ini sebagai penunjang aktivitas kerja maupun sekedar *lifestyle* atau gaya hidup. Telepon cerdas (*smartphone*) adalah telepon genggam yang mempunyai kemampuan tingkat tinggi, kadang-kadang dengan fungsi yang menyerupai komputer. Belum ada standar pabrik yang menentukan arti telepon cerdas. Bagi beberapa orang, telepon pintar merupakan telepon yang bekerja menggunakan seluruh perangkat lunak sistem operasi yang menyediakan hubungan standar dan mendasar bagi pengembang aplikasi. Bagi yang lainnya, telepon cerdas hanyalah merupakan sebuah telepon yang menyajikan fitur canggih seperti surel (surat elektronik), internet dan kemampuan membaca buku elektronik (*e-book*) atau terdapat papan

ketik (baik sebagaimana jadi maupun dihubung keluar) dan penyambung VGA. Dengan kata lain, telepon cerdas merupakan komputer kecil yang mempunyai kemampuan sebuah telepon.

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam piranti bergerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat piranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan piranti keras, piranti lunak, dan piranti telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia. (Safaat, 2011)

Beberapa fitur Android antara lain:

1. *Application Framework*, yakni digunakan untuk membangun aplikasi Android.
2. *Integrated Browser*, Android menyertakan *browser* berbasis WebKit sebagai aplikasi standar.
3. *Optimized Graphic*, Android mempunyai pustaka grafik 2D dan menyertakan pustaka grafik 3D Open GLES.
4. *Sqlite*, adalah aplikasi basis data Sqlite yang disertakan dalam android.
5. *Media Support*, dukungan untuk memutar format multimedia yang banyak.
6. *GSM telephony support*, adalah kemampuan android untuk mengakses langsung *hardware* untuk komunikasi GSM

7. *Bluetooth, EDGE, 3G, 4G dan WiFi*, dukungan untuk banyak jenis *wireless*.
8. *Camera, GPS, compass, NFC dan accelerometer*, dukungan untuk *hardware* tersebut, tersedia API untuk mengakses *hardware* tersebut.
9. *Rich development environment*, tersedia *software development* yang lengkap.

2.1.2.2 MIT App Inventor



Gambar 2.2. Logo MIT App Inventor

(Sumber: <http://georgepavlides.info/wp-content/uploads/2015/06/app-inventor.jpg>)

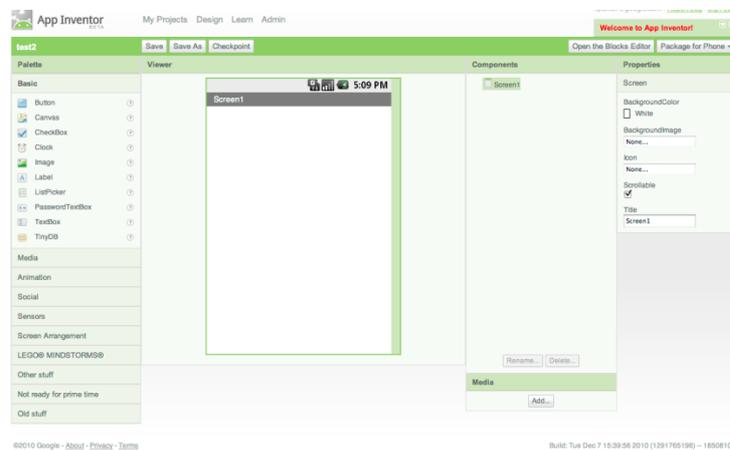
MIT App Inventor adalah sebuah *tool* untuk membuat aplikasi android, aplikasi ini menggunakan *tool* berbasis visual block programming tanpa ada satu pun kode *sintaks* program di dalamnya (Hariyanto, 2014).

MIT App Inventor adalah aplikasi yang disediakan oleh Google dan sekarang dikelola oleh *Massachusetts Institute of Technology (MIT)*. MIT App Inventor memungkinkan setiap orang (termasuk orang-orang yang tidak mempunyai *basic programming*) untuk membuat aplikasi perangkat lunak untuk sistem operasi Android. MIT App Inventor menggunakan antarmuka grafis yang memungkinkan pengguna untuk *drag and drop* sebuah objek visual untuk

menciptakan aplikasi yang dapat berjalan pada sistem android, yang pada saat ini dipakai oleh banyak perangkat *handphone*.

Aplikasi MIT App Inventor ini harus diakses secara online pada sebuah web Browser. MIT App Inventor memiliki dua komponen utama yaitu:

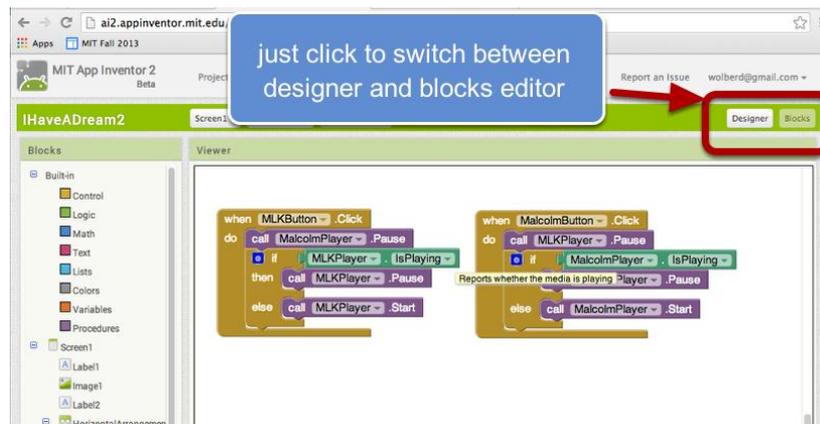
1. *The App Inventor Designer* adalah aplikasi dimana pengguna melakukan perancangan antarmuka untuk aplikasi yang akan dibangun.



Gambar 2.3. Tampilan App Inventor Designer

(Sumber: <http://appinventor.mit.edu/explore/sites/all/files/SetupAssets/ai2.png>)

2. *The App Inventor Blocks Editor* adalah aplikasi dimana pengguna merakit blok program yang menentukan bagaimana komponen harus bersikap. Dengan instruksi pemrograman yang sudah di kelompokkan dan disusun layaknya seperti *puzzle*.



Gambar 2.4. Tampilan App Inventor Block Editor

(Sumber: http://www.appinventor.org/screensteps/ai2Changes/images/App_Inventor_2_Changes/media_1386079849118_lg.png)

Keuntungan menggunakan MIT App Inventor adalah pengembang tidak memerlukan pengetahuan java untuk mengembangkan aplikasi android, sehingga setiap pemilik perangkat telepon genggam berbasis android dapat menciptakan aplikasinya sendiri, tanpa perlu mengeluarkan uang sepeser pun dan tanpa perlu mempelajari bahasa pemrograman java.

Pada pengembangan konvensional menggunakan java, pengembang aplikasi harus melakukan *coding* untuk tampilan *user interface*, sedangkan pada aplikasi MIT App Inventor, *user interface* sudah tersedia sehingga mempersingkat waktu untuk pengembangan aplikasi multimedia.

Pada penelitian ini MIT App Inventor digunakan sebagai *software* untuk membuat aplikasi *Remote Control* pada *smartphone* Android yang digunakan untuk mengontrol dan memonitori sistem putaran motor mobil RC.

2.1.2.3 Mikrokontroler Arduino Mega2560

Mikrokontroler AVR merupakan pengontrol utama standar industri dan riset saat ini. Hal ini dikarenakan berbagai kelebihan yang dimilikinya

dibandingkan mikroprosesor, yaitu murah, dukungan *software* dan dokumentasi yang memadai, dan memerlukan komponen pendukung yang sangat sedikit (Budiharto, 2008).

Arduino Mega adalah salah satu produk berlabel Arduino yang sebenarnya adalah papan elektronik yang mengandung mikrokontroler Atmega2560 (sebuah keping yang secara fungsional bertindak seperti sebuah komputer). Piranti ini dapat dimanfaatkan untuk mewujudkan rangkaian elektronik dari yang sederhana hingga yang kompleks (Kadir, 2013). Secara fisik, Arduino Mega 2560 memiliki *board* berukuran lebih besar dibanding arduino tipe lainnya. Hal tersebut dikarenakan *board* ini memiliki pin analog, pin digital, serta pin komunikasi yang lebih banyak dibanding arduino tipe lainnya.



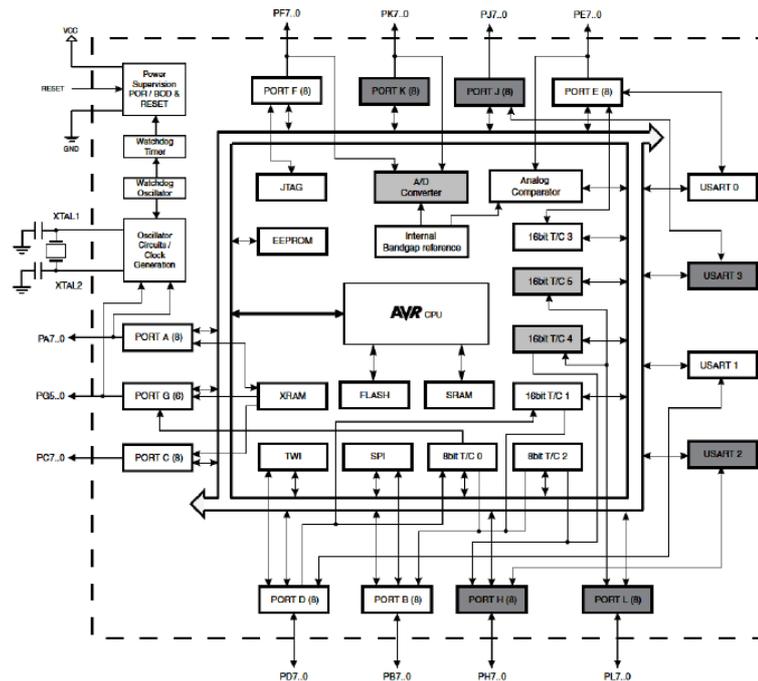
Gambar 2.5. Arduino Mega2560

(Sumber: <http://www.electroschematics.com/wp-content/uploads/2013/01/Arduino-Mega-2560-Pinout.jpg>)

Pada **Gambar 2.5**, merupakan Arduino Mega2560. Arduino tipe ini memiliki pin analog sebanyak 16 pin, pin digital I/O sebanyak 54 pin, serta pin komunikasi serial sebanyak 4 pasang pin.

1. Keterangan Spesifikasi dari Arduino Mega 2560

- a. Diagram I/O Atmega 2560



Gambar 2.6. Blok Diagram ATmega 2560

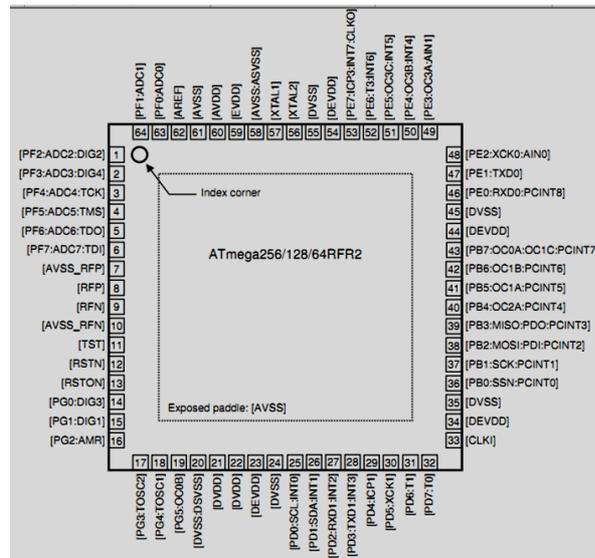
Blok diagram gambar 2.5 merupakan diagram alir dan jalur data serta port – port yang terdapat pada ATmega 2560.

Tabel 2.1. Spesifikasi Arduino Mega 2560 (Arduino.cc, 2016)

Microcontroller	ATmega2560
<i>Operating Voltage</i>	5V
<i>Input Voltage (recommended)</i>	7-12V
<i>Input Voltage (limits)</i>	6-20V
<i>Digital I/O Pins</i>	54 (of wich 15 provide PWM output)
<i>Analog Input Pins</i>	16
<i>DC Current per I/O Pin</i>	40mA
<i>DC Current for 3.3V Pin</i>	50mA
<i>Flash Memory</i>	256 KB of wich 8 KB used by bootloader
SRAM	8 KB

EEPROM	4 KB
<i>Clock Speed</i>	16 Mhz

b. Diagram Pin ATmega2560



Gambar 2.7 Konfigurasi pin Arduino Mega2560

Beberapa fungsi konfigurasi dari pin ATmega2560 adalah :

1. VCC, Dihubungkan ke sumber tegangan 4,5 – 5 volt
2. GND, Dihubungkan ke ground.
3. Reset, Mengembalikan kondisi kerja mikrokontroler pada posisi awal pin ini harus bernilai 1 agar fungsi pin ini dapat bekerja .
4. XTAL1, Masukan ke penguat *inverting osilator* dan masukan ke rangkaian *clock timer*
5. XTAL2, Keluaran dari penguat *inverting osilator*.

2.1.2.4 Mikrokontroler Arduino Nano328

Mikrokontroller lebih dari sekedar sebuah mikroprosesor karena sudah terdapat ROM (Read-Only Memory), RAM (Read-Write Memory), beberapa port masukan maupun keluaran, dan beberapa peripheral seperti pencacah/pewaktu,

diperlukan lagi pada papan Arduino versi terakhir karena pemilihan sumber daya eksternal atau USB dilakukan secara otomatis.

4. 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 *microcontroller* agar melakukan sebuah operasi untuk setiap detak-nya. Kristal ini dipilih yang berdetak 16 juta kali per detik (16MHz).

6. 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 *microcontroller*.

7. *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.

8. *IC Microcontroller Atmega*

Komponen utama dari papan Arduino, di dalamnya terdapat CPU, ROM dan RAM.

9. X1 – Sumber daya eksternal

Jika hendak disuplai dengan sumber daya eksternal, papan Arduino dapat diberikan tegangan DC antara 9-12V.

10. 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.

Terdapat satu buah Arduino Nano yang digunakan pada penelitian kali ini yang dipasang pada robot untuk mengolah data serial yang dikirim dari *smartphone* dan mengontrol aktuator (motor servo).

2.1.2.5 *Software* Arduino IDE

Software ini digunakan untuk menulis program pada board arduino. IDE (*Integrated Development Environment*) adalah *software* yang sangat berperan untuk menulis program, meng-compile menjadi kode biner dan mengupload ke dalam *memory* mikrokontroler (Syahwil, 2013).

Arduino IDE merupakan *software* khusus ntuk memprogram *board* arduino dengan bahasa C sebagai dasar pemrogramannya. Arduino IDE dibuat khusus untuk memudahkan dalam pembuatan *sintaks* program arduino yang sifatnya *open source* dengan menyediakan berbagai *library* yang dapat di unduh secara gratis di situs resmi arduino, yaitu www.arduino.cc.

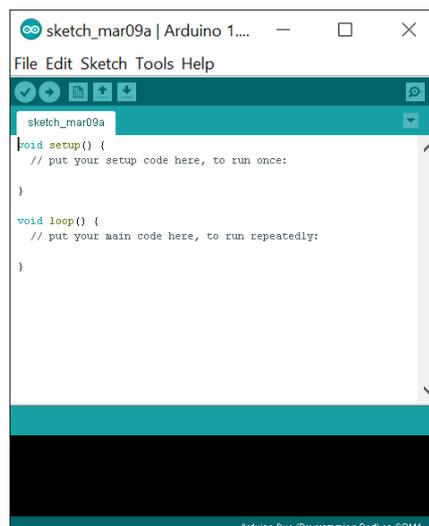
Software IDE Arduino yang digunakan pada penelitian kali yaitu Arduino IDE versi 1.6.8, **Gambar 2.9**.menunjukkan layar Splash Arduino IDE versi 1.6.8.



Gambar 2.9. Form *Splash* Arduino Versi 1.6.8

(Sumber: <https://blog.arduino.cc/wp-content/uploads/2015/11/Screenshot-2015-11-03-12.27.18.png>)

Arduino IDE sangat mudah digunakan dan banyak sekali pengembang di seluruh dunia yang terus berkontribusi baik dalam penyempurnaan Arduino IDE maupun *library-library* yang dapat digunakan di dalam Arduino IDE. *Library* adalah istilah untuk suatu program fungsi yang dapat digunakan untuk memanipulasi data maupun digunakan untuk menggunakan hardware tertentu. Tampilan layar program Arduino IDE dapat dilihat pada **Gambar 2.10.**



Gambar 2.10. Tampilan Arduino IDE versi 1.6.8

(Sumber: <https://blog.arduino.cc/wp-content/uploads/2016/03/before.png>)

2.1.3 Komponen Sensor

Menurut Kamus Besar Bahasa Indonesia, Sensor adalah elemen yg mengubah sinyal fisik menjadi sinyal elektronik yg dibutuhkan *computer*.

Secara umum sensor didefinisikan sebagai alat yang mampu menangkap fenomena fisika atau kimia kemudian mengubahnya menjadi sinyal elektrik baik arus listrik ataupun tegangan (Electronic,2015). Fenomena fisik yang mampu menstimulus sensor untuk menghasilkan sinyal elektrik meliputi temperatur, tekanan, gaya, medan magnet cahaya, pergerakan dan sebagainya. Penggunaan sensor pada penelitian ini menggunakan satu buah sensor, yaitu sensor optocoupler.

