



**RANCANG BANGUN ROBOT AVOIDER SEGALA MEDAN BERBASIS  
ARDUINO MEGA 2560**

**PROYEK AKHIR**

oleh

**Wahyu Hidayat**

**NIM 131903102018**

**PROGRAM STUDI DIPLOMA III  
JURUSAN TEKNIK ELEKTRONIKA  
FAKULTAS TEKNIK  
UNIVERSITAS JEMBER  
2016**



**RANCANG BANGUN ROBOT AVOIDER SEGALA MEDAN  
BERBASIS ARDUINO MEGA 2560**

**PROYEK AKHIR**

diajukan guna melengkapi tugas akhir dan memenuhi syarat-syarat  
untuk menyelesaikan Program Studi Teknik Elektro (DIII)  
dan mencapai gelar Ahli Madya (Amd)

Oleh  
**Wahyu Hidayat**  
NIM 131903102018

**PROGRAM STUDI DIPLOMA III TEKNIK  
JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNIK  
UNIVERSITAS JEMBER  
2016**

**PERSEMBAHAN**

*Proyek akhir ini merupakan sebuah awal, langkah kecil menuju lompatan besar guna menggapai kesuksesan yang lebih baik lagi. Untuk itu saya ucapkan rasa syukur dan terima kasih sebesar-besarnya kepada...*

*Allah SWT, dengan segala Keagungan dan Keajaiban-Nya yang senantiasa mendengar do'a ku, menuntunku dari dari kegelapan, serta senantiasa menaungiku dengan rahmat dan hidayah-Nya dan junjunganku Nabi Besar Muhammad SAW yang telah menjadi penerang di dunia dan suri tauladan bagi kita semua;*

*Ibunda Sarni, Ayahanda Jambidi, Kakakku Adi Febriyanto, terima kasih atas segala kasih sayang, dukungan, semangat, dan doa selama ini;*

*Seluruh teman dan sahabat seperjuangan Teknik Elektro angkatan 2013, kalian sebagai tempat berbagi suka dan duka yang tidak akan terlupakan. Aku menjadikan kalian semua bagian dari diriku dan aku sangat menyayangi kalian semua;*

*Seluruh Dosen Jurusan Teknik Elektro yang terhormat, terima kasih telah banyak memberikan ilmu dan mendidik dengan penuh kesabaran;*

*Buat semua teman – teman Teknik Elektro semua angkatan, Serta semua pihak yang belum tertulis dalam lembar persembahan ini, Terima kasih atas segalanya;*

***Almamater Fakultas Teknik Universitas Jember.***

## MOTTO

“Demi masa. Sesungguhnya manusia itu benar-benar dalam kerugian,  
kecuali orang-orang yang beriman dan mengerjakan amal saleh dan nasehat  
menasehati supaya mentaati kebenaran dan nasehat menasehati supaya  
menetapi kesabaran”

(QS: Al Ashr 1-3)

“Jangan taat pada manusia. Taatlah pada kebenaran yang kau temukan  
sendiri dengan kedaulatanmu”

(Emha Ainun Nadjib)

“Harmonisasi hidup akan membawa kenikmatan hidup ”

(Wahyu Hidayat)

“Berlaku seperti lautan yang menerima apapun yang datang”

(Wahyu Hidayat)

## PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Wahyu Hidayat

NIM : 131903102018

Menyatakan dengan sesungguhnya bahwa proyek akhir yang berjudul: *“Rancang Bangun Robot Avoider Segala Medan Berbasis Arduino Mega 2560”* adalah benar-benar hasil karya sendiri, kecuali jika dalam pengutipan substansi disebutkan sumbernya dan belum pernah diajukan pada institusi mana pun serta bukan karya jiplakan. Saya bertanggung jawab atas keabsahan dan kebenaran isinya sesuai dengan sikap ilmiah yang harus dijunjung tinggi.

Demikian pernyataan ini saya buat dengan sebenarnya, tanpa adanya tekanan dan paksaan dari pihak manapun serta bersedia mendapat sanksi akademik jika ternyata di kemudian hari pernyataan ini tidak benar.

Jember, 28 Juni 2016

Yang menyatakan,

Wahyu Hidayat  
NIM 131903102018

**PROYEK AKHIR**

**RANCANG BANGUN ROBOT AVOIDER SEGALA MEDAN BERBASIS  
ARDUINO MEGA 2560**

Oleh  
Wahyu Hidayat  
NIM 131903102018

Pembimbing :

Dosen Pembimbing Utama : M. Agung Prawira N, S.T., M.T.

Dosen Pembimbing Anggota : Dr. Triwahju Hardianto, ST., MT.

