



**PEMBUKA KAP DAN BAGASI MOBIL MENGGUNAKAN
SMARTPHONE BERBASIS *BLUETOOTH***

PROYEK AKHIR

**Diajukan Kepada Fakultas Teknik Universitas Negeri Yogyakarta
Untuk Memenuhi Sebagian Persyaratan
Guna Memperoleh Gelar Ahli Madya**



OLEH:

IVANA YUNISA CAHYONO

NIM.14507134003

PROGRAM STUDI TEKNIK ELEKTRONIKA

JURUSAN PENDIDIKAN TEKNIK ELEKTRONIKA DAN INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS NEGERI YOGYAKARTA

2018

LEMBAR PERSETUJUAN

PROYEK AKHIR

**PEMBUKA KAP DAN BAGASI MOBIL MENGGUNAKAN
SMARTPHONE BERBASIS *BLUETOOTH***

Oleh

IVANA YUNISA CAHYONO

14507134003

Telah diperiksa dan disetujui oleh pembimbing:

Untuk diuji

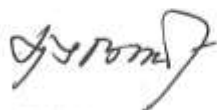
Yogyakarta, 10 Januari 2018

Mengetahui,

Kaprodi Teknik Elektronika

Menyetujui,

Pembimbing Proyek Akhir



Dr. Sri Waluyanti

NIP. 19581218 198603 2 001



Dessy Irmawati ST,MT

NIP. 19791214 201012 2 002

Dipersiapkan dan Disusun Oleh:

IVANA YUNISA CAHYONO

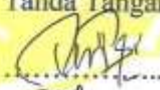
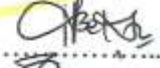
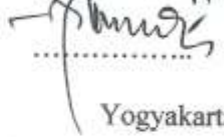
14507134003

Telah dipertahankan didepan Dewan Penguji Proyek Akhir
FAKULTAS TEKNIK UNIVERSITAS NEGERI YOGYAKARTA

Pada tanggal 10 Januari 2018

Dan Dinyatakan Telah Memenuhi Syarat Guna Memperoleh Gelar
Ahli Madya Teknik

SUSUNAN DEWAN PENGUJI

Nama	Jabatan	Tanda Tangan	Tanggal
1. Dessy Irmawati, ST,MT.	Ketua Penguji		23/01/2018
2. Bkti Wulandari, M.pd.	Sekretaris Penguji		23/01/2018
3. Dr. Pramudi Utomo, M.Si.	Penguji		23/01/2018

Yogyakarta, 10 Januari 2018

Dekan Fakultas Teknik UNY



Dr. Widarto, M.Pd.

NIP. 19631230 198812 1 0010

LEMBAR PERNYATAAN KEASLIAN

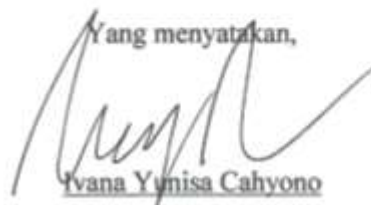
Yang bertanda tangan dibawah ini:

Nama : Ivana Yunisa Cahyono
NIM : 14507134003
Program Studi : Teknik Elektronika D-III
Judul Proyek Akhir : Pembuka Kap dan Bagasi Mobil Menggunakan
Smartphone Berbasis Bluetooth

Menyatakan bahwa Proyek Akhir ini adalah hasil pekerjaan saya sendiri, dan sepanjang pengetahuan saya, tidak berisi materi yang ditulis oleh orang lain sebagai persyaratan penyelesaian studi di Universitas Negeri Yogyakarta atau perguruan tinggi lain, kecuali bagian-bagian tertentu saya ambil sebagai acuan dengan mengikuti kaidah penulisan karya ilmiah yang benar. Jika ternyata terbukti pernyataan ini tidak benar, sepenuhnya menjadi tanggung jawab saya.

Yogyakarta, 10 Januari 2018

Yang menyatakan,



Ivana Yunisa Cahyono
NIM. 14507134003

PERSEMBAHAN

Proyek akhir ini penulis persembahkan untuk:

Allah SWT yang telah memberikan kehidupan dan kesempatan

Ibu dan Ayah atas segala doa, perhatian, kasih sayang dan dukungan

Adikku yang senangtiasa menyemangati dan memotifasi

Teman-teman Teknik Elektronika 2014 tetap semangat

Untuk semua orang yang telah banyak membantu untuk proyek akhir ini, terima kasih

MOTTO

“Sesuatu yang bisa dirimu bayangkan adalah nyata”

“Janganlah menganggap setiap momen kehidupan adalah suatu hal wajar dan memang sudah sepantasnya kita dapatkan, akan tetapi berpikirlah setiap momen kehidupan adalah hal-hal paling berharga yang setiap saat bisa hilang”

Hal yang paling menakutkan di dunia ini adalah “WAKTU”

PROYEK AKHIR

PEMBUKA KAP DAN BAGASI MOBIL MENGGUNAKAN *SMARTPHONE* BERBASIS *BLUETOOTH*

Oleh : Ivana Yunisa Cahyono

NIM : 14507134003

ABSTRAK

Tujuan pembuatan alat ini adalah untuk (1) merancang dan membuat pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth* agar memudahkan pengguna dalam membuka kap dan bagasi mobil, (2) mengetahui sistem kerja pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth* pada industri atau masyarakat umum, (3) mampu mengimplementasikan rancangan sistem pada mobil keluaran tahun 70-an.

Metode pengembangan pembuka kap dan bagasi mobil menggunakan *smartphone* berbasis *bluetooth* terdiri dari tahap identifikasi kebutuhan, analisis kebutuhan, blok diagram rangkaian, perencanaan sistem, langkah pembuatan alat, *flowchart* program, pengujian alat dan pengambilan data. Sistem ini memakai arduino uno (AT-Mega 328) sebagai kontrol utama, *smartphone* android sebagai media masukan *virtual button*. Alat ini diujikan pada mobil Honda Civic First Generation tahun 1979 dengan berat penutup Kap mesin sebesar 10 kg dan berat pintu Bagasi sebesar 6 kg. Alat pembuka kap dan bagasi mobil menggunakan tiga buah hidrolik sebagai penyangga dengan dua buah motor *dynamo starter* dan relay 2 channel masing-masing 5V dan 12V.

Hasil untuk pengujian catu daya aki (accumulator) 12V dan Regulator 9V tanpa beban ketika mesin mobil mati dengan rata-rata *error* sebesar 5,32% dan 3,4%. Saat dengan beban ketika mesin mobil mati dengan rata-rata *error* sebesar 5,7% dan 3,7%. Saat tanpa beban ketika mesin mobil menyala dengan rata-rata *error* sebesar 15,68% dan 3,4%. Pengujian ada beban ketika mesin mobil menyala dengan rata-rata *error* sebesar 14,8% dan 3,8%. Pengujian *delay* waktu *virtual button* menggunakan lima jenis sistem operasi *smartphone* yang berbeda dengan rata-rata *delay* 0,15 detik. Jangkauan jarak *bluetooth* maksimal dari 1 sampai 30 meter dengan dan tanpa penghalang, Untuk pengujian unjuk kerja sistem dapat bekerja sesuai dengan fungsi dan tujuannya, selain itu dalam menjalankan algoritma program alat ini mempunyai tingkat dengan keberhasilan 100%.

Kata Kunci: Pembuka Kap dan Bagasi, *Smartphone*, Motor *Dynamo Starter*

KATA PENGANTAR

Assalamualaikum wr. wb.

Puji syukur penulis panjatkan ke hadirat Allah SWT sehingga dengan rahmat dan hidayah-Nya Laporan Proyek Akhir ini dapat terselesaikan tanpa halangan yang berarti. Sholawat serta salam tercurah pada Rasulullah SAW, keluarga, sahabat dan orang-orang yang istiqomah di jalan-Nya.

Dalam menyusun Laporan Proyek Akhir ini penulis merasa banyak kekurangan karena terbatasnya kemampuan dan pengetahuan penulis. Oleh karena itu pada kesempatan ini penulis menyampaikan ucapan terima kasih dan penghargaan yang sebesar-besarnya kepada:

1. Dessy Irmawati ST, MT. selaku Dosen Pembimbing Penyusunan Laporan Proyek akhir.
2. Dr. Sri Waluyanti, M.Pd. selaku Ketua Program Studi Diploma III dan Koordinator Proyek Akhir Jurusan Pendidikan Teknik Elektronika dan Informatika Universitas Negeri Yogyakarta.
3. Dr. Fatchul Arifin, M.T. selaku Ketua Jurusan Pendidikan Teknik Elektronika dan Informatika Universitas Negeri Yogyakarta.
4. Dr. Widarto, M.Pd. selaku Dekan Fakultas Teknik Universitas Negeri Yogyakarta.
5. Keluarga yang telah memberikan semangat, dukungan, dan motivasi.
6. Seluruh Dosen Pengajar Teknik Elektronika dan Informatika Universitas Negeri Yogyakarta atas bekal ilmu yang diberikan kepada penulis.
7. Bkti Wulandari, M.Pd. yang telah membimbing dalam perkuliahan dan pembimbing Praktik Industri penulis.

8. Teman-teman Fakultas Teknik UNY khususnya Teknik Elektronika kelas B 2014 yang telah memberikan bantuan sehingga pembuatan proyek akhir ini dapat terselesaikan.
9. Bapak Budi yang telah membantu mengelas dan memasang komponen tugas akhir.
10. Rifan, Agung dan Priyo yang selalu memberikan motivasi dan semangat.
11. Semua pihak yang tidak dapat penulis sebutkan satu persatu yang telah membantu hingga terselesainya laporan ini.

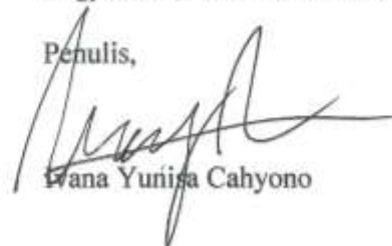
Pada orang yang penulis kasihi dan sayangi. Ayah penulis Ony Setyo Cahyono dan ibu penulis Karsinah yang telah memberi penulis cinta dan kasih sayang hingga penulis tumbuh dewasa saat ini serta selalu memberi semangat bahkan membantuku memberi saran dalam mengerjakan tugas akhir ini. Adik penulis Sharfina yang senang tiasa memotivasi. Tiada mungkin penulis balas hanya dengan selembur kertas kata cinta. Semoga ini menjadi awal langkah penulis untuk membuat ayah dan ibu bahagia.

Akhir kata penulis mohon maaf yang sebesar-besarnya apabila ada kekeliruan di dalam penulisan laporan ini.

Wassalamu'alaikum wr. wb.

Yogyakarta, 10 Januari 2018

Penulis,



Wana Yunisa Cahyono

DAFTAR ISI

HALAMAN JUDUL.....	i
PERSETUJUAN	ii
PENGESAHAN	iii
PERNYATAAN KEASLIAN.....	iv
PERSEMBAHAN	v
MOTTO.	v
ABSTRAK	vi
KATA PENGANTAR	vii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN	1
A. Latar Belakang	1
B. Identifikasi Masalah	3
C. Batasan Masalah.....	3
D. Rumusan Masalah	4
E. Tujuan	4
F. Manfaat	5
G. Keaslian Gagasan	6
BAB II PENDEKATAN PEMECAHAN MASALAH	7
A. Kap Mobil	7
B. Bagasi Mobil	8
C. Arduino Uno	10
D. Modul <i>bluetooth</i> HC-05	20
E. Sistem Operasi Android	21
F. App Inventor	26

G. Relay	27
H. Hidrolik	30
I. Motor Dynamo Starter	31
J. Baterai Aki Mobil	32
BAB III KONSEP PERANCANGAN.....	40
A. Identifikasi Kebutuhan	40
B. Analisis Kebutuhan	41
C. Perancangan Sistem	44
D. Blok Diagram Rangkaian.....	51
E. Langkah Pembuatan Alat	52
F. Spesifikasi Alat	57
G. Pengujian Alat.....	58
H. Tabel Uji Alat.....	58
I. Pengoperasian Alat.....	67
BAB IV PENGUJIAN DAN PEMBAHASAN	69
A. Hasil Pengujian	69
B. Pembahasan.....	75
BAB V PENUTUP.....	85
A. Kesimpulan	85
B. Keterbatasan Alat	86
C. Saran.....	86
DAFTAR PUSTAKA	87
LAMPIRAN	89

DAFTAR TABEL

Tabel 1. Tingkatan versi android	22
Tabel 2. Kebutuhan alat dan bahan	44
Tabel 3. Rencana pengujian catu daya tanpa beban saat mesin mobil mati	60
Tabel 4. Rencana pengujian catu daya dengan beban saat mesin mobil mati.....	60
Tabel 5. Rencana pengujian catu daya tanpa beban saat mesin mobil menyala	61
Tabel 6. Rencana pengujian catu daya dengan beban saat mesin mobil menyala	61
Tabel 7. Rencana pengujian jangkauan <i>bluetooth</i>	62
Tabel 8. Rencana pengujian motor <i>dynamo starter</i> saat mesin mobil mati	63
Tabel 9. Rencana pengujian motor <i>dynamo starter</i> saat mesin mobil menyala	63
Tabel 10. Rencana pengujian modul relay 2 channel 5V dan relay 12V	64
Tabel 11. Rencana pengujian perintah <i>virtual button</i>	65
Tabel 12. Rencana pengujian <i>button</i> manual <i>emergency</i>	66
Tabel 13. Rencana pengujian unjuk kerja	67
Tabel 14. Pengujian catu daya tanpa beban saat mesin mobil mati	69
Tabel 15. Pengujian catu daya dengan beban saat mesin mobil mati	70
Tabel 16. Pengujian catu daya tanpa beban saat mesin mobil menyala.....	70
Tabel 17. Pengujian catu daya dengan beban saat mesin mobil menyala.....	71
Tabel 18. Pengujian jangkauan <i>bluetooth</i>	71
Tabel 19. Pengujian motor <i>dynamo starter</i> saat mesin mobil mati.....	72
Tabel 20. Pengujian motor <i>dynamo starter</i> saat mesin mobil menyala	72
Tabel 21. Pengujian modul relay 2 channel 5V dan relay 12V	73
Tabel 22. Pengujian perintah <i>virtual button</i>	73
Tabel 23. Pengujian <i>button</i> manual <i>emergency</i>	74

DAFTAR GAMBAR

Gambar 1. Kap mobil.....	7
Gambar 2. Bagasi mobil.....	8
Gambar 3. Arduino uno.....	11
Gambar 4. Pin <i>power</i> dan <i>ground</i>	14
Gambar 5. Tampilan jendela arduino.....	15
Gambar 6. <i>Bluetooth module</i> HC-05.....	21
Gambar 7. Tampilan app inventor pada web browser	27
Gambar 8. Relay 12 volt	28
Gambar 9. Struktur sederhana relay.....	28
Gambar 10. Modul relay 2 channel.....	29
Gambar 11. Komponen hidrolik	31
Gambar 12. Motor <i>dynamo starter</i>	32
Gambar 13. Prinsip kerja motor starter satu siklus	33
Gambar 14. Kontruksi sel Aki	35
Gambar 15. Kontruksi block sel.....	35
Gambar 16. Hubungan block sel.....	36
Gambar 17. Batteray terisi penuh.....	36
Gambar 18. Batteray dalam keadaan dipakai.....	37
Gambar 19. Batteray dalam keadaan kosong.....	38
Gambar 20. Block diagram perancangan alat secara umum	40
Gambar 21. Pemasangan hidrolik pada kap.....	45
Gambar 22. Pemasangan hidrolik pada bagasi	46
Gambar 23. Arduino uno R3.....	46
Gambar 24. Rangkaian konfigurasi komponen.....	47
Gambar 25. Inisialisasi pin pada program Arduino UNO	48
Gambar 26. <i>Flowchart</i> aplikasi android.....	50
Gambar 27. <i>Flowchart</i> program arduino	51

Gambar 28. Blok diagram rangkaian keseluruhan	52
Gambar 29. Tampilan login aplikasi	54
Gambar 30. Tampilan menu pemilihan	54
Gambar 31. Tampilan blok aplikasi	55
Gambar 32. Tampilan rangkaian regulator	56
Gambar 33. Tampilan titik pengukuran catu daya	60
Gambar 34. Tampilan <i>button</i> manual <i>emergency</i>	66

DAFTAR LAMPIRAN

Lampiran 1. Skema rangkaian keseluruhan	90
Lampiran 2. Layout.....	91
Lampiran 3. Daftar komponen	92
Lampiran 4. <i>Listing</i> program.....	93
Lampiran 5. Gambar alat.....	94
Lampiran 6. Tabel 13 hasil uji unjuk kerja	95
Lampiran 7. Datasheet Arduino uno	97
Lampiran 8. Data sheet <i>bluetooth</i> HC-05	105

BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Inovasi-inovasi pada dunia otomotif menjadi daya minat untuk masyarakat pada mobil-mobil baru. Namun, hal ini tidak menutup kemungkinan pada pecinta mobil kuno tahun 70-an untuk berinovasi menambah fitur-fitur mobil yang terbatas. Perkembangan otomotif terbaru seperti pembuka bagasi menggunakan hidrolis diaplikasikan untuk meringankan beban angkat pada bagasi mobil yang terbilang berat. Namun, pembuka bagasi ini sayangnya tetaplah melalui tombol pada kabin mobil atau *central lock remote*. Cara mengoperasikan juga tetap membutuhkan tenaga manusia untuk memberikan dorongan sedikit sehingga hidrolis dapat terbuka dengan sendirinya. Mobil kuno tahun 70-an, pembuka bagasi masih manual tidak ada penyangga hidrolis hanya sebuah besi penyangga yang pengoperasiannya harus dilakukan oleh pengguna mobil sendiri.

Pembuka bagasi pada mobil baru memang sekarang terbilang lebih maju daripada pembuka kap mesin. Pada mobil baru ataupun mobil kuno tahun 70-an, pembuka kap mesin masih ditangani secara manual dan tidak ada penyangga hidrolis yang terpasang. Pengguna pernah atau sering tertimpa pintu kap atau bagasi saat lalai membuka kap atau bagasi mobil, pemicu hal ini cukup beragam seperti, kelelahan se usai mengemudi, kecerobohan pengguna, salah menempatkan besi penyangga pada mobil atau kerusakan yang terjadi tiba-tiba pada besi penyangga pintu kap atau bagasi. Seperti survey yang telah penulis lakukan pada

beberapa responden. Beberapa responden juga berpendapat bahwa pengoperasian yang masih manual tentu saja tidak praktis, membutuhkan waktu dan energi yang banyak pula.

Berdasarkan latar belakang di atas, penulis memilih pembuka kap dan bagasi mobil karena selama ini pengguna direpotkan dengan tingkat kepraktisan yang kurang saat membuka kap dan bagasi ketika mobil dalam keadaan tertutup. Pengguna diharuskan membuka kunci mobil atau *central lock* kemudian menekan tombol pada kabin. Permasalahan juga terjadi pada biaya variasi mobil yang terbilang mahal dan pengoperasian yang sulit membuat para pengguna mobil kuno tahun 70-an tidak berpikir untuk menambahkan hidrolis pada bagasi atau kap mesin mobil demi keamanan mereka.

Penulis berpendapat bahwa untuk memaksimalkan kinerja hidrolis sebagai pembuka otomatis dan penyangga kap serta bagasi mobil perlu dilakukan. Selain itu, penggunaan *bluetooth* dan *smartphone* untuk mengontrol pembuka otomatis kap dan bagasi perlu pula diujikan. Perangkat tersebut diharapkan dapat membuka kap dan bagasi secara tepat. Untuk itu perlu dilakukan pembuatan proyek akhir dengan judul Pembuka kap dan bagasi mobil menggunakan *smartphone* berbasis *bluetooth*.

B. Identifikasi masalah

Berdasarkan latar belakang di atas dapat diidentifikasi beberapa permasalahan antara lain:

1. Sering terjadinya *human error* dengan teknologi pembuka kap dan bagasi mobil yang masih manual
2. Tingkat keamanan rendah dengan teknologi pembuka kap dan bagasi mobil yang masih manual
3. Tingkat kepraktisan dan ketepatan yang kurang dari pembuka kap dan bagasi yang masih manual
4. Harga dan jasa pemasangan hidrolis yang masih cukup mahal
5. Belum adanya pemanfaatan hidrolis secara maksimal pada pintu bagasi ataupun pada kap mobil
6. Belum adanya pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth*

C. Batasan masalah

Berdasarkan identifikasi masalah di atas permasalahan dibatasi pada *point* 5 dan 6 yaitu, belum adanya pemanfaatan hidrolis secara maksimal pada pintu bagasi ataupun pada kap mobil dan pembuka kap dan bagasi yang menggunakan *smartphone* berbasis *bluetooth*. Pembuatan proyek akhir ini menggunakan Arduino Uno sebagai pengendali utama, motor *dynamo starter* untuk *unlock* kunci pintu kap dan bagasi, Modul *bluetooth* sebagai penghubung dan *smartphone* android sebagai media masukan tombol pembuka.

D. Rumusan masalah

Berdasarkan latar belakang masalah yang ada, maka dapat ditentukan rumusan permasalahan sebagai berikut:

1. Bagaimana merancang *software* pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth*?
2. Bagaimana merancang *hardware* pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth*?
3. Bagaimana sistem kerja pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth*?