2.1.3.1 *Sensor Optocoupler*

Optocoupler adalah suatu piranti yang terdiri dari 2 bagian yaitu *transmitter* dan *receiver*, yaitu antara bagian cahaya dengan bagian deteksi sumber cahaya terpisah. Biasanya optocoupler digunakan sebagai saklar elektrik, yang bekerja secara otomatis. Pada dasarnya Optocoupler adalah suatu komponen penghubung (*coupling*) yang bekerja berdasarkan picu cahaya optic. Optocoupler terdiri dari dua bagian yaitu:

1. Pada *transmitter* dibangun dari sebuah LED infra merah. Jika dibandingkan dengan menggunakan LED biasa, LED infra merah memiliki ketahanan yang lebih baik terhadap sinyal tampak. Cahaya yang dipancarkan oleh LED infra merah tidak terlihat oleh mata telanjang.

2. Pada bagian *receiver* dibangun dengan dasar komponen *Photodiode*. Photodiode merupakan suatu transistor yang peka terhadap tenaga cahaya. Suatu sumber cahaya menghasilkan energi panas, begitu pula dengan spektrum infra merah. Karena spektrum inframerah mempunyai efek panas yang lebih besar dari cahaya tampak, maka *Photodiode* lebih peka untuk menangkap radiasi dari sinar infra merah.

Oleh karena itu Optocoupler dapat dikatakan sebagai gabungan dari LED infra merah dengan fototransistor yang terbungkus menjadi satu chips. Cahaya infra merah termasuk dalam gelombang elektromagnetik yang tidak tampak oleh mata telanjang. Sinar ini tidak tampak oleh mata karena mempunyai panjang gelombang, berkas cahaya yang terlalu panjang bagi tanggapan mata manusia. Sinar infra merah mempunyai daerah frekuensi 1×10^{12} Hz sampai dengan 1×10^{14} GHz atau daerah frekuensi dengan panjang gelombang $1\mu\text{m} - 1\text{mm}$.

LED infra merah ini merupakan komponen elektronika yang memancarkan cahaya infra merah dengan konsumsi daya sangat kecil. Jika diberi bias maju, LED infra merah yang terdapat pada optocoupler akan mengeluarkan panjang gelombang sekitar 0,9 mikrometer.

Proses terjadinya pancaran cahaya pada LED infra merah dalam optocoupler adalah sebagai berikut. Saat dioda menghantarkan arus, elektron lepas dari ikatannya karena memerlukan tenaga dari catu daya listrik. Setelah elektron lepas, banyak elektron yang bergabung dengan lubang yang ada di sekitarnya (memasuki lubang lain yang kosong). Pada saat masuk lubang yang lain, elektron melepaskan tenaga yang akan diradiasikan dalam bentuk cahaya, sehingga dioda

akan menyala atau memancarkan cahaya pada saat dilewati arus. Cahaya infra merah yang terdapat pada optocoupler tidak perlu lensa untuk memfokuskan cahaya karena dalam satu chip mempunyai jarak yang dekat dengan penerimanya. Pada optocoupler yang bertugas sebagai penerima cahaya infra merah adalah fototransistor. Fototransistor merupakan komponen elektronika yang berfungsi sebagai detektor cahaya infra merah. Detektor cahaya ini mengubah efek cahaya menjadi sinyal listrik, oleh sebab itu fototransistor termasuk dalam golongan detektor optik.

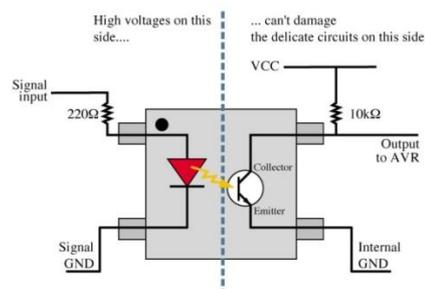
Fototransistor memiliki sambungan kolektor–basis yang besar dengan cahaya infra merah, karena cahaya ini dapat membangkitkan pasangan lubang elektron. Dengan diberi bias maju, cahaya yang masuk akan menimbulkan arus pada kolektor. Fototransistor memiliki bahan utama yaitu germanium atau silikon yang sama dengan bahan pembuat transistor. Tipe fototransistor juga sama dengan transistor pada umumnya yaitu PNP dan NPN. Perbedaan transistor dengan fototransistor hanya terletak pada dindingnya yang memungkinkan cahaya infra merah mengaktifkan daerah basis, sedangkan transistor biasa ditempatkan pada dinding logam yang tertutup.

Ditinjau dari penggunaannya, fisik optocoupler dapat berbentuk bermacam-macam. Bila hanya digunakan untuk mengisolasi level tegangan atau data pada sisi transmitter dan sisi receiver, maka optocoupler ini biasanya dibuat dalam bentuk solid (tidak ada ruang antara LED dan Photodiode). Sehingga sinyal listrik yang ada pada input dan output akan terisolasi. Dengan kata lain optocoupler ini digunakan sebagai optoisolator jenis IC.

Prinsip kerja dari optocoupler adalah :

1. Jika antara *Photodiode* dan LED terhalang maka *Photodiode* tersebut akan off sehingga *output* dari kolektor akan berlogika high.
2. Sebaliknya jika antara *Photodiode* dan LED tidak terhalang maka *Photodiode* dan LED tidak terhalang maka *Photodiode* tersebut akan on sehingga outputnya akan berlogika low.

Sebagai piranti elektronika yang berfungsi sebagai pemisah antara rangkaian power dengan rangkaian control. Komponen ini merupakan salah satu jenis komponen yang memanfaatkan sinar sebagai pemicu on/off-nya. Opto berarti optic dan coupler berarti pemicu. Sehingga bisa diartikan bahwa optocoupler merupakan suatu komponen yang bekerja berdasarkan picu cahaya optic opto-coupler termasuk dalam sensor, dimana terdiri dari dua bagian yaitu *transmitter* dan *receiver*. Dasar rangkaian dapat ditunjukkan seperti pada gambar dibawah ini:



Gambar 2.11. Sensor Optocoupler

(Sumber: <http://id.pinterest.com/dinesvion/optocoupler-chip-circuit-circuits-ctr-opto-coupler>)

Sebagai pemancar atau transmitter dibangun dari sebuah led infra merah untuk mendapatkan ketahanan yang lebih baik daripada menggunakan led biasa. Sensor ini bisa digunakan sebagai isolator dari rangkaian tegangan rendah

kerangkaian tegangan tinggi. Selain itu juga bisa dipakai sebagai pendeteksi adanya penghalang antara *transmitter* dan *receiver* dengan memberi ruang uji dibagian tengah antara led dengan photo transistor. Penggunaan ini bisa diterapkan untuk mendeteksi putaran motor atau mendeteksi lubang penanda disket pada disk *drive computer*. Tapi pada alat yang penulis buat optocoupler untuk mendeteksi putaran. Berikut tampilan fisik sensor optocoupler:



Gambar 2.12. Bentuk Fisik Sensor Optocoupler

(Sumber: http://static3.tme.eu/products_pics/0/d/e/0deca0712deffba24a0a347d0e6c1a65/482368.jpg)

Adapun jenis sensor optocoupler yang peneliti gunakan adalah Beam Photoelectric Sensor. Berikut adalah bentuk fisik sensor yang digunakan.



Gambar 2.13. Bentuk Fisik Beam Photoelectric Sensor

(Sumber: http://img.dxcn.com/productimages/sku_140554_2.jpg)

Sensor Optocoupler dapat dikendalikan oleh arduino dengan mudah menggunakan *sketch* yang telah disediakan oleh *website* dari sensor optocoupler

<https://brainy-bits.com/tutorials/speed-sensor-with-arduino/>. Dengan konfigurasi pin 2 sebagai output sensor optocoupler dan pin 3, 4 sebagai input motor DC. Berikut ini adalah contoh program arduino sensor optocoupler dari *website* resmi sensor optocoupler.

```
int encoder_pin = 2; // The pin the encoder is connected
unsigned int rpm; // rpm reading
volatile byte pulses; // number of pulses
unsigned long timeold;
// The number of pulses per revolution
// depends on your index disc!!
unsigned int pulsesperturn = 20;
int motor = 3;
void counter()
{
  //Update count
  pulses++;
}

void setup()
{
  Serial.begin(9600);
  //Use statusPin to flash along with interrupts
  pinMode(encoder_pin, INPUT);
  pinMode (motor, OUTPUT);

  //Interrupt 0 is digital pin 2, so that is where the IR detector is connected
  //Triggers on FALLING (change from HIGH to LOW)
  attachInterrupt(0, counter, FALLING);
  // Initialize
  pulses = 0;
  rpm = 0;
  timeold = 0;
}
void loop()
{
  if (millis() - timeold >= 1000){ /*Uptade every one second, this will be equal to
  reading frecuency (Hz).*/

  //Don't process interrupts during calculations
  detachInterrupt(0);
  //Note that this would be 60*1000/(millis() - timeold)*pulses if the interrupt
  //happened once per revolution
```

```

rpm = (40 * 1000 / pulsesperturn )/ (millis() - timeold)* pulses;
timeold = millis();
pulses = 0;

//Write it out to serial port
Serial.print("RPM = ");
Serial.println(rpm);
//Restart the interrupt processing
attachInterrupt(0, counter, FALLING);
analogWrite (motor, 200);
}
}

```

2.1.4 Komponen Penggerak

2.1.4.1 Motor

Motor merupakan perangkat elektromagnetik yang dapat mengubah energi listrik menjadi energi mekanik. Perubahan ini dilakukan dengan merubah tenaga listrik menjadi magnet yang disebut sebagai elektromagnet. Sebagaimana kita ketahui bahwa : kutub-kutub dari magnet yang senama akan tolak-menolak dan kutub-kutub tidak senama, tarik-menarik. Maka kita dapat memperoleh gerakan jika kita menempatkan sebuah magnet pada sebuah poros yang dapat berputar, dan magnet yang lain pada suatu kedudukan yang tetap. Dengan cara inilah energi listrik dapat diubah menjadi energi mekanik.

Energi mekanik ini digunakan untuk, misalnya memutar impeler pompa, fan atau blower, menggerakkan kompresor, mengangkat bahan, dll. Motor listrik digunakan juga dirumah (mixer, bor listrik, afan angin) dan industri. Motor listrik kadangkala disebut “kuda kerja” nya industri sebab diperkirakan bahwa motor-motor menggunakan sekitar 70% beban total industry

Secara umum motor listrik dapat dibagi menjadi motor ac dan motor dc, pembagian ini berdasarkan pada arus listrik yang digunakan untuk

menggerakkannya. Namun penulis pada bagian ini kita hanya membahas mengenai motor dc.

2.1.4.2 Motor Servo



Gambar 2.14. Bentuk Fisik Motor Servo

(Sumber: *Datasheet motor servo tower Pro MG90s*)

Motor servo merupakan motor DC yang memiliki rangkaian kontrol elektronik dan internal gear untuk mengendalikan pergerakan dan sudut sudutnya (Iswanto, 2011).

Motor servo dikemas dalam berbentuk kotak segi empat dengan sebuah output shaft motor dan konektor dengan 3 kabel yaitu *power*, *control*, dan *ground*. *Gear motor servo* ada yang terbuat dari plastik, metal, atau titanium. Di dalam motor servo terdapat potensiometer yang digunakan sebagai sensor posisi. Potensiometer tersebut dihubungkan dengan output shaft untuk mengetahui posisi aktual shaft. Ketika motor dc berputar, maka *output shaft* juga berputar dan sekaligus memutar potensiometer. Rangkaian kontrol kemudian dapat membaca kondisi potensiometer tersebut untuk mengetahui posisi aktual *shaft*. Jika posisinya sesuai dengan yang diinginkan, maka motor dc akan berhenti. Sudut posisi motor servo (*Operating Angle*) bervariasi tergantung jenis motor servo (Andrianto, 2008). Sistem mekanis motor servo tampak pada **Gambar 2.15**.



Gambar 2.15. Sistem Mekanis Servo

(Sumber : <http://www.kelasrobot.com/2014/11/macam-macam-actuator-motor-robot.html>)

Ada dua jenis motor servo yaitu:

- *Motor Servo Standard*

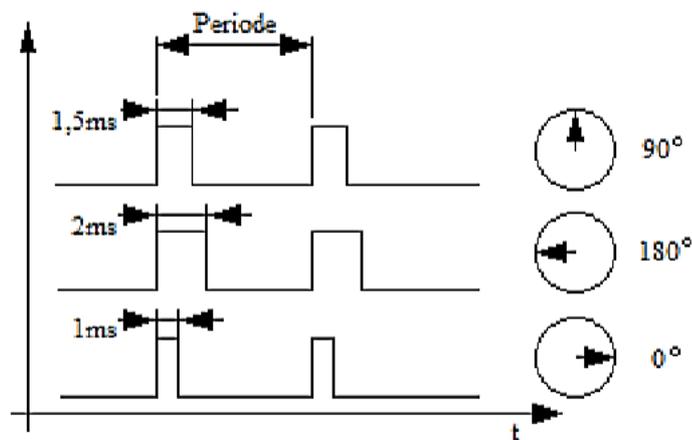
Yaitu motor servo yang mampu bergerak CW (*Clock Wise* atau searah jarum jam) dan CCW (*Counter Clock Wise* atau berlawanan arah jarum jam) dengan sudut operasi tertentu, misalnya 60, 90, atau 180 derajat. Motor servo standard 180 derajat merupakan motor dc yang digunakan dalam penelitian ini.

- *Motor Servo Continuous*

Yaitu motor servo yang mampu bergerak CW dan CCW tanpa batasan sudut operasi (berputar secara kontinyu).

Motor servo biasanya menggunakan tegangan suplai 4,8 hingga 7,2 volt. Motor servo dikendalikan dengan cara mengirimkan sebuah pulsa yang lebar pulsanya bervariasi. Pulsa tersebut dimasukkan melalui kabel kontrol motor servo. Biasanya lebar pulsanya antara 1ms sampai 2ms dengan periode lebar pulsa sebesar 20 ms. Lebar pulsa akan mengakibatkan perubahan posisi pada servo.

Misalnya sebuah pulsa 1.5 ms akan memutar motor pada posisi 90 derajat (posisi netral). Agar posisi servo tetap pada posisi ini, maka pulsa harus terus diberikan pada servo. Jadi meskipun ada gaya yang melawan, servo akan tetap bertahan pada posisinya. Gaya maksimum servo tergantung dari rentang torsi servo (Andrianto, 2008).



Gambar 2.16. Lebar Pulsa dan Posisi Servo

(Sumber: Pemrograman mikrokontroler AVR ATmega16, 2008)

Pada penelitian kali ini, peneliti menggunakan satu buah motor servo yang di letakan di bagian depan mobil berfungsi sebagai aktuator penggerak untuk memberikan aktuasi arah baik arah kanan maupun kiri.

Motor servo dapat dikendalikan oleh arduino dengan mudah menggunakan *library* “servo.h” yang telah disediakan pada arduino IDE. Pada contoh pengendalian motor servo menggunakan arduino, pin 9 arduino dihubungkan dengan pin signal motor servo. Berikut ini merupakan contoh program motor servo yang terdapat pada bagian “*Example*” dari arduino IDE.

```
/* Sweep
by BARRAGAN <http://barraganstudio.com>
This example code is in the public domain.
modified 8 Nov 2013
```

by Scott Fitzgerald

<http://www.arduino.cc/en/Tutorial/Sweep>

*/

```
#include <Servo.h>
Servo myservo; // create servo object to control a servo
// twelve servo objects can be created on most boards
int pos = 0; // variable to store the servo position
void setup() {
  myservo.attach(9); // attaches the servo on pin 9 to the servo object
}
void loop() {
  for (pos = 0; pos <= 180; pos += 1) { // goes from 0 degrees to 180 degrees
    // in steps of 1 degree
    myservo.write(pos); // tell servo to go to position in variable 'pos'
    delay(15); // waits 15ms for the servo to reach the position
  }
  for (pos = 180; pos >= 0; pos -= 1) { // goes from 180 degrees to 0 degrees
    myservo.write(pos); // tell servo to go to position in variable 'pos'
    delay(15); // waits 15ms for the servo to reach the position
  }
}
```

2.1.4.3 Motor DC

Motor DC adalah motor yang menggunakan sumber tegangan DC dan digunakan untuk mengubah *energy* listrik menjadi *energy* mekanik. Komponen ini bekerja dengan prinsip elektromagnetik. Ketika sumber tegangan diberikan, medan magnet dibagian yang diam atau disebut stator akan terbentuk. Medan magnet ini membuat rotor atau bagian yang bergerak berputar dan tentu saja dapat dimanfaatkan untuk memutar benda lain, misalnya roda.

Kecepatan putaran motor DC ditentukan oleh besar tegangan. Semakin tinggi tegangannya, semakin cepat putarannya. Namun, tentu saja tegangan yang diberikan ke motor DC ada batasannya. Tegangan yang terlampau tinggi, yang melampaui batas maksimumnya, dapat membuat motor terbakar.