## LEMBAR PENGESAHAN

Laporan Proyek Akhir berjudul “*Rancang Bangun Robot Avoider Segala Medan Berbasis Arduino Mega 2560*” oleh Wahyu Hidayat NIM: 131903102018 telah diuji dan disahkan oleh Fakultas Teknik Elektro Universitas Jember pada;

Hari : Selasa  
Tanggal : 28 Juni 2016  
Tempat : Fakultas Teknik Universitas Jember

Pembimbing Utama,

Pembimbing Anggota,

M. Agung Prawira N, S.T., M.T  
NIP 19871217 201212 1 003

Dr. Triwahju Hardianto, S.T., M.T.  
NIP 197008261997021001

Penguji I,

Penguji II,

Satryo Budi Utomo, S.T., M.T.  
NIP 198501262008011002

Sumardi, S.T.,M.T.  
NIP. 196701131998021001

Mengesahkan  
Dekan,

Dr.Ir. Entin Hidayah M.U.M  
NIP. 19661215 199503 2 001

**RANCANG BANGUN ROBOT AVOIDER SEGALA MEDAN BERBASIS  
ARDUINO MEGA 2560**

**Wahyu Hidayat**

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember

**ABSTRAK**

Dalam berkembangnya teknologi khususnya bidang robotika, semua hal yang berhubungan dengan kehidupan dan aktifitas manusia yang bersifat konstan. Robot datang sebagai solusi utama dalam membantu kinerja manusia. Misalnya dalam penyelamatan korban bencana reruntuhan gedung digunakan robot sebagai pendeteksi medan yang akan dilalui manusia. Oleh karena itu dirancang sebuah robot *avoider* segala medan untuk mempermudah relawan dalam pencarian korban. Robot digerakkan dengan *gearbox* motor DC, sensor *ultrasonic* sebagai kendali robot. Sensor yang digunakan sebanyak 6 buah untuk mempermudah pergerakan. Robot ini dikontrol otomatis menggunakan Arduino. Tingkat keberhasilan robot dalam menyusuri labirin sebesar 77,78 %.

Kata Kunci : Arduino, Robot *Avoider*, *Gearbox* Motor DC, Sensor *Ultrasonic*

## ***Design Build Avoider All Terrain Robot Based Arduino Mega 2560***

*Wahyu Hidayat*

*Department of Electrical Engineering, Faculty of Engineering, University of Jember*

### **ABSTRACT**

*In the development of technology especially in the field of robotics, all matters relating to human life and activity which is constant. Robot comes as a major solution in helping human performance. For example in the rubble of disaster rescue robot used as the detection of human terrain to be traversed. Therefore designed a avoider all-terrain robot to facilitate the volunteers in the search for survivors. Robot gearbox driven by a DC motor, ultrasonic sensors as robot control. Sensors are used seanyak 6 pieces for ease of movement. The robot is controlled automatically using Arduino. The robot is controlled automatically using Arduino. The success rate of the robot in the labyrinth of 77.78%.*

*Keywords: Arduino, Robot Avoider, Gearbox Motor DC, Ultrasonic Sensor*

## RINGKASAN

### **Rancang Bangun Robot *Avoider* Segala Medan Berbasis Arduino Mega 2560;**

Wahyu Hidayat; 131903102018; 2016; 78 Halaman; Diploma III Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Semakin tahun dunia teknologi mengalami perkembangan yang pesat. Khususnya dalam bidang robotika untuk membantu kehidupan sehari-hari pekerjaan manusia. Robot dapat membantu kinerja pekerjaan manusia secara konstan dan aman. Misalnya dalam upaya penyelamatan korban gempa pada reruntuhan gedung. Robot yang digunakan robot beroda yang dapat bergerak di segala medan. Penggunaan robot beroda tersebut dapat meningkatkan keamanan dari relawan maupun menyingkat waktu dalam penyusuran gedung.

Sistem robot *avoider* ini dikontrol menggunakan Arduino. Robot digerakkan dengan menggunakan *gearbox* motor DC dan dikendalikan oleh sensor *ultrasonic* yang berfungsi sebagai mata dari robot itu sendiri. Untuk driver mosfet berfungsi sebagai kontrol *gearbox* motor DC agar bergerak dengan kecepatan tertentu. Sensor *ultrasonic* akan memberi perintah jika terdapat halangan robot akan bergerak menghindari begitu seterusnya. Sensor yang digunakan sebanyak 6 buah dengan masing-masing 1 buah didepan dan belakang serta pada samping kanan-kiri masing-masing 2 buah.

Pada pengujian robot dilakukan pengujian pada sistem kerja sensor *ultrasonic* dan bekerjanya robot dalam lintasan labirin. Pada pengujian sensor *ultrasonic* mengalami tingkat keberhasilan 83,34 %. Sedangkan kinerja robot keseluruhan mengalami tingkat keberhasilan 77,78 %. Hal itu terjadi dikarenakan salah satu *gearbox* motor DC mengalami kendala mekanik.