E. Tujuan

Adapun tujuan dari tugas akhir ini adalah:

1. Merancang dan membuat pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth* agar memudahkan pengguna dalam mengakses kap dan bagasi mobil.
2. Mengetahui sistem kerja pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth* pada industri atau masyarakat umum.
3. Mampu mengimplementasikan rancangan sistem ini agar dapat digunakan dalam kegiatan menggunakan mobil sehari-hari

F. Manfaat

Dalam pembuatan proyek akhir ini, diharapkan dapat memberikan manfaat sebagai berikut:

1. Bagi Mahasiswa
 - a. Mampu merealisasikan teori yang didapatkan selama mengikuti perkuliahan.
 - b. Sebagai wujud kontribusi terhadap universitas.
2. Sebagai wujud kontribusi terhadap universitas bagi Jurusan Pendidikan Teknik Elektronika
 - a. Sebagai wujud partisipasi dalam pengembangan dibidang IPTEK.
 - b. Sebagai tolak ukur daya serap mahasiswa yang bersangkutan selama menempuh pendidikan dan kemampuan ilmunya secara praktis.
3. Bagi Dunia Industri
 - a. Terwujudnya alat sebagai sarana peningkatan teknologi dan alat pembantu yang dapat dimanfaatkan dalam dunia otomotif.
 - b. Sebagai bentuk kontribusi terhadap masyarakat dalam mewujudkan pengembangan teknologi.
 - c. Sebagai inovasi terhadap dunia industri otomotif dalam mewujudkan perkembangan teknologi

G. Keaslian Gagasan

Proyek akhir dengan judul “Pembuka kap dan bagasi mobil menggunakan *smartphone* berbasis *bluetooth*” ini dibuat untuk mempermudah para pengguna mobil untuk membuka kap dan bagasi mobil yang selama ini masih menimbulkan *human error* dan kurangnya efektifitas dalam membuka kap dan bagasi. Berikut ini beberapa penelitian relevan yang dapat dijadikan acuan untuk karya proyek akhir ini diantaranya adalah:

1. *Design of car lifthing mechanism using sensitive hydraulic system*, Yeoh Li Ching, 2016 pada karya ini menggunakan sensitive hidrolik sebagai pembuka dan penyangga kap mesin
2. Implementasi sistem *bluetooth* menggunakan android dan arduino untuk kendali peralatan elektronik, Rahmiati dkk, 2014 karya ini digunakan untuk kendali peralatan elektronik menggunakan *speech recognition* dan sentuhan tombol *virtual* pada aplikasi android yang dikirimkan melalui inframerah.
3. Rancang Bangun Sistem Printer Tanpa Kabel Berbasis *bluetooth* dan WIFI, Ahmad Irfan Yusuf, 2014 pada karya ini menggunakan *Blueetooth* untuk mengendalikan *printer*

BAB II

PENDEKATAN PEMECAHAN MASALAH

A. Kap Mobil

Kap menurut kamus besar Bahasa Indonesia adalah tudung mesin atau atap mobil. Kap mesin dibuat secara aerodinamis untuk meminimalkan efek udara. Kap umumnya digunakan untuk melindungi bagian-bagian penting mobil seperti radiator, mesin dan banyak bagian lainnya. Di dunia otomotif optimalisasi bobot memainkan peran penting dalam perubahan material kap dari berat menjadi ringan, dinamis dan kaku. Seperti yang terlihat pada Gambar 1.



Gambar 1. Kap Mobil

(Sumber : <https://www.quora.com/What-is-the-function-of-a-car-bonnet>)

Alat pembuka kap biasanya terletak di bawah lingkaran kemudi atau di samping kursi pengemudi. Saat pengemudi menarik penutup maka kunci panel kap mesin terlepas dan memungkinkan akses ke kompartemen mesin. Pada mobil balap atau mobil dengan kap aftermarket (yang tidak menggunakan sistem gerendel pabrik) kap mesin bisa ditebuk dengan pin kap. Pada sebuah kap biasanya terdapat *hood*

ornament, hood scoop, power bulge, dan wiper jet. Kap biasanya terbuat dari baja, namun aluminium sekarang lebih menjadi tren di perusahaan otomotif.

B. Bagasi Mobil

Bagasi paling sering terletak di bagian belakang kendaraan. Desain awal bagasi yaitu ruang kosong berada di bagian belakang kendaraan yang memungkinkan untuk memasang pintu bagasi. Bagasi didesain untuk area penyimpanan ke dalam kendaraan dan seiring waktu bagasi tidak hanya untuk area penyimpanan tetapi juga untuk memberikan tampilan ramping pada bagian belakang mobil. Bagasi sebagai penyimpanan barang utama biasanya disediakan di ujung kendaraan yang berlawanan dengan lokasi mesin. Seperti yang terlihat pada gambar 2.



Gambar 2. Bagasi Mobil

1. Pintu Bagasi

Pintu bagasi berengsel di bagian atas, samping, atau bawah. Jika pintu berengsel di bagian bawah itu disebut truk bak, terutama di Amerika Serikat. Pintu bawah adalah pintu umum di *station wagon*, truk *pickup*, atau kendaraan *sport* (SUV). Karena potensi asap karbon monoksida, jendela *tailgate* pada *station*

wagon harus ditutup setiap kali mesin menyala. *Tailgates* adalah ruang untuk keperluan penyimpanan dalam mobil. Mekanisme pintu dan engsel dari *tailgate* 3 arah dirancang dengan pegangan khusus untuk dibuka pada arah yang dipilih. Pada akhir 1970-an, itu adalah pengaturan bagasi *station wagon* yang paling umum.

Pada pintu bagasi dipasang engsel agar dapat terbuka dan sebuah peyangga besi juga dipasang agar pintu bagasi agar dapat terangkat dan tetap terbuka. hidrolis atau pegas juga menjadi alternatif yang digunakan untuk membantu mengangkat dan menahan tutup bagasi. Pada mobil dengan bagasi di belakang, penutupnya kadang-kadang disertai dengan lampu rem depan yang terpasang di tengah. Pada mobil modern, tutup bagasi bisa dibuka dengan *fob* kunci mobil.

2. *Lock*

a. Tradisional *Lock*

Kunci bagasi menjaga bagasi yang terkunci untuk perlindungan. Kunci ini sedikit berbeda dari kunci pintu mobil dan kunci umum kendaraan lainnya. Sebagian besar dibagi menjadi tiga bagian yang berbeda: mekanisme kunci itu sendiri, mekanisme kunci *Emergency Releases* dan mekanisme *Automatic Releases*. Komponen utama dari kunci ini adalah silinder dan *strike*. Silinder memegang *tumblers* dan mekanisme lain yang diperlukan untuk membuka kuncian, sementara *strike* memasukkan kait yang menahan bagasi saat terkunci. Sebagian besar mobil memiliki silinder dan penyangga yang terletak di tutup bagasi. Ada banyak desain kunci bagasi yang berbeda, dan beberapa mungkin memisahkan silinder dan *strike* di berbagai bagian bagasi.

b. *Lock and Emergency Release*

Lock and Emergency Release adalah kunci yang menggunakan silinder mekanisme *six-to 10-tumbler*. Semakin banyak *tumblers* yang dimiliki bagasi, maka akan semakin aman mobil tersebut. Perangkat *Emergency Release* adalah mekanisme sederhana yang terletak di dalam bagasi, biasanya dalam bentuk pegangan atau tuas. Perangkat ini membuka bagasi dari dalam bahkan jika bagasi sepenuhnya tertutup dan dalam posisi terkunci. Perangkat ini adalah termasuk desain keselamatan sehingga siapapun yang terkunci di bagasi bisa keluar dengan mencari pegangan darurat. Perangkat ini mungkin bisa dikatakan kunci terpisah yang membuka bagian dalam mobil.

c. *Automatic Release*

Mekanisme *Automatic Releases* adalah fitur dalam mobil yang digunakan pengemudi untuk membuka bagasi dari jarak jauh saat berada di dalam mobil, biasanya dengan menarik tuas. Ini mengaktifkan *solenoid*, yang menggunakan medan elektromagnetik untuk bergerak dan mengoperasikan perangkat. Solenoida juga digunakan dalam sistem kunci yang populer saat ini. Mekanisme pelepasan di dalam mobil pada awalnya dibuat untuk efisiensi, sehingga pengemudi bisa membuka bagasi sebelum keluar dan mengaksesnya, tanpa membuang waktu menggunakan kunci untuk membuka kunci bagasi.

C. Arduino uno

Arduino Uno adalah papan sirkuit berbasis mikrokontroler ATmega328. IC (+integrated circuit) ini memiliki 14 *input/output* digital (6 output untuk PWM), 6 analog input, resonator kristal keramik 16 MHz, Koneksi USB, soket adaptor, pin header ICSP, dan tombol reset. Hal inilah yang dibutuhkan untuk mendukung mikrokontroler, sederhana saja, hanya dengan menghubungkannya ke komputer dengan kabel USB atau sumber tegangan adaptor AC-DC dan *battery* (Prawoto, 2015). Selain itu dengan menggunakan ATmega328 ini jauh lebih murah dibanding dengan ATmega16. Sehingga digunakan ATmega328 ini untuk memproses *input* dan output pada alat ini. Seperti yang terlihat pada gambar 3 di bawah.



Gambar 3. Arduino Uno

(Sumber : <http://www.hobbytronics.co.uk/arduino-uno-r3>)

1. Fitur Arduino

Fitur yang dimiliki Arduino Uno sebagai berikut:

- a. Arduino Uno memiliki kemampuan tinggi dengan daya rendah.
- b. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16MHz.

- c. Memiliki kapasitas *Flash* memori 32 KByte, EEPROM 1 KByte, dan SRAM 2 KByte.
- d. Arduino memiliki fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya.
- e. Terdapat 14 *DigitalPin* yang dapat menjadi Input dan Output.
- f. Unit interupsi *Eksternal* dan *Internal*.
- g. *Port* USART untuk komunikasi serial.
- h. Fitur *peripheral*
- i. Tiga buah *Timer/Counter* dengan kemampuan pembandingan
- j. Dua buah *Timer/Counter* 8 bit dengan *Prescaler* terpisah dan *Mode Compare*.
- k. 1 (satu) buah *Timer/Counter* 16 bit dengan *Prescaler* terpisah, *Mode Compare* dan *Mode Capture*.
- l. 6 *channel PWM*.
- m. 6 *channel*, 10-bit ADC.
- n. *Byte-oriented Two-wired Serial Interface*.
- o. *Programmable* Serial USART.
- p. Antarmuka SPI.
- q. *Hardware* TWI.
- r. *Watchdog Timer* dengan *Oscillator Internal*.
- s. *Analog Comparator*.
- 2. *Input dan Output (I/O)*

Arduino Uno memiliki 14 buah digital pin yang dapat digunakan sebagai input atau output, dengan menggunakan fungsi `pinMode()`, `digitalWrite()`, dan

`digitalRead()`. Pin-pin tersebut bekerja pada tegangan 5V, dan setiap pin dapat menyediakan atau menerima arus 20mA, dan memiliki tahanan pull-up sekitar 20-50k ohm (secara *default* dalam posisi *disconnect*). Nilai maximum adalah 40mA, yang sebisa mungkin dihindari untuk menghindari kerusakan chip mikrokontroller.

Beberapa pin memiliki fungsi khusus :

- a. **Serial**, terdiri dari 2 pin : pin 0 (RX) dan pin 1 (TX) yang digunakan untuk menerima (RX) dan mengirim (TX) data serial.
- b. **External Interrupt**, yaitu pin 2 dan pin 3. Kedua pin tersebut dapat digunakan untuk mengaktifkan interrupt. Gunakan fungsi `attachInterrupt()`.
- c. **PWM**, Pin 3, 5, 6, 9, 10, dan 11 menyediakan output PWM 8-bit dengan menggunakan fungsi `analogWrite()`.
- d. **SPI** : Pin 10(SS), 11(MOSI), 12(MISO), dan 13(SCK) mendukung komunikasi SPI dengan menggunakan SPI Library.
- e. **LED** : Pin 13. Pada pin 13 terhubung built-in led yang dikendalikan oleh digital pin no 13.
- f. **TWI** : Pin A4(SDA) dan pin A5(SCL) yang mendukung komunikasi TWI dengan menggunakan *Wire Library*.

Arduino Uno memiliki 6 buah input analog, yang diberi tanda dengan A0, A1, A2, A3, A4, A5. Masing-masing pin analog tersebut memiliki resolusi 10 bits (jadi bisa memiliki 1024 nilai). Secara default, pin-pin tersebut diukur dari ground ke 5V, namun bisa juga menggunakan pin AREF dengan menggunakan fungsi `analogReference()`.

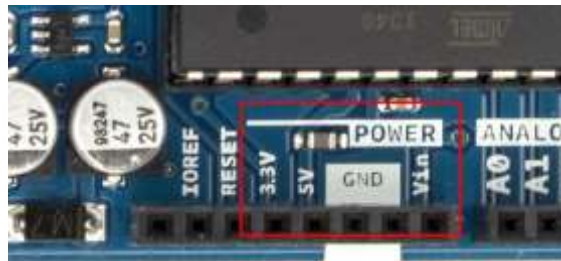
Beberapa input lainnya pada board ini adalah :

1. **AREF.** Sebagai referensi tegangan untuk input analog.
2. **Reset.** Hubungkan ke LOW untuk melakukan reset terhadap mikrokontroller.

Sama dengan penggunaan tombol reset yang tersedia.

3. Power Supply

Board Arduino Uno dapat ditenagai dengan power yang diperoleh dari koneksi kabel USB, atau via power supply eksternal. Pilihan power yang digunakan akan dilakukan secara otomatis. Seperti yang terlihat pada gambar 4



Gambar 4. Pin Power dan Ground

(Sumber : <http://www.caratekno.com/2015/07/pengertian-arduino-uno-mikrokontroler.html>)

External power supply dapat diperoleh dari adaptor AC-DC atau bahkan baterai, melalui jack DC yang tersedia, atau menghubungkan langsung GND dan pin Vin yang ada di board. Board dapat beroperasi dengan power dari external power supply yang memiliki tegangan antara 6V hingga 20V.

Beberapa pin power pada Arduino Uno :

- a. **GND.** Ini adalah ground atau negatif.
- b. **Vin.** Ini adalah pin yang digunakan jika anda ingin memberikan power langsung ke board Arduino dengan rentang tegangan yang disarankan 7V – 12V.
- c. **Pin 5V.** Ini adalah pin output dimana pada pin tersebut mengalir tegangan 5V yang telah melalui regulator.
- d. **3V3.** Ini adalah pin output dimana pada pin tersebut disediakan tegangan 3.3V yang telah melalui regulator.
- e. **IOREF.** Ini adalah pin yang menyediakan referensi tegangan mikrokontroller. Biasanya digunakan pada board shield untuk memperoleh tegangan yang sesuai, apakah 5V atau 3.3V.

4. Bahasa Pemrograman Arduino

Arduino IDE menggunakan bahasa pemrograman C++ dengan versi yang telah disederhanakan dengan bantuan pustaka-pustaka (*libraries*) Arduino, sehingga lebih mudah dalam belajar pemrograman.

Mikrokontroler diprogram menggunakan bahasa pemrograman arduino yang memiliki kemiripan *syntax* dengan bahasa pemrograman C. (Hendriono Dede, 2014)



Gambar 5. Tampilan Jendela Arduino.

Gambar 5 adalah tampilan jendela arduino yang di dalamnya terdapat beberapa baris program.

5. Struktur

Setiap program arduino (biasa disebut sketch) mempunyai dua buah fungsi yang harus ada, yaitu:

- a. `Void setup() { ... } ,`

Semua kode didalam kurung kurawal akan dijalankan hanya satu kali ketika program arduino dijalankan untuk pertama kalinya.

- b. `Void loop() { ... } ,`

Fungsi ini dijalankan setelah setup (fungsi void setup) selesai. Setelah dijalankan satu kali fungsi ini akan dijalankan kembali, dan lagi secara terus menerus sampai catu daya (*power*) dilepaskan.

6. Syntax

Berikut ini adalah elemen bahasa c yang dibutuhkan untuk format penulisan:

- a. `//` (komentar satu baris)

kadang diperlukan untuk memberi catatan pada diri sendiri apa arti dari kode-kode yang dituliskan. Cukup menuliskan dua buah garis miring dan apapun yang kita ketikan dibelakangnya akan diabaikan oleh program.

- b. `/*`(komentar banyak baris)

Jika anda mempunyai banyak catatan, maka hal tersebut dapat dituliskan pada beberapa baris sebagai komentar. Semua hal yang terletak di antara dua simbol tersebut akan diabaikan oleh program.

- c. `{ ... }` atau kurung kurawal

Digunakan untuk mendefinisikan kapan blok program mulai dan berakhir (digunakan juga pada fungsi dan pengulangan).

- d. `;` (titik koma),

Setiap baris kode harus diakhiri dengan tanda titik koma (jika ada titik koma yang hilang maka program tidak akan bisa dijalankan).

7. Variabel

Sebuah program secara garis besar didefinisikan sebagai instruksi untuk memindahkan angka dengan cara yang cerdas. Variabel inilah yang digunakan untuk memudahkannya.

- a. Int (integer), digunakan untuk menyimpan angka dalam 2 byte (16 bit). Tidak mempunyai angka desimal dan menyimpan nilai dari -23.768 s/d 32.767.
- b. Long, digunakan ketika integer tidak mencukupi lagi. Memakai 4 byte (32 bit) dari memori RAM dan mempunyai rentang nilai dari -2.147.648 s/d 2.147.483.647.

- c. Boolean, variabel sederhana yang digunakan untuk menyimpan nilai TRUE (benar) atau FALSE (salah). Sangat berguna karena hanya menggunakan 1 bit dari RAM.
- d. Float, digunakan untuk angka desimal (floating point). Memakai 4 byte (32 bit) dari RAM dan mempunyai rentang nilai dari $-3,4028235E+38$ s/d $3,4028235E+38$.
- e. Char (character), menyimpan 1 karakter menggunakan kode ASCII (misalnya 'A' = 65). Hanya memakai 1 byte (8 bit) dari RAM.

8. Operator Matematika

Operator yang digunakan untuk memanipulasi angka (bekerja seperti matematika yang sederhana).

- a. = (sama dengan), membuat sesuatu menjadi sama dengan nilai yang lain (misalnya: $x = 10 * 2$, $x = 20$).
- b. % (persen), menghasilkan sisa dari hasil pembagian suatu angka yang lain (misalnya : $12 \% 10$, ini akan menghasilkan angka 2).
- c. + (plus), merupakan operasi penjumlahan.
- d. - (minus), operasi pengurangan.
- e. * (asteris), operasi perkalian.
- f. / (garis miring), operasi pembagian.

9. Operator Pembandingan

Digunakan untuk mebandingkan nilai logika.

a. `==` (sama dengan)

misalnya: `12 == 10` adalah FALSE (salah) atau `12 == 12` adalah TRUE (benar).

b. `!=` (tidak sama dengan)

misalnya: `12 != 10` adalah TRUE (benar) atau `12 != 12` adalah FALSE (salah).

c. `<` (lebih kecil dari)

misalnya: `12 < 10` adalah FALSE (salah) atau `12 < 12` adalah FALSE (salah) atau `12 < 14` adalah TRUE (benar).

d. `>` (lebih besar dari)

misalnya: `12 > 10` adalah TRUE (benar) atau `12 > 12` adalah FALSE (salah) atau `12 > 14` adalah FALSE (salah).

10. Struktur Pengaturan

Program sangat tergantung pada pengaturan apa yang akan dijalankan berikutnya. Berikut ini adalah elemen dasar pengaturan.

a. `If ... else`, dengan format seperti berikut ini:

```
If(kondisi) { ... }  
Else if(kondisi) { ... }  
Else { ... }
```

Dengan struktur seperti diatas program akan menjalankan kode yang ada di dalam kurung kurawal jika kondisinya TRUE, dan jika tidak (FALSE) maka akan diperiksa apakah kondisi pada else if dan jika kondisinya FALSE maka kode pada else yang akan dijalankan.

b. `For`, dengan format penulisan sebagai berikut:

```
For(int i = 0; i < #pengulangan; i++) { ... }
```

Digunakan bila Anda ingin melakukan pengulangan kode program di dalam kurung kurawal beberapa kali, ganti #pengulangan dengan jumlah pengulangan yang diinginkan. Melakukan perhitungan ke atas (++) atau ke bawah (--).

11. Digital

a. `pinMode(pin, mode)`

Digunakan untuk menetapkan mode dari suatu pin, pin adalah nomor pin yang akan digunakan sebagai port dari 0 s/d 19 (pin analog 0 s/d 5 adalah 14 s/d 19). Mode yang bisa digunakan adalah INPUT atau OUTPUT.

b. `digitalWrite(pin, value)`

Ketika sebuah pin ditetapkan sebagai OUTPUT, pin tersebut dapat dijadikan HIGH (+5 volt) atau LOW (ground).

c. `digitalRead(pin)`

Ketika sebuah pin ditetapkan sebagai INPUT, maka Anda dapat menggunakan kode ini untuk mendapatkan nilai pin tersebut apakah HIGH (+5 volt) atau LOW (ground).

12. Analog

Arduino adalah mesin digital tetapi mempunyai kemampuan untuk beroperasi di dalam analog.

a. `Analog Write(pin, value)`

Beberapa pin pada arduino mendukung PWM (*pulse width modulation*) yaitu pin 3, 5, 6, 9, 10, 11. Ini dapat merubah pin hidup (on) atau mati (off) dengan sangat cepat sehingga membuatnya dapat berfungsi layaknya keluaran analog.

Value (nilai) pada format kode tersebut adalah angka antara 0 (0% duty cycle ~ 0 volt) dan 255 (100% duty cycle ~ 5 volt).

b. `Analog Read(pin)`

Ketika pin analog ditetapkan sebagai INPUT Anda dapat membaca keluaran voltase-nya. Keluarannya berupa angka antara 0 (untuk 0 volt) dan 1024 (untuk 5 volt).