Gambar 2.17. Motor DC

(Sumber: <http://ecx.images-amazon.com/images/I/41No3C5HTgL.SY300.jpg>)

Ketika pasokan tegangan ke motor DC dihentikan, medan magnet berangsur-angsur menghilang, sekaligus menghasilkan tegangan balik. (Abdul Kadir, 2013)

Sering kali gerakan-gerakan pada robot membutuhkan torsi yang cukup besar dan hal ini tidak dapat dilakukan oleh motor dc saja. Untuk memperbesar torsi dibutuhkan rangkaian gir yang mereduksi kecepatan motor sekaligus meningkatkan torsi. Proses pengurangan kecepatan dan pengurangan torsi ini berbanding terbalik dan dihitung berdasarkan perbandingan gigi (Nalwan, 2012). *Gear Ratio* (Perbandingan gigi) menentukan perbandingan kecepatan motor dc dengan persamaan berikut (ITB, 2011).

$$GR = \frac{R}{r} = \frac{w1}{w2}$$

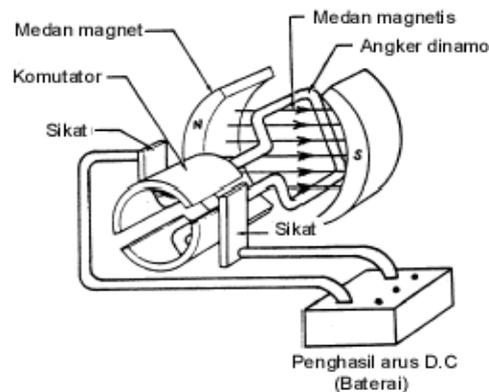
- ❖ R = Jari-jari gigi yang terhubung ke roda
- ❖ r = Jari-jari gigi yang terhubung ke motor dc
- ❖ w1 = kecepatan sudut atau putaran motor dc
- ❖ w2 = kecepatan sudut atau putaran roda.

Roda dalam hal ini dapat diartikan sebagai benda yang diputar oleh motor dc *gearbox*. Pada penelitian kali ini, peneliti menggunakan 1 (satu) buah motor DC *gearbox* yang berfungsi sebagai aktuator roda mobil RC.

Karena kecepatan motor berbanding lurus dengan potensial yang diberikan maka dapat mengatur kecepatan dengan mengecilkan dan membesarkan potensial tegangan pada motor. Yakni dengan cara menggunakan teknik *PWM (Pulse Width Modulation)*. Teknik ini sering digunakan untuk kontrol motor aktuator. Teknik ini mengatur pengaturan lebar pulsa yang diberikan kepada motor, semakin besar (lebar pulsa) maka kecepatan motor akan melaju maksimal.

2.1.4.3.1 Putaran Motor DC

Putaran Motor DC memerlukan suplai tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik. Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Jika terjadi putaran pada kumparan jangkar dalam pada medan magnet, maka akan timbul tegangan (GGL) yang berubah-ubah arah pada setiap setengah putaran, sehingga merupakan tegangan bolak-balik. Prinsip kerja dari arus searah adalah membalik fasa tegangan dari gelombang yang mempunyai nilai positif dengan menggunakan komutator, dengan demikian arus yang berbalik arah dengan kumparan jangkar yang berputar dalam medan magnet. Bentuk motor paling sederhana memiliki kumparan satu lilitan yang bisa berputar bebas di antara kutub-kutub magnet permanen.



Gambar 2.18. Putaran Motor DC

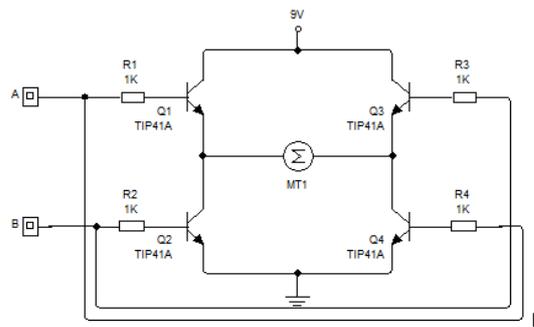
(Sumber: <http://3.bp.blogspot.com/-TTMYhP1qEy4/UHrj9bcINwI/AAAAAAAAAKA/t4O6ykLrris/s1600/Motor-DC-Sederhana.jpg>)

Catu tegangan dc dari baterai menuju ke lilitan melalui sikat yang menyentuh komutator, dua segmen yang terhubung dengan dua ujung lilitan. Kumbaran satu lilitan pada gambar di atas disebut angker dinamo. Angker dinamo adalah sebutan untuk komponen yang berputar di antara medan magnet.

2.1.4.3.2 Driver Motor DC H-Bridge

Motor DC tipe H-Bridge menggunakan power driver berupa transistor. Rangkaian driver motor DC H-Bridge transistor ini dapat mengendalikan arah putaran motor DC dalam 2 arah dan dapat dikontrol dengan metode PWM (pulse Width Modulation) maupun metode sinyal logika dasar TTL (High) dan (Low). Untuk pengendalian motor DC dengan metode PWM maka dengan rangkaian driver motor DC ini kecepatan putaran motor DC dapat dikendalikan dengan baik. Apabila menggunakan metode logika TTL 0 dan 1 maka rangkaian ini hanya dapat mengendalikan arah putaran motor DC saja dengan kecepatan putaran motor DC maksimum. Rangkaian driver motor DC H-Bridge ini menggunakan rangkaian jembatan transistor 4 unit dengan proteksi impuls tegangan induksi

motor DC berupa dioda yang dipasang paralel dengan masing-masing transistor secara reverse bias. Rangkaian driver motor DC secara detail dapat dilihat pada gambar berikut.



Gambar 2.19. Rangkaian Driver Motor DC H-Bridge Transistor

(Sumber: <http://3.bp.blogspot.com/-rKQYaOq5oho/U8UZT0WNsSI/AAAAAAAAAr4/5i9t8AUcUaw/s1600/H-bridge.png>)

Proses mengendalikan motor DC menggunakan rangkaian driver motor DC H-Bridge diatas dapat diuraikan dalam beberapa bagian sebagai berikut:

Driver Motor DC dengan metode logika TTL (0 dan 1) atau *High* dan *Low* hanya dapat mengendalikan arah putar motor DC dalam 2 arah tanpa pengendalian kecepatan putaran (kepatan maksimum). untuk mengendalikan motor DC dalam 2 arah dengan rangkaian driver motor dc h-bridge diatas konfigurasi kontrol pada jalur input adalah dengan memberikan input berupa logika TTL ke jalur input A dan B.

- ✓ Untuk mengendalikan arah putar searah jarum jam adalah dengan memberikan logika TTL 1 (*high*) pada jalur input A dan logika TTL 0 (*low*) pada jalur input B.

- ✓ Untuk mengendalikan arah putar berlawanan arah jarum jam adalah dengan memberikan logika TTL 1 (*high*) pada jalur input B dan logika TTL 0 (*low*) pada jalur input A.

Driver motor DC dengan metode PWM (*Pulse Width Modulation*) dapat mengendalikan arah putaran motor DC dan kecepatan motor DC menggunakan pulsa PWM yang diberikan ke jalur input A dan B, dimana konfigurasi sinyal kontrol sebagai berikut.

- ✓ Untuk mengendalikan arah putar motor DC searah jarum jam dengan kecepatan dikendalikan pulsa PWM maka jalur input B selalu diberikan logika TTL 0 (*Low*) dan jalur input A diberikan pulsa PWM.
- ✓ Untuk mengendalikan arah putar motor DC berlawanan arah jarum jam dengan kecepatan dikendalikan pulsa PWM maka jalur input A selalu diberikan logika TTL 0 (*Low*) dan jalur input B diberikan pulsa PWM.

Kecepatan putaran motor DC dikendalikan oleh persentasi *ton-duty cycle* pulsa PWM yang diberikan ke jalur input rangkaian driver motor DC h-bridge transistor diatas.

Motor DC dapat dikendalikan oleh arduino dengan mudah menggunakan program yang bisa dibuat sendiri pada *software* arduino IDE. Pada contoh pengendalian motor DC menggunakan arduino, pin 3 dan 4 dihubungkan dengan pin input dari driver motor DC. Berikut ini merupakan contoh program motor DC yang dapat diatur kecepatan putar menggunakan delay.

```
// MotorDC
int cepat = 3;
int lambat = 4;
```

```
void setup()
{
  Serial.begin (9600);
  //Kontrol Input & Output
  pinMode(cepat, OUTPUT);
  pinMode(lambat, OUTPUT);
}
void loop()
{
  Serial.println(cepat);
  Serial.println(lambat);
  {
  //Kendali Putaran RPM
  {
for(int fadeValue = 0 ; fadeValue <= 100; fadeValue +=5)
  analogWrite(cepat, fadeValue);
  delay (10000);
for(int fadeValue = 0 ; fadeValue <= 150; fadeValue +=5)
  analogWrite(cepat, fadeValue);
  delay (10000);
for(int fadeValue = 0 ; fadeValue <= 200; fadeValue +=5)
  analogWrite(cepat, fadeValue);
  delay (10000);
for(int fadeValue = 0 ; fadeValue <= 255; fadeValue +=5)
  analogWrite(cepat, fadeValue);
  delay (10000);
for(int fadeValue = 255 ; fadeValue >= 200; fadeValue -=5)
  analogWrite(lambat, fadeValue);
  delay (10000);
for(int fadeValue = 200 ; fadeValue <= 150; fadeValue +=5)
  analogWrite(lambat, fadeValue);
```

```

    delay (10000);
for(int fadeValue = 150 ; fadeValue >= 100; fadeValue -=5)
analogWrite(lambat, fadeValue);
delay (10000);
for(int fadeValue = 100 ; fadeValue >= 0; fadeValue -=5)
analogWrite(lambat, fadeValue);
}
}
}

```

2.1.5 Perangkat Komunikasi Data

Komunikasi data adalah transmisi data elektronik melalui beberapa media yang dapat berupa kabel koaksial, fiber optic, microwave, dan sebagainya (Gunadarma, 1994). Untuk terjadinya komunikasi data, perangkat harus menjadi bagian dari sistem komunikasi, yang terdiri dari komunikasi hardware (perangkat keras) dan program / *software* (perangkat lunak). Dalam penelitian kali ini, komunikasi data yang digunakan adalah *Bluetooth*.

2.1.5.1 Definisi Bluetooth

Sebuah sistem *Bluetooth* adalah sebuah saluran nirkabel jarak pendek berbasis gelombang radio yang perangkat kesistemannya dikembangkan dalam bentuk sebuah *microchip* (Stallings, 2007).

Adapun menurut (Supriyanto, 2006) *Bluetooth* adalah sebuah teknologi komunikasi *wireless* (tanpa kabel) yang beroperasi dalam pita frekuensi 2,4 GHz *unlicensed ISM (Industrial, Scientific and Medical)* dengan menggunakan sebuah *frequency hopping transceiver* yang mampu menyediakan layanan komunikasi data dan suara secara *real-time* antara *host-host Bluetooth* dengan jarak jangkauan yang terbatas.

Pada dasarnya teknologi *Bluetooth* ini dibuat bukan untuk menggantikan penggunaan media kabel dalam melakukan pertukaran data atau informasi, tetapi juga mampu menawarkan fitur yang baik dalam hal teknologi *mobile wireless* dengan biaya dan konsumsi daya yang relatif rendah, mudah dalam pengoperasiannya serta mampu menyediakan berbagai macam

Teknik *frequency-hopping* (lompatan antar-pita frekuensi) dengan jarak antar-frekuensi pembawa selebar 1 MHz. Satuan jaringan terkecil dalam skema *Bluetooth* adalah sebuah piconet yang terdiri dari sebuah perangkat yang berfungsi sebagai *master* (majikan) dan satu layanan.

Sebuah sistem *Bluetooth* menerapkan hingga tujuh buah perangkat lainnya yang berfungsi sebagai *slave* (budak). Perangkat radio yang berperan sebagai *master* akan menentukan kanal mana (melalui skema *frequency-hopping*) dan fasa mana (yaitu waktu transmisi) yang akan digunakan oleh tiap-tiap perangkat *slave* yang ada dibawah perangkat komandonya. Perangkat *master* akan menggunakan alamat perangkatnya sendiri sebagai salah satu parameter dalam membuat keputusan-keputusan dimana perangkat-perangkat *slave* harus dapat melakukan sinkronisasi dengan kanal dan fasa sang *master*. Sebuah *slave* hanya dapat berkomunikasi dengan perangkat *master* ketika diberi izin oleh sang *master*.

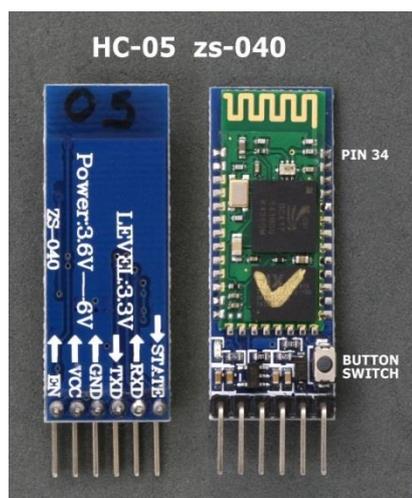
Adapun dua jenis kanal fisik yang dapat dibangun diantara sebuah *master* dan sebuah *slave* adalah (Stallings, 2007):

1. Berorientasi koneksi, sinkron (*synchronous connection oriented-SCO*), yaitu mengalokasikan sejumlah tetap *bandwidth* untuk setiap jalur radio yang menghubungkan *master* dengan sebuah *slave*.

2. Tanpa koneksi, asinkron (*asynchronous connectionless-ACL*) yaitu sebuah koneksi satu-titik-ke-banyak-titik (*point-to-multipoint*) yang menghubungkan perangkat *master* dengan semua *slave* yang ada dalam piconet.

2.1.1.1 Modul Bluetooth HC-05

Pada penelitian kali ini, peneliti menggunakan modul Bluetooth HC-05 sebagai perangkat komunikasi data antara robot pengambil benda dengan *smartphone* yang dijadikan sebagai *remote control*. Modul Bluetooth HC-05 adalah *bluetooth* yang memiliki komunikasi serial UART dalam penerimaan dan pengiriman datanya. *Bluetooth* HC-05 memungkinkan dapat berkomunikasi langsung dengan arduino melalui komunikasi serial. Pada dasarnya, *Bluetooth* HC-05 dapat dikonfigurasi sebagai *master* dan *slave*, yang artinya *Bluetooth* HC-05 ini dapat digunakan untuk mengirim dan menerima data. Berbeda dengan seri *Bluetooth* bernomer genap seperti HC-06 yang hanya dapat digunakan sebagai *slave* atau penerima data. Berikut adalah bentuk fisik dari *Bluetooth* HC-05.



Gambar 2.20. Bentuk Fisik Modul Bluetooth HC-05

(Sumber: *Datasheet* modul *Bluetooth* HC-05)

Modul *Bluetooth* HC-05 memiliki spesifikasi dalam penggunaannya antara lain:

- Sensitivitas -80dBm (*Typical*)
- Daya transmisi RF sampai dengan +4dBm
- Operasi daya rendah 1,8V – 3,6V I/O.
- Kontrol PIO.
- Antarmuka UART dengan *baudrate* yang dapat diprogram.

Bluetooth HC-05 memiliki perintah set dalam melakukan perubahan *baudrate*, nama *Bluetooth*, perubahan pasword dan yang lainnya dengan memanfaatkan komunikasi serial. Konfigurasi dilakukan pada PC dengan menggunakan *hyper terminal* dan *Bluetooth* yang sudah terkoneksi dengan PC yang telah melalui rs232. Berikut adalah perintah set utama yang digunakan antara lain:

1. *Command* “AT”

Command AT digunakan untuk melakukan tes *Bluetooth*. Untuk mengetahui jika *Bluetooth* dapat berfungsi atau tidak, ketika command “AT” dikirimkan 30 maka akan mendapatkan respon balik, atau *Bluetooth* akan mengirimkan command “OK” melalui jalur TX *Bluetooth*.

2. *Command* “AT+BAUD”

Untuk melakukan perubahan baud rate yang digunakan dengan mengirimkan “AT+BAUD”. Sebagai contoh “AT+BAUD1”, “1” setelah baud mengartikan baud rate yang digunakan. Baud rate yang disediakan oleh *Bluetooth* yaitu:

- a. (1200)

- b. (2400)
- c. (4800)
- d. (9600)
- e. (19200)
- f. (38400)
- g. (57600)
- h. (115200)

Respon yang akan diterima ketika proses penggantian baud rate selesai yaitu *Bluetooth* akan mengirimkan "OK" melalui jalur TX *Bluetooth*.

3. *Command* "AT+NAME (device name)"

Command "AT+NAME (device name)" digunakan untuk melakukan perubahan nama device *bluetooth*, sebagai contoh "AT+NAMETEST" yang berarti bahwa *Bluetooth* tersebut bernama test ketika di deteksi oleh perangkat lain. Ketika command telah berhasil dikirimkan maka respon balik yang akan di dapatkan adalah "OK set name" namun, jika tidak berhasil atau gagal maka respon yang diterima adalah "FAIL".

4. *Command* "AT+PINxxxx"

Command "AT+PINxxxx" digunakan untuk melakukan perubahan pin. Pin *Bluetooth* akan muncul ketika hardware lain akan melakukan koneksi ke *Bluetooth* HC05. Proses setting hanya bisa dilakukan pada saat *Bluetooth* module dalam kondisi tidak terhubung/paired dengan device lain, hal ini bisa dilihat dari nyala led pada modul. Jika led menyala berkedip berarti *bluetooth* module ini tidak terkoneksi dengan device *bluetooth* lain.