## SUMMARY

### ***Design Build Avoider All Terrain Robot Based Arduino Mega 2560;***

*Wahyu Hidayat; 131903102018; 2016; 78 pages; Diploma in Electrical Engineering Department, Faculty of Engineering University of Jember.*

*The more the world of technology has improved message. Especially in the field of robotics to assist the daily life of human work. Robots can help humans work performance constantly and safely. For example in efforts to rescue earthquake victims in the rubble. Robots are used wheeled robot that can move in all fields. The use of a wheeled robot that can improve the security of volunteers as well as saving valuable time in the traverse of the building.*

*Avoider robot system is controlled using the Arduino. Robot is moved using a DC motor and gearbox is controlled by ultrasonic sensor that serves as the eyes of the robot itself. For mosfet driver function sebagai gearbox control a DC motor that moves at a certain speed. Ultrasonic sensors will give the order if there is obstacle avoidance robot will move and so on. Sensors are used as much as 6 pieces with 1 piece each in front and rear and on the right and left side of each 2 pieces.*

*In testing the robotic testing system ultrasonic sensors work and the operation of the robot in the maze path. On testing ultrasonic sensors experience a success rate of 83.34%. While the overall performance of the robot experienced a 77.78% success rate. It happened due to one DC motor gearbox experiencing mechanical problems.*

## PRAKATA

*Bismillahirrohmanirrohim*

Puji syukur ke hadirat Allah SWT atas segala rahmat dan hidayah-Nya sehingga laporan proyek akhir yang berjudul “*Rancang Bangun Robot Avoider Segala Medan Berbasis Arduino Mega 2560*” dapat terselesaikan dengan baik. Laporan proyek akhir ini disusun untuk memenuhi salah satu syarat dalam menyelesaikan pendidikan Diploma Tiga (DIII) pada Jurusan Teknik Elektro Fakultas Teknik Universitas Jember.

Terselesainya laporan proyek akhir ini tidak terlepas dari bantuan berbagai pihak, oleh karena itu disampaikan ucapan terima kasih kepada:

1. Allah Swt yang telah melimpahkan rahmat dan rizki-Nya serta memberi kelancaran dan kemudahan sehingga terselesainya proyek akhir ini.
2. Nabi Muhammad SAW yang telah membawa kita ke peradaban manusia yang lebih baik.
3. Bapak/Ibu, telah memberikan dorongan semangat, motivasi, dukungan dan doanya demi terselesainya proyek akhir ini.
4. Ibu Dr. Ir. Entin Hidayah M.U.M selaku Dekan Fakultas Teknik Universitas Jember.
5. Bapak Dr. Ir. Bambang Sri Kaloko, S.T., M.T. selaku Ketua Jurusan Fakultas Teknik Elektro Universitas Jember.
6. Bapak Satriyo Budi Utomo, S.T., M.T. selaku Ketua Prodi D3 Fakultas Teknik Elektro Universitas Jember.
7. Bapak M. Agung Prawira N, S.T., M.T. selaku dosen pembimbing utama dan Bapak Dr. Triwahju Hardianto, ST., MT. selaku dosen pembimbing anggota yang telah meluangkan waktu dan pikiran guna memberikan bimbingan dan pengarahan dalam penyusunan proyek akhir ini.
8. Seluruh Dosen yang ada di Fakultas Teknik khususnya Teknik Elektro beserta karyawan.

9. Keluarga besar Teknik Elektro khususnya angkatan 2013 INTEL UNEJ, terimakasih atas dukungan dan motivasi yang kalian berikan.
10. Teman – teman seperjuangan DEGAN UNEJ 2013 yang selalu mendukung selama menjalani masa kuliah sampai terselesaikannya proyek akhir ini, kenangan dan pengalaman tak akan pernah terlupakan.
11. Para penghuni Patrang yang telah meluangkan waktu membantu secara rohani maupun fisik.
12. Para penghuni Kontrakan B5/03 yang ikut mendukung baik rohani maupun jasmani.
13. Semua pihak yang tidak dapat saya sebutkan satu persatu, terima kasih atas dukungan dan motivasi kalian dalam penyusunan proyek akhir ini.

Penulis menyadari bahwa sebagai manusia biasa tidak terlepas dari keterbatasan, yang biasanya akan mewarnai kadar ilmiah dari proposal proyek akhir ini. Oleh karena itu penulis selalu terbuka terhadap masukan dan saran dari semua pihak yang sifatnya membangun untuk mendekati kesempurnaan. Tidak lupa penulis menyampaikan permohonan maaf yang sebesar-besarnya jika terdapat kesalahan dan kekeliruan. Akhir kata penulis berharap laporan ini dapat memberikan manfaat bagi pembaca dan dapat menjadi bahan acuan yang bermanfaat di kemudian hari.