D. Modul *Bluetooth* HC-05

Bluetooth Module HC-05 merupakan module komunikasi nirkabel pada frekuensi 2.4GHz dengan pilihan koneksi bisa sebagai *slave*, ataupun sebagai master. Sangat mudah digunakan dengan mikrokontroler untuk membuat aplikasi *wireless*. *Interface* yang digunakan adalah serial RXD, TXD, VCC dan GND. *Built in* LED sebagai indikator koneksi *bluetooth*.

Tegangan input antara 3.6 ~ 6V, jangan menghubungkan dengan sumber daya lebih dari 7V. Arus saat *unpaired* sekitar 30mA, dan saat *paired* (terhubung) sebesar 10mA. 4 pin interface 3.3V dapat langsung dihubungkan ke berbagai macam mikrokontroler (khusus Arduino, 8051, 8535, AVR, PIC, ARM, MSP430, etc.). Jarak efektif jangkauan sebesar 10 meter, meskipun dapat mencapai lebih dari 10 meter, namun kualitas koneksi makin berkurang. Seperti yang dapat dilihat pada gambar 6



Gambar 6. *Bluetooth* Module HC-05

(<http://www.geraicerdas.com/mikrokontroler/module/bluetooth-module-hc-05-detail>)

E. Sistem Operasi Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. Android, Inc. didirikan di Palo Alto, California, pada bulan Oktober 2003 oleh Andy Rubin (pendiri Danger), Rich Miner (pendiri Wildfire Communications, Inc.), Nick Sears (mantan VP T-Mobile), dan Chris White (kepala desain dan pengembangan antarmuka WebTV) untuk mengembangkan "perangkat seluler pintar yang lebih sadar akan lokasi dan preferensi penggunanya". Tujuan awal pengembangan Android adalah untuk mengembangkan sebuah sistem operasi canggih yang diperuntukkan bagi kamera digital, namun kemudian disadari bahwa pasar untuk perangkat tersebut tidak cukup besar, dan pengembangan Android lalu dialihkan bagi pasar telepon pintar untuk menyaingi Symbian dan Windows Mobile (iPhone Apple belum dirilis pada saat itu). Meskipun para pengembang Android adalah pakar-pakar teknologi yang berpengalaman, Android Inc. dioperasikan secara

diam-diam, hanya diungkapkan bahwa para pengembang sedang menciptakan sebuah perangkat lunak yang diperuntukkan bagi telepon seluler.

Tabel 1. Tingkatan Versi Android

Nama	Versi	Peluncuran
Cupcake	1.5	27 April 2009
Donut	1.6	15 September 2009
Eclair	2.0 – 2.1	26 Oktober 2009
Froyo	2.2 – 2.2.3	20 Mei 2010
Gingerbread	2.3 – 2.3.7	6 Desember 2010
Honeycomb	3.0–3.2.6	22 Pebruari 2011
Ice Cream Sandwich	4.0 – 4.0.4	18 Oktober 2011
Jelly Bean	4.1 – 4.3.1	9 Juli 2012
KitKat	4.4 – 4.4.4	31 Oktobe 2013
Lollipop	5.0 – 5.1.1	12 November 2014

(Sumber : <https://haiwiki.info/aplikasi/tingkatan-versi-android/>)

1. Android 1.0 (API level 1)

Android 1.0, Versi komersil dirilis pada 23 September 2008, dengan menggunakan device HTC DREAM.

2. Android 1.1

Pada maret 2009 google merilis Android versi 1.1 pada versi ini Android sudah dilengkapi dengan pembaharuan estis pada aplikasi, jam, alarm, speech search, pengirim pesan dan gmail, serta pembaharuan email.

3. Android 1.5 (cupcake)

Pada 27 April 2009, Android 1.5 dirilis, menggunakan kernel Linux 2.6.27. versi ini adalah rilis pertama yang secara resmi menggunakan nama kode berdasarkan makanan pencucui mulut, nama yang kemudian digunakan untuk semua ver4si rilis selanjutnya. Pembaruan pada versi ini termasuk beberapa fitur baru dan perubahan UI.

4. Android 1.6 (donut)

Android Donut menampilkan proses pencarian yang lebih baik dibanding sebelumnya, penggunaan baterai indikator dan kontrol applet VPN. Fitur lainnya adalah galeri yang memungkinkan pengguna untuk memilih foto yang akan dihapus; kamera, camcorder dan galeri yang dintegrasikan; CDMA / EVDO, 802.1x, VPN, Gestures, dan Text-to-speech engine; kemampuan dial kontak; teknologi text to change speech (tidak tersedia pada semua ponsel pengadaan resolusi VWGA).

5. Android 2.0/2.1 (Éclair)

Android Éclair menambahkan perubahan yang dilakukan pada pengoptimalan hardware, peningkatan Google Maps 3.1.2, perubahan UI dengan browser baru dan dukungan HTML5, daftar kontak yang baru, dukungan flash untuk kamera 3,2 MP, digital Zoom, dan *Bluetooth* 2.1.

6. Android 2.2 (Froyo)

Perubahan android Froyo terhadap versi-versi sebelumnya antara lain adalah dukungan Adobe Flash 10.1, kecepatan kinerja dan aplikasi 2 sampai 5 kali lebih cepat, integrasi V8 JavaScript engine yang dipakai Google Chrome yang mempercepat kemampuan rendering pada browser, pemasangan aplikasi dalam SD Card, kemampuan WiFi Hotspot portabel, dan kemampuan auto update dalam aplikasi Android Market.

7. Android 2.3 (gingerbread)

Perubahan pada android Gingerbread antara lain peningkatan kemampuan permainan (gaming), peningkatan fungsi copy paste, layar antar muka (User 7

Interface) didesain ulang, dukungan format video VP8 dan WebM, efek audio baru (reverb, equalization, headphone virtualization, dan bass boost), dukungan kemampuan Near Field Communication (NFC), dan dukungan jumlah kamera yang lebih dari satu.

8. Android 3.0/3.1 (honeycomb)

Android Honeycomb dirancang khusus untuk tablet. Android versi ini mendukung ukuran layar yang lebih besar. User Interface pada Honeycomb juga berbeda karena sudah didesain untuk tablet. Honeycomb juga mendukung multi prosesor dan juga akselerasi perangkat keras (hardware) untuk grafis. Tablet pertama yang dibuat dengan menjalankan Honeycomb adalah Motorola Xoom. Perangkat tablet dengan platform Android 3.0 akan segera hadir di Indonesia. Perangkat tersebut bernama Eee Pad Transformer produksi dari Asus.

9. Android 4.0 (ice cream sandwich)

Android Ice Cream Sanwitch membawa fitur Honeycomb untuk *smartphone* dan menambahkan fitur baru termasuk membuka kunci dengan pengenalan wajah, jaringan data pemantauan penggunaan dan kontrol, terpadu kontak jaringan sosial, perangkat tambahan fotografi, mencari email secara offline, dan berbagi informasi dengan menggunakan NFC.

10. Android 4.1 (Jelly Bean)

Pada tahun 2012, android Jelly Bean dirilis untuk komputer tablet dan mendukung untuk digunakan pada sistem operasi PC atau Komputer. Sehingga rumornya kemunculan Android Jelly Bean ini untuk menyaingi rilis terbaru Windows 8 yang juga akan segera dirilis. Karena kita ketahui bersama

perbincangan versi Android sebelumnya yaitu Android Ice Cream Sandwhich pun masih hangat di telinga.

11. Android 4.2 (Jelly Bean API level 17)

Dirilis pada 13 november 2012, Jelly Bean versi 4.2 melengkapi kekurangan maupun bugs yang sering terjadi pada JB 4.1, seperti perbaikan bug pada aplikasi ‘people’, penambahan tampilan nirkabel (*miracast*), perbaikan aksesibilitas, VPN yang selalut terhubung dan lain – lain.

12. Android 4.4 (KitKat) 8

Dirilis pada 31 Oktober 2013, KitKat versi 4.4 memiliki antarmuka terbaru dengan status bar dan navigasi transparan pada layar depan, webviews berbasis *Chromium*, mendukung media komunikasi Infra merah yang memungkinkan devices bisa menjadi remote untuk smart tv.

13. Android 5.0 (Lollipop)

Pada android Lollipop google selaku pengembang membuat sebuah desain antarmuka terbaru yang dinamakan “Material Design”, serta mendukung 64bit ART compiler, dan menambahkan system keamanan yang bernama ‘*factory reset protection*’ yang berfungsi ketika *smartphone* hilang, ia tidak bisa direset ulang tanpa memasukkan id dan password akun google.

F. App Inventor

App Inventor adalah aplikasi web sumber terbuka yang awalnya dikembangkan oleh Google dan saat ini dikelola oleh Massachusetts Institute of Technology (MIT). App Inventor memungkinkan pengguna baru untuk memprogram komputer untuk menciptakan aplikasi perangkat lunak bagi sistem

operasi Android. App Inventor menggunakan antarmuka grafis, serupa dengan antarmuka pengguna pada Scratch dan StarLogo TNG, yang memungkinkan pengguna untuk men-drag-and-drop obyek visual untuk menciptakan aplikasi yang bisa dijalankan pada perangkat Android. Dalam menciptakan App Inventor, Google telah melakukan riset yang berhubungan dengan komputasi edukasional dan menyelesaikan lingkungan pengembangan *online* Google. Seperti yang terlihat pada gambar 7



Gambar 7. Tampilan App Inventor pada Web Browser

G. Relay

Menurut Kho (2017) Modul relay pada dasarnya adalah saklar (switch) yang menyambungkan atau memutus kontak tegangan sambung secara mekanik jika diberi tegangan listrik maka relay akan bekerja dan relay akan langsung menutup (terhubung), jika relay tidak mendapatkan tegangan maka relay tidak dapat beroperasi (terputus). Karena relay bersifat normali close (NC) dan normali (NO).

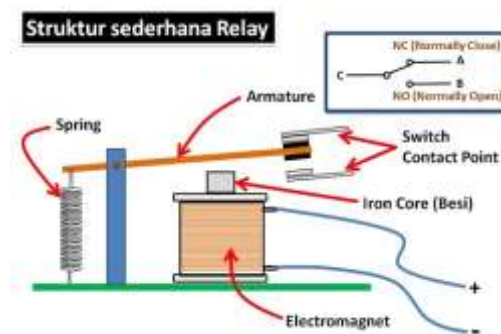
1. Relay 12 volt



Gambar 8. Relay 12 Volt

(Sumber: catalog.hella.com/electrics/relays.html)

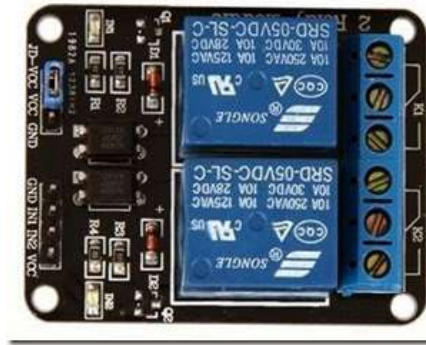
Relay memiliki 2 komponen utama yaitu electromagnet (coil) dan mekanikal (seperangkat kontak saklar/switch). Relay memiliki bagian-bagian mekanik dan elektronik yang dapat dilihat pada gambar 9.



Gambar 9. Struktur Sederhana Relay

(Sumber: Kho, 2015)

2. Modul Relay 2 Channel



Gambar 10. Modul Relay 2 Channel

(Sumber: <https://forum.arduino.cc/index.php?topic=281634.0>.)

Gambar 10 merupakan modul relay yang memiliki 2 channel untuk mengendalikan 2 alat. Adapun spesifikasi dari modul relay 2 channel, sebagai berikut:

- a. System power supply positive
- b. IN1 – IN2: relay control ports
- c. Active low, relay signal input voltage range: 0 – 5V
- d. Relay maksimum output: DC 30V/10A, AC 250V/10A
- e. Energization status indicator light, release status LED is off
- f. Relay of quality, SPDT. A common terminal, a normally open, a normally close
- g. Dimension: 50.6mm x 38.8mm x 19.3mm

H. Hidrolik

Menurut Rabie (2009) disebutkan bahwa energi cairan hidrolik dalam sistem hidrolik akan terjadi saat hidrolik sistem tenaga transfer ke tenaga mekanik. Namun menurut Trevor(1996) Umumnya, perbedaan utama antara hidrodinamik dan hidrostatik bersifat hidrodinamik berkaitan dengan fluida bergerak dimana gaya diproduksi dengan gerakan sedangkan hidrostatik berhubungan dengan cairan stasioner dimana gaya diproduksi dengan gerakan.

1. Sistem power hidrolik

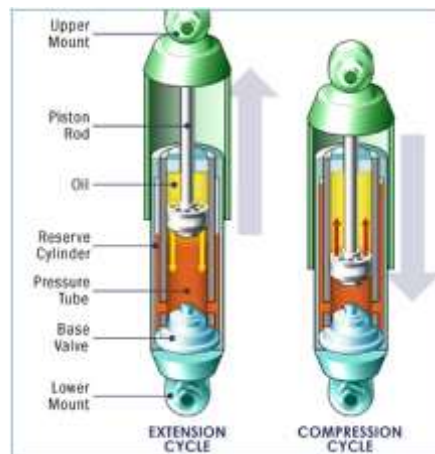
Hidrolik melibatkan gerak cairan ketika cairan berdampak pada benda dan melepaskan energinya untuk melakukan tekanan. Sistem tenaga hidrodinamika terjadi ketika energi kinetik cairan hidrolik meningkat. Menurut Rabie (2009) Hidrodinamik sistem tenaga listrik dapat berupa kopling hidrolik dan konverter torsi dalam sistem hidrolik. Perancangan struktur piranti kopling hidrolik berada dalam area tertutup seperti sekitarnya Dengan dinding memungkinkan cairan bisa lewat dengan cepat ke bagian lain.

Komponen dalam kopling hidrolik adalah turbin dan pompa sentrifugal. Daya transfer dari *input* ke *output shaft*. Pertama, pompanya drive diputar dan memungkinkan cairan seperti minyak melewati turbin. Ketika Minyak melewati turbin, tabrakan akan terjadi karena energi kinetik hilang. Menurut Newnes (2005) hidrostatik terjadi dimana cairan bertekanan. Kekuatannya adalah Transfer ketika energi tekanan cairan meningkat dianggap sebagai tenaga hidrostatik sistem. Ini bisa digunakan di area industri, peralatan bergerak, ruang udara dan banyak lagi. Sistem tenaga hidrodinamik terhubung ke penggerak utama dan pompa sedangkan

motor hidrolik terhubung ke beban. Komponen dalam Sistem tenaga hidrodinamik dapat menghasilkan aliran ke motor hidrolik.

2. Tipe komponen hidrolik

Meski sistem hidrolik bisa diaplikasikan di berbagai aplikasi, tetapi Jumlah dan susunan dari komponen akan berbeda dan tergantung aplikasinya. Komponen dasar di sistem hidrolik adalah *reservoir*, pompa hidrolik, aktuator hidrolik, katup kontrol, perpipaan dan cairan hidrolik. Menurut Mobley (2000) masing-masing komponen terdiri dari fungsi tertentu. Seperti yang terlihat pada gambar 11



Gambar 11. komponen Hirolik

(sumber: <https://auto.howstuffworks.com/car-suspension2.htm>)

3. Reservoir

Biasanya, reservoir adalah menyimpan cairan seperti minyak mineral, air dan cairan jenis lainnya. Wadah tertutup Reservoir tidak hanya menyimpan cairan namun bisa juga melibatkan proses filtrasi cairan sebelum melewati komponen lain dalam sistem. Menurut Vaxjo (2007) salah satu cara kerja terpenting adalah reservoir menghilangkan gelembung udara yang dihasilkan oleh pompa hidrolik.

I. Motor Dinamo Starter

Sistem starter adalah bagian dari sistem pada kendaraan untuk memberikan putaran awal bagi engine agar dapat menjalankan siklus kerjanya. Dengan memutar fly wheel, engine mendapat putaran awal dan selanjutnya dapat bekerja memberikan putaran dengan sendirinya melalui siklus pembakaran pada ruang bakar. Engine yang digunakan dalam kendaraan berat komersial, khususnya engine diesel, memerlukan starter dengan output yang jauh lebih tinggi dari pada kendaraan ringan pada umumnya. Starter merupakan alat untuk mengubah energi listrik menjadi energi mekanis dan outputnya dalam kilowatt akan selalu lebih kecil dari inputnya.

Salah satu cara untuk meningkatkan jumlah daya output pada motor starter kendaraan berat adalah merancang motor starter dengan tegangan yang lebih tinggi. Mengingat daya berbanding lurus dengan tegangan dan arus, maka apabila input arus sama, output akan meningkat ketika tegangan yang diterapkan meningkat.

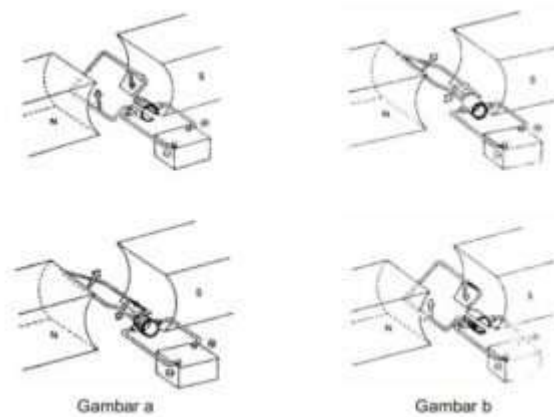


Gambar 12. Motor *Dynamo Starter*

(sumber: <http://totalotomotif.com/prinsip-kerja-motor-starter/>)

Prinsip kerja Motor starter satu siklus dengan kumparan anker tunggal dijelaskan sebagai berikut Arus listrik mengalir dari positif baterai → sikat positif komutator → sikat negatif → baterai.

1. Sisi kumparan arus menjauhi kita membentuk medan magnet dengan garis gaya magnet searah putaran jarum jam.
2. Medan magnet yang timbul diantara kutup-kutub, magnet saling berinteraksi dengan medan magnet yang timbul pada kumparan menghasilkan gaya magnet yang mengarah kebawah.



Gambar 13. Perputaran Motor *Dynamo Starter* Satu Siklus

(sumber: <http://totalotomotif.com/prinsip-kerja-motor-starter/>)

3. Sisi kumparan dengan arus mendekati kita membentuk medan magnet, dengan garis gaya magnet berlawanan arah putaran jarum jam.
4. Medan magnet yang timbul antara kutub-kutub magnet saling berinteraksi dengan medan magnet pada kumparan dan menghasilkan gaya magnet mengarah keatas. Akibat dari arah kedua gaya magnet yang berlawanan tersebut maka anker akan berputar setengah putaran searah jarum terlihat pada gambar 13.

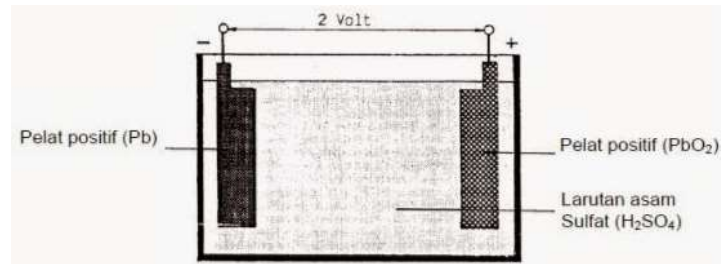
Bila arah arus pada kumparan yang memotong kutub magnet diarahkan hanya satu arah melalui lamel komutator, maka akan menghasilkan putaran motor yang teratur secara terus menerus atau kontinyu. Torsi yang terjadi akan tergantung pada kuat medan magnet, dan panjang kumparan yang berada dalam medan magnet. Dalam motor yang sebenarnya terdapat beberapa set atau pasangan kumparan yang disebut armature untuk menjamin putaran motor yang lebih teratur seperti yang terlihat pada gambar 12.

J. Baterai Aki Mobil

Baterai adalah alat untuk menyimpan sumber dari tenaga listrik dengan melalui proses elektrokimia sehingga sumber dari tenaga listrik dapat diubah menjadi tenaga kimia dan sebaliknya tenaga kimia menjadi tenaga listrik.

Fungsi baterai adalah untuk memberikan sumber tenaga listrik yang cukup pada sebuah peralatan misalnya untuk menghidupkan mobil (starter) serta melayani proses pada sistem pengapian hingga melayani penerangan lampu dan kebutuhan lainnya pada mobil atau motor.

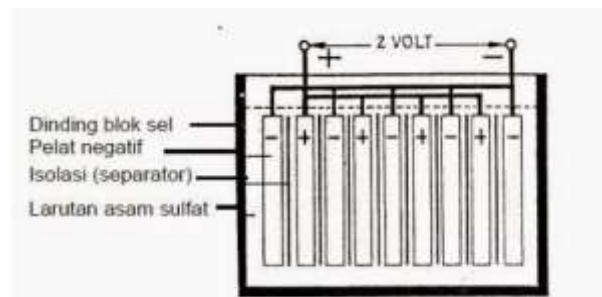
- a. Konstruksi sebuah sel Baterai terdiri dari beberapa sel dan setiap sel terdiri dari pelat positif dan pelat negatif dan sel ini dibuat dari pelat logam timbel berpori, dengan maksud dan tujuan untuk mempermudah reaksi kimia pada permukaan berpori tersebut sedangkan bahan aktif dari pelat positif adalah timbel dioksida (PbO_2) berwarna coklat dan untuk pelat negatif adalah timbel (Pb) berwarna abu – abu. Seperti yang terlihat pada gambar 14.