5. *Command* “AT+VERSION”

Command “AT+VERSION” digunakan untuk mengetahui versi *Bluetooth*. Ketika *command* dikirimkan maka *bluetooth* akan mengirim respon balik yaitu dengan mengirimkan versi *bluetoothnya*, jika *bluetooth* yang digunakan adalah HC06 maka respon yang dikirimkan adalah “Linvor1.5”.

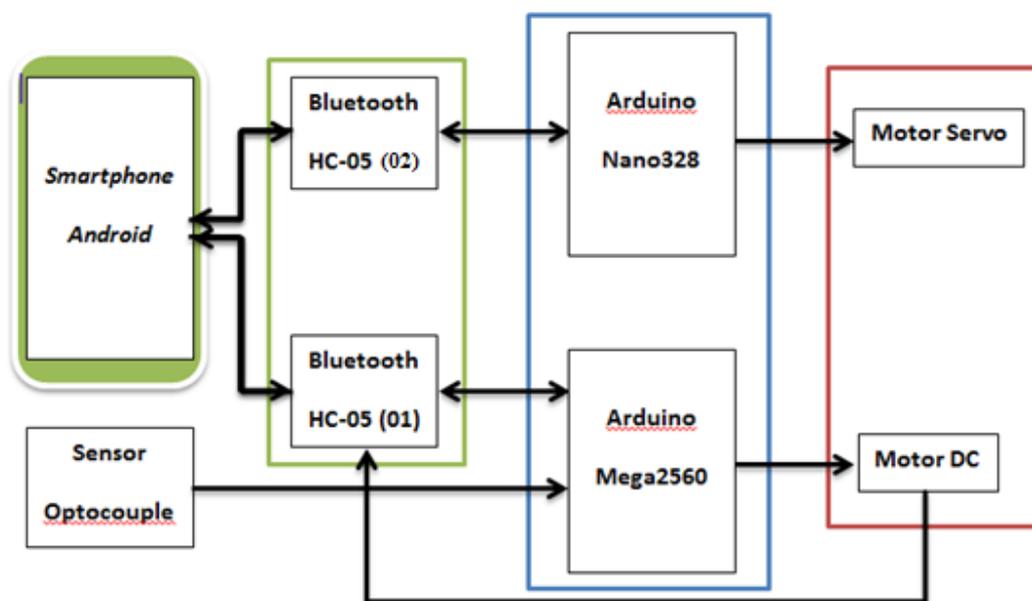
2.1.6 **Kerangka Berpikir**

2.1.6.1 *Blok Diagram*

Blok diagram merupakan suatu tahapan dari proses dalam pembuatan prototipe. Blok diagram prototipe digunakan untuk menentukan komponen penyusun dari suatu alat yang akan dibuat, sehingga hasil akhirnya sesuai dengan yang diinginkan. Mempermudah dalam proses pembuatan prototipe, karena setiap blok memiliki fungsi dan spesifikasi tertentu.

Setiap blok dihubungkan sehingga terbentuk sistem alat yang diharapkan. Prototipe sistem kontrol dan monitoring kecepatan mobil RC (*remote control*) berbasis *smartphone android* dibangun dari beberapa blok masukan, blok pengolahan data dan blok keluaran. Agar mendapatkan hasil pembuatan yang diinginkan dibagi menjadi dua macam perancangan dari segi perangkat keras (*Hardware*) dan perancangan dari segi perangkat lunak (*software*). Secara garis besar perancangan perangkat keras meliputi *board arduino mega 2560* yang berfungsi sebagai pengolah data masukan selanjutnya digunakan untuk menentukan proses pengendalian. Sistem modul sensor optocoupler berfungsi sebagai masukan *arduino*.

Perancangan perangkat lunak pada proses pembuatan prototipe ini meliputi pemrograman pengendali pada arduino dari mengolah data masukan dari sensor, menjadi suatu aksi yang berupa berputarnya motor DC dan langsung memberikan notifikasi ke aplikasi android. Perancangan pemrograman untuk mengatur kecepatan putaran motor. Berikut ini secara garis besar rancangan blok diagram sistem protoipe peringatan dan evakuasi bahaya kebakaran gedung berbasis arduino mega 2560.



Gambar 2.21. Diagram Blok Sistem Kontrol Dan Monitoring Putaran Motor Mobil RC (Remote Control) Berbasis Smartphone Android

Sistem kontrol dan monitoring mobil RC berbasis *smartphone* android, terdiri dari masukan sistem berupa sensor optocoupler. Unit proses berupa sebuah mikrokontroller Arduino Mega2560 dan Arduino Nano328 serta keluaran berupa motor DC dan motor servo dan Bluetooth HC-05 sebagai koneksi yang saling terintegrasi seperti pada **gambar 2.21**.

2.1.6.2 *Flowchart*

Flowchart adalah representasi grafis dan langkah-langkah yang harus diikuti dalam menyelesaikan suatu permasalahan yang terdiri dari sekumpulan simbol, dimana masing-masing simbol mempresentasikan kegiatan tertentu. Flowchart menolong analisis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif dalam pengoperasian. Flowchart diawali dengan penerimaan input dan diakhiri dengan penampilan output. Flowchart adalah suatu gambaran yang menjelaskan urutan: Pembacaan data, Pemrosesan data, Pengambilan keputusan terhadap data, dan penyajian hasil pemrosesan data.

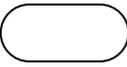
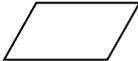
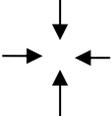
Flowchart terdapat lima jenis yaitu:

1. Pertama flowchart sistem yang menunjukkan alur kerja atau apa yang sedang dikerjakan di dalam sistem secara keseluruhan, dan menjelaskan urutan prosedur-prosedur yang ada dalam sistem.
2. Kedua flowchart dokumen digunakan untuk menelusuri alur dari data yang ditulis melalui sistem.
3. Ketiga flowchart skematik digunakan untuk menggambarkan suatu sistem atau prosedur.
4. Keempat flowchart program yang dihasilkan dari flochart sistem, dengan keterangan yang lebih rinci tentang bagaimana setiap langkah program atau presedur sesungguhnya dilaksanakan.

5. Kelima flowchart proses merupakan penggambaran rekayasa industrial yang memecahkan dan menganalisis langkah-langkah selanjutnya dalam suatu prosedur atau sistem.

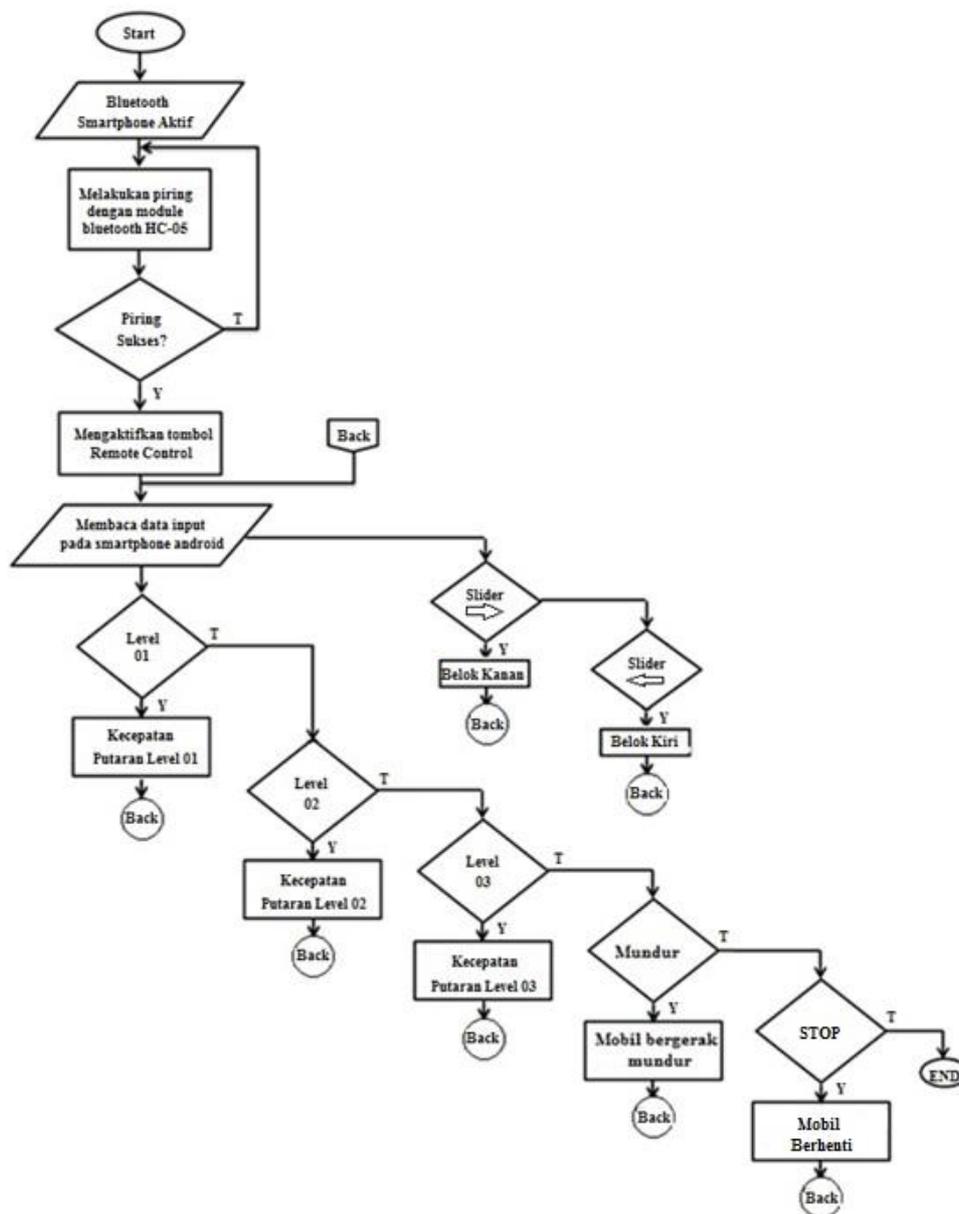
Dalam flowchart terdapat simbol-simbol standart yang dikeluarkan oleh ANSI dan ISO. Berikut simbol-simbol yang digunakan untuk menyusun *flowchart*.

Tabel 2.2. Simbol-Simbol Flowchart

Simbol	Nama	Keterangan
	<i>Terminator</i>	Menandai awal dan akhir suatu <i>flowchart</i>
	<i>Input-output</i>	Mempresentasikan sebuah fungsi <i>input</i> atau <i>output</i>
	Pecabangan / keputusan	Melakukan suatu percabangan, pemeriksa terhadap suatu kondosi
	Proses/penugasan	Kegiatan pemrosesan input
	Preparasi/persiapan	Pemberian harga awal
	Arah aliran	Menghubungkan setiap langkah dalam <i>flowchart</i> , serta menunjukkan kemana arah aliran.
	Konektor <i>on page</i>	Digunakan untuk menghubungkan satu langkah dengan langkah lain dalam <i>flowchart</i> pada satu halaman.

	Konektor <i>off page</i>	Digunakan untuk menghubungkan satu langkah dengan langkah lain dalam flowchart pada halaman yang berbeda.
---	--------------------------	---

1. Flowchart sistem kontrol dan monitoring monil RC (*Remote Control*) berbasis *Smartphone* Android



Gambar 2.22. Flowchart mobil RC

2.1.7 Kontrol dan Monitoring

2.1.7.1 Kontrol

Menurut Kamus Besar Bahasa Indonesia (KBBI) Kontrol adalah pengawasan, pemeriksaan, pengendalian.

Pengawasan (*Controlling*) adalah proses pengamatan, penentuan standar yang akan dicapai, menilai pelaksanaan, dan jika perlu mengambil tindakan korektif sehingga pelaksanaan dapat berjalan sesuai dengan rencana yang telah ditetapkan sebelumnya. Tujuan utama dari kegiatan pengawasan adalah membuat kegiatan-kegiatan manajemen dinamis dan berhasil secara efektif dan efisien. Sesuai dengan perannya dalam sebuah organisasi, pengawasan memiliki beberapa fungsi.

Fungsi pengawasan yang dimaksud antara lain sebagai berikut:

- Mencegah penyimpangan-penyimpangan.
- Memperbaiki kesalahan atau kelemahan, dan menindak penyalahgunaan serta penyelewengan.

Secara umum, pengawasan dapat dilakukan dengan beberapa langkah, antara lain sebagai berikut:

- Menetapkan standar untuk pengawasan.
- Meneliti, memeriksa, dan menilai hasil yang dapat dicapai.
- Membandingkan hasil dengan standar.
- Memperbaiki penyimpangan dengan tindakan koreksi.

Pengawasan dapat dilakukan dengan beberapa metode, antara lain sebagai berikut:

- **Komparatif:** Komparatif yaitu sistem pengawasan yang dilakukan dengan cara membandingkan hasil dengan rencana.
- **Inspektif:** Inspektif artinya sistem pemeriksaan setempa berguna untuk mengetahui secara langsung keadaan sebenarnya mengenai pelaksanaan suatu pekerjaan.
- **Verifikatif:** Verifikatif artinya sistem pengawasan secara pemeriksaan, biasanya menyangkut bidang keuangan dan material.
- **Investigatif:** Investigatif artinya pengawasan.

2.1.7.2 *Monitoring*

Monitoring didefinisikan sebagai siklus kegiatan yang mencakup pengumpulan, peninjauan ulang, pelaporan, dan tindakan atas informasi suatu proses yang sedang diimplementasikan (Mercy, 2005). Umumnya, *monitoring* digunakan dalam *checking* antara kinerja dan target yang telah ditentukan.

Monitoring ditinjau dari hubungan terhadap manajemen kinerja adalah proses terintegrasi untuk memastikan bahwa proses berjalan sesuai rencana (*on the track*). *Monitoring* dapat memberikan informasi keberlangsungan proses untuk menetapkan langkah menuju ke arah perbaikan yang berkesinambungan. Pada pelaksanaannya, *monitoring* dilakukan ketika suatu proses sedang berlangsung. Level kajian sistem *monitoring* mengacu pada kegiatan per kegiatan dalam suatu bagian (Wrihatnolo, 2008), misalnya kegiatan pemesanan barang pada *supplier* oleh bagian *purchasing*. Indikator yang menjadi acuan *monitoring* adalah *output* per proses/per kegiatan.

Umumnya, pelaku *monitoring* merupakan pihak-pihak yang berkepentingan dalam proses, baik pelaku proses (*self monitoring*) maupun atasan/*supervisor* pekerja. Berbagai macam alat bantu yang digunakan dalam pelaksanaan sistem *monitoring*, baik observasi/*interview* secara langsung, dokumentasi maupun aplikasi visual (Chong, 2005).

Pada dasarnya, *monitoring* memiliki dua fungsi dasar yang berhubungan, yaitu *compliance monitoring* dan *performance monitoring* (Mercy, 2005). *Compliance monitoring* berfungsi untuk memastikan proses sesuai dengan harapan/rencana. Sedangkan, *performance monitoring* berfungsi untuk mengetahui perkembangan dalam pencapaian target yang diharapkan.

Umumnya, *output monitoring* berupa *progress report* proses. *Output* tersebut diukur secara deskriptif maupun non-deskriptif. *Output* monitoring bertujuan untuk mengetahui kesesuaian proses telah berjalan. *Output* monitoring berguna pada perbaikan mekanisme proses / kegiatan di mana *monitoring* dilakukan.

BAB III

METODOLOGI PENELITIAN

3.1 Tempat dan Waktu Penelitian

Penelitian ini dilaksanakan lebih banyak di rumah peneliti, Jl. Lodaya Kp. Baru, Purwakarta dan sesekali dilaksanakan di Jurusan Teknik Elektro Fakultas Teknik Universitas Negeri Jakarta. Waktu penelitian dilaksanakan pada semester Ganjil Tahun Akademik 2016-2017.