Jember, 28 Juni 2016

Penulis

DAFTAR ISI

	Halaman
<b>HALAMAN SAMBUNG</b> .....	i
<b>HALAMAN JUDUL</b> .....	ii
<b>HALAMAN PERSEMBAHAN</b> .....	iii
<b>HALAMAN MOTTO</b> .....	iv
<b>HALAMAN PERNYATAAN</b> .....	v
<b>HALAMAN PEMBIMBING</b> .....	vi
<b>HALAMAN PENGESAHAN</b> .....	vii
<b>ABSTRAK</b> .....	viii
<b>ABSTRACT</b> .....	ix
<b>RINGKASAN</b> .....	x
<b>SUMMARY</b> .....	xi
<b>PRAKATA</b> .....	xii
<b>DAFTAR ISI</b> .....	xiv
<b>DAFTAR GAMBAR</b> .....	xvii
<b>DAFTAR TABEL</b> .....	xix
<b>BAB 1. PENDAHULUAN</b> .....	1
<b>1.1 Latar Belakang</b> .....	1
<b>1.2 Rumusan Masalah</b> .....	4
<b>1.3 Tujuan Penelitian</b> .....	4
<b>1.4 Batasan Masalah</b> .....	4
<b>1.5 Manfaat Penelitian</b> .....	5
<b>BAB 2. TINJAUAN PUSTAKA</b> .....	6
<b>2.1 Mobile Robot</b> .....	6
<b>2.2 Sensor Ultrasonic HC-SR04</b> .....	8
2.2.1 Spesifikasi Sensor HC-SR04 .....	8

2.2.2 Cara Kerja Sensor HC-SR04 .....	9
<b>2.3 LCD (<i>Liquid Crystal Display</i>)</b> .....	10
2.3.1 Kontroler LCD ( <i>Liquid Crystal Display</i> ) .....	11
<b>2.4 Arduino Mega 2560</b> .....	13
2.4.1 Sumber Daya .....	14
2.4.2 Memori .....	16
2.4.3 <i>Input dan Output</i> .....	16
2.4.4 Komunikasi .....	17
2.4.5 <i>Programing</i> .....	18
2.4.6 <i>Software</i> Arduino .....	18
2.4.7 Perangkat Lunak ( <i>Arduino IDE</i> ) .....	19
<b>2.5 Transistor</b> .....	19
2.5.1 Cara Kerja Transistor .....	20
2.5.2 Jenis-Jenis Transistor .....	20
<b>2.6 MOSFET</b> .....	22
<b>2.7 Driver Motor <i>H-Bridge</i> MOSFET</b> .....	23
2.7.1 Teori <i>H-Bridge</i> MOSFET .....	23
2.7.2 Konfigurasi <i>H-Bridge</i> MOSFET .....	24
<b>2.8 Motor DC</b> .....	25
<b>2.9 Gearbox Motor</b> .....	26
<b>BAB 3. METODOLOGI PENELITIAN</b> .....	28
<b>3.1 Tempat Penelitian</b> .....	28
<b>3.2 Alat dan Bahan</b> .....	28
3.2.1 Pembuatan Sensor .....	28
3.2.2 Pembuatan Tampilan LCD .....	28
3.2.3 Pembuatan <i>Driver</i> Motor DC .....	28
3.2.4 Pembuatan Catu Daya .....	28
3.2.5 <i>Software</i> .....	29
3.2.6 <i>Output</i> .....	29

3.2.7 Alat .....	29
<b>3.3 Blok Diagram .....</b>	<b>29</b>
<b>3.4 Perancangan Mekanik .....</b>	<b>31</b>
3.4.1 Desain Mekanik Robot .....	31
3.4.2 Bahan dan Bentuk Robot .....	32
3.4.3 Hasil Perancangan Mekanik Robot <i>Avoider</i> .....	33
<b>3.5 Perancangan Elektronik .....</b>	<b>38</b>
3.5.1 Rangkaian <i>Shield</i> Arduino Mega 2560 .....	38
3.5.2 Rangkaian Sensor <i>Ultrasonic</i> HC-SR04 .....	39
3.5.3 Rangkaian <i>Driver</i> Motor MOSFET .....	39
3.5.4 Rangkaian LCD .....	40
3.5.5 Rangkaian <i>Power Supply</i> .....	40
3.5.6 <i>Gearbox</i> Motor .....	41
<b>3.6 Perancangan Software .....</b>	<b>42</b>
3.6.1 <i>Flowchart</i> .....	42
3.6.2 Program Arduino .....	44
<b>3.7 Desain Lintasan .....</b>	<b>46</b>
<b>BAB 4. HASIL DAN PEMBAHASAN .....</b>	<b>47</b>
4.1 Pengujian Sensor Jarak <i>Ultrasonic</i> HC-SR04 .....	47
4.2 Pengujian Sensor <i>Ultrasonic</i> HC-SR04 pada jenis Halangan .....	50
4.3 Pengujian Robot dalam Keadaan Bekerja .....	54
4.4 Pengujian Robot dalam Menyusuri Lintasan .....	59
<b>BAB 5. PENUTUP.....</b>	<b>63</b>
5.1 Kesimpulan .....	63
5.2 Saran .....	63
<b>DAFTAR PUSTAKA .....</b>	<b>64</b>
<b>LAMPIRAN.....</b>	<b>66</b>