Gambar 14. Kontruksi Sel Aki

(sumber: <http://www.dasarotomotif.info/2014/08/fungsi-dan-konstruksi-baterai-aki.html>)

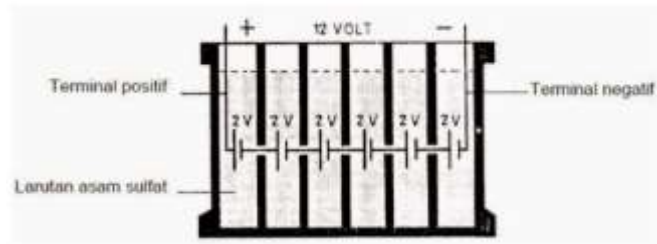
- b. Konstruksi blok sel Batang penghubung sel-sel adalah pelat – pelat yang tergabung di dalam blok blok sel dan pelat positif dibatasi oleh isolasi (separator) yang terbuat dari ebonit atau pelastik kemudian blok – blok sel ini dimasukkan dalam blok baterai yang diisi larutan asam sulfat (H_2SO_4) serta setiap blok sel menghasilkan tegangan sebesar 2 Volt seperti gambar 15.



Gambar 15. Kontruksi Block Sel

(sumber: <http://www.dasarotomotif.info/2014/08/fungsi-dan-konstruksi-baterai-aki.html>)

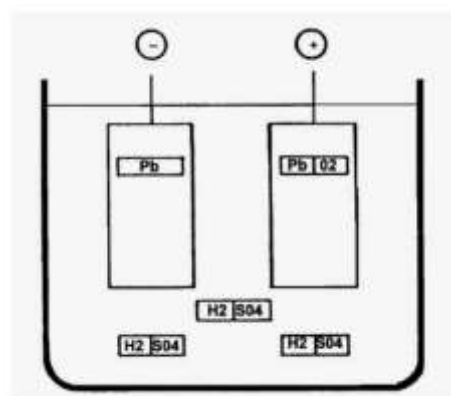
- c. Hubungan blok sel Tujuan dari menghubungkan blok – blok sel secara seri adalah untuk memperoleh tegangan yang lebih tinggi misalnya untuk memperoleh tegangan 12 Volt, baterai membutuhkan 6 blok sel yang masing – masing bertegangan 2 Volt seperti yang terlihat pada gambar 16.



Gambar 16. Hubungan Block Sel

(sumber: <http://www.dasarotomotif.info/2014/08/fungsi-dan-konstruksi-baterai-aki.html>)

- d. Proses elektrokimia Aki mobil :
 - a. Baterai dalam keadaan saat di isi air dan diberi arus penuh seperti yang terlihat pada gambar 17



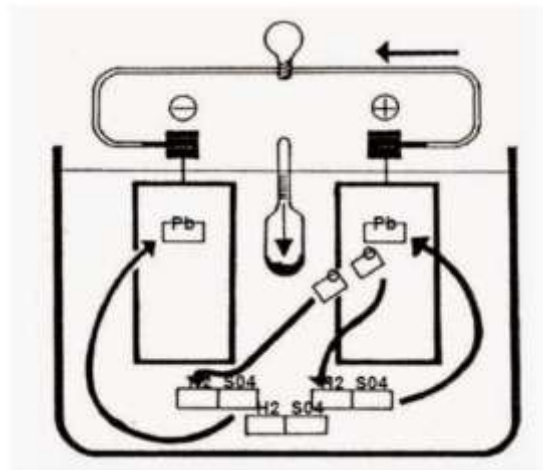
Gambar 17. Battery Terisi Penuh

(sumber: <http://www.dasarotomotif.info/2014/08/fungsi-dan-konstruksi-baterai-aki.html>)

Saat baterai berisikan air asam sulfat dan sudah dalam keadaan diberi arus penuh maka pada saat temperatur 20°C , berat jenis air baterai = 1,285 Kg/l

dan dalam keadaan ini bahan aktif pada pelat positif adalah timbel dioksida (PbO_2) berwarna coklat sedangkan pada pelat negatif timbel (Pb) berwarna abu-abu.

- b. Baterai dalam keadaan dipakai seperti yang terlihat pada gambar 18

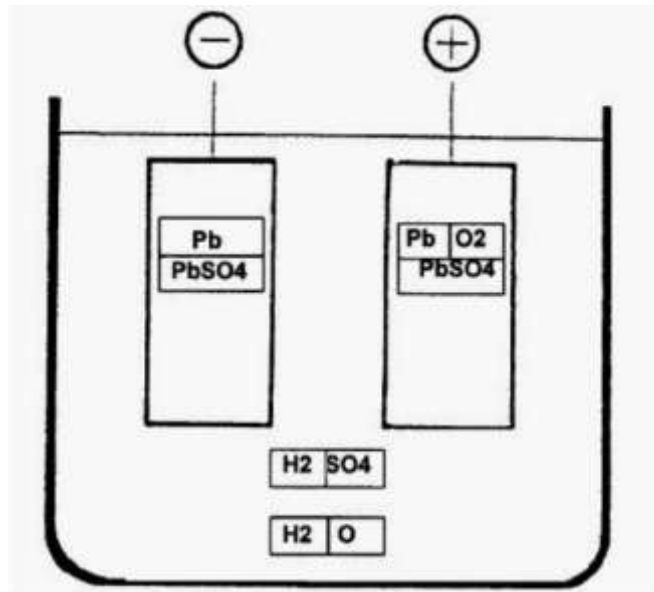


Gambar 18. Battery dalam Keadaan Dipakai

(sumber: <http://www.dasarotomotif.info/2014/08/fungsi-dan-konstruksi-baterai-aki.html>)

Saat baterai dalam keadaan dipakai maka Oksigen (O_2) yang berada pada pelat positif bereaksi dengan hidrogen (H) dan membentuk air (H_2O) dan pada waktu yang bersamaan membuat timbel Pb pada pelat positif bereaksi dengan sisa asam (SO_4) sehingga menjadi timbel sulfat (PbSO_4) sedangkan pada pelat negatif juga mengalami reaksi dengan sisa asam (SO_4) sehingga berubah menjadi timbel sulfat (PbSO_4) pula.

c. Baterai dalam keadaan kosong



Gambar 19. Battery dalam Keadaan Kosong

(sumber: <http://www.dasarotomotif.info/2014/08/fungsi-dan-konstruksi-baterai-aki.html>)

Saat baterai dalam keadaan kosong maka bila reaksi berlangsung terus menerus membuat arus listrik akan habis sehingga asam sulfat terbagi menjadi dua bagian, satu bagian membentuk air (H_2O) dan bagian lain bereaksi dengan bahan pelat dan membentuk timbel sulfat (PbO_4) yang menyebabkan berat jenis elektrolit menurun 1,08 kg/l. 4. Baterai dalam keadaan pengisian arus listrik Saat baterai dalam pengisian arus listrik membuat keadaan akan terbalik dikarenakan oksigen dalam asam baterai bereaksi dengan timbel pada pelat positif sehingga sisa asam terurai dari pelat – pelat dan bereaksi dengan hidrogen di dalam asam baterai seperti yang terlihat pada gambar 19.

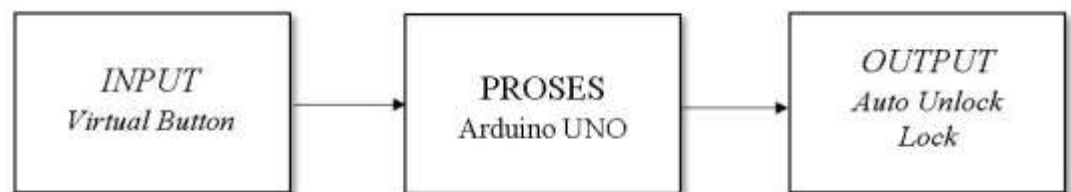
Maka hal ini akan menambah besarnya berat jenis air baterai yang mana penambahan ini akan berlangsung selama pengisian sampai berat jenis mencapai 1,285 kg/l sehingga dalam keadaan ini baterai telah terisi penuh.

BAB III

KONSEP PERANCANGAN

Pembuka kap dan bagasi mobil menggunakan *smartphone* berbasis *bluetooth* ini dirancang untuk membuka otomatis kap mesin dan pintu bagasi mobil dengan hidrolik sebagai penyangga yang membutuhkan beberapa langkah untuk merancang sistem ini yaitu diperlukan kebutuhan komponen, mendesain rancangan miniatur rumah, membuat sistem mekanik, pemrograman, pembuatan aplikasi *virtual button*, dan tahap terakhir melakukan pengujian alat sehingga didapatkan hasil alat dengan kinerja yang akurat sesuai dengan apa yang diharapkan.

Proses perancangan alat secara umum dapat dilihat pada gambar 20.



Gambar 20. Block Diagram Perancangan Alat Secara Umum

A. Identifikasi Kebutuhan

Identifikasi kebutuhan proyek akhir pembuka kap dan bagasi mobil menggunakan *smartphone* berbasis *bluetooth* dibagi menjadi 2 sebagai berikut:

1. *Hardware*
 - a. Catu Daya

Dibutuhkan Rangkaian regulator dan *switching* sebagai catu daya untuk menunjang pengoperasian alat.

- b. Tiga buah penyangga Hidrolik dibutuhkan untuk meyangga kap dan bagasi
- c. Dua buah motor *dynamo starter* dibutuhkan untuk membuka kap dan bagasi
- d. Satu buah relay 2 channel 5 volt dibutuhkan untuk mengirimkan sinyal perintah pada relay 12 volt
- e. Dua buah relay 1 channel 12 volt dibutuhkan untuk *switching* motor *dynamo starter*
- f. Satu buah Aki mobil dibutuhkan untuk catu daya
- g. Dua buah saklar dibutuhkan untuk *switching* manual
- h. Dua buah *button* manual dibutuhkan untuk tombol *emergency*
- i. *Bluetooth* HC05 dibutuhkan untuk menghubungkan *smartphone* dengan arduino
- j. Arduino UNO dibutuhkan untuk kendali utama

2. *Software*

- a. *Smartphone* dibutuhkan untuk menjalankan program
- b. App Inventor dibutuhkan untuk mendesain dan membuat program

B. Analisis Kebutuhan

Berdasarkan identifikasi kebutuhan di atas, maka diperoleh beberapa analisis kebutuhan terhadap sistem yang akan dirancang adalah sebagai berikut:

1. *Hardware*

a. Catu Daya

Dibutuhkan aki (accumulator) mobil untuk menghidupkan rangkaian pembuka kap dan bagasi mobil keluaran awalnya adalah 12V. Tegangan 12V digunakan untuk menghidupkan motor *dynamo starter* dan relay 12V.

Kemudian kelurgan aki diturunkan menggunakan IC regulator 78012 dan 7809 untuk menghidupkan Arduino.

b. Tiga buah penyangga Hidrolik

Adanya hidrolik pada proyek akhir ini selain untuk peyangga adalah untuk pembuka otomatis. Dibutuhkan 3 buah hidrolik nantinya akan dipasang 1 untuk kap mobil dan 2 untuk pintu bagasi.

c. Dua buah motor *dynamo starter*

Dibutuhkan 1 buah motor *dynamo starter* untuk menarik *lock door* kap serta *helper* pendorong pintu kap mesin, dan 1 buah lagi untuk menarik *lock door* pintu bagasi.

d. Dua buah saklar

Dibutuhkan dua buah saklar untuk menyalakan atau mematikan rangkaian Arduino serta rangkaian motor *dynamo starter*.

e. Dua buah *button* manual

Adanya *button* manual sebagai tombol *emergency* bagi kap dan bagasi ketika alat pembuka kap dan bagasi menggunakan *smartphone* tidak berjalan dengan baik.

f. *Bluetooth* HC05

Dibutuhkan *bluetooth* HC05 sebagai alat komunikasi utama antara *smartphone* dan Arduino.

g. Arduino UNO

Arduino uno digunakan sebagai pengendali utama yang akan melakukan pemrosesan data dari masukan untuk selanjutnya mengatur keputusan dari

jalannya alat yang akan dikirimkan ke bagian keluaran. Pemilihan arduino uno sebagai pengendali utama karena kemudahan dalam akses karena memiliki 14 buah pin digital yang dapat digunakan untuk jalur *input* maupun *output* yang sifatnya dapat diprogram ulang (*programmable*).

h. Relay 2 channel 5 volt

Relay 2 channel 5 volt digunakan untuk *switching lock* tahap pertama dimana relay ini hanya sebagai jembatan Arduino untuk menghidupkan relay 1 channel 2 volt.

i. Relay 1 channel 12 volt

Relay 1 channel 12 volt digunakan untuk *switching lock* langsung antara sumber aki mobil dengan motor *dynamo starter*

2. *Software*

a. *Smartphone*

Dibutuhkan *smartphone* adalah sebagai sarana utama tempat menjalankan program dari App Inventor.

b. App Inventor

App Inventor dibutuhkan untuk membuat program pembuka kap dan bagasi yang nantinya akan menggunakan *virtual button* sebagai pengambilan keputusan pengguna aplikasi.

Kebutuhan komponen baik utama maupun penunjang dapat dilihat pada tabel

2.

Tabel 2. kebutuhan alat dan bahan

No	Rangkaian	Komponen	Spesifikasi
1	Catu daya	aki	- Tegangan <i>input primer</i> 12V - Arus <i>output</i> 60 Ampere
2	<i>System Minimum</i>	Arduino	Arduino Uno
3	<i>Input</i>	Handphone	Android 5.1 (Lollipop)
4	<i>Output</i>	Modul Relay 2 channel	- Tegangan <i>input</i> 5V - 50.6mm x 38.8mm x 19.3mm
		Relay	Tegangan <i>input</i> 12V
		Motor Dinamo Starter	Tegangan input 12V
5	DII	Baut	3mm
		Box	21cm x 12cm x 6cm
		Kabel Pelangi	Secukupnya
		Hidrolik	60 cm
		Helper	Besi 20 cm
		IC 78012	- IC regulator <i>input</i> 12V - <i>Output</i> tegangan 12V
		IC 7809	- IC regulator <i>input</i> 12V - <i>Output</i> tegangan 9V
		Saklar	<i>Switch manual</i>
		Voltmeter mini	
		Button manual	

C. Perancangan sistem

Perancangan sistem pembuka kap dan bagasi mobil menggunakan *smartphone* berbasis *bluetooth* terdiri dari perancangan *hardware* dan *software*.

1. Hardware

Pada tugas akhir ini dibutuhkan perancangan *hardware* yang meliputi perancangan mekanik pada kap dan perancangan mekanik pada bagasi.

a. Rancangan Mekanik pada Kap mesin

Rancangan mekanik pada kap mesin memerlukan beberapa komponen yaitu, 1 buah hidrolik, 1 helper, dan 1 buah motor *dynamo starter*. Hidrolik di pasang pada tengah-tengah pintu kap seperti yang terlihat pada gambar 21.



Gambar 21. Pemasangan Hidrolik pada Kap

Hidrolik dipasang pada tengah-tengah pintu kap dimaksudkan agar tidak mengganggu kerja komponen lain mesin mobil dan menjadi titik tumpu beban angkat pintu kap yang terbilang cukup berat.

Untuk membantu hidrolik mengangkat pintu kap mesin, dibutuhkan sebuah *Helper*. *Helper* berfungsi untuk mendorong pintu kap ketika *lock* ditarik oleh motor *dynamo starter*. *Helper* sendiri memiliki lock tersendiri. Ketika *lock* untuk pintu bagasi dilepas, *lock* helper akan dilepas bersamaan.

b. Rancangan Mekanik pada Bagasi

Rancangan mekanik pada bagasi tidak jauh berbeda dengan kap, hanya saja bagasi tidak membutuhkan helper dan menggunakan 2 buah hidrolik untuk penyangga seperti yang terlihat pada gambar 22.



Gambar 22. Pemasangan Hidrolik pada Bagasi

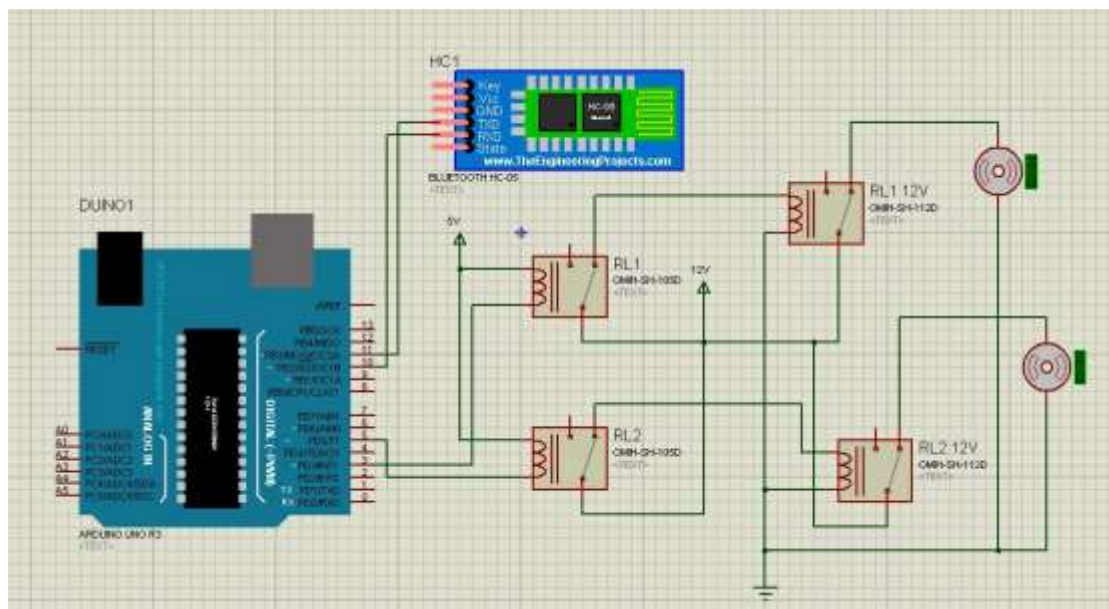
c. Arduino uno



Gambar 23. Arduino Uno R3
(Sumber : Luglio, 2014)

Pada gambar 23 adalah mikrokontroller Arduino Uno R3 yang dapat bekerja dan memproses datagram yang dikirimkan dari Aplikasi android hanya jika didalamnya sudah dimasukkan listing program, program yang dimasukkan kedalam Arduino dibuat dan diupload ke Arduino menggunakan tools

pemrograman Arduino IDE. Fungsi program disini antara lain yaitu, menginisialisasi pin -pin mana saja yang akan menjadi *output* atau *input*, mengubah datagram yang dikirim dari Android menjadi perintah Logika “*HIGH*” atau “*LOW*” yang akan mengaktifkan atau mematikan relay dan *output – output* pendukung lainnya, serta menginisialisasi alamat IP *bluetooth* yang akan menjadi alamat tujuan pengiriman datagram dari Android. Untuk lebih jelasnya dapat dilihat pada rangkaian konfigurasi komponen gambar 24.