3.2 Alat dan Bahan Penelitian

Instrumen yang digunakan untuk membuat alat ini terdiri dari:

1. Sistem komputer yang digunakan di dalam penelitian ini adalah dengan spesifikasi sebagai berikut:
 - a. Intel(R) Celeron(R) CPU 1007U @1.50Hz 1.50GHz
 - b. Memory RAM 4.00 GB
 - c. Sistem operasi Microsoft Windows 8 Pro Ultimate 64 bit
2. *Smartphone* Android yang digunakan dalam menunjang penelitian dengan spesifikasi sebagai berikut:
 - a. Android O.S. Android Kitkat 4.4.2
 - b. RAM 1 GB
 - c. Dimensi Layar IPS LCD, 480 x 854 pixels, 4.5 inches (~218 ppi pixel density), Oleophobic coating

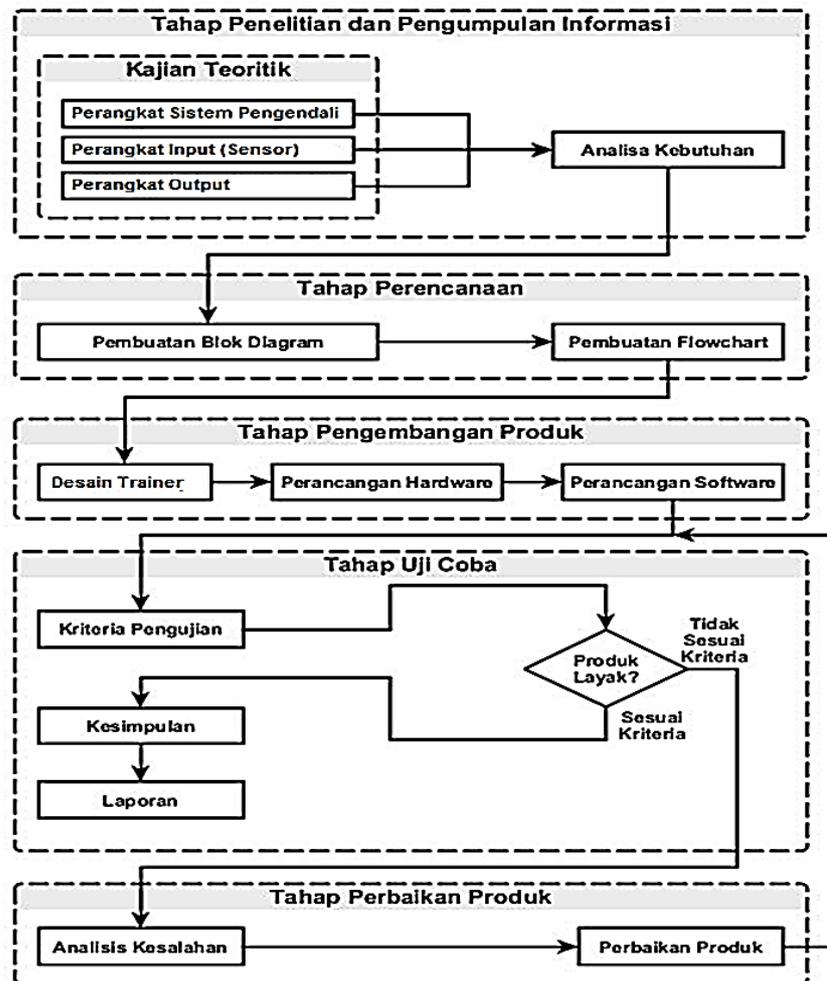
- d. Konektivitas HSDPA, 42.2 Mbps; HSUPA, 5.76 Mbps, Wi-Fi 802.11 b/g/n, Wi-Fi Direct, Wi-Fi hotspot, Bluetooth v4.0, microUSB v2.0, USB OTG
 - e. CPU Intel Atom Z2520 Dual-core 1.2 GHz
3. Perangkat lunak (*Software*) yang digunakan:
- a. Arduino IDE 1.6.8, digunakan untuk memprogram Arduino Mega2560 dan Arduino Nano328.
 - b. APP Inventor, aplikasi *online* ini digunakan untuk membuat aplikasi pada *smartphone* android
 - c. EAGLE Layout Editor 6.4.0, digunakan untuk membuat skematik dan *layout* rangkaian.
 - d. Microsoft Word 2010, digunakan untuk penulisan laporan
 - e. *Snipping tools*, digunakan untuk mengambil gambar pada layar laptop.
4. Perangkat keras yang digunakan
- a. Solder listrik.
 - b. Mini *electric drill* (bor tangan kecil)
 - c. Tang potong
 - d. Tang jepit
 - e. Charger battery Li-ion dan Li-po
 - f. Macam – macam obeng
5. Alat ukur yang digunakan
- a. Multimeter analog

- b. Multimeter digital
- c. Serial Monitor Arduino IDE 1.6.8
- d. Tachometer

3.3 Diagram Alir Penelitian

Metodologi penelitian adalah langkah-langkah yang dilakukan dalam penelitian ilmiah yang bertujuan untuk mendapatkan hasil sehingga tujuan dari penelitian tersebut dapat terpenuhi. Metodologi penelitian yang digunakan dalam menyelesaikan penelitian ini adalah menggunakan metodologi penelitian dan pengembangan (*Research and Development*). Borg and Gall menyatakan bahwa penelitian dan pengembangan (*Research and Development*), merupakan metode penelitian yang digunakan untuk mengembangkan atau memvalidasi produk-produk yang digunakan dalam pendidikan dan pembelajaran (M.D.Gall, 1989). berikut adalah lima rumusan langkah-langkahnya:

1. Penelitian dan pengumpulan informasi (*Research and Information Collecting*)
2. Perencanaan (*Planning*)
3. Pengembangan bentuk awal produk (*Develop Preliminary Form of Product*)
4. Uji lapangan (*Prototype Test*)
5. Perbaikan produk (*Product Revision*)



Gambar 3.1. Diagram Alir Penelitian

3.3.1 Tahap Penelitian dan Pengumpulan Informasi

Pada tahap *research and information* merupakan analisis kebutuhan dari suatu sistem dimana kebutuhan yang pada umumnya yaitu *input* dan *output* yang digunakan dalam sistem tersebut.

Berdasarkan kajian pustaka dan hasil survey lapangan untuk membuat sistem kontrol dan monitoring putaran motor mobil RC (*Remote Control*) berbasis android. Perangkat input yang digunakan pada sistem ini adalah sensor Ouptocopler dimana sensor ini dapat mendeteksi putaran motor.

Menganalisa perangkat proses yang digunakan pada sistem, peneliti menggunakan arduino mega 2560 dan arduino nano328 sebagai pengendali dan pemroses data inputan sensor dan *Bluetooth* HC-05 sebagai komunikasi serial antara Arduino mega2560 dan arduino nano328 dengan *smartphone*. Sedangkan perangkat *output* yang digunakan berupa motor DC dan motor servo.

3.3.2 Tahap Perancangan

Perancangan sistem pada penelitian sistem kontrol dan monitoring putaran motor mobil RC (*Remote Control*) berbasis android adalah sebuah sistem yang dapat digunakan untuk mengukur putaran motor ketika mobil sedang melaju.

3.3.3 Tahap Pengembangan Produk

Pada tahap pengembangan bentuk awal produk, peneliti melakukan tiga tahapan utama yaitu perancangan desain alat, perancangan perangkat keras dan perancangan perangkat lunak.

3.4 Tahap Uji Coba

Pada tahap pengujian peneliti melakukan uji coba, uji coba pertama peneliti menguji mekanik mobil RC, baik itu motor DC maupun motor servo. Pada tahap kedua dilakukan pengujian pada rangkaian elektronik, apakah sudah sesuai besaran tegangan dan arus yang dibutuhkan serta terhubung dengan baik antar semua komponen dan modul. Pada tahap ketiga dilakukan pengujian aplikasi *remote control* yang sudah dibuat, apakah dapat mengendalikan mobil RC sesuai fungsinya dengan menguji komunikasi serial antara *smartphone* android dengan mobil RC. Pada tahap terakhir dilakukan pengujian sensor optocoupler yang digunakan untuk membaca putaran motor.

3.4.1 Tahap Perbaikan Produk

Tahap perbaikan produk dilakukan ketika hasil uji coba tidak sesuai dengan perencanaan yang bertujuan untuk mencari kesalahan dan kekurangan pada sistem agar dapat diperbaiki sehingga mendapatkan hasil yang sesuai dengan perencanaan dan kriteria yang telah ditentukan.

3.5 Perancangan Alat

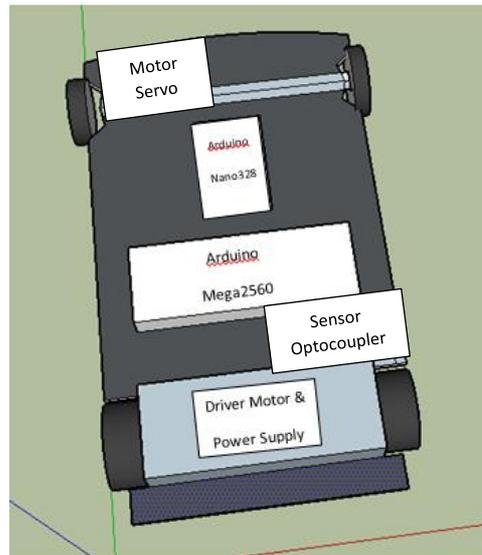
Perancangan alat merupakan suatu rencana yang komprehensif dan memiliki tujuan yang terarah dalam melakukan penelitian untuk menghasilkan karya sesuai dengan yang diinginkan. Hal yang dilakukan dalam merealisasikan sistem control dan monitoring putaran motor mobil RC terdiri dari beberapa tahapan.

3.5.1 Perancangan Perangkat Keras

Perancangan perangkat keras (*hardware*) mobil RC bertujuan agar mobil RC memiliki ketepatan dalam menjalankan fungsinya yaitu dapat mengontrol dan memonitoring putaran motor. Terdiri dari desain mekanik mobil RC, perancangan rangkaian elektronik dan perancangan integrasi komponen.

3.5.1.1 Desain Perakitan Mekanik

Dalam mendesain sistem control dan monitoring putaran motor, teknik yang digunakan adalah mendesain peletakan part-part komponen hingga menjadi bentuk utuh. Adapun dalam mendesain bagian mekanik mobil RC, bahan yang digunakan adalah *chasis* mobil RC yang bisa didapatkan di toko mainan. Adapun desain mekanik mobil RC dapat dilihat pada **Gambar 3.2**.



Gambar 3.2. Desain Perakitan Mekanik Mobil RC

Dalam perakitan sistem control dan monitoring putaran motor Mobil RC dibagi kedalam beberapa bagian, antara lain:

1. Bagian *steering*

Bagian *steering* ini berfungsi sebagai penyearah laju mobil atau dalam arti kata lain sebagai setir mobil yang bertujuan untuk mengarahkan kemana laju mobil bergerak apakah akan melaju dengan arah lurus atau belok kanan ataupun belok kiri. Bagian *steering* ini digerakkan oleh satu buah motor servo. Sebagaimana ditampilkan pada **gambar 3.3** sebagai berikut:



Gambar 3.3. Steering Mobil RC

2. Bagian dasar mobil

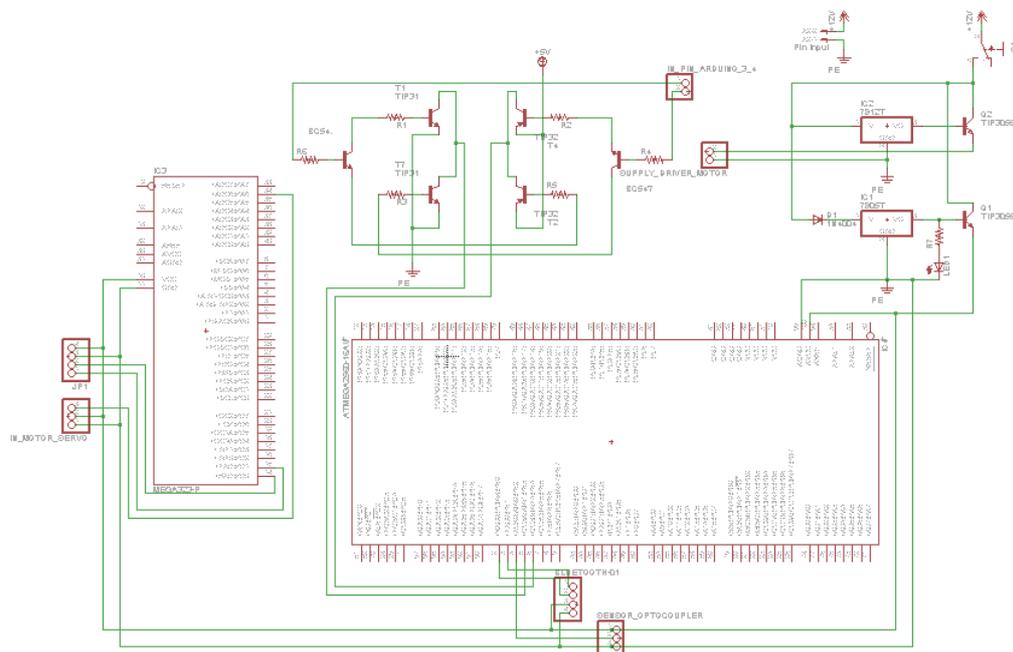
Bagian dasar mobil (Chasis) berfungsi sebagai tempat atau dudukan motor DC, motor Servo, Arduino Nano328, Arduino Mega2560, batre dan PCB rangkaian elektronika. Motor DC berfungsi sebagai aktuator roda mobil dalam bergerak.



Gambar 3.4 Bagian Dasar Mobil RC

3.5.1.2 Desain Elektronik

Pada suatu sistem hal yang menjadi inti adalah skema elektronik, karena berfungsi sebagai jalur-jalur penghubung antara semua komponen sistem, baik itu mekanik, sensor maupun aktuator. Dalam penelitian kali ini, peneliti menggunakan *software* EAGLE 6.4.0 profesional untuk membuat skema rangkaian yang dapat dilihat pada **Gambar 3.5**.



Gambar 3.5 Skematik Rangkaian Control Dan Monitoring Putaran Motor Mobil RC

3.5.2 Perancangan Perangkat Lunak

Perancangan perangkat lunak merupakan sebuah perancangan program yang dibuat untuk mendukung sistem control dan monitoring putaran motor mobil RC agar dapat bekerja dengan baik. Peneliti merancang dua buah perangkat lunak agar mobil RC dapat bekerja sesuai dengan tujuannya, kedua perangkat lunak tersebut dapat saling berkomunikasi. Berikut adalah penjelasan perangkat lunak tersebut.

3.5.2.1 Perancangan Program Arduino

Perangkat lunak pertama adalah program arduino yang dirancang menggunakan perangkat lunak lain bernama Arduino IDE versi 1.6.8. Perancangan program dibuat berdasarkan prinsip kerja sistem control dan monitoring putaran motor mobil RC berbasis *smartphone* android. Berikut

merupakan detail konfigurasi pin-pin *input* pada mikrokontroller arduino Mega2560 dan arduino Nano328 dengan perangkat *input* yang digunakan pada mobil RC, dapat dilihat pada **Tabel 3.1**, **Tabel 3.2**, **Tabel 3.3** dan **Tabel 3.4**.

Tabel 3.1 Penggunaan Pin *Input* Pada Arduino Mega2560

No	Nama Perangkat <i>Input</i>	Pin Perangkat <i>Input</i>	Pin Arduino Mega2560
1	Bluetooth HC-05 (01)	Tx	<i>Communication 1</i>
2	Sensor Optocoupler	-	Pwm 2

Adapun untuk penggunaan pin *output* pada Arduino Mega2560 dengan perangkat keluaran dapat dilihat pada **Table 3.2**.

Tabel 3.2 Penggunaan Pin *Output* Pada Arduino Mega 2560

No	Nama Perangkat <i>Output</i>	Pin Perangkat <i>Output</i>	Pin Arduino Mega2560
1	Bluetooth HC-05 (01)	Rx	<i>Communication 0</i>
2	Driver Motor	Clock	Pwm 3
		Clock	Pwm 4

Adapun untuk penggunaan pin *input* pada Arduino Nano328 dengan perangkat masukan dapat dilihat pada **Tabel 3.3**.

Tabel 3.3. Penggunaan Pin *Input* Pada Arduino Nano328

No	Nama Perangkat <i>Input</i>	Pin Perangkat <i>Input</i>	Pin Arduino Mega2560
1	Bluetooth HC-05 (02)	Tx	<i>D1</i>

Adapun untuk penggunaan pin *output* pada Arduino Nano328 dengan perangkat keluaran dapat dilihat pada **Table 3.4**.

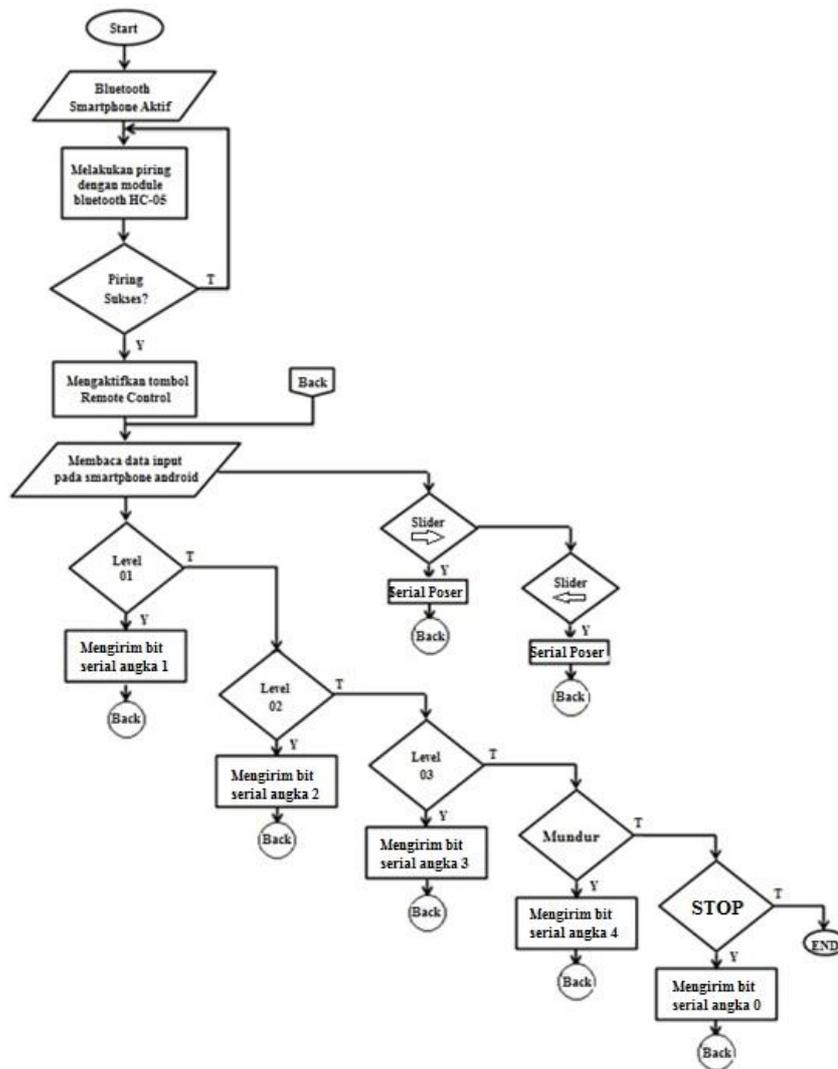
Tabel 3.4. Penggunaan Pin *Output* Pada Arduino Nano328

No	Nama Perangkat <i>Output</i>	Pin Perangkat <i>Output</i>	Pin Arduino Mega2560
1	Bluetooth HC-05 (02)	Rx	D0
2	Motor Servo	Clock	D6

3.5.2.2 Perancangan Aplikasi *Remote Control* Pada *Smartphone Android*

Dalam pembuatan aplikasi android peneliti menggunakan perangkat lunak MIT App Inventor 2 yang dioperasikan secara *online*. Pembuatan aplikasi android ini bertujuan sebagai *remote control* untuk mengendalikan mobil RC dengan mengontrol dan monitoring putaran motor.