DAFTAR GAMBAR

	Halaman
2.1 Skema Roda dengan <i>Timing Belt (Tank)</i> .....	7
2.2 Sensor <i>Ultrasonic</i> HC-SR04 .....	8
2.3 Diagram Waktu Sensor HC-SR04 .....	9
2.4 Diagram Koneksi HC-SR04 dengan Arduino UNO .....	10
2.5 LCD ( <i>Liquid Cristal Display</i> ) 16x2 .....	11
2.6 Arduino Mega 2560 .....	13
2.7 Jenis-Jenis Transistor .....	21
2.8 Simbol Transistor MOSFET Mode <i>Depletion</i> .....	22
2.9 Simbol Transistor MOSFET Mode <i>Enhancement</i> .....	23
2.10 Rangkaian <i>H-Bridge</i> MOSFET .....	23
2.11 <i>H-Bridge</i> konfigurasi MOSFET A&D <i>on</i> , B&C <i>off</i> .....	24
2.12 <i>H-Bridge</i> konfigurasi MOSFET A&D <i>off</i> , B&C <i>on</i> .....	25
2.13 Motor DC .....	26
2.14 <i>Gearbox</i> Motor .....	27
3.1 Blok Diagram .....	29
3.2 Perancangan Mekanik Robot <i>Avoider</i> Segala Medan.....	31
3.3 Bentuk Robot <i>Avoider</i> .....	33
3.4 Bodi Robot <i>Avoider</i> Nampak Atas .....	34
3.5 Bodi Robot <i>Avoider</i> Nampak Samping .....	35
3.6 Bodi Robot Bagian Bawah .....	35
3.7 Sensor Kanan dan Kiri .....	36
3.8 Sensor Depan .....	37
3.9 Sensor Belakang.....	37
3.10 Aktuator atau Roda Robot .....	38
3.11 Rangkaian <i>Shield</i> Arduino Mega 2560 .....	38
3.12 Rangkaian Sensor <i>Ultrasonic</i> HC-SR04 .....	39

3.13 Rangkaian <i>Driver</i> MOSFET .....	40
3.14 Rangkaian LCD .....	40
3.15 Rangkaian <i>Power Supply</i> .....	41
3.16 <i>Gearbox</i> Motor 1 .....	41
3.17 <i>Gearbox</i> Motor 2 .....	41
3.18 <i>Flowchart</i> Robot Maju .....	42
3.19 <i>Flowchart</i> Robot Mundur .....	43
3.20 Lintasan Labirin .....	46
4.1 Pengujian Sensor <i>Ultrasonic</i> HC-SR04 .....	54
4.2 Denah Rute Lintasan Robot .....	59

**DAFTAR TABEL**

	Halaman
2.1 Tabel Deskripsi Arduino Mega 2560 .....	14
4.1 Tabel Pengujian Sensor <i>Ultrasonic</i> HC-SR04 .....	47
4.2 Tabel Pengujian Sensor <i>Ultrasonic</i> HC-SR04 pada Jenis Halangan .	51
4.3 Tabel Pengujian Robot Bergerak Maju .....	55
4.4 Tabel Pengujian Robot Bergerak Belok Kanan .....	55
4.5 Tabel Pengujian Robot Bergerak Belok Kiri .....	56
4.6 Tabel Pengujian Robot Bergerak Mundur .....	56
4.7 Tabel Pengujian Robot Bergerak dalam Tanjakan-Turunan .....	57
4.8 Tabel Pengujian Robot dalam Lintasan Tidak Rata .....	58
4.9 Tabel Pengujian Robot Ketika Terdapat Pertigaan .....	58
4.10 Tabel Pengujian Robot dalam Menyusuri Lintasan .....	60

## BAB 1. PENDAHULUAN

### 1.1 Latar Belakang

Bencana alam adalah bencana yang diakibatkan oleh peristiwa atau serangkaian peristiwa yang disebabkan oleh gejala-gejala alam yang dapat mengakibatkan kerusakan lingkungan, kerugian materi, maupun korban manusia. (Kamadhis UGM, 2007)

Bencana alam dapat terjadi secara tiba-tiba maupun melalui proses yang berlangsung secara perlahan. Beberapa jenis bencana seperti gempa bumi, hampir tidak mungkin diperkirakan secara akurat kapan, dimana akan terjadi dan besaran kekuatannya. Sedangkan beberapa bencana lainnya seperti banjir, tanah longsor, kekeringan, letusan gunung api, tsunami dan anomali cuaca masih dapat diramalkan sebelumnya. Meskipun demikian kejadian bencana selalu memberikan dampak kejutan dan menimbulkan banyak kerugian baik jiwa maupun materi. Kejutan tersebut terjadi karena kurangnya kewaspadaan dan kesiapan dalam menghadapi ancaman bahaya. (BARKONAS PB, 2007)