Gambar 24. Rangkaian Konfigurasi Komponen

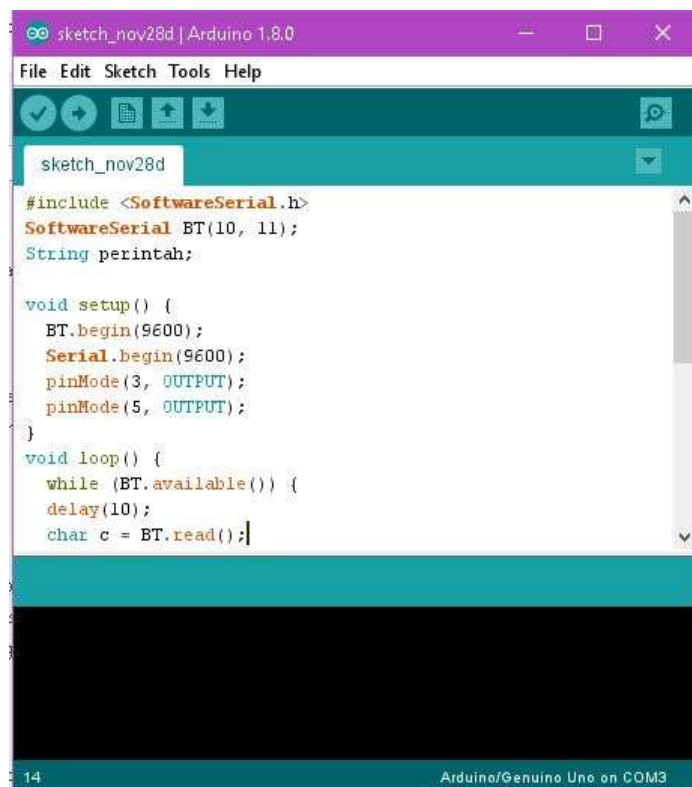
2. Software

Pada perancangan *software* dibagi menjadi 2 yaitu perancangan program arduino uno dan aplikasi *virtual button*.

a. Software arduino IDE

Dalam pemrograman Arduino ini sendiri menggunakan bahasa pemrograman C. Listing program Arduino ini dikenal dengan nama *sketch*. Dalam setiap *sketch*

memiliki dua buah fungsi penting yaitu “*void setup() {}*” dan “*void loop() {}*”. Pembuat program Arduino ini sendiri dimulai dengan menginisialisasi pin – pin mana saja yang akan digunakan oleh system, gambar 25 merupakan potongan coding pemrograman Arduino:



Gambar 25. Inisialisasi Pin pada Program Arduino UNO

Keterangan:

Buka_Kap untuk relay 1 menggunakan pin 3

Buka_Bagasi untuk relay 2 menggunakan pin 5

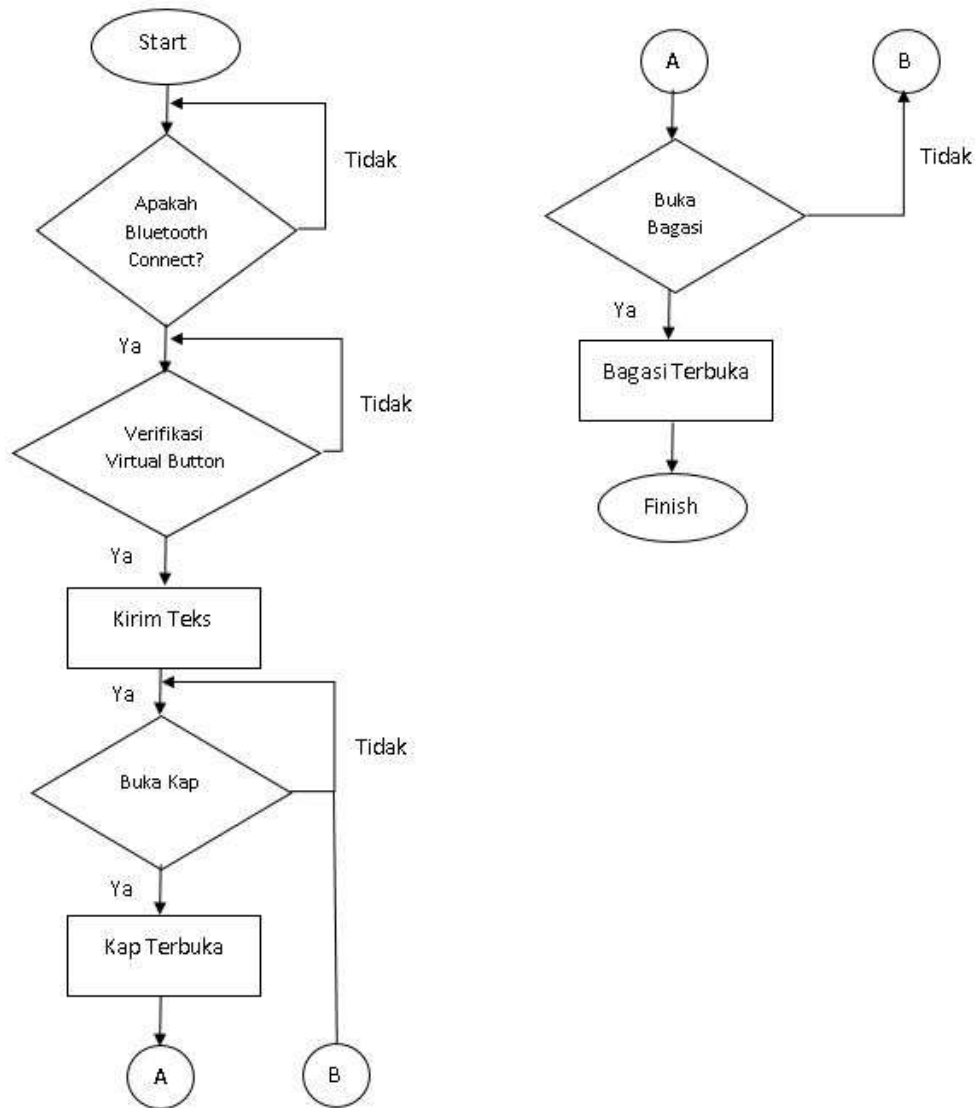
b. App inventor

Dalam pembuatan sistem ini, di gunakan ponsel atau *smartphone* dengan sistem operasi android untuk pengontrolannya. App inventor memungkinkan pengguna baru untuk memprogram komputer untuk menciptakan aplikasi

perangkat lunak bagi sistem operasi Android. App inventor menggunakan antarmuka grafis, serupa dengan antarmuka pengguna pada scratch dan star logo TNG, yang memungkinkan pengguna untuk men-drag-and-drop obyek visual untuk menciptakan aplikasi yang bisa dijalankan pada perangkat Android. Perangkat dengan sistem operasi android ini mengirim perintah untuk mengontrol hidup dan mati lampu dalam ruangan dan membuka atau menutup pintu gerbang, dengan *bluetooth* (yang sudah terhubung dengan perangkat dan program arduino) memancarkan sinyal untuk komunikasi data dan di pair oleh aplikasi. Uji coba dari perancangan dan pembuatan aplikasi android ini digunakan *smartphone* dengan sistem operasi android, dalam tugas akhir ini menggunakan tipe *smartphone* terbaru.

c. Diagram alur (*flowchart*)

Pada pembuatan proyek akhir ini, dibutuhkan suatu teknik perancangan yang mempunyai struktur yang baik, biasanya diawali dengan pembuatan diagram alur (*flowchart*). Diagram alur digunakan untuk menggambarkan terlebih dahulu apa yang harus dikerjakan sebelum mulai merancang atau membuat suatu system seperti yang akan dijelaskan dibawah ini. Gambar 26 adalah diagram alur (*flowchart*) dari aplikasi android dan gambar 27 diagram alur program Arduino yang akan dibuat.

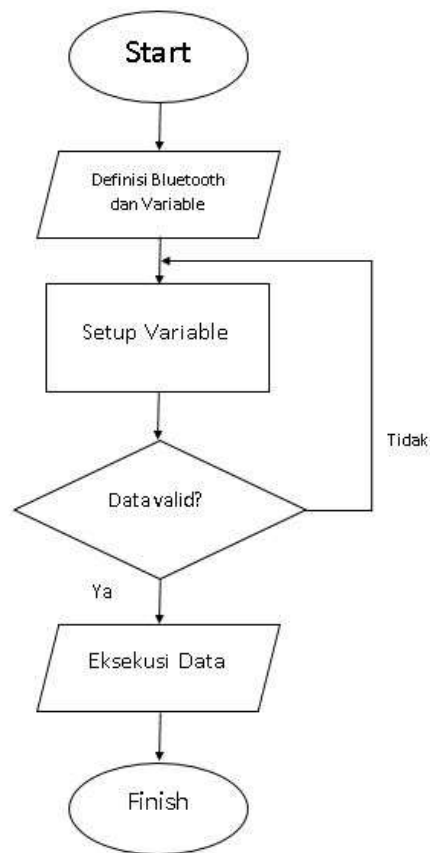


Gambar 26. Flow Chart Aplikasi Android

Alur algoritma aplikasi:

1. Start
2. Apakah *bluetooth connect*?
3. Jika tidak kembali ke step 2, jika ya verifikasi *virtual button*
4. Kirim teks
5. Buka Kap atau Bagasi, jika yam aka kap atau bagasi akan terbuka jika tidak kembali ke step 4

6. Finish

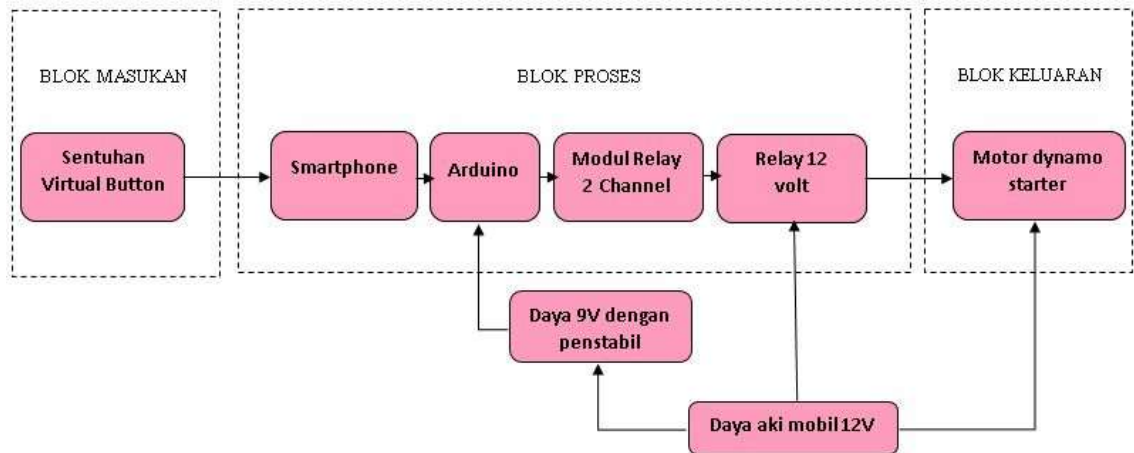


Gambar 27. Flowchart Program Arduino

Alur algoritma aplikasi:

1. Start
2. Mendefinisikan perangkat *bluetooth* dan membuat variable-variable.
3. Lalu membuat atau mensetting *bluetooth* dan variable-variablenya.
4. Kemudian membuat nilai-nilai untuk membaca dan mengecek data apakah sudah valid, jika tidak kembali ke step 3 jika ya
5. Mengeksekusi data dan variable
6. Finish

D. Blok diagram rangkaian



Gambar 28. Blok Diagram Rangkaian Keseluruhan

Gambar 28 merupakan blok diagram rangkaian sistem keseluruhan yang diimplementasikan pada pembuatan alat ini yang meliputi blok masukan, blok proses, blok keluaran, serta catu daya. Penjelasan bagian-bagian blok pada gambar 28.

1. Blok masukan

Pada bagian ini sentuhan pengguna pada *virtual button* dalam aplikasi adalah sumber masukan utama untuk mengendalikan pembuka kap dan bagasi otomatis. Pengguna hanya perlu menyentuh *virtual button* “Buka Kap” atau “Buka Bagasi”.

2. Blok Proses

Masukan sentuhan yang telah diproses pada program berubah menjadi teks yang dikirimkan kepada Arduino kemudian diterima relay untuk menyalakan motor *dynamo starter*.

3. Blok Keluaran

Pada blok ini terdapat 2 motor *dynamo starter* yang masing-masing terpasang 1 pada kap dan bagasi mobil.

E. Langkah Pembuatan Alat

Tugas akhir pembuka kap dan bagasi mobil menggunakan *smartphone* berbasis *bluetooth* ini diperlukan beberapa tahapan dalam pembuatannya adapun tahapannya sebagai berikut:

1. Pembuatan aplikasi android menggunakan *App Inventor*

Aplikasi virtual button dibutuhkan sebagai media masukan sentuhan melalui *smartphone* yang dibuat menggunakan *app inventor*.

Berikut merupakan langkah-langkah dalam pembuatan aplikasi virtual button:

1. Mendesain tampilan login aplikasi



Gambar 29. Tampilan Login Aplikasi

Gambar 29 menunjukkan window pembuatan screen login aplikasi menggunakan app inventor yang berisi *username* dan *password* aplikasi didesain sebagai hak akses dan menjaga keamanan aplikasi dari pengguna lain.

2. Mendesain tampilan Menu Pemilihan



Gambar 30. Tampilan Menu Pemilihan

Gambar 30 adalah menu pemilihan aplikasi pembuka kap dan bagasi yang didalamnya terdapat *button* pilih *bluetooth* untuk menyambungkan alat dengan aplikasi. Button Buka Kap untuk membuka kap secara otomatis dan Buka Bagasi untuk membuka bagasi secara otomatis.

Setelah seluruh desain aplikasi dibuat selanjutnya memprogram aplikasi dengan menekan tombol blocks yang ada di kanan atas aplikasi. Tampilan blok aplikasi app inventor dapat dilihat pada gambar 31:



Gambar 31. Tampilan Blok Aplikasi

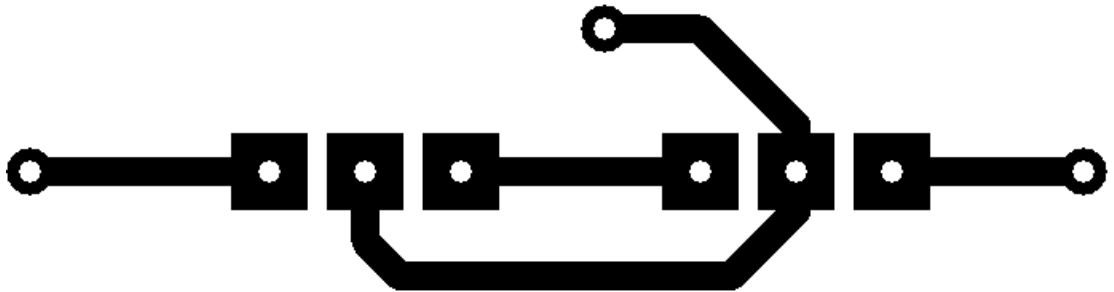
Gambar 31 menunjukkan tampilan blok pemrograman aplikasi pembuka kap dan bagasi. Terdapat beberapa baris program seperti *puzzle* yang sudah tersusun dengan cara *drag and drop* sesuai dengan *screen* yang akan diprogram gambar 30 merupakan contoh program dari *screen* kendali.

2. Merangkai komponen dengan project board untuk mencobanya
3. Membuat listing program menggunakan tools pemrograman arduino IDE.
4. Memasukan program ke arduino UNO.
5. Membuat rangkaian regulator

Rangkaian regulator ini berguna untuk menstabilkan sekaligus menurunkan tegangan aki dari 12 volt menjadi 9 volt. Rangkaian regulator ini menggunakan IC

78012 dan 7809. Langkah-langkah pembuatan rangkaian regulator adalah sebagai berikut:

- a. Membuat rangkaian PCB regulator



Gambar 32. Tampilan PCB Regulator

Seperti yang terlihat pada gambar 32 IC 78012 dirangkai seri dengan IC 7809.

- b. Menyiapkan papan PCB berlubang
- c. Menyolder kaki-kaki komponen
- d. Menyolder kaki keluaran rangkaian regulator ke kabel adaptor
- e. Menyolder kaki masukan rangkaian regulator ke kabel aki
6. Setelah rangkaian regulator selesai, kemudian rangkaian tersebut dipasang pada aki yang kemudian akan menjadi *powersupply* pada arduino UNO.
7. Melakukan pengujian koneksi antara modul *bluetooth* dengan *smartphone* android.
8. Merakit semua komponen rangkaian ke dalam box.
9. Memasang box alat kedalam mobil
10. Melakukan pengujian alat

F. Spesifikasi Alat

Spesifikasi pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth* yaitu:

1. Alat ini diujikan pada mobil Honda Civic *First Generation* tahun 1979 dengan berat penutup Kap mesin sebesar 10 kg dan berat pintu Bagasi sebesar 6 kg yang terbuat dari besi.
2. Satu box pengendali yang terbuat dari besi berdimensi 21cm x 12cm x 6cm terpasang pada *dashboard* kabin dalam mobil
3. Sumber tegangan yang digunakan pada motor *dynamo starter* 12 volt berasal dari aki mobil
4. Sumber tegangan yang digunakan pada Arduino adalah 9 volt berasal dari aki yang sudah diturunkan
5. *Smartphone* android versi lollipop sebagai uji coba.
6. Kendali sistem menggunakan arduino UNO R3.
7. *Virtual button* sebagai masukan pada alat
8. Dua saklar manual sebagai pemutus dan penyambung rangkaian Arduino serta motor *dynamo starter* dari aki untuk menghemat daya aki saat mobil tidak digunakan
9. Sistem kendali jarak menggunakan *bluetooth* seri HC-05
10. Motor *dynamo starter* sebagai pembuka *lock* kap dan bagasi
11. *Helper* sebagai alat bantu pembuka kap mobil

G. Pengujian Alat

1. Uji fungsional

Pada pengujian ini dilakukan dengan cara menguji setiap bagian dari masing-masing fungsi alat. Tujuan dari pengujian ini yaitu untuk mengetahui setiap bagian dari perangkat telah bekerja sesuai dengan fungsinya masing-masing atau tidak seperti menghubungkan *bluetooth* dengan *smartphone*, mengecek *powersupply* dan mengaktifkan alat

2. Uji unjuk kerja

Pada pengujian unjuk kerja alat ini dilakukan dengan cara membuka aplikasi pembuka kap dan bagasi pada *smartphone* android dan menghubungkan *bluetooth* antara *smartphone* android dengan alat apakah dapat terhubung atau tidak untuk melihat unjuk kerja alat. Hal-hal yang perlu di amati antara lain: apakah *smartphone* terhubung dengan alat atau tidak, pastikan *bluetooth* pada keduanya aktif.

H. Tabel Uji Alat

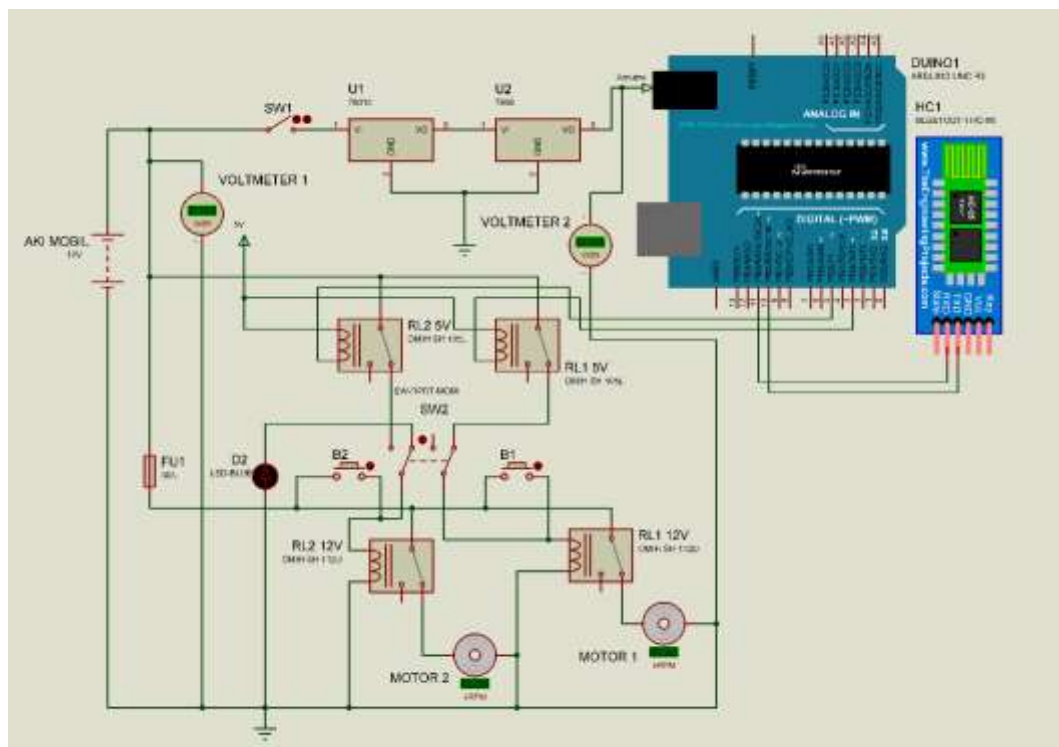
1. Uji Fungsional

Dalam tugas akhir ini diperlukan beberapa pengujian yaitu:

a. Pengujian catu daya aki 12V dan Regulator 9V

Pengujian rangkaian catu daya diperlukan agar masukan tegangan yang diberikan pada alat dapat sesuai dengan spesifikasi dari masing-masing komponen seperti arduino UNO membutuhkan tegangan 9 volt DC dan motor *dynamo starter* yang membutuhkan tegangan 12 volt. Pengujian dilakukan dengan mengukur keluaran dari *power supply* dan memasukan hasilnya pada

tabel 3, 4, 5 dan 6. Pengukuran rangkaian *output power supply* dilakukan sebanyak lima kali agar dapat diketahui besaran data ukur yang valid. Pengukuran tegangan power supply aki 12V dan Regulator 9V saat tanpa beban juga saat dengan beban ketika mesin mobil mati, juga ketika mesin mobil hidup saat tanpa beban saat dengan beban Arduino UNO, Motor *dynamo starter*, dan *relay*. Dengan titik pengukuran seperti gambar 33.



Gambar 33. Tampilan Titik Pengukuran Catu Daya

Tabel 3. Rencana Pengujian Catu Daya Tanpa Beban saat Mesin Mobil Mati

NO	Pengukuran pada	Pengukuran ke-	V-Out (volt)	V-Out Terukur (volt)	Selisih tegangan	<i>Error (%)</i>	Rata-rata <i>error (%)</i>
1	Catu daya aki 12V	1					
		2					
		3					
		4					
		5					
2	Rangkaian Regulator 9V	1					
		2					
		3					
		4					
		5					

Tabel 4. Rencana Pengujian Catu Daya dengan Beban saat Mesin Mati

NO	Pengukuran pada	Pengukuran ke-	V-Out (volt)	V-Out Terukur (volt)	Selisih tegangan	<i>Error (%)</i>	Rata-rata <i>error (%)</i>
1	Catu daya aki 12V	1					
		2					
		3					
		4					
		5					
2	Rangkaian Regulator 9V	1					
		2					
		3					
		4					
		5					

Tabel 5. Rencana Pengujian Catu Daya Tanpa Beban saat Mesin Mobil Menyala

NO	Pengukuran pada	Pengukuran ke-	V-Out (volt)	V-Out Terukur (volt)	Selisih tegangan	Error (%)	Rata-rata error (%)
1	Catu daya aki 12V	1					
		2					
		3					
		4					
		5					
2	Rangkaian Regulator 9V	1					
		2					
		3					
		4					
		5					

Tabel 6. Rencana Pengujian Catu Daya dengan Beban saat Mesin Menyala

NO	Pengukuran pada	Pengukuran ke-	V-Out (volt)	V-Out Terukur (volt)	Selisih tegangan	Error (%)	Rata-rata error (%)
1	Catu daya aki 12V	1					
		2					
		3					
		4					
		5					
2	Rangkaian Regulator 9V	1					
		2					
		3					
		4					
		5					

b. Pengujian *Bluetooth*

Pengujian *bluetooth* dimaksudkan agar dapat mengetahui jangkauan kinerja pada *bluetooth*. Pengujian dilakukan dengan menyambungkan modul *bluetooth* dengan *smartphone* dari jarak tertentu dan mengoperasikanya

dengan alat, sehingga diharapkan dapat mengetahui jarak maksimal kinerja *bluetooth*. Tabel rencana pengujian dapat dilihat pada tabel 7.