Adapun cara kerja aplikasi *Remote Control* dapat dilihat pada Gambar dibawah ini:



Gambar 3.7. Flowchart Cara Kerja Aplikasi Remote Control

3.6 Teknik dan Prosedur Pengumpulan Data

Prosedur penelitian dan pengumpulan data sistem control dan monitoring putaran motor mobil RC berbasis Smartphone Android dilakukan melalui beberapa tahap:

1. Perancangan desain dan perakitan mobil RC

Peneliti merancang desain mekanik perakitan mobil RC dengan dibantu oleh aplikasi SketchUp 2014. Dalam membuat rancangan mekanik mobil RC agar dapat mengontrol dan memonitori putaran motor, Peneliti menggunakan *chasis* mobil RC yang dibeli di toko mainan dan perakitan *part-part* komponen yang disesuaikan dengan lebar dan panjangnya *chasis* mobil RC.

2. Pembuatan rangkaian elektronika robot.

Dalam membuat rangkaian elektronika, peneliti menggunakan *software* Eagle versi 6.4.0 yang umum digunakan dalam merancang dan membuat *layout* rangkain elektronika. Peneliti merancang dan membuat rangkaian yang mengintegrasikan semua modul dan komponen pendukung menjadi satu dalam papan pcb *single layer*. Adapun modul elektronika yang digunakan adalah modul *Bluetooth HC-05* dan modul sensor optocoupler.

3. Pembuatan aplikasi *remote control* robot.

Aplikasi *remote control* mobil RC dibuat menggunakan aplikasi *online* MIT App Inventor 2 yang hasilnya berbentuk file aplikasi android dengan extension .apk. Aplikasi *remote control* ini berfungsi sebagai pengontrol sekaligus dapat memonitori putaran motor mobil RC. Aplikasi *remote control* diatur dengan tampilan *landscape* pada *smartphone* android agar memudahkan

dalam penggunaannya, terdiri dari slider agar mobil dapat bergerak baik ke kanan maupun ke kiri dan kombinasi tombol kontrol motor DC, serta tombol untuk dapat mengkoneksikan Bluetooth dari smartphone dengan modul bluetooth yang di letakkan di bagian atas. Semua tombol berada dalam satu *screen* atau layar *smartphone* android agar memudahkan peneliti dalam menguji coba mobil RC.

4. Pembuatan program arduino

Peneliti menggunakan *software* arduino IDE versi 1.6.8 untuk memprogram mikrokontroler arduino Mega2560 dan arduino Nano328. Program arduino ini dirancang khusus untuk mengintegrasikan modul-modul dan rangkaian elektronika dengan aktuator motor DC dan motor servo agar dapat berfungsi sebagaimana fungsinya.

5. Uji coba aplikasi *remote control* dalam mengendalikan mobil RC

Uji coba dilakukan jika perangkat keras dan perangkat lunak telah teruji dan komunikasi dapat berjalan sesuai instruksi yang diberikan.

3.7 Teknik Analisis Data

Teknik analisis data adalah perbandingan antara hasil pengujian dengan kriteria yang ditetapkan oleh peneliti untuk menguji keberhasilan sebuah sistem. Pada kriteria pengujian yang diajukan peneliti terdapat beberapa pengujian, yaitu pengujian regulator tegangan, pengujian *driver* motor DC, pengujian motor servo, sensor optocoupler, pengujian koneksi *smartphone* dengan mobil RC, pengujian pembacaan data komunikasi serial *remote control*, pengujian kendali *remote*

control terhadap mobil RC, dan pengujian jarak konektivitas *Bluetooth* mobil RC dan *Bluetooth smartphone*.

3.7.1. Pengujian *Driver Motor*

Pengujian *driver* motor DC bertujuan untuk mengetahui apakah *driver* motor yang mengendalikan arah putaran motor dc *gearbox* dapat dikendalikan oleh arduino Mega2560. Pengujian ini dilakukan dengan memberikan inputan HIGH dan LOW pada pin arduino Mega2560 yang terhubung dengan *driver* motor DC *gearbox* yaitu pin 3 dan 4.

Tabel 3.5. Pengujian *Driver Motor* DC

No	Arduino Mega2560		Kriteria Pengujian	Arah Putaran Motor
	Pin Pwm 3	Pin Pwm 4		
1	HIGH	LOW	Arah putaran motor searah jarum jam	
2	LOW	HIGH	Arah putaran motor berlawanan arah jarum jam	

3.7.2. Pengujian Motor Servo

Pengujian *motor servo* bertujuan untuk mengetahui apakah *motor servo* dapat dikendalikan oleh arduino Nano328. Pengujian ini dilakukan dengan memberikan inputan nilai derajat pada smartphone android yang terhubung dengan *motor servo* yaitu pin D6.

Tabel 3.6. Pengujian Motor Servo

No	Jenis Motor	Rotasi Motor per-derajat	Hasil Pengujian	
			Foto	Derajat yang ditempuh
1	Motor Servo	30 ⁰		
		40 ⁰		

		50 ⁰		
		60 ⁰		
		70 ⁰		
		80 ⁰		
		90 ⁰		

3.7.3. Pengujian Pembacaan Sensor Optocoupler

Pengujian sensor optocoupler bertujuan untuk mengetahui apakah sensor optocoupler dapat membaca putaran motor dengan cara menguji pembacaan putaran motor per menit dan diukur dengan menggunakan alat ukur Tachometer. Pengujian ini dilakukan dengan memberikan nilai putaran terhadap motor DC.

Tabel 3.7. Pengujian Pembacaan Sensor Optocoupler

No	Kriteria pengujian	Nilai input putaran	Hasil	
			Foto pengukuran Tachometer	Pembacaan Sensor Pada Aplikasi
1	<i>Push button</i> Level 01 ditekan	300		
	<i>Push button</i> Level 02 ditekan	400		
	<i>Push button</i> Level 03 ditekan	500		
	<i>Push Button</i> Mundur ditekan	320		

Adapun pengujian tegangan pada sensor optocoupler ketika sensor mendeteksi (tertutup) adanya penghalang dan ketika sensor tidak mendeteksi (terbuka) dari penghalang.

Tabel 3.8. Pengukuran Tegangan Pada Sensor Optocoupler

No	Nama Sensor	Kriteria pengujian	Foto pengukuran avometer digital	keterangan
1	Sensor optocoupler	Tertutup		
2		Tidak tertutup		

3.7.4. Pengujian Konektivitas *Smartphone* Dengan Mobil RC

Pengujian konektivitas bertujuan untuk menguji parameter kesuksesan perangkat *Bluetooth smartphone* android apakah dapat terhubung dengan Mobil RC atau tidak. Pengujian ini dilakukan dengan melakukan *paring Bluetooth* antara *smartphone* android dengan modul *Bluetooth* pada Mobil RC.

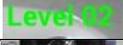
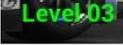
Tabel 3.9. Kriteria Pengujian Konektivitas *Smartphone* Dengan Mobil RC

No	Kondisi modul Bluetooth pada mobil RC	Tampilan aplikasi remote control	Indicator piring
1	ON		Berhasil
2	OFF		Tidak Berhasil

3.7.5. Prngujian pembacaan data komunikasi serial aplikasi *remote control* mobil RC

Pengujian pembacaan data komunikasi serial *remote control* bertujuan untuk mengetahui data bit serial angka yang dikirimkan oleh aplikasi *remote control* pada *smartphone* android. Pengujian ini dilakukan dengan menekan tombol-tombol pada *remote control*, apakah angka bit data serial yang dikirimkan sesuai dengan yang diinginkan pada *flowchart* cara kerja aplikasi *remote control*. Untuk mengetahui nilai angka bit data serial yang dikirimkan, dilakukan dengan melihat pada serial monitor software arduino IDE berapa saja bit data serial angka yang diterima oleh arduino Mega2560 dan arduino Nano328.

Tabel 3.10. Pengujian Pembacaan Data Komunikasi Serial *Remote Control*

No	Tombol yang ditekan	Bit serial yang dikirimkan	Hasil pembacaan diserial monitor	Keterangan
1		1		
2		2		
3		3		

4	Mundur	4		
5	STOP	0		

3.7.6. Pengujian kendali *remote control* terhadap mobil RC

Pengujian kendali *remote control* terhadap mobil bertujuan untuk mengetahui keberhasilan aplikasi *remote control* dalam mengendalikan mobil RC. Pengujian ini dilakukan dengan mengkoordinasikan setiap tombol yang ada pada aplikasi *remote control* dengan gerak mobil, apakah gerak mobil sesuai dengan fungsi setiap tombol yang ditekan pada aplikasi *remote control* tersebut atau tidak.

Tabel 3.11. Pengujian Kendali *Remote Control* Terhadap Mobil RC

No	Tombol Yang Ditekan	Aksi Yang Harus Dilakukan Oleh Mobil	Hasil Gerakan Mobil	Keterangan
1		Mobil bergerak maju pelan		
2		Mobil bergerak maju sedang		
3		Mobil bergerak maju cepat		
4		Mobil bergerak mundur		
5		Mobil berhenti		
6		Mobil berbelok ke kiri		
7		Mobil berbelok ke kanan		

3.7.7. Pengujian jarak konektivitas *Bluetooth HC-05* dengan *Bluetooth smartphone*

Pengujian jarak konektivitas *Bluetooth* mobil RC dengan *Bluetooth smartphone* bertujuan untuk mengetahui seberapa jauh jarak mobil dapat dikendalikan oleh aplikasi *remote control* pada *smartphone* dengan menggunakan komunikasi *Bluetooth*. Pengujian ini dilakukan dengan mengendalikan mobil dengan kondisi bergerak lurus maju sejauh mungkin, dengan mengukur jarak tempuh robot menggunakan meteran.

Tabel 3.12. Pengujian Konektivitas *Bluetooth HC-05* dengan *Bluetooth Smartphone*

No	Jarak Komunikasi Antara Mobil RC dengan Smartphone Android	Konektivitas <i>Bluetooth</i> robot dengan <i>Bluetooth smartphone</i>
1	1 Meter	
2	2 Meter	
3	3 Meter	
4	4 Meter	
5	5 Meter	
6	6 Meter	
7	7 Meter	
8	8 Meter	
9	9 Meter	
10	10 Meter	

BAB IV

HASIL PENELITIAN

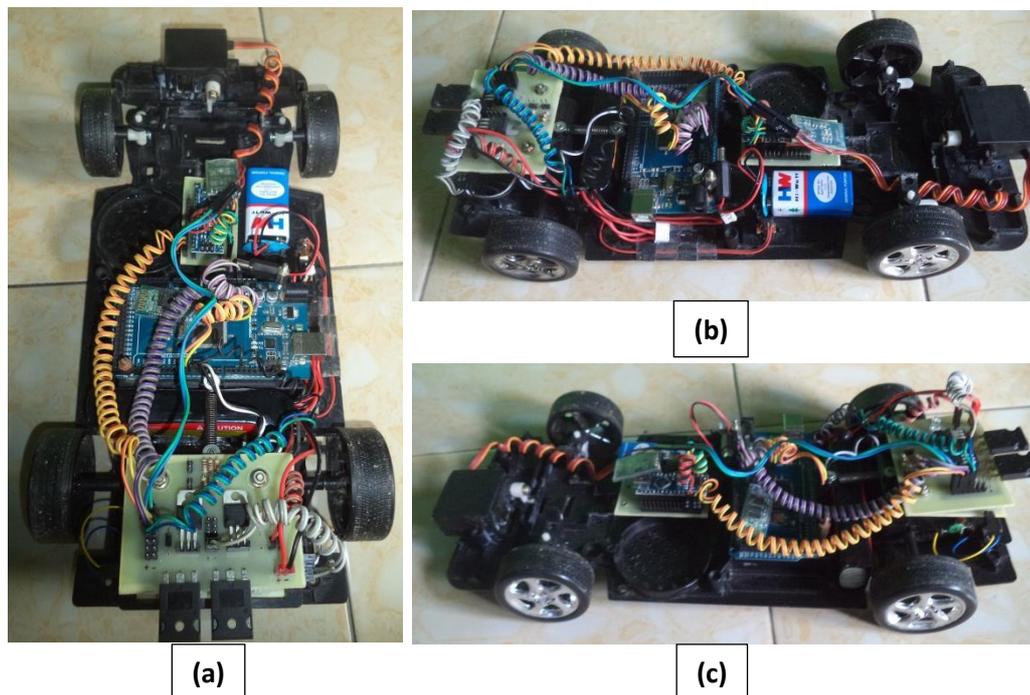
4.1. Deskripsi Hasil Penelitian

Hasil pengujian yang telah dilakukan terhadap indikator-indikator penelitian dari “Sistem Kontrol dan Monitoring Putaran Motor Mobil RC (*Remote Control*) Berbasis Smartphone Android” dapat dikategorikan sebagai berikut:

- 3.8 Hasil pembuatan dan pengujian perangkat keras (*hardware*) yaitu mobil RC.
- 3.9 Hasil pembuatan perangkat lunak (*software*) yaitu aplikasi *remote control* mobil RC.
- 3.10 Hasil pengujian *remote control* Mobil RC dalam mengendalikan gerak dan putaran motor mobil RC.

4.1.1. Hasil Pembuatan dan Pengujian Perangkat Keras (*Hardware*)

Berdasarkan perancangan alat pada BAB III, penelitian ini menghasilkan desain mekanik mobil RC dengan memberikan *steering* motor yang berfungsi sebagai penyearah mobil untuk bergerak ke kanan, ke kiri maupun bergerak lurus. dilengkapi dengan 2 buah roda yang membuat mobil RC dapat di control sekaligus dimonitori dengan *smartphone* android. Adapun bentuk mobil RC yang dibuat dapat dilihat pada **gambar 4.1**.



Gambar 4.1. (a) Bentuk Mobil RC (b) Mobil RC Tampak Samping Kanan (c) Bentuk Mobil RC Tampak Samping Kiri

Adapun spesifikasi dari control dan monitoring putaran motor Mobil RC adalah sebagai berikut:

Kontrol dan monitoring mobil RC memiliki 2 bagian kerja mekanik utama yaitu roda dan *steering*. Bagian roda mobil menggunakan motor DC yang berfungsi agar mobil dapat bergerak maju, sedangkan bagian *steering* menggunakan motor servo yang berfungsi sebagai penyearah mobil baik ke arah kanan, ke arah kiri maupun bergerak lurus. Untuk sumber tegangan mobil RC, digunakan *battery* Li-po dan menggunakan *battery* ABC kotak yang disusun secara seri yang memiliki tegangan sebesar 11.1 Volt dan arus sebesar 2200mAh. Juga terdapat sensor optocoupler yang dipasang di bagian belakang yang berfungsi sebagai pembaca putaran (RPM). Sebagai otak mobil RC digunakan 2 buah mikrokontroler yaitu mikrokontroler arduino Mega2560 dan arduino

Nano328 yang masing-masingnya berfungsi sebagai berikut, mikrokontroler arduino mega 2560 digunakan untuk menggerakkan motor DC dan sensor optocoupler sedangkan arduino Nano328 digunakan untuk menggerakkan motor servo. Control dan monitoring mobil RC juga memiliki switch power yang berfungsi untuk menghidupkan dan mematikan mobil RC.

Adapun hasil pengujian perangkat keras (*hardware*) yang dilakukan adalah sebagai berikut:

4.1.1.1. Hasil Pengujian Driver Motor

Pengujian *driver* motor DC bertujuan untuk mengetahui apakah *driver* motor yang mengendalikan arah putaran motor dc *gearbox* dapat dikendalikan oleh arduino Mega2560. Pengujian ini dilakukan dengan memberikan inputan HIGH dan LOW pada pin arduino Mega2560 yang terhubung dengan *driver* motor DC *gearbox* yaitu pin 3 dan 4. Cara pengujian *driver* motor DC dapat dilihat pada **gambar 4.2**. Adapun Hasil pengujian *driver* motor DC dapat dilihat pada **tabel 4.1**.