Gempa bumi disebabkan oleh adanya pelepasan energi regangan elastis batuan pada litosfir. Semakin besar energi yang dilepas semakin kuat gempa yang terjadi. (Kamadhis UGM, 2007). Sehingga menimbulkan kerusakan yang parah ketika kekuatan gempa mengalami peningkatan. Kerusakan tersebut dapat meruntuhkan sebuah gedung dengan konstruksi yang kuat sekalipun. Akibat gempa tersebut menimbulkan adanya korban dari pihak manusia, yaitu ketika manusia tertimpa reruntuhan gedung yang kemudian menimbulkan kematian.

Suatu bencana alam tidak dapat diprediksi dan datang dengan tiba-tiba. Terlebih lagi disebagian besar wilayah Indonesia sering terjadi bencana alam. Misalkan saja gempa bumi yang dapat meruntuhkan gedung-gedung dan lain sebagainya. Bahkan gempa bumi dapat memicu terjadinya tsunami yang terjadi di Aceh pada tahun 2004 lalu. (Kamadhis UGM, 2007). Bencana tersebut merenggut

korban hingga ratusan ribu jiwa. Dalam pencarian korban bencana tersebut mengalami kesulitan akibat reruntuhan gedung ataupun rumah. Sehingga untuk penyelamatan korban yang masih hidup dan pencarian korban mengalami kesulitan. Relawan harus memindahkan puing-puing gedung dengan hati-hati, jikalau masih ada korban yang selamat. Sistem pencarian tersebut membutuhkan waktu yang lama, pencarian tersebut juga dapat mengancam jiwa dari relawan itu sendiri. Pertama, relawan dapat terserang penyakit dari mayat yang sudah membusuk. Kedua, relawan bisa saja tertimpa reruntuhan gedung yang diakibatkan gempa sebelumnya.

Robot *avoider* ini adalah pengembangan robot *avoider* di CV. Rekayasa Nusantara yang merupakan tempat Kerja Praktek untuk melaksanakan matakuliah Kerja Praktek. Robot *avoider* di CV. Rekayasa Nusantara hanya menggunakan satu sensor yang berada didepan dengan digerakkan kekanan kekiri menggunakan sebuah servo. Robot ini hanya bisa mendeteksi bidang dinding biasa, untuk beberapa jenis dinding yang lain masih belum dilakukan pengujian. Desain robot ini hanya dibuat untuk bidang rata saja, sebab menggunakan roda biasa. Roda depan menggunakan roda bebas sedangkan robot digerakan dengan menggunakan *black gearbox* motor 200rpm saja. Sehingga robot *avoider* di CV. Rekayasa Nusantara tidak memungkinkan digunakan untuk segala medan.

Robot penghindar halangan (*avoider*) ialah robot yang sengaja dirancang untuk dapat menghindari penghalang yang berada disekitarnya. (Nauriana, 2009). Sehingga robot dapat menghindari terjadi benturan terhadap penghalang disekitarnya. (Maulana Mujahidin, 2008). Dengan demikian robot tersebut dapat mencari jalan keluar yang tidak terdapat halangannya. Robot ini akan mengitari lingkungan disekitarnya, kemudian akan mencari jalan keluar dari sebuah labirin yang sudah ada. Robot *avoider* ini memiliki 2 buah motor, yang kemudian akan dioperasikan dalam segala medan dengan menambahkan roda seperti *tank*. Agar menyerupai keadaan aslinya, dengan desain robot menggunakan roda yang besar dan motor DC yang memiliki torsi yang besar kemudian dilengkapi desain roda seperti *tank*.

Robot *avoider* ini juga dibantu dengan sensor HC-SR04 yang memiliki fungsi sebagai sensor jarak. Sehingga jarak halangan terhadap pusat robot dapat diketahui dengan menggunakan sensor *ultrasonic*. Dengan pembacaan jarak ini robot dapat menghindari halangan berupa dinding, kaca, maupun dinding yang tidak rata. Yang kemudian dirubah menjadi beberapa perintah yang dilaksanakan robot *avoider* sesuai fungsinya.

Dari hasil riset sensor *ultrasonic* tidak mengalami kendala untuk pembacaan sensor jarak dalam bidang datar maupun bidang tidak rata. Hasil pengujian terhadap obyek benda hitam, putih, dinding biasa, kaca tidak mengalami perubahan yang signifikan sedangkan pengujian terhadap obyek dengan bidang tidak rata mengalami pengukuran dengan jarak terjauh dari obyek dengan permukaan tidak rata. Dengan hasil penelitian tersebut dapat diambil kesimpulan bahwa sensor *ultrasonic* dapat mendeteksi obyek tanpa terpengaruh perbedaan warna benda ataupun kaca dan akan mendeteksi jarak terjauh dari posisi obyek didepan sensor. Secara umum semakin jauh jarak yang diukur, semakin besar persen kesalahan yang terjadi.