Tabel 7. Rencana Pengujian Jangkauan *Bluetooth*

NO.	Kondisi	Jarak	Hasil transmisi		Delay (detik)
			diterima	Ditolak	
1	Tanpa penghalang	1-10 meter			
		11-15 meter			
		16-20 meter			
		21-25 meter			
		26-30 meter			
		34 meter			
2	Ada penghalang	1-10 meter			
		11-15 meter			
		16-20 meter			
		21-25 meter			
		26-30 meter			
		34 meter			

c. Pengujian motor *dynamo starter*

Pengujian motor *dynamo starter* dimaksudkan untuk mengetahui apakah motor telah bekerja dengan baik atau belum pada saat mesin mobil sedang mati ataupun mesin mobil menyala. Pengujian dilakukan dengan menghubungkan motor dynamo starter pada aki 12V juga dihubungkan pada rangkaian Arduino untuk memberi masukan logika. Tabel rencana pengujian dapat dilihat pada tabel 8 dan 9.

Tabel 8. Rencana Pengujian Motor *Dynamo Starter* saat Mesin Mobil Mati

no	Motor pada	Pengukuran ke-	Kondisi logika	V-In (Volt)	Error (%)	keterangan
1	Kap Mobil	1	1			
		2	0			
		3	1			
		4	0			
		5	1			
		6	0			
2	Bagasi Mobil	1	1			
		2	0			
		3	1			
		4	0			
		5	1			
		6	0			

Tabel 9. Rencana Pengujian Motor *Dynamo Starter* saat Mesin Mobil Menyala

no	Motor pada	Pengukuran ke-	Kondisi logika	V-In (Volt)	Error (%)	keterangan
1	Kap Mobil	1	1			
		2	0			
		3	1			
		4	0			
		5	1			
		6	0			
2	Bagasi Mobil	1	1			
		2	0			
		3	1			
		4	0			
		5	1			
		6	0			

d. Pengujian modul *relay* 2 channel 5V dan *relay* 12V

Pengujian ini dimaksudkan untuk mengetahui apakah modul *relay* 2 channel 5V dan *relay* 12V telah bekerja dengan baik atau belum dalam keadaan mesin mobil mati ataupun hidup. Pengujian dilakukan dengan memberikan sinyal *input high* dan *low* dari pin arduino uno yang dihubungkan dengan pin pada modul *relay* 2 channel 5V. Tabel rencana pengujian dapat dilihat pada tabel 10.

Tabel 10. Rencana Pengujian modul relay 2 channel 5V dan relay 12V

No	Kondisi mesin mobil	Pengukuran ke-	Sinyal input	Kondisi relay 5V	Kondisi relay 12V	Tegangan V-in (Volt)		Keterangan
						Relay 5V	Relay 12V	
1	Mati	1	High					
			Low					
		2	High					
			Low					
		3	High					
			Low					
2	Hidup	1	High					
			Low					
		2	High					
			Low					
		3	High					
			Low					

e. Pengujian perintah *virtual button*

Pengujian ini dimaksudkan untuk mengetahui apakah aplikasi pembuka kap dan bagasi dalam *smartphone* dapat bekerja dengan baik meski berbeda *smartphone* dan juga berbeda versi android. Pengujian dilakukan dengan menginstal aplikasi pembuka kap dan bagasi pada *smartphone* beberapa responden dan mencoba membuka kap serta bagasi melalui *smartphone* responden. Tabel rencana pengujian dapat dilihat pada tabel 11.

Tabel 11. Rencana Pengujian Perintah *Virtual Button* Menggunakan Variasi

Sistem Operasi Android

No	<i>Virtual Button</i> pada	Hasil respon		Waktu delay system (detik)	Waktu respon rata-rata	Ket
		Kap	Bagasi			
1	Jelly Bean					
2	Kitkat					
3	Lollipop					
4	Marshmallow					
5	Nougat					

f. Pengujian *button* manual *emergency*

Pengujian ini dimaksudkan untuk mengetahui apakah tombol emergency B1 dan B2 yang dirancang sudah berjalan dengan baik atau belum. Seperti yang terlihat pada gambar 34.



Gambar 34. Tampilan *Button* Manual *Emergency*

Pengujian dilakukan sebanyak lima kali dengan selang waktu 30 menit antar pengujian. Tabel rencana pengujian dapat dilihat pada tabel 12.

Tabel 12. Rencana Pengujian *Button Manual Emergency*

No	Pengujian tombol pada	Pengujian ke-	Respon tombol	keterangan
1	Kap	1		
		2		
		3		
		4		
		5		
2	Bagasi	1		
		2		
		3		
		4		
		5		

2. Uji unjuk kerja

Pengujian unjuk kerja alat dilakukan dengan cara mengoperasikan sistem rangkaian alat yang telah dibuat sesuai dengan tujuannya. Adapun rencana pengujian unjuk kerja tabel 13 dan hasil dari unjuk kerja itu sendiri terdapat dalam lampiran 6 tabel 24.

Tabel 13. Rencana Pengujian Unjuk Kerja

No	Perintah virtual button	Tampilan	Respon perintah	keterangan
1	Buka kap			
2	Buka bagasi			

I. Pengoperasian Alat

Berikut ini adalah langkah pengoperasian alat pembuka kap dan bagasi menggunakan *smartphone* berbasis *bluetooth*:

1. Mengubungkan box rangkaian alat dengan aki mobil
2. Menghidupkan saklar rangkaian Arduino
3. Setelah beberapa saat hidupkan saklar untuk *relay* 12V dan motor *dynamo starter*
4. Membuka aplikasi pembuka kap dan bagasi yang telah terinstall pada *smartphone*
5. Memasukan username dengan 'admin' dan password dengan '1234'
6. Menekan masuk untuk login
7. Ketika muncul *box* dialog apakah ingin menyalakan *bluetooth*, pilih Allow
8. Ketika masuk pada menu pilihan tekan pilih *bluetooth* dan pilih *bluetooth HC-*

05

9. Menekan ‘Buka Kap’ jika ingin membuka kap atau menekan ‘Buka Bagasi’ bila ingin membuka bagasi
10. Alat akan merespon dengan *virtual button* yang anda tekan
11. Bila ingin mematikan kinerja alat agar aki mobil tidak habis, maka matikan terlebih dahulu saklar rangkaian *relay* 12V dan motor *dynamo starter*
12. Mematikan saklar rangkaian Arduino

BAB IV

PENGUJIAN DAN PEMBAHASAN

A. Hasil Pengujian

Pengujian alat untuk mengetahui kinerja dari masing-masing komponen yang sudah dirangkai sesuai dengan spesifikasinya. Hasil dari pengujian ini diharapkan dapat mampu menghasilkan data yang benar dan alat bekerja sesuai dengan fungsinya.

1. Pengujian Catu daya aki mobil 12V dan Rangkaian Regulator 9V

Tabel 14. Pengujian Catu Daya Tanpa Beban saat Mesin Mobil Mati

NO	Pengukuran pada	Pengukuran ke-	V-Out (volt)	V-Out Terukur (volt)	Selisih tegangan	Error (%)	Rata-rata Error (%)
1	Catu daya aki 12V	1	12	12,6	0,6	5	5,32%
		2	12	12,7	0,7	5,83	
		3	12	12,7	0,7	5,83	
		4	12	12,7	0,7	5,83	
		5	12	12,7	0,7	5,83	
2	Rangkaian Regulator 9V	1	9	8,69	0,31	3,4	3,4%.
		2	9	8,69	0,31	3,4	
		3	9	8,69	0,31	3,4	
		4	9	8,69	0,31	3,4	
		5	9	8,69	0,31	3,4	

Tabel 15. Pengujian Catu Daya dengan Beban saat Mesin Mati

NO	Pengukuran pada	Pengukuran ke-	V-Out (volt)	V-Out Terukur (volt)	Selisih tegangan	Error (%)	Rata-rata Error (%)
1	Catu daya aki 12V	1	12	12,4	0,4	3,3	5,7%
		2	12	12,7	0,7	5,83	
		3	12	12,7	0,7	5,83	
		4	12	12,7	0,7	5,83	
		5	12	12,7	0,7	5,83	
2	Rangkaian Regulator 9V	1	9	8,68	0,32	3,56	3,7%.
		2	9	8,67	0,33	3,67	
		3	9	8,66	0,34	3,78	
		4	9	8,66	0,34	3,78	
		5	9	8,66	0,34	3,78	

Tabel 16. Pengujian Catu Daya Tanpa Beban saat Mesin Mobil Menyala

NO	Pengukuran pada	Pengukuran ke-	V-Out (volt)	V-Out Terukur (volt)	Selisih tegangan	Error (%)	Rata-rata Error (%)
1	Catu daya aki 12V	1	12	13,3	1,3	10,89	15,68%.
		2	12	13,6	1,6	13,33	
		3	12	13,9	1,9	15,83	
		4	12	14,1	2,1	17,5	
		5	12	14,5	2,5	20,83	
2	Rangkaian Regulator 9V	1	9	8,69	0,31	3,4	3,4%
		2	9	8,69	0,31	3,4	
		3	9	8,69	0,31	3,4	
		4	9	8,69	0,31	3,4	
		5	9	8,69	0,31	3,4	

Tabel 17. Pengujian Catu Daya dengan Beban saat Mesin Menyala

NO	Pengukuran pada	Pengukuran ke-	V-Out (volt)	V-Out Terukur (volt)	Selisih tegangan	Error (%)	Rata-rata Error (%)
1	Catu daya aki 12V	1	12	13,3	1,3	10,89	14,8%
		2	12	13,5	1,5	12,5	
		3	12	13,7	1,7	14,17	
		4	12	14	2	16,67	
		5	12	14,4	2,4	20	
2	Rangkaian Regulator 9V	1	9	8,67	0,33	3,67	3,8%.
		2	9	8,66	0,34	3,78	
		3	9	8,65	0,35	3,89	
		4	9	8,65	0,35	3,89	
		5	9	8,65	0,35	3,89	

2. Pengujian jarak *bluetooth*

Tabel 18. Pengujian Jangkauan *Bluetooth*

NO.	Kondisi	Jarak	Hasil transmisi		Delay (detik)
			diterima	Ditolak	
1	Tanpa penghalang	1-10 meter	√		0
		11-15 meter	√		0
		16-20 meter	√		0,5
		21-25 meter	√		5
		26-30 meter	√		8
		34 meter		√	-
2	Ada penghalang	1-10 meter	√		0
		11-15 meter	√		0
		16-20 meter	√		0,7
		21-25 meter	√		6
		26-30 meter	√		9
		34 meter		√	-

3. Pengujian motor *dynamo starter*

Tabel 19. Pengujian Motor *Dynamo Starter* saat Mesin Mobil Mati

no	Motor pada	Pengukuran ke-	Kondisi logika	V-In (Volt)	Error (%)	keterangan
1	Kap Mobil	1	1	12,6	-	Hidup
		2	0	0	-	Mati
		3	1	12,7	-	Hidup
		4	0	0	-	Mati
		5	1	12,7	-	Hidup
		6	0	0	-	Mati
2	Bagasi Mobil	1	1	12,6	-	Hidup
		2	0	0	-	Mati
		3	1	12,7	-	Hidup
		4	0	0	-	Mati
		5	1	12,7	-	Hidup
		6	0	0	-	Mati

Tabel 20. Pengujian Motor *Dynamo Starter* saat Mesin Mobil Menyala

no	Motor pada	Pengukuran ke-	Kondisi logika	V-In (Volt)	Error (%)	keterangan
1	Kap Mobil	1	1	13,3	-	Hidup
		2	0	0	-	Mati
		3	1	13,7	-	Hidup
		4	0	0	-	Mati
		5	1	14,4	-	Hidup
		6	0	0	-	Mati
2	Bagasi Mobil	1	1	13,3	-	Hidup
		2	0	0	-	Mati
		3	1	13,7	-	Hidup
		4	0	0	-	Mati
		5	1	14,4	-	Hidup
		6	0	0	-	Mati

4. Pengujian modul *relay* 2 channel 5V dan *relay* 12V

Tabel 21. Pengujian modul relay 2 channel 5V dan relay 12V

No	Kondisi mesin mobil	Pengukuran ke-	Sinyal input	Kondisi relay 5V	Kondisi relay 12V	Tegangan V-in (Volt)		Keterangan
						Relay 5V	Relay 12V	
1	Mati	1	High	Hidup	Hidup	4,92	12,6	Benar
			Low	Mati	Mati	0	0	Benar
		2	High	Hidup	Hidup	4,93	12,7	Benar
			Low	Mati	Mati	0	0	Benar
		3	High	Hidup	Hidup	4,95	12,7	Benar
			Low	Mati	Mati	0	0	Benar
2	Hidup	1	High	Hidup	Hidup	4,9	13,3	Benar
			Low	Mati	Mati	0	0	Benar
		2	High	Hidup	Hidup	4,86	13,7	Benar
			Low	Mati	Mati	0	0	Benar
		3	High	Hidup	Hidup	4,85	14,4	Benar
			Low	Mati	Mati	0	0	Benar

5. Pengujian perintah *virtual button*

Tabel 22. Pengujian Perintah *Virtual Button* Menggunakan Variasi Sistem Operasi

Android

No	Virtual Button	Hasil respon		Waktu delay system (detik)	Waktu respon rata-rata	Ket
		Kap	Bagasi			
1	Jelly Bean	Terbuka	Terbuka	0,1	0,15	Sesuai
2	Kitkat	Terbuka	Terbuka	0,15		Sesuai
3	Lollipop	Terbuka	Terbuka	0,1		Sesuai
4	Marshmallow	Terbuka	Terbuka	0,1		Sesuai
5	Nougat	Terbuka	Terbuka	0,3		Sesuai

6. Pengujian *button manual emergency*

Tabel 23. Pengujian *Button Manual Emergency*

No	Pengujian tombol pada	Pengujian ke-	Respon tombol	keterangan
1	Kap	1	Terbuka	Benar
		2	Terbuka	Benar
		3	Terbuka	Benar
		4	Terbuka	Benar
		5	Terbuka	Benar
2	Bagasi	1	Terbuka	Benar
		2	Terbuka	Benar
		3	Terbuka	Benar
		4	Terbuka	Benar
		5	Terbuka	Benar

7. Pengujian secara keseluruhan

Pada pengujian ini seluruh rangkaian sistem termasuk komponen *hardware* dan *software* dioperasikan. Seluruh komponen dirangkai dalam satu kesatuan dan komponen-komponen tersebut telah diuji satu persatu sebelum digabungkan. Tujuan dari pengujian ini adalah untuk mengetahui alat bekerja baik dan benar sesuai yang diharapkan. Adapun hasil unjuk kerja secara keseluruhan dapat dilihat pada lampiran 6 tabel 24.

B. Pembahasan

Berdasarkan hasil pengamatan dan pengujian beberapa rangkaian dan komponen pada proyek akhir ini, maka dapat disimpulkan bahwa seluruh rangkaian dapat bekerja dengan baik dan sesuai dengan fungsi dari masing-masing komponen. Pada pengujian pengukuran rangkaian sistem terdapat sedikit perbedaan dengan adanya selisih dari hasil pengukuran dengan apa yang diperoleh dari teori *datasheet* komponen. Perbedaan tersebut dipengaruhi oleh banyak faktor seperti alat ukur yang digunakan, nilai komponen yang tidak sesuai dengan labelnya, toleransi nilai komponen dari pabrik yang memproduksi komponen, dan kurang telitinya dalam pengukuran.

1. Analisis pengujian catu daya aki mobil 12V dan rangkaian regulator 9V

Catu daya system pembuka kap dan bagasi ini menggunakan aki mobil karena aki adalah sumber listrik semua komponen di mobil dan juga mesin pada mobil terdapat alternator untuk pengisian aki. Jadi penggunaan aki sebagai catu daya memungkinkan rangkaian untuk mendapatkan sumber listrik dari dalam mobil secara terus-menerut, selama mesin mobil menyala.

Pengujian dilakukan dengan empat versi yaitu untuk yang pertama catu daya diuji tanpa diberi beban pada *output* sebanyak lima kali percobaan ketika mesin mobil mati, kedua dengan memberikan beban berupa semua komponen yang digunakan pada alat ini ketika mesin mobil dalam keadaan mati, ketiga dengan catu daya diuji tanpa diberi beban sebanyak lima kali percobaan ketika mesin mobil menyala, keempat dengan memberikan beban berupa semua komponen yang digunakan pada alat ini ketika mesin mobil dalam keadaan

menyala. Berdasarkan pengujian tersebut, maka dihasilkan pengukuran sebagai berikut:

a. Tanpa beban ketika mesin mobil mati

Pengukuran daya tanpa beban dilakukan sebanyak lima kali pengukuran agar data *valid*. Dapat dilihat pada tabel 14 pengujian catu daya aki (accumulator) mobil 12V dan rangkaian regulator 9V menunjukkan banyaknya angka konstan yaitu 12,7 volt dan 8,69 volt. Tidak bulat sebesar 12V dan 9V karena tegangan yang terukur pada multimeter memiliki selisih dengan pengujian sebenarnya. Sebagai contoh untuk mengetahui besar *error* pada pengujian, maka penulis mengambil contoh dari pengujian pertama tabel 14 catu daya yang menghasilkan perhitungan sebagai berikut:

$$\frac{0,6}{12} \times 100\% = 5\% \text{ dan } \frac{0,31}{9} \times 100\% = 3,4\%$$

Oleh karena itu, rata-rata *error* yang dihasilkan sebesar 5,32% dan 3,4%. Adanya *error* tersebut dikarenakan arus yang masuk ke *power supply* tidak stabil dan kondisi regulasi dan filter pada rangkaian *power supply* yang kurang baik.

b. Dengan beban ketika mesin mobil mati

Pengukuran daya dengan beban dilakukan sebanyak lima kali pengukuran agar data *valid*. Dapat dilihat pada tabel 15 pengujian aki (accumulator) 12V dan rangkaian regulator 9V menghasilkan data rata-rata 12,7 volt dan 8,66 volt. Setelah ada bebanpun tegangan tidak bulat sebesar 12V dan 9V karena tegangan yang Terukur pada multimeter memiliki selisih dengan pengujian

sebenarnya. Sebagai contoh untuk mengetahui besar *error* pada pengujian, maka penulis mengambil contoh dari pengujian pertama tabel 15 catu daya yang menghasilkan perhitungan sebagai berikut:

$$\frac{0,4}{12} \times 100\% = 3,3\% \text{ dan } \frac{0,32}{9} \times 100\% = 3,56\%$$

Oleh karena itu, rata-rata *error* yang dihasilkan sebesar 5,7% dan 3,7%. Adanya *error* tersebut dikarenakan arus yang masuk ke *power supply* tidak stabil dan kondisi regulasi dan filter pada rangkaian *power supply* yang kurang baik.

c. Tanpa beban ketika mesin mobil menyala

Pengukuran daya tanpa beban dilakukan sebanyak lima kali pengukuran agar data *valid*. Dapat dilihat pada tabel 16 pengujian catu daya aki (accumulator) mobil 12V menunjukkan data kenaikan tegangan yang konstan dari 13,3 hingga 14,4 volt. Sebagai contoh untuk mengetahui besar *error* pada pengujian, maka penulis mengambil contoh dari pengujian pertama tabel 16 catu daya yang menghasilkan perhitungan sebagai berikut:

$$\frac{2,5}{12} \times 100\% = 20,83\% \text{ dan } \frac{0,31}{9} \times 100\% = 3,4\%$$

Hal ini disebabkan oleh alternator mobil yang akan otomatis menyala ketika mesin mobil dinyalakan dan pelonjakan angka voltase menjadi 14,5 volt dikarenakan kelistrikan mobil yang dipakai mengalami masalah. Voltase ideal yang harusnya keluar ketika mesin mobil menyala adalah 13,8 volt. Oleh karena itu rata-rata *error* yang dihasilkan aki (accumulator) mobil ketika mesin menyala adalah 15,68%.

Sedangkan untuk rangkaian regulator 9V menghasilkan angka tetap yaitu 8,69 seperti pada percobaan tanpa beban ketika mesin mati dengan *error* sebesar 3,4%

d. Dengan beban ketika mesin mobil menyala

Pengukuran daya dengan beban dilakukan sebanyak lima kali pengukuran agar data valid. Dapat dilihat pada tabel 17 pengujian aki (accumulator) 12V menunjukkan data kenaikan tegangan yang konstan dari 13,3 hingga 14 volt. Sebagai contoh untuk mengetahui besar *error* pada pengujian, maka penulis mengambil contoh dari pengujian pertama tabel 17 catu daya yang menghasilkan perhitungan sebagai berikut:

$$\frac{2}{12} \times 100\% = 16,67\% \text{ dan } \frac{0,33}{9} \times 100\% = 3,67\%$$

Hal ini disebabkan oleh alternator mobil yang akan otomatis menyala ketika mesin mobil dinyalakan dan pelonjakan angka voltase menjadi 14 volt dikarenakan kelistrikan mobil yang dipakai mengalami masalah. Voltase ideal yang harusnya keluar ketika mesin mobil menyala adalah 13,8 volt. Oleh karena itu rata-rata *error* yang dihasilkan aki (accumulator) mobil ketika mesin menyala adalah 14,8%

Sedangkan untuk rangkaian regulator 9V menunjukan beberapa data konstan yaitu 8,65 volt dengan rata-rata besar *error* 3,8%. Berdasarkan hasil pengujian di atas dapat disimpulkan bahwa daya yang di *supply* oleh masing-masing catu daya sudah cukup baik dan sudah memenuhi kebutuhan daya pada sistem alat.