Gambar 4.2. Cara Pengujian Driver Motor DC

Tabel 4.1. Hasil Pengujian Driver Motor DC

No	Arduino Mega2560		Kriteria Pengujian	Arah Putaran Motor
	Pin Pwm 3	Pin Pwm 4		
1	HIGH	LOW	Arah putaran motor searah jarum jam	Searah dengan jarum jam
2	LOW	HIGH	Arah putaran motor berlawanan arah jarum jam	Berlawanan dengan arah jarum jam

Berdasarkan hasil pengujian *driver* motor DC pada **tabel 4.1** didapatkan bahwa arah putaran motor DC telah sesuai dengan kriteria pengujian. Hal ini menunjukkan bahwa *driver* motor yang mengendalikan arah putaran motor DC dapat dikendalikan oleh arduino Mega2560 dan pengujian dinyatakan berhasil.

4.1.1.2. Hasil Pengujian Motor Servo

Pengujian *motor servo* bertujuan untuk mengetahui apakah *motor servo* dapat dikendalikan oleh arduino Nano328. Pengujian ini dilakukan dengan memberikan inputan nilai derajat pada *smartphone* android yang terhubung dengan *motor servo* yaitu pin D6.

Tabel 4.2. Hasil Pengujian Motor Servo

No	Jenis Motor	Input Rotasi Motor Perderajat Dari Smartphone	Hasil Pengujian	
			Foto	Derajat yang ditempuh
1	Motor Servo			30 ⁰

				40 ⁰
				50 ⁰
				60 ⁰
				70 ⁰
				80 ⁰
				90 ⁰

Berdasarkan hasil pengujian motor servo dengan input derajat dari smartphone android pada table 4.2 didapatkan bahwa motor servo bekerja dengan baik sesuai kriteria pengujian.

4.1.1.3. Hasil Pengujian Sensor Optocoupler

Pengujian sensor optocoupler bertujuan untuk mengetahui apakah sensor optocoupler dapat membaca putaran motor dengan cara menguji pembacaan putaran motor per menit dan diukur dengan menggunakan alat ukur Tachometer. Pengujian ini dilakukan dengan memberikan nilai putaran terhadap motor DC.

Tabel 4.3. Hasil Pengujian Sensor Optocoupler

No	Kriteria Pengujian	Nilai Input Putaran	Hasil	
			Pembacaan Pada Tachometer (RPM)	Pembacaan Sensor Pada Aplikasi (RPM)
1	<i>Push button</i> Level 01 ditekan	300		
			294.9	294
	<i>Push button</i> Level 02 ditekan	500		
			492.8	491
	<i>Push button</i> Level 03 ditekan	600		
			583.2	584

	<i>Push Button Mundur</i>	320		
			320	316

Berdasarkan hasil pengujian pembacaan banyaknya putaran motor per menit (RPM) dengan menggunakan alat ukur Tachometer didapatkan bahwa sensor optocoupler dapat membaca putaran motor dengan baik sesuai kriteria pengujian.

Tabel 4.4. Pengukuran Tegangan Pada Sensor Optocoupler

No	Nama Sensor	Kriteria pengujian	Foto pengukuran avometer digital	Keterangan
1	Sensor optocoupler	Tertutup		Berhasil, tegangan 4.09 Volt ketika sensor mendeteksi
2		Tidak tertutup		Berhasil, tegangan 0.20 Volt ketika sensor tidak mendeteksi

Berdasarkan hasil pengujian pembacaan tegangan ketika di tutup dan ketika tidak ditutup berbeda, ketika ditutup didapatkan 4.09 volt, sedangkan ketika tidak tertutup didapatkan 0.20 volt maka dengan ini dinyatakan pengujian berhasil.

4.1.2. Hasil pembuatan dan pengujian perangkat lunak (*software*)

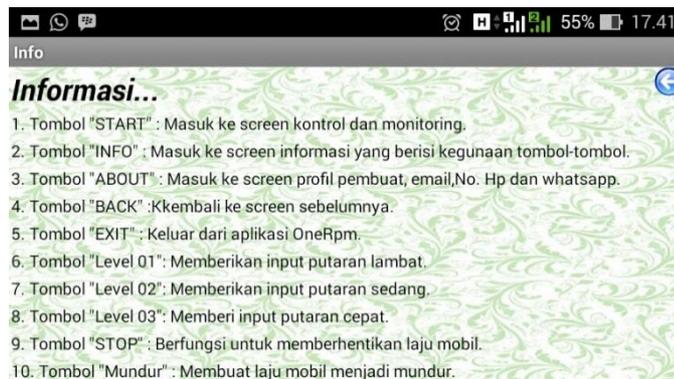
Berdasarkan perancangan alat pada BAB III, penelitian ini menghasilkan sebuah aplikasi *smartphone* android yang berfungsi sebagai *remote control* yang dapat mengendalikan dan memonitori mobil RC. Adapun tampilan aplikasi *remote control* robot yang dibuat dapat dilihat pada **gambar 4.3**, **gambar 4.4**, **gambar 4.5**, **gambar 4.6** dan **gambar 4.7**.



Gambar 4.3. Tampilan Awal Aplikasi Remote Control



Gambar 4.4. Tampilan Screen Kontrol Dan Monitoring



Gambar 4.5. Tampilan Screen Informasi Aplikasi



Gambar 4.6. Tampilan Screen Profil Pembuat Aplikasi



Gambar 4.7. Tampilan Screen Jika Ingin Keluar Aplikasi

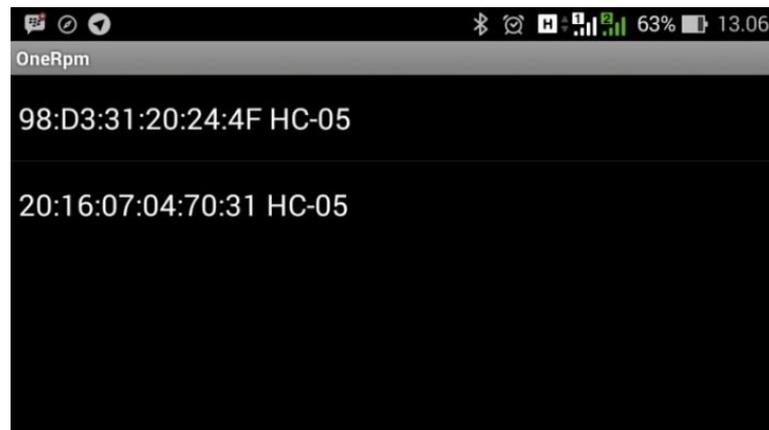
Aplikasi *remote control* yang dibuat memiliki 5 *screen* (halaman tampilan aplikasi) yaitu *screen* pertama berupa tampilan awal aplikasi, *screen* kedua berupa tampilan navigasi kontrol dan monitoring mobil RC, *screen* ketiga berupa informasi aplikasi, *screen* keempat berupa tampilan profil pembuat aplikasi dan *screen* kelima berupa tampilan jika ingin keluar dari aplikasi. Pada

screen tampilan awal aplikasi disajikan beberapa tombol yang masing-masing memiliki fungsi yang berbeda antara lain tombol START, INFO, ABOUT dan EXIT kemudian pada *screen* kedua berupa *screen* navigasi dan disajikan beberapa tombol yang masing-masing memiliki fungsi berbeda antara yaitu: tombol konektivitas *Bluetooth*, rangkaian tombol kendali mobil RC yang terdiri dari kontrol pengaturan kecepatan putaran motor dan *slider* arah. Kemudian *screen* ketiga yakni *screen* informasi aplikasi yang berisi informasi kegunaan aplikasi *remote control* serta satu tombol *back* yang berfungsi mengembalikan ke *screen* tampilan awal jika ditekan, dan selanjutnya adalah *screen* keempat yang berisi identitas pembuat aplikasi serta satu tombol *back* yang berfungsi mengembalikan ke *screen* tampilan awal jika ditekan dan yang terakhir yakni *screen exit* yang berfungsi untuk keluar dari aplikasi jika tombol ditekan.

Adapun hasil pengujian perangkat lunak (*software*) yang dilakukan adalah sebagai berikut:

4.1.2.1. Hasil Pengujian Konektivitas *Smartphone* dengan Mobil RC

Pengujian konektivitas bertujuan untuk menguji parameter kesuksesan perangkat *Bluetooth smartphone* android apakah dapat terhubung dengan *Bluetooth* mobil RC atau tidak. Pengujian ini dilakukan dengan melakukan *paring Bluetooth* antara *smartphone* android dengan modul *Bluetooth* pada mobil. Cara dan hasil pengujian konektivitas *smartphone* dengan mobil RC dapat dilihat pada **gambar 4.8** dan **tabel 4.5**.



Gambar 4.8. Tampilan pencarian konektivitas Bluetooth

Tabel 4.5. Pengujian Konektivitas *Smartphone* Dengan Mobil RC

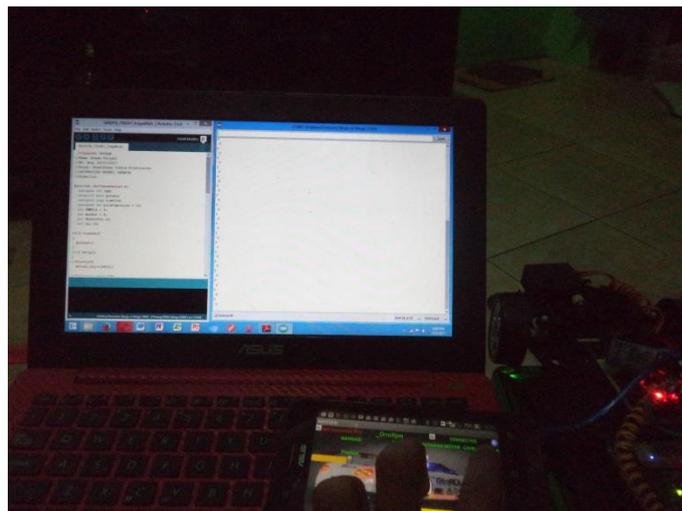
No	Kondisi modul Bluetooth pada mobil RC	Tampilan aplikasi remote control	Indicator piring
1	ON		Berhasil
2	OFF		Tidak Berhasil

Berdasarkan hasil pengujian konektivitas *smartphone* dengan mobil RC didapatkan bahwa aplikasi *remote control* berhasil melakukan *pairing Bluetooth* jika konektivitas *Bluetooth* antara *smartphone* dan *modul Bluetooth* mobil RC

keduanya aktif atau dalam kondisi ON yang dibuktikan dengan indikator status koneksi yang terhubung pada layar *smartphone*.

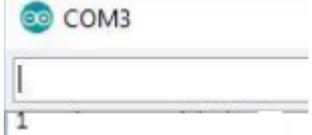
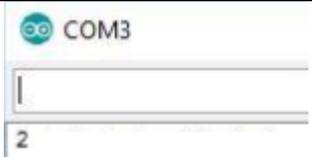
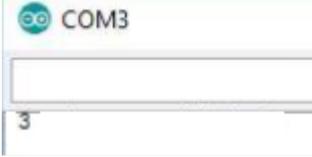
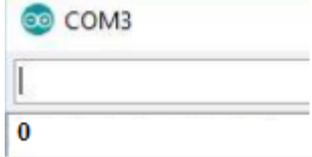
4.1.2.2. Hasil Pengujian Pembacaan Data Komunikasi Serial Aplikasi Remote Control

Pengujian pembacaan data komunikasi serial *remote control* bertujuan untuk mengetahui data bit serial angka yang dikirimkan oleh aplikasi *remote control* pada *smartphone* android. Pengujian ini dilakukan dengan menekan tombol-tombol pada *remote control*, apakah angka bit data serial yang dikirimkan sesuai dengan yang diinginkan pada *flowchart* cara kerja aplikasi *remote control* pada BAB III. Untuk mengetahui nilai angka bit data serial yang dikirimkan, dilakukan dengan melihat pada serial monitor software arduino IDE berapa saja bit data serial angka yang diterima oleh arduino Mega2560. Adapun cara dan hasil pengujian pembacaan data komunikasi serial *remote control* dapat dilihat pada **gambar 4.9** dan **tabel 4.6**.



Gambar 4.9. Pengujian Pembacaan Data Bit Serial Remote Control

Tabel 4.6. Hasil Pengujian Pembacaan Bit Serial Data *Remote Control*

No	Tombol yang ditekan	Bit serial yang dikirimkan	Hasil pembacaan diserial monitor	Keterangan
1		1		Berhasil
2		2		Berhasil
3		3		Berhasil
4		4		Berhasil
5		0		Berhasil

Berdasarkan hasil pengujian pembacaan data komunikasi serial *remote control* didapatkan bahwa bit-bit data yang terbaca pada serial monitor arduino IDE sesuai dengan bit-bit data saat tombol-tombol aplikasi *remote control* pada *smartphone* ditekan.

4.1.3. Hasil Pengujian *Remote Control* dalam Mengendalikan Gerak Mobil RC

Pengujian terakhir pada penelitian ini adalah apakah aplikasi *remote control* yang telah dibuat dapat mengendalikan mobil RC agar sesuai dengan

fungsi dan kegunaannya. Adapun hasil pengujian *remote control* robot dalam mengendalikan mobil RC adalah sebagai berikut:

4.1.3.1. Hasil Pengujian Kendali *Remote Control* Terhadap Mobil RC

Pengujian kendali *remote control* terhadap mobil RC bertujuan untuk mengetahui keberhasilan aplikasi *remote control* dalam mengendalikan mobil RC. Pengujian ini dilakukan dengan mengkoordinasikan setiap tombol yang ada pada aplikasi *remote control* dengan gerak mobil, apakah gerak mobil sesuai dengan fungsi setiap tombol yang ditekan pada aplikasi *remote control* tersebut atau tidak. Cara dan hasil pengujian kendali *remote control* terhadap mobil dapat dilihat pada gambar 4.10 dan tabel 4.7.



Gambar 4.10. Kondisi Awal Mobil Sebelum Dikendalikan

Tabel 4.7. Hasil Pengujian Kendali *Remote Control* Terhadap Mobil RC

No	Tombol Yang Ditekan	Aksi Yang Harus Dilakukan Oleh Mobil	Hasil Gerakan Mobil	Keterangan
1		Mobil bergerak maju pelan		Berhasil, mobil bergerak lambat
2		Mobil bergerak maju sedang		Berhasil, mobil bergerak sedang
3		Mobil bergerak maju cepat		Berhasil, mobil bergerak cepat

4	Mundur	Mobil bergerak mundur		Berhasil, mobil bergerak mundur
5	STOP	Mobil berhenti		Berhasil, mobil berhenti
6		Mobil berbelok ke kiri		Berhasil, steering berbelok ke kiri
7		Mobil berbelok ke kanan		Berhasil, steering berbelok ke kanan

Berdasarkan data hasil pengujian kendali *remote control* terhadap gerak mobil didapatkan bahwa mobil bergerak sesuai dengan yang dikendalikan oleh aplikasi *remote control* dan pengujian dinyatakan berhasil.

4.1.3.2. Pengujian Jarak Konektivitas *Bluetooth HC-05* dengan *Bluetooth*

Smartphone

Pengujian jarak konektivitas *Bluetooth* mobil RC dengan *Bluetooth smartphone* bertujuan untuk mengetahui seberapa jauh jarak mobil RC dapat dikendalikan oleh aplikasi *remote control* pada *smartphone* dengan menggunakan komunikasi *Bluetooth*. Pengujian ini dilakukan dengan mengendalikan mobil RC dengan kondisi bergerak lurus maju sejauh mungkin, dengan mengukur jarak tempuh mobil menggunakan meteran. Adapun cara dan hasil pengujian jarak Konektivitas *Bluetooth* Robot dengan *Bluetooth Smartphone* dapat dilihat pada gambar 4.11 dan tabel 4.8.



Gambar 4.11. Pengukuran Jarak Tempuh Mobil RC Menggunakan Meteran

Tabel 4.8. Hasil Pengujian Konektivitas *Bluetooth HC-05* Dengan *Bluetooth Smartphone*

No	Jarak Komunikasi Antara Mobil RC dengan Smartphone Android	Konektivitas <i>Bluetooth</i> robot dengan <i>Bluetooth smartphone</i>
1	1 Meter	Tersambung

2	2 Meter	Tersambung
3	3 Meter	Tersambung
4	4 Meter	Tersambung
5	5 Meter	Tersambung
6	6 Meter	Tersambung
7	7 Meter	Tersambung
8	8 Meter	Tersambung
9	9 Meter	Tersambung
10	10 Meter	Tersambung
11	11 Meter	Tersambung, Terdapat Delay
12	12 Meter	Tersambung, Terdapat Delay
13	13 Meter	Tersambung, Terdapat Delay
14	14 Meter	Tersambung, Terdapat Delay
15	15 Meter	Tersambung, Terdapat Delay
16	16 Meter	Tersambung, Terdapat Delay
17	17 Meter	Tersambung, Terdapat Delay\
18	18 Meter	Tersambung, Terdapat Delay

4.2. Pembahasan

Setelah melakukan pengujian terhadap komponen-komponen pembangun sistem, maka sistem yang dibuat bisa dikatakan memenuhi rencana awal, seperti *flowchart* dan diagram blok. Juga penelitian yang dilakukan pun sesuai dengan diagram alir penelitian yang telah dijabarkan pada BAB III, sehingga dalam melakukan pengujian sistem dalam hal ini berarti “Sistem Control Dan Monitoring Putaran Motor Mobil Rc (*Remote Control*) Berbasis *Smartphone* Android” dapat berjalan dengan baik sesuai harapan peneliti.

Pada bagian awal, perancangan dan pengujian perangkat keras (*hardware*) berhasil sesuai perencanaan yaitu dengan berhasil dirakitnya mobil RC yang dapat di *control* dan di monitoring dimana terdiri dari dua bagian yaitu steering yang berfungsi sebagai sistem mekanik penyearah arah laju mobil serta motor DC yang membuat mobil dapat bergerak maju. Adapun pengujian sistem elektronik mobil RC juga berhasil sesuai perencanaan peneliti, dengan menggabungkan rangkaian driver motor dengan power supply serta merangkai modul sensor optocoupler dan modul Bluetooth HC-05.

Pada bagian kedua, pembuatan dan pengujian perangkat lunak (*software*) juga berhasil sesuai perencanaan yaitu dengan berhasil dibuatnya aplikasi remote control mobil RC pada smartphone android dan pemrograman mikrokontroler arduino Mega2560 dan arduino Nano328 untuk control dan monitoring mobil RC. Dimana uji coba bit serial yang dikirimkan oleh aplikasi remote control juga sesuai dengan tampilan pada serial monitor software arduino IDE.

Pada bagian akhir, pengujian aplikasi *remote control* terhadap gerak mobil RC dapat dikatakan berhasil karena control dan monitoring mobil RC dapat bergerak sesuai arahan yang diberikan melalui aplikasi *remote control* robot ini.

4.3. Kekurangan Alat

Dalam penelitian ini masih terdapat kekurangan alat, diantaranya sebagai berikut:

1. Proses pengiriman informasi maupun hasil monitoring terbatas oleh jarak, jarak yang dapat ditempuh adalah kurang dari 20 meter.
2. Kondisi *output* pada *prototype* ini diasumsikan selalu dalam kondisi ideal sehingga proses pengendalian dan monitoring pada aplikasi

ini tidak dapat memberikan notifikasi apabila kondisi output dalam keadaan rusak.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah melakukan perancangan, implementasi, pengujian dan analisa dapat diambil kesimpulan sebagai berikut:

1. Sistem kontrol dan monitoring putaran motor mobil RC (*Remote Control*) menggunakan sebuah sensor optocoupler. Sensor ini dijadikan masukkan kepada protoipe. Sistem kontrol dan monitoring putaran motor mobil RC (*Remote Control*) memiliki beberapa satu kesatuan sistem yang terdiri dari perangkat keras dan perangkat lunak.
2. Optocoupler dengan kombinasi LED-Phototransistor adalah Optocoupler yang terdiri dari sebuah komponen LED (*Light Emitting Diode*) yang memancarkan cahaya infra merah (IR LED) dan sebuah komponen semikonduktor yang peka terhadap cahaya (Phototransistor) sebagai bagian yang digunakan untuk mendeteksi cahaya infra merah yang dipancarkan oleh IR LED.
3. Sistem pengendali dalam *prototype* menggunakan mikrokontroller Arduino Mega 2560 dan Arduino Nano328, beberapa komponen pendukung seperti sensor optocoupler, *Bluetooth* HC-05. Pada *prototype* dengan inputan dari sensor optocoupler untuk membaca putaran motor (rpm) yang akhirnya dapat diproses menjadi sebuah informasi dan informasi tersebut di kirim ke *smartphone* melalui modul *Bluetooth*.

4. Sistem perangkat lunak pada sistem *control* dan monitoring mobil RC (*Remote Control*) dibuat menggunakan *software* APP Inventor. APP Inventor digunakan sebagai pembuat aplikasi. Aplikasi ini dibuat untuk mendukung kinerja alat sebagai monitoring jarak jauh melalui *smartphone*.

5.2 Saran

Perancangan prototipe sistem kontrol dan monitoring putaran motor mobil RC (*Remote Control*), memiliki beberapa saran diantaranya:

1. Adanya notifikasi baterai di mobil agar dapat lebih terkontrol dari segi sumber tegangan.
2. Adanya notifikasi pada aplikasi *remote control* apabila terjadi kerusakan komponen pada mobil.
3. Adanya notifikasi “PUTARAN MOTOR STABIL” pada aplikasi *remote control* ketika putaran motor telah mencapai putaran yang telah ditentukan.
4. Komunikasi yang digunakan agar jarak tempuh lebih jauh bias menggunakan sistem IP Address dengan menggunakan module NodemCu.

DAFTAR PUSTAKA

- Andrianto, H. (2008). *Pemrograman Mikrokontroler AVR ATmega16*. Bandung: Informatika Bandung.
- Bambang Murdaka Eka Jati, *Fisika Dasar*, ANDI, Yogyakarta, 2008.
- Djuandi, F. (2011). *Pengenalan Arduino*. *Pengenalan Arduino*, 9.
- Gunadarma. (1994). *Pengantar Komunikasi Data*. Jakarta: Penerbit Gunadarma.
- Ibrahim M.PD.I , (2016, November, Kamis), Pengertian, Sejarah. Diambil kembali dari: <http://www.terwujud.com/2014/12/mobilpengertian-sejarah-inovasi.html>
- Isianto, J. E. (2014). *Pengantar Elektronika dan Instrumentasi, Pembekalan Project Arduino dan Android*. Yogyakarta: ANDI.
- Kadir, A. (2013). *Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemrogramannya Menggunakan Arduino*. Yogyakarta: ANDI.
- Raymond A. serway, *Fisika untuk Sains dan Teknik*, Salemba Teknika. Jakarta, 2009.
- Safaat, N. (2011). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika Bandung.
- Setiawan, (2016, November, Selasa). Mobil. Diambil kembali dari Kamus Besar Bahasa Indonesia: <http://kbbi.web.id/mobil>
- Setiawan, (2016, November, Selasa). Kontrol. Diambil kembali dari Kamus Besar Bahasa Indonesia: <http://kbbi.web.id/kontrol>
- Setiawan, (2016, November, Selasa). Monitor. Diambil kembali dari Kamus Besar Bahasa Indonesia: <http://kbbi.web.id/monitor>
- Stallings, W. (2007). *Komunikasi & Jaringan Nirkabel Edisi kedua Jilid 2*. Jakarta: Erlangga.
- Supriyanto, A. (2006). *Tinjauan Teknis Perangkat Wireless dan Standar Keamanannya*. Semarang: Universitas Stikubank.
- Yusro, M., Djatmiko, W., Premono, A., Prabawati, M., Suryasumirat, D. S., Yuliatmojo, P., . . . Sukarno, R. (2015). *Buku Panduan Penyusunan Skripsi dan Non Skripsi*. Jakarta: FT UNJ.
- Zaka (2016, November, Jum'at). Pengertian pengawasan dan *controlling*. Diambil kembali dari: <http://www.pengertianahli.com/2014/01/pengertian-pengawasan-controlling.html>

LAMPIRAN

1. Lampiran 1 List Program Arduino Mega2560

```
//Created: OneRpm                               Serial.begin(9600);
//Nama :Arwan Faisyal
//No. Reg: 5215110327                             //Deklarasi untuk RPM
//Prodi: Pendidikan Teknik                       attachInterrupt(0, counter,
Elektronika                                       FALLING);
//UNIVERSITAS NEGERI                             pulses = 0;
JAKARTA                                           rpm = 0;
//Bismillah                                       timeold = 0;

#include <SoftwareSerial.h>
unsigned int rpm;
volatile byte pulses;
unsigned long timeold;
unsigned int pulsesperturn = 12;
int PWMPin = 3;
int mundur = 4;
int SensorPin =2;
int val =0;

void counter()
{
  pulses++;
}
void setup()
{
  //Bluetooth
  //Kontrol Input & Output
  pinMode(PWMPin, OUTPUT);
  pinMode (mundur, OUTPUT);
}
void loop()
{
  if(Serial.available(>0)
  if (millis() - timeold >= 1000)
  val = Serial.read();
  Serial.println(PWMPin);
  {
  //Baca RPM
  detachInterrupt(0);
  rpm =(30*1000/pulsesperturn) /
  (millis()-timeold) * pulses;
  rpm = map (rpm, 0, 255, 0, rpm);
```

```

timeold = millis ();
pulses = 0;
Serial.print(" ");
Serial.println(rpm, DEC);
attachInterrupt(0, counter,
FALLING);
    delay(1000);
    }

//Kendali
    {
    if(val==1)
for(int fadeValue = 0 ; fadeValue <=
100; fadeValue +=5)
analogWrite(PWMPin, fadeValue);

else if(val==2)
for(int fadeValue = 0 ; fadeValue <=
170; fadeValue +=5)
analogWrite(PWMPin, fadeValue);
else if(val==3)
for(int fadeValue = 0 ; fadeValue <=
255; fadeValue +=5)
analogWrite(PWMPin, fadeValue);

else if(val==0)
for(int fadeValue = 255 ; fadeValue
>= 0; fadeValue -=5)
analogWrite(PWMPin, fadeValue);
    }
    {
if (val==4)
for(int fadeValue = 0 ; fadeValue <=
140; fadeValue +=5)
analogWrite(mundur, fadeValue);

else if(val==5)
for(int fadeValue = 140 ; fadeValue
>= 0; fadeValue -=5)
analogWrite(mundur, fadeValue);
    }
    }

```

2. Lampiran 2 List Program Arduino Nano328

```
#include<SoftwareSerial.h>
```

```
#include<Servo.h>
```

```
3
```

```
Servo ser;
```

```
int poser = 60;
```

```
int value;
```

```
void setup()
```

```
{
```

```
Serial.begin(9600);
```

```
ser.attach(6);
```

```
}
```

```
void loop()
```

```
{
```

```
if(Serial.available()>0)
```

```
while(Serial.available())
```

```
{
```

```
value = Serial.read();
```

```
poser = value ;
```

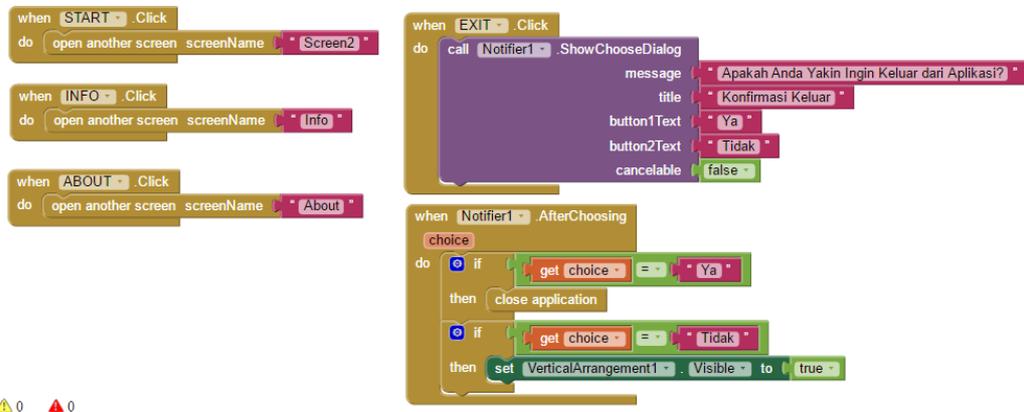
```
ser.write(poser);
```

```
delay(15); }
```

```
}
```

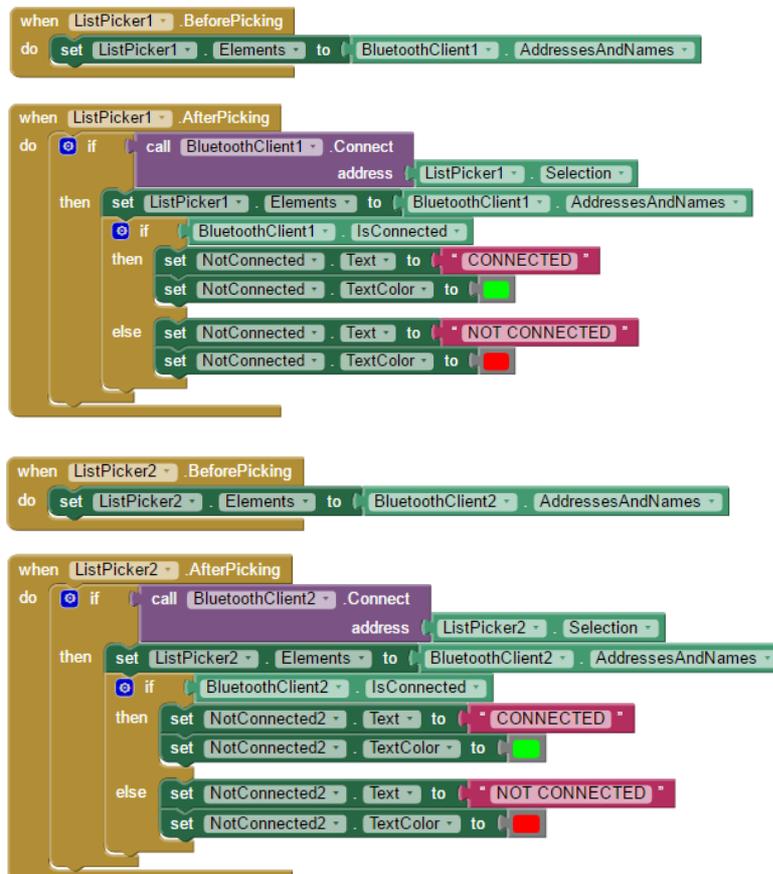
3. Lampiran 3 Program Android App Inventor

A. Program Screen 1 (Home)



0 0

B. Program Screen 2 (Navigasi)



The image shows several Scratch code blocks:

- when Level01 . TouchUp**: do call BluetoothClient1 . Send1ByteNumber number " 1 "
- when Level02 . TouchUp**: do call BluetoothClient1 . Send1ByteNumber number " 2 "
- when Level03 . TouchUp**: do call BluetoothClient1 . Send1ByteNumber number " 3 "
- when STOP . TouchUp**: do call BluetoothClient1 . Send1ByteNumber number " 0 "
- when mundur . TouchDown**: do call BluetoothClient1 . Send1ByteNumber number " 4 "
- when mundur . TouchUp**: do call BluetoothClient1 . Send1ByteNumber number " 5 "
- when Slider1 . PositionChanged**: thumbPosition do set Label11 . Text to round Slider1 . ThumbPosition call BluetoothClient2 . Send1ByteNumber number round Slider1 . ThumbPosition
- when Clock1 . Timer**: do if BluetoothClient1 . IsConnected then set Output . Text to call BluetoothClient1 . ReceiveText numberOfBytes call BluetoothClient1 . BytesAvailableToReceive
- when Button5 . Click**: do close screen

C. Program Screen 3 (Info)

when Button1 . Click
do close screen

D. Program Screen 4 (About)

when Button1 . Click
do close screen

Daftar Riwayat Hidup



Arwan Faisyal lahir di Purwakarta, 16 Maret 1994, anak pertama dari dua bersaudara, penulis dibesarkan dari keluarga biasa namun dengan cara yang luar biasa oleh kedua orang tua yaitu Mahmud Widada dan Tini Sumiantini. Bertempat tinggal di Jl. Lodaya Kp. Baru RT.04/Rw.0 Kel. Nagri Tengah, Kec. Purwakarta, Kab. Purwakarta, Jawa Barat (4111). Penulis pernah melaksanakan Program Kerja Lapangan (PKL) di PT. TOYOTA MOTOR MANUFACTURING INDONESIA (TMIIN) yang beralamat di Jl. Laks. Yos Sudarso, Sunter II Jakarta 14330-Indonesia pada bulan 18 Agustus s/d 12 September 2014. Dan juga pernah melaksanakan Praktik Keterampilan Mengajar di SMKN 29 JAKARTA yang beralamat di Jl. Prof. Joko Sutono SH No.1, RT.15/RW.6, Petogogan, Kby. Baru, Kota Jakarta Selatan, DKI Jakarta 12170 pada bulan 16 Februari s/d 16 Mei 2015.

Riwayat Pendidikan: Pendidikan formal dimulai di TKA Alhidayah (1997-1998), kemudian melanjutkan sekolah di TPA Alhidayah (1998-1999), SDN Singawinata III Purwakarta(1999-2005), SMPN 2 Pasawahan Kab. Purwakarta (2005-2008), SMKN 1 Purwakarta (2008-2011). Setelah itu melanjutkan pendidikan di Program Studi Pendidikan Teknik Elektronika, Fakultas Teknik, Universitas Negeri Jakarta pada tahun 2011.

Riwayat Organisasi: Selama kuliah ada beberapa organisasi yang penulis pernah ikuti diantaranya pernah menjabat sebagai Staff Keuangan Badan Perwakilan Mahasiswa (BPM UNJ) periode 2011-2012, Staff Departemen Syiar Lembaga Dakwah Kampus Sahabat Muslim (LDK Salim UNJ) 2015-2016, Ketua Bidang Kekeluargaan Perkumpulan Mahasiswa Purwakarta (PERMATA) periode 2011-2013.