2. Pengujian jarak *bluetooth*

Pengujian jarak *bluetooth* dilakukan dengan memberikan dua kondisi dimana kondisi yang pertama tanpa penghalang dan kondisi kedua dengan menggunakan penghalang. Pengujian tanpa penghalang dilakukan disebuah lapangan terbuka tanpa adanya hambatan apapun dan pengujian menggunakan penghalang dilakukan disebuah tempat parkir sebuah swalayan yang ramai mobil terparkir sebagai penghalang yang dicoba untuk mengetahui jangkauan transmisi *bluetooth* dengan jarak tertentu. Hasil pengujian jangkauan *bluetooth* dapat dilihat pada tabel 18.

Ketika tanpa penghalang, *bluetooth* dari jarak 1-30 meter masih dapat menerima perintah yang diberikan *smartphone* akan tetapi mulai menunjukan delay ketika memasuki jarak 16 meter dan *delay* semakin lama ketika jarak *bluetooth* bertambah. Saat memasuki jarak 34 meter *bluetooth* tidak dapat lagi menangkap perintah yang diberikan *smartphone*.

Ketika *bluetooth* dioperasikan dengan penghalang beberapa mobil lain dalam tempat parkir swalayan, jarak jangkauan *bluetooth* sama seperti saat *bluetooth* dioperasikan tanpa adanya penghalang. Akan tetapi, perbedaan *delay* muncul pada jarak 16 meter. Saat terdapat penghalang, *delay bluetooth* untuk menerima perintah *smartphone* dari jarak 16-30 meter lebih lama dibanding ketika tidak ada penghalang dan ketika memasuki jarak 34 meter *bluetooth* mengalami *lost connection*.

Dari pengujian tersebut dapat disimpulkan bahwa jarak maksimal *bluetooth* pada alat yaitu 1 sampai 30 meter dengan adanya penghalang dan tanpa penghalang.

3. Pengujian motor *dynamo starter*

Pengujian motor *dynamo starter* yang terpasang pada kap dan bagasi ini dilakukan sebanyak enam kali dengan tiga kali kondisi logika 1 (benar) dan tiga kali kondisi logika 0 (salah) saat mesin mobil mati dan enam kali dengan tiga kali kondisi logika 1 (benar) dan tiga kali kondisi logika 0 (salah) saat mesin mobil menyala. Motor *dynamo starter* ini digunakan untuk menarik tuas pengungkit yang berfungsi untuk membuka *lock* pada kap dan bagasi.

Ketika mesin mobil dalam keadaan mati, voltase yang didapat oleh motor *dynamo starter* kap dan bagasi sama dengan rata-rata yaitu 12,7 volt. Dengan voltase 12,7 motor *dynamo starter* berputar dengan semestinya. Percobaan pengamatan ini dapat dilihat pada tabel 19.

Ketika mesin mobil dalam keadaan nyala, voltase yang didapat oleh motor *dynamo starter* kap dan bagasi sama dengan kenaikan signifikan seperti pada percobaan catu daya ada beban ketika mesin mobil dinyalakan yaitu 13,3 volt hingga 14 volt. Hal ini tidak mempengaruhi motor *dynamo starter* dalam bekerja. Ketika mendapat perubahan voltase, motor *dynamo starter* tetap berputar dengan semestinya saat diberikan logika 1 (benar) dan berhenti ketika diberikan logika 0 (salah). Hal ini dapat dilihat pada percobaan pengamatan tabel 20.

Dari percobaan yang telah dilakukan kita dapat melihat bahwa perbedaan voltase saat mesin mobil mati dan saat mesin mobil hidup tidak mempengaruhi motor *dynamo starter*.

4. Pengujian modul *relay* 2 channel 5V dan *relay* 12V

Pengujian modul relay 2 channel 5V dan relay 12V ini untuk mengetahui apakah perbedaan tegangan mempengaruhi kinerja relay. Relay 5V digunakan untuk memberikan tegangan kepada relay 12V saat dibutuhkan. Sedangkan relay 12V digunakan untuk menghidupkan motor *dynamo starter* yang membutuhkan arus ampere besar. Relay 12V juga dipakai sebagai pengaman ketika menerima arus tinggi untuk menyalakan motor *dynamo starter*. Pengujian tegangan ini dilakukan enam kali pada saat mesin mobil mati dan enam kali pada saat mesin mobil menyala agar data yang dihasilkan *valid*. Percobaan pengujian ini dapat dilihat pada tabel 21.

Ketika mesin mobil dalam keadaan mati, dapat dilihat tidak ada *error* yang berarti ketika tegangan masuk dari relay modul 2 channel 5V dan relay 12V. walaupun terdapat kenaikan tegangan pada V-In relay 5V disetiap setelah logika low di berikan.

Ketika mesin mobil dalam keadaan hidup, dapat dilihat pula tidak ada *error* yang berarti ketika tegangan masuk dari relay modul 2 channel 5V dan relay 12V. walaupun terdapat kenaikan tegangan V-In pada relay 12V dan penurunan tegangan akibat beban mesin pada relay 5V.

Dari percobaan tabel 21 dapat dilihat bahwa perbedaan tegangan saat mesin mobil hidup dan mesin mobil mati tidak ada *error* yang menyebabkan relay menjadi tidak bekerja dengan baik.

5. Pengujian perintah *virtual button*

Pengujian *virtual button* ini dilakukan sebanyak lima kali dengan cara menginstall aplikasi pembuka kap dan bagasi pada *smartphone* dengan jenis *operating system* berbeda dan dengan jangkauan *bluetooth* antara 1-10 meter. Percobaan pengujian ini dapat dilihat pada tabel 22.

Pada pengujian ini melakukan uji perintah *virtual button* dengan *operating system* Jelly Bean atau android 4.2.2. berdasarkan percobaan yang telah dilakukan, hasil yang didapatkan sesuai dengan harapan. Aplikasi berjalan dengan baik dan perintah yang dikirimkan dari *smartphone* menuju *bluetooth* dapat diterima dengan baik. Pada pengujian tersebut terdapat delay waktu respon sebesar 0,1 detik.

Pada pengujian kedua, melakukan uji perintah *virtual button* dengan *operating system* KitKat atau android 4.4.2. berdasarkan percobaan yang telah dilakukan, hasil yang didapatkan sesuai dengan harapan. Aplikasi berjalan dengan baik dan perintah yang dikirimkan dari *smartphone* menuju *bluetooth* dapat diterima dengan baik. Pada pengujian tersebut terdapat delay waktu respon sebesar 0,15 detik.

Pada pengujian ketiga, melakukan uji perintah *virtual button* dengan *operating system* Lolipop atau android 5.0. berdasarkan percobaan yang telah dilakukan, hasil yang didapatkan sesuai dengan harapan. Aplikasi berjalan

dengan baik dan perintah yang dikirimkan dari *smartphone* menuju *bluetooth* dapat diterima dengan baik. Pada pengujian tersebut terdapat delay waktu respon sebesar 0,1 detik.

Pada pengujian keempat, melakukan uji perintah *virtual button* dengan *operating system* Marshmallow atau android 6.0. berdasarkan percobaan yang telah dilakukan, hasil yang didapatkan sesuai dengan harapan. Aplikasi berjalan dengan baik dan perintah yang dikirimkan dari *smartphone* menuju *bluetooth* dapat diterima dengan baik. Pada pengujian tersebut terdapat delay waktu respon sebesar 0,1 detik.

Pada pengujian kelima, melakukan uji perintah *virtual button* dengan *operating system* Nougat atau android 7.0. berdasarkan percobaan yang telah dilakukan, hasil yang didapatkan sesuai dengan harapan. Aplikasi berjalan dengan baik dan perintah yang dikirimkan dari *smartphone* menuju *bluetooth* dapat diterima dengan baik. Pada pengujian tersebut terdapat delay waktu respon sebesar 0,3 detik.

Dari percobaan tabel 22 dapat dilihat perbedaan jenis *operating system* android tidak berpengaruh terhadap kinerja *virtual button* dan system rangkaian pembuka kap dan bagasi yang menghasilkan rata-rata delay sebesar 0,15 detik.

6. Pengujian button manual emergency

Pengujian button emergency ini dilakukan dengan cara menekan tombol secara berkala pada selang waktu 15 menit, 30menit, 2 jam, 1 hari kemudian dan 3 hari kemudian. Hal ini dilakukan untuk mengetahui apakah tombol

sudah bekerja dengan baik dalam jangka waktu tertentu. Dari respon yang dihasilkan pengoperasian tombol *emergency* ini sudah baik dan tidak mengalami *error*.

7. Analisis pengujian secara keseluruhan

Dari hasil pengujian alat secara keseluruhan didapatkan bahwa alat sudah bekerja dengan semestinya meskipun masih terdapat banyak keterbatasan. Alat sudah baik dalam merespon dan sudah bekerja dengan apa yang diharapkan. Banyak faktor yang mempengaruhi kinerja alat seperti tata cara menyalakan saklar saat akan menggunakan alat ataupun mematikan ketika tidak dibutuhkan terkadang masih menimbulkan kesalahan ataupun dari alat itu sendiri baik dari *hardware* maupun *software*.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

1. Pembuatan *software virtual button* menggunakan App Inventor untuk mendesain dan memprogram aplikasi secara *online*. Program yang digunakan untuk menjalankan mikrokontroler arduino uno menggunakan arduino IDE dan pembuatan *source code* program menggunakan bahasa C.
2. Pembuatan *hardware* menggunakan tiga buah hidrolis, sebuah helper, dua buah motor dynamo starter, Arduino Uno, modul relay 2 channel 5 volt dan 12 volt. Hasil pengujian catu daya aki (accumulator) 12V dan Regulator 9V tanpa beban ketika mesin mobil mati dengan rata-rata *error* sebesar 5,32% dan 3,4%. Pengujian dengan beban ketika mesin mobil mati dengan rata-rata *error* sebesar 5,7% dan 3,7%. Pengujian tanpa beban ketika mesin mobil menyala dengan rata-rata *error* sebesar 15,68% dan 3,4%. Pengujian ada beban ketika mesin mobil menyala dengan rata-rata *error* sebesar 14,8% dan 3,8%. Pengujian *delay* waktu *virtual button* menggunakan lima jenis sistem operasi *smartphone* yang berbeda dengan rata-rata *delay* 0,15 detik. Jangkauan jarak *bluetooth* maksimal dari 1 sampai 30 meter dengan dan tanpa penghalang,
3. Hasil unjuk kerja pembuka kap dan bagasi mobil menggunakan *smartphone* berbasis *bluetooth* sudah bekerja sesuai dengan fungsinya dengan keberhasilan 100%.

B. Keterbatasan Alat

Pembuka kap dan bagasi mobil menggunakan *smartphone* berbasis *bluetooth* memiliki keterbatasan dalam sistem kerjanya antara lain:

1. Mekanik pembuka kap dan bagasi masih sangat kasar dan suaranya sangat kencang
2. Alat ini tidak dilengkapi dengan menutup kap dan bagasi otomatis
3. Pada keamanan pada menu login aplikasi belum maksimal karena masih menggunakan *username* dan *password universal* belum menggunakan *database*

C. Saran

Berdasarkan keterbatasan waktu, kemampuan dan dana, masih banyak kekurangan dalam pengerjaan alat yang dibuat ini, maka dari itu penulis menyarankan sebagai berikut:

1. Alat sebaiknya menggunakan peredam suara pada *motor dynamo* starters sehingga suara saat motor berputar untuk membuka kunci tidak kasar
2. Hidrolis pada alat sebaiknya diganti dengan pneumatik sehingga baik membuka ataupun menutup kap dan bagasi dapat dilakukan

DAFTAR PUSTAKA

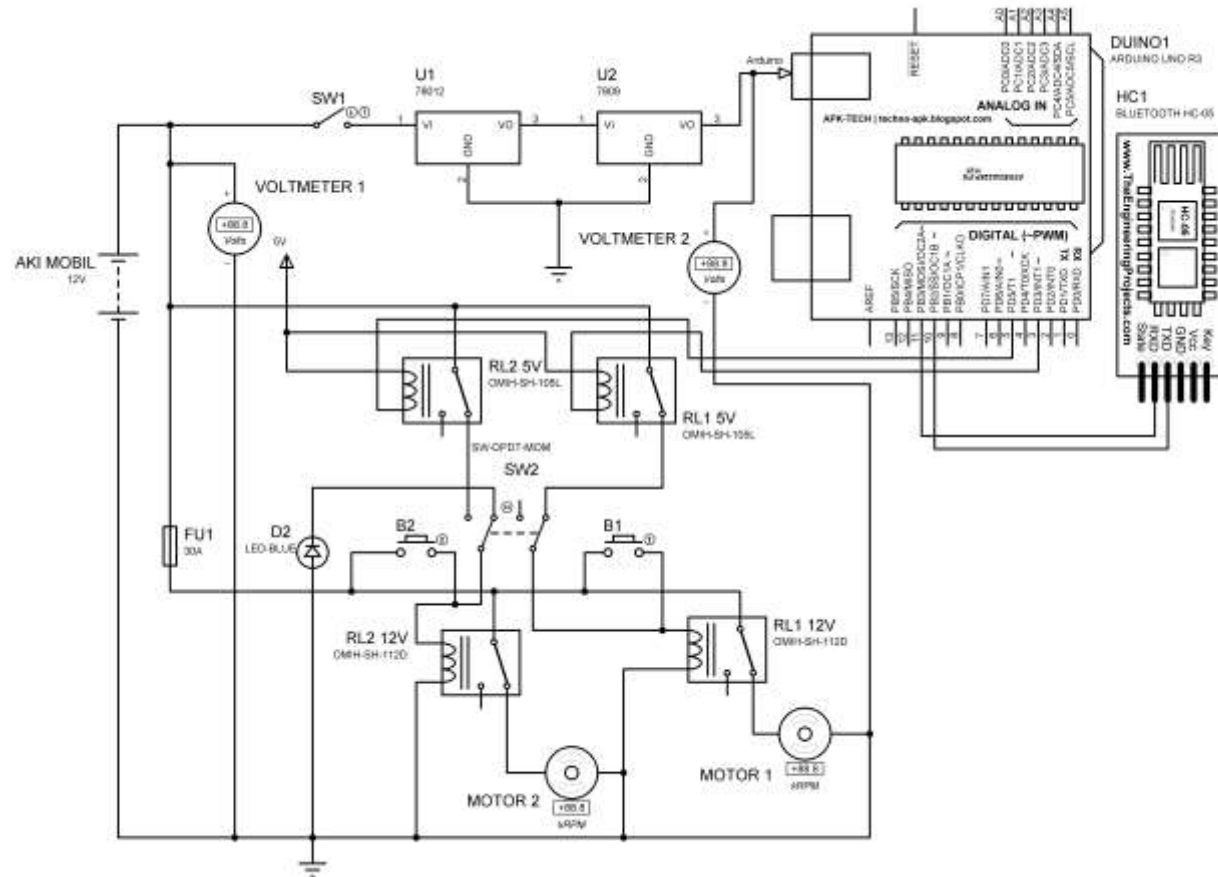
- Baykus, Burak dkk (2011). *Design and kinematic analysis of a parallel mechanism to be utilized a luggagw door by an analogy to a four-bar mechanism*. Diambil pada 16 Oktober 2017 dari <http://www.SciRP.org/journal/eng>
- Bhaskar, N. dan P. Rayudu (2015). *Design and analysis of car bonnet*. Diambil pada 16 Oktober 2017 dari <http://inpressco.com/category/ijcet>
- Drive, Kelly (2011). *Relays, horns and switches hella electrics*. Diambil pada 14 Desember 2017 dari www.hellausa.com
- Harris, William (2016). *How car suspensions work*. Diambil pada 12 November 2017 dari <https://auto.howstuffworks.com/car-suspension2.htm>
- Hendriono, D (2014). *Apa itu arduino*. Diambil pada 1 oktober 2017 dari <http://www.hendriono.com/blog/post/apa-itu-arduino#isi1>
- Julianto, G. (2015). *Pengertian dan tingkatan versi android*. Diambil pada 27 september 2017 dari <http://seputarti.com/android/pengertian-dan-tingkatan-versi-android.html>
- Kho, D. (2017). *Pengertian relay*. Diambil pada 15 desember 2017 dari <http://teknikelektronika.com/pengertian-relay-fungsi-relay/>
- Mujiono, j. (2015). *Pengertian dan fungsi bluetoooth*. Diambil pada 27 september 2017 dari <http://www.teorikomputer.com/2015/10/pengertian-dan-fungsi-bluetooth.html>
- Prawoto, I. (2015). *Pengertian arduino uno mikrokontroler atmega328*. Diambil pada 14 oktober 2017 dari <https://www.caratekno.com/2015/07/pengertian-arduino-uno-mikrokontroler.html>
- Purwanto, Dwi (2014). *Fungsi dan konstruksi baterai aki*. Diambil pada 15 Desember 2017 dari <http://www.dasarotomotif.info/2014/08/fungsi-dan-konstruksi-baterai-aki.html>
- Santoso, Leo Willyanto dan Yulia (2004). *Studi dan uji coba teknologi Bluetooth sebagai alternative komunikasi data nirkabel*. Diambil pada 18 Oktober 2017 dari <http://puslit.petra.ac.id/journals/informatics/>


Soares, C. Guedes (2015). Sistem kerja hidrolik. *Renewable Energies Offshore*, 272.

Yeoh, Li Ching (2016) *Design of car lifthing mechanism using sensitive hydraulic system*. Diambil pada 20 september 2017 dari <http://eprints.utm.edu.my/16403/>

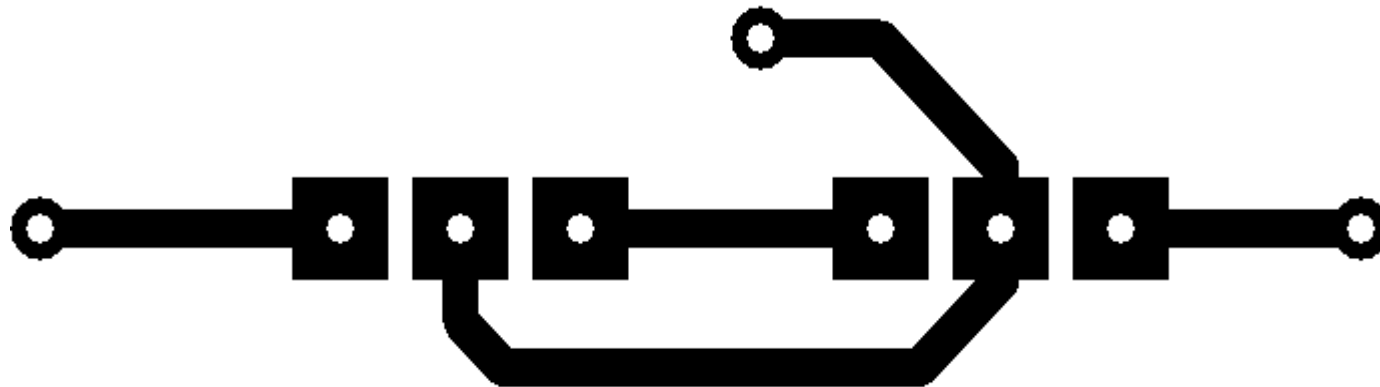
LAMPIRAN

Lampiran 1. Skema Rangkaian Keseluruhan




RANGKAIAN KESELURUHAN			KETERANGAN	
 TEKNIK ELEKTRONIKA - D3 FAKULTAS TEKNIK UNIV. NEGERI YOGYAKARTA	SKA. 1:1	DIS.DESSY IRMAWATI	A4	No. 01
	DIG. IVANA Y.C.	DIP.DESSY IRMAWATI	NIM. 14507134003	

Lampiran 2. Layout



labcenter
Electronics
<http://www.fischl.de>

RANGKAIAN REGULATOR				KETERANGAN	
	TEKNIK ELEKTRONIKA - D3 FAKULTAS TEKNIK UNIV. NEGERI YOGYAKARTA	SKA. 1:1	DIS.DESSY IRMAWATI	A4	No. 02
		DIG. IVANA Y.C.	DIP.DESSY IRMAWATI		
				NIM. 14507134003	

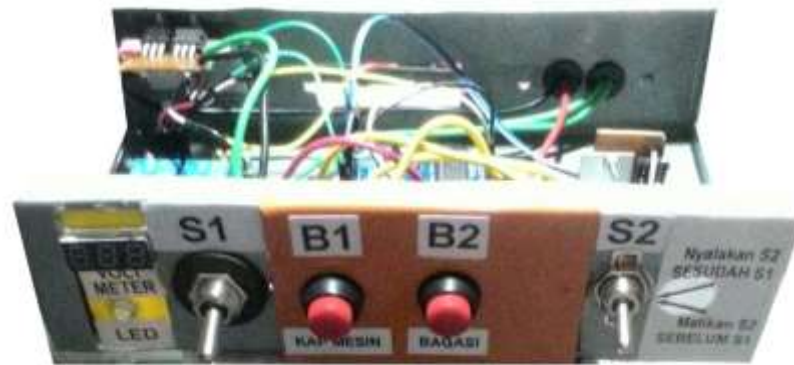
Lampiran 3. Daftar Komponen

1. Aki
2. Arduino
3. Handphone
4. Modul Relay 2 channel
5. Relay
6. Motor Dinamo Starter
7. Baut
8. *Box*
9. *Kabel Pelangi*
10. *Hidrolik*
11. *Helper*
12. *IC 78012*
13. *IC 7809*
14. *Saklar*
15. *Voltmeter mini*
16. *Button manual*

Lampiran 4. *Listing* Program

```
if(perintah.length() > 0) {  
  Serial.println(perintah);  
  if(perintah == "buka kap")  
  { digitalWrite(3, HIGH); }  
  delay(200);  
  { digitalWrite(3, LOW); }  
  
  if(perintah == "buka bagasi")  
  { digitalWrite(5, HIGH); }  
  delay(200);  
  { digitalWrite(5, LOW); }  
  
  perintah="";}}  
#include <SoftwareSerial.h>  
SoftwareSerial BT(10, 11);  
String perintah;  
  
void setup() {  
  BT.begin(9600);  
  Serial.begin(9600);  
  pinMode(3, OUTPUT);  
  pinMode(5, OUTPUT);  
}  
void loop() {  
  while (BT.available()) {  
    delay(10);  
    char c = BT.read();  
    perintah += c;  
  }  
}
```

Lampiran 5. Gambar Alat



Lampiran 6. Tabel 24 Hasil Uji Unjuk Kerja

No	Perintah virtual button	Tampilan	Respon perintah	keterangan
1	Buka kap			Kap terbuka

2	Buka bagasi			Bagasi terbuka
---	-------------	--	---	----------------

Lampiran 7. Data Sheet Arduino Uno



Product Overview

The Arduino Uno is a microcontroller board based on the ATmega328 ([datasheet](#)). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega8U2 programmed as a USB-to-serial converter.

"Uno" means one in Italian and is named to mark the upcoming release of Arduino 1.0. The Uno and version 1.0 will be the reference versions of Arduino, moving forward. The Uno is the latest in a series of USB Arduino boards, and the reference model for the Arduino platform; for a comparison with previous versions, see the [index of Arduino boards](#).

Index

Technical
Specifications

Page 2

How to use Arduino
Programming Environment, Basic Tutorials

Page 6

Terms &
Conditions

Page 7

Environmental Policies
half sqm of green via Impatto Zero®

Page 7



radiospares

RADIONICS

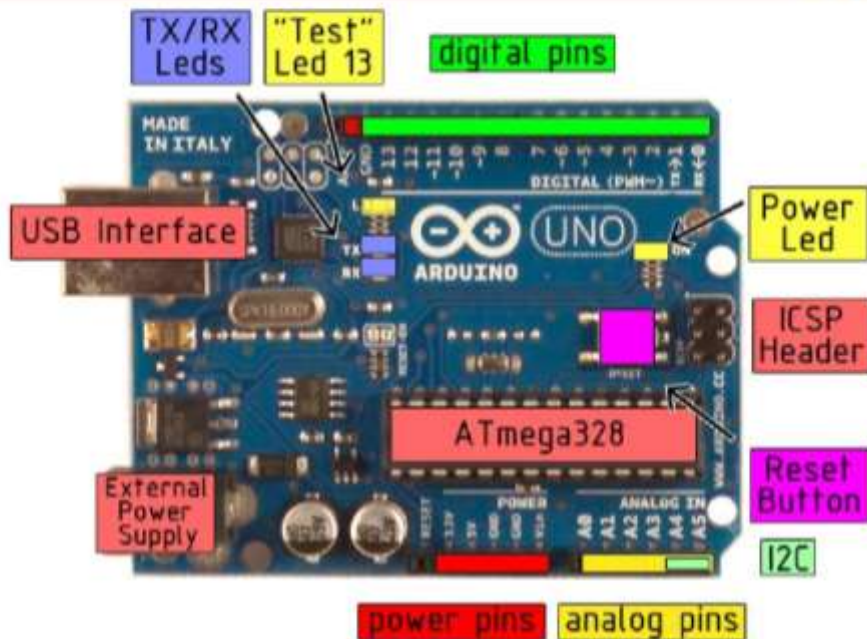


EAGLE files: [arduino-duemilanove-uno-design.zip](#) Schematic: [arduino-uno-schematic.pdf](#)

Summary

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (of which 6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB of which 0.5 KB used by bootloader
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz

the board



The Uno has 6 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the AREF pin and the [analogReference\(\)](#) function. Additionally, some pins have specialized functionality:

- **I²C: 4 (SDA) and 5 (SCL).** Support I²C (TWI) communication using the [Wire library](#).

There are a couple of other pins on the board:

- **AREF.** Reference voltage for the analog inputs. Used with [analogReference\(\)](#).
- **Reset.** Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

See also the [mapping between Arduino pins and Atmega328 ports](#).

Communication

The Arduino Uno has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An ATmega8U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '8U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an *.inf file is required..

The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serial communication on pins 0 and 1).

A [SoftwareSerial library](#) allows for serial communication on any of the Uno's digital pins.

The ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus; see the [documentation](#) for details. To use the SPI communication, please see the ATmega328 datasheet.

Programming

The Arduino Uno can be programmed with the Arduino software ([download](#)). Select "Arduino Uno w/ ATmega328" from the **Tools > Board** menu (according to the microcontroller on your board). For details, see the [reference](#) and [tutorials](#).

The ATmega328 on the Arduino Uno comes preburned with a [bootloader](#) that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol ([reference](#), [C header files](#)).

You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header; see [these instructions](#) for details.

The ATmega8U2 firmware source code is available . The ATmega8U2 is loaded with a DFU bootloader, which can be activated by connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2. You can then use [Atmel's FLIP software](#) (Windows) or the [DFU programmer](#) (Mac OS X and Linux) to load a new firmware. Or you can use the ISP header with an external programmer (overwriting the DFU bootloader).



RADIOSPARES

RADIONICS



Power

The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector.

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **VIN.** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V.** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3.** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND.** Ground pins.

Memory

The Atmega328 has 32 KB of flash memory for storing code (of which 0,5 KB is used for the bootloader); it has also 2 KB of SRAM and 1 KB of EEPROM (which can be read and written with the [EEPROM library](#)).

Input and Output

Each of the 14 digital pins on the Uno can be used as an input or output, using [pinMode\(\)](#), [digitalWrite\(\)](#), and [digitalRead\(\)](#) functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the [attachInterrupt\(\)](#) function for details.
- **PWM: 3, 5, 6, 9, 10, and 11.** Provide 8-bit PWM output with the [analogWrite\(\)](#) function.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- **LED: 13.** There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.



radiospares

RADIONICS



Automatic (Software) Reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Uno is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the ATmega8U2 is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. This means that the bootloader can have a shorter timeout, as the lowering of DTR can be well-coordinated with the start of the upload.

This setup has other implications. When the Uno is connected to either a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Uno. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

The Uno contains a trace that can be cut to disable the auto-reset. The pads on either side of the trace can be soldered together to re-enable it. It's labeled "RESET-EN". You may also be able to disable the auto-reset by connecting a 110 ohm resistor from 5V to the reset line; see [this forum thread](#) for details.

USB Overcurrent Protection

The Arduino Uno has a resettable polyfuse that protects your computer's USB ports from shorts and overcurrent. Although most computers provide their own internal protection, the fuse provides an extra layer of protection. If more than 500 mA is applied to the USB port, the fuse will automatically break the connection until the short or overload is removed.

Physical Characteristics

The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. Three screw holes allow the board to be attached to a surface or case. Note that the distance between digital pins 7 and 8 is 160 mil (0.16"), not an even multiple of the 100 mil spacing of the other pins.



radiospares

RADIONICS



How to use Arduino



Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the [Arduino programming language](#) (based on [Wiring](#)) and the Arduino development environment (based on [Processing](#)). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, MaxMSP).

Arduino is a cross-platform program. You'll have to follow different instructions for your personal OS. Check on the [Arduino site](http://arduino.cc/en/Guide/HomePage) for the latest instructions. <http://arduino.cc/en/Guide/HomePage>

Linux Install

Windows Install

Mac Install

Once you have downloaded/unzipped the arduino IDE, you can Plug the Arduino to your PC via USB cable.

Blink led

Now you're actually ready to "burn" your first program on the arduino board. To select "blink led", the physical translation of the well known programming "hello world", select

**File>Sketchbook>
Arduino-0017>Examples>
Digital>Blink**

Once you have your skecth you'll see something very close to the screenshot on the right.

In **Tools>Board** select

Now you have to go to
Tools>SerialPort
and select the right serial port, the one arduino is attached to.



Done compiling

Press Compile button
(to check for errors)



Upload



TX RX Flashing



Blinking Led!

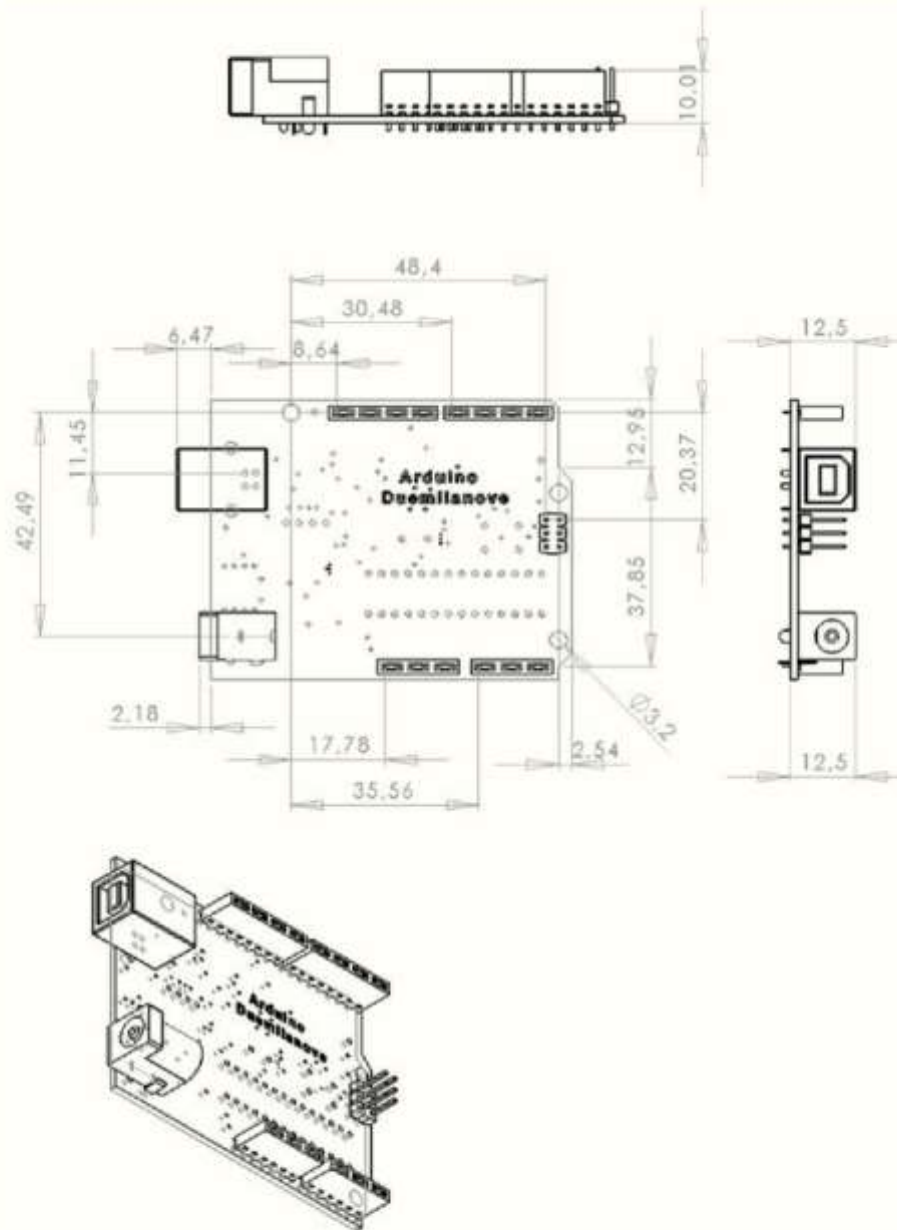


radiospares

RADIONICS



Dimensioned Drawing



radiospares

RADIONICS



Terms & Conditions



1. Warranties

1.1 The producer warrants that its products will conform to the Specifications. This warranty lasts for one (1) years from the date of the sale. The producer shall not be liable for any defects that are caused by neglect, misuse or mistreatment by the Customer, including improper installation or testing, or for any products that have been altered or modified in any way by a Customer. Moreover, The producer shall not be liable for any defects that result from Customer's design, specifications or instructions for such products. Testing and other quality control techniques are used to the extent the producer deems necessary.

1.2 If any products fail to conform to the warranty set forth above, the producer's sole liability shall be to replace such products. The producer's liability shall be limited to products that are determined by the producer not to conform to such warranty. If the producer elects to replace such products, the producer shall have a reasonable time to replacements. Replaced products shall be warranted for a new full warranty period.

1.3 EXCEPT AS SET FORTH ABOVE, PRODUCTS ARE PROVIDED "AS IS" AND "WITH ALL FAULTS." THE PRODUCER DISCLAIMS ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, REGARDING PRODUCTS, INCLUDING BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

1.4 Customer agrees that prior to using any systems that include the producer products, Customer will test such systems and the functionality of the products as used in such systems. The producer may provide technical, applications or design advice, quality characterization, reliability data or other services. Customer acknowledges and agrees that providing these services shall not expand or otherwise alter the producer's warranties, as set forth above, and no additional obligations or liabilities shall arise from the producer providing such services.

1.5 The Arduino™ products are not authorized for use in safety-critical applications where a failure of the product would reasonably be expected to cause severe personal injury or death. Safety-Critical Applications include, without limitation, life support devices and systems, equipment or systems for the operation of nuclear facilities and weapons systems. Arduino™ products are neither designed nor intended for use in military or aerospace applications or environments and for automotive applications or environment. Customer acknowledges and agrees that any such use of Arduino™ products which is solely at the Customer's risk, and that Customer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

1.6 Customer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products and any use of Arduino™ products in Customer's applications, notwithstanding any applications-related information or support that may be provided by the producer.

2. Indemnification

The Customer acknowledges and agrees to defend, indemnify and hold harmless the producer from and against any and all third-party losses, damages, liabilities and expenses it incurs to the extent directly caused by: (i) an actual breach by a Customer of the representation and warranties made under this terms and conditions or (ii) the gross negligence or willful misconduct by the Customer.

3. Consequential Damages Waiver

In no event the producer shall be liable to the Customer or any third parties for any special, collateral, indirect, punitive, incidental, consequential or exemplary damages in connection with or arising out of the products provided hereunder, regardless of whether the producer has been advised of the possibility of such damages. This section will survive the termination of the warranty period.

4. Changes to specifications

The producer may make changes to specifications and product descriptions at any time, without notice. The Customer must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." The producer reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The product information on the Web Site or Materials is subject to change without notice. Do not finalize a design with this information.



Environmental Policies



The producer of Arduino™ has joined the Impatto Zero® policy of LifeGate.it. For each Arduino board produced is created / looked after half squared Km of Costa Rica's forest's.



radiospares

RADIONICS



Lampiran 8. Data Sheet *Bluetooth* HC-05

DATASHEET BLUETOOTH TO SERIAL PORT MODULE HC05



Overview

HC-05 module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup.

Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Bluecore 04-External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature). It has the

footprint as small as 12.7mmx27mm. Hope it will simplify your overall design/development cycle.

www.electronica60norte.com
electronica60norte@hotmail.com

Specifications

Hardware features

- Typical -80dBm sensitivity.
- Up to +4dBm RF transmit power.
- Low Power 1.8V Operation, 3.3 to 5 V I/O.
- PIO control.
- UART interface with programmable baud rate.
- With integrated antenna.
- With edge connector.

Software features

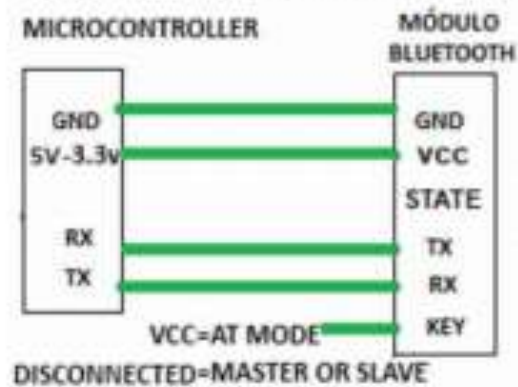
- Slave default Baud rate: 9600, Data bits:8, Stop bit:1,Parity:No parity.
- PIO9 and PIO8 can be connected to red and blue led separately. When master and slave are paired, red and blue led blinks 1time/2s in interval, while disconnected only blue led blinks 2times/s.
- Auto-connect to the last device on power as default.
- Permit pairing device to connect as default.
- Auto-pairing **PINCODE:"1234"** as default.
- Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

www.electronica60norte.com
electronica60norte@hotmail.com

Pin out configuration



Typical Application Circuit

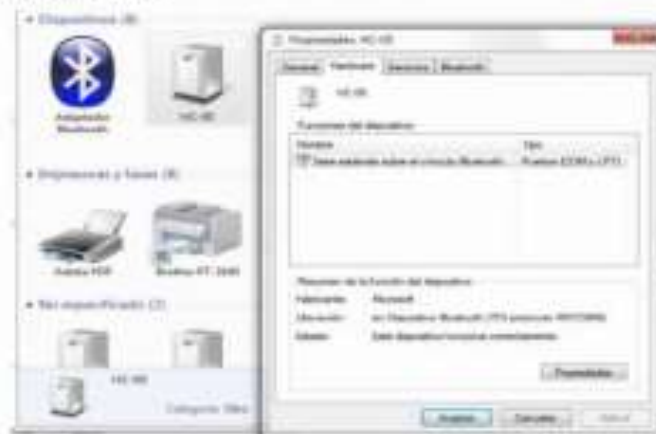


www.electronica60norte.com
electronica60norte@hotmail.com

Print to PDF without this message by purchasing novaPDF (<http://www.novapdf.com/>)

After connect the Bluetooth module, scan for new devices from the PC and you will find the module with the device name "HC-05", after that, click to connect, if some message appears asking about "Pairing code" just put "1234" as default code.

BLUE LED = ACTIVE (Blinking 500ms period inactive connection, change 1seg with active connection)



Open a serial terminal and select the serial COM x port number that assigned Windows to Bluetooth Module.

Configure the serial terminal with these parameters:

- Baud rate: 9600.
- Data bits:8.
- Stop bit:1.
- Parity: No parity.

www.electronica60norfe.com
electronica60norfe@hotmail.com

Open connection and you will be ready to send and receive data from module Bluetooth like Serial Port COM



AT COMMANDS

How to get to AT COMMAND mode

- 1: Connect KEY pin to VCC.
- 2: Supply power to module. Then the module will enter into AT MODE. In this mode you have to use baud rate at 38400. In this way, user should change the baud rate for SLAVE AND MASTER mode.

How to set this module as "Master - Host" role

- 1: Input high level to KEY.
- 2: Supply power to the module. And the module will enter to AT COMMAND.
- 3: Set the parameters of the hyper terminal or the other serial tools (baud rate: 38400, data bit:8, stop bit:1, no parity bit, no Flow Control).
- 4: Sent the characters "AT+ROLE=1\r\n" through serial, then receive the characters "OK\r\n". Here, "\r\n" is the CRLF.
- 5: Sent the characters "AT+CMODE=1\r\n" through serial, then receive the characters "OK\r\n". Here, "\r\n" is the CRLF.
- 6: Default factory password passkey is: 1243, this must be the same in the Bluetooth slave module if you want to pair it.
To read passkey use this command: "AT+PSWD?".
To Reset the password command sent the characters "AT+PSWD=XXXX".
The password must be 4-bits.

7: Leave free KEY, and supply power to the module again. Then this module will become master role and search the other module (slave role) automatically to build the connection (baud rate:9600, data bit:8, stop bit:1, no parity bit, no Flow Control).

How to set this module be the "Slave - Device" role

- 1: Input high level to KEY.
- 2: Supply power to the module. And the module will enter to AT COMMAND.
- 3: Set the parameters of the super terminal or the other serial tools (baud rate: 38400, data bit:8, stop bit:1, no parity bit, no Flow Control).
- 4: Sent the characters "AT+ROLE=0\r\n" through serial, then receive the characters "OK\r\n". Here, "\r\n" is the CRLF.
- 5: Sent the characters "AT+CMODE=0\r\n" through serial, then receive the characters "OK\r\n". Here, "\r\n" is the CRLF.
- 6: Default factory password passkey is: 1243, this must be the same in the Bluetooth master module if you want to pair it.
To read passkey sent the characters "AT+PSWD?".
To Reset the password command sent the characters "AT+PSWD=XXXX".
The password must be 4-bits.
- 7: Leave free KEY, and supply power to the module again. Then this module will become slave role and wait to be discover it by the other module (master role) automatically to build the connection (baud rate:38400, data bit:8, stop bit:1, no parity bit, no Flow Control).

How to get to the standard communication mode

- 1: Leave free KEY, don't connect it to VDD neither GND.
- 2: Supply power to the module. Then the module will enter to communication mode. It can be used for pairing.

Notes

- (1) HC-05's command should end up with "\r\n". It means when you finish programming, you should add terminator ("ENTER" or "0x0d 0x0a") to the program.
- (2) The most common commands for HC-05 are: AT+ROLE (set master-slave), AT+CMODE (set address pairing), AT+PSWD (set password).
If you want the master module has the function of remembering slave module, the most simply way is: First, set AT+CMODE=1. Make the master module pair with the slave module. Second, set AT+CMODE=0. Then the master module just can make pair with that specified slave module.

www.electronica60norte.com

electronica60norte@hotmail.com