Prinsip kerja sebuah modul sensor *ultrasonic* yaitu mendeteksi objek dengan cara mengirimkan gelombang *ultrasonic* dan kemudian menerima pantulan gelombang tersebut. Sensor hanya akan mengirimkan gelombang *ultrasonic* ketika ada pulsa *trigger* dari mikrokontroler (*Pulsa high* selama 5 $\mu$ S). Gelombang *ultrasonic* dengan frekuensi sebesar 40KHz akan dipancarkan selama 200 $\mu$ S. Gelombang ini akan merambat di udara dengan kecepatan 344.424 m / detik (atau 1 cm setiap 29.034 $\mu$ S), mengenai objek untuk kemudian terpantul kembali ke *ultrasonic*. (Hadijaya Pratama, 2012)

Dari hasil pembacaan sensor tersebut kemudian perintah yang sudah ada dalam Arduino Mega dijalankan sesuai pembacaan sensor. Robot melakukan belok kiri maupun kanan ataupun mundur. Hasil pembacaan sensor juga ditampilkan juga dalam LCD sebagai hasil pembacaan sensor jarak atau ultrasonik.

Robot *avoider* digunakan untuk menarik orang keluar dari bahaya, buruk, atau lingkungan kerja berat dengan kapasitas kerja otonom.(Xue Yang, 2014) Dengan

demikian robot tersebut dalam menyelamatkan seseorang dalam mencari jalan keluar dari labirin ataupun bangunan yang dapat membingungkan beberapa orang.

## 1.2 Rumusan Masalah

Berdasarkan uraian latar belakang, maka permasalahan yang diteliti dalam proyek akhir ini dapat dirumuskan sebagai berikut:

1. Bagaimana desain robot *avoider* segala medan berbasis Arduino Mega 2560?
2. Bagaimana cara mendesain robot *avoider* menggunakan sensor *ultrasonic* ?

## 1.3 Tujuan Penelitian

Tujuan dibuatnya alat ini adalah sebagai berikut:

1. Menghasilkan robot *avoider* segala medan berbasis Arduino Mega 2560.
2. Menghasilkan robot *avoider* dengan menggunakan sensor *ultrasonic*.

## 1.4 Batasan Masalah

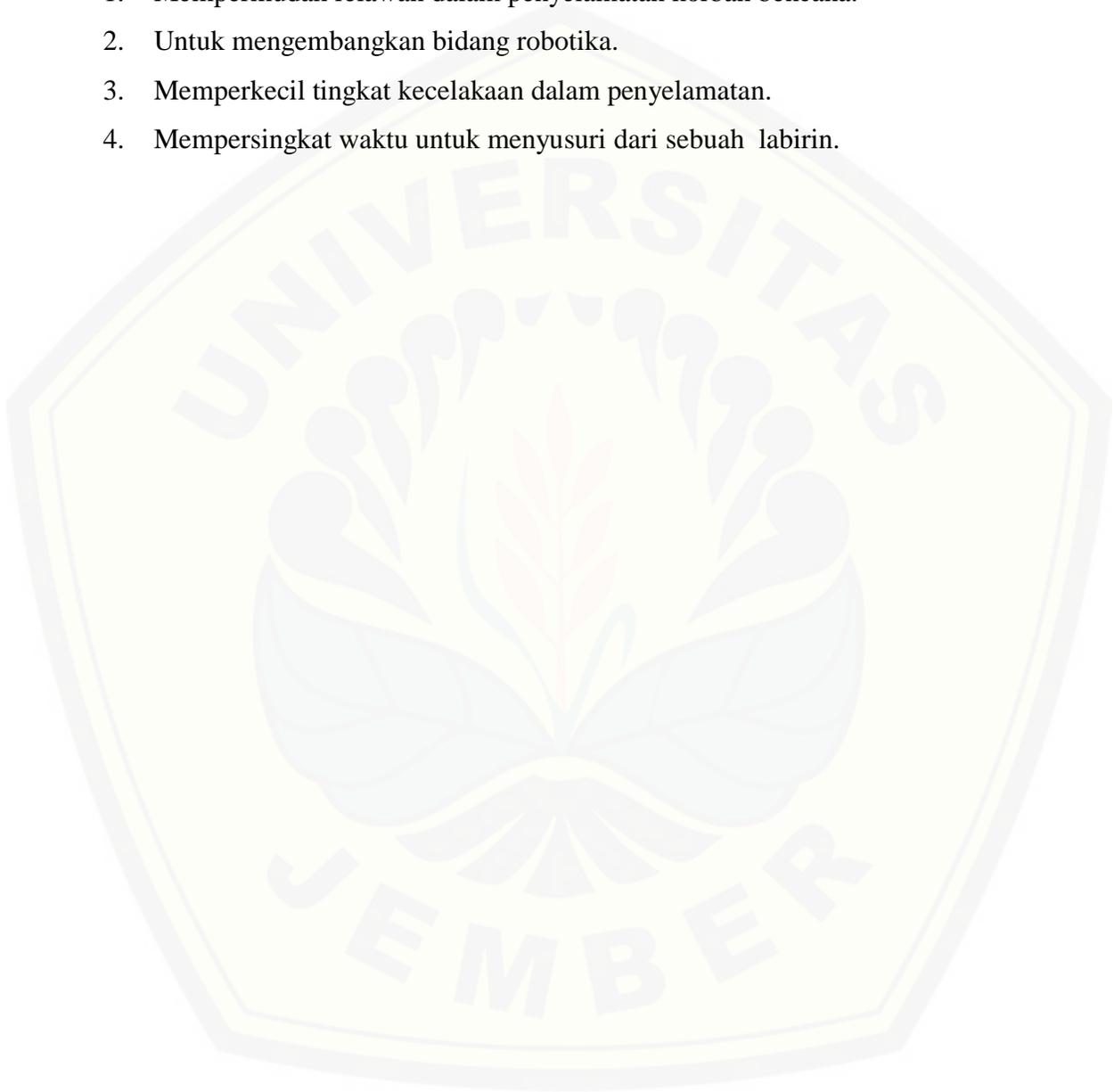
Ruang lingkup penelitian dalam proyek akhir ini sebagai berikut:

1. Sensor yang digunakan dalam prototipe robot *avoider* adalah sensor jarak HC-SR04.
2. Sistem kontrol yang digunakan berbasis Arduino.
3. Alat ini bekerja pada dinding biasa, dinding berwarna hitam maupun putih, kaca, dinding biasa dan permukaan tidak rata.
4. Robot hanya berjalan di beberapa rintangan yang sudah tersedia pada labirin.
5. Robot berjalan dari rute satu ke rute lainnya, yang sudah ditentukan.
6. Diaplikasikan dalam medan labirin.
7. Parameter yang digunakan robot untuk menyusuri dari sebuah labirin tidak lebih dari 5 menit.

### **1.5 Manfaat Penelitian**

Manfaat yang diharapkan dengan adanya alat ini adalah:

1. Mempermudah relawan dalam penyelamatan korban bencana.
2. Untuk mengembangkan bidang robotika.
3. Memperkecil tingkat kecelakaan dalam penyelamatan.
4. Mempersingkat waktu untuk menyusuri dari sebuah labirin.



## BAB 2. TINJAUAN PUSTAKA

### 2.1 *Mobile Robot*

Kata “robot” berasal dari bahasa Chech (Ceko) yang berarti pekerja. Saat ini secara sadar atau tidak sadar, robot telah hadir di dalam kehidupan manusia dalam bentuk yang bermacam-macam. Terdapat bentuk disain robot yang sederhana untuk mengerjakan kegiatan mudah berulang-ulang. Ada pula robot yang dirancang untuk “berperilaku” sangat kompleks dan sampai batas tertentu untuk mengontrol dirinya sendiri.

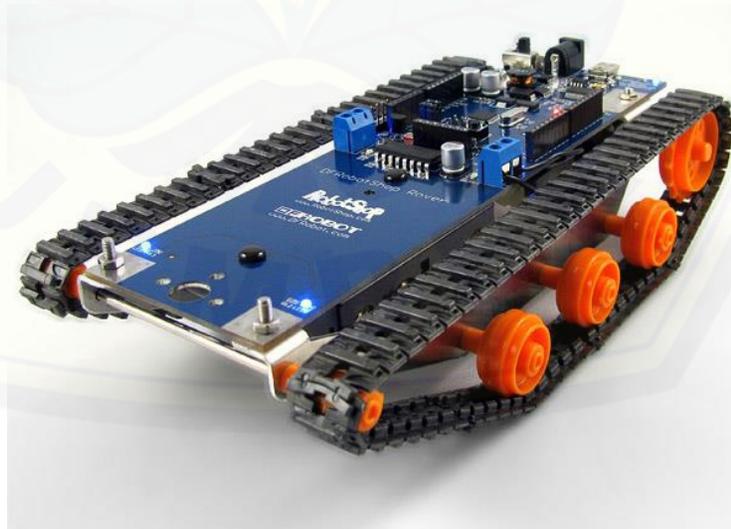
Robot memiliki karakter utama yaitu *sensing*, robot memiliki sensor untuk dapat merasakan keadaan lingkungan sekitarnya. *Movement*, robot memiliki roda ataupun kaki untuk dapat bergerak di lingkungan sekitarnya. *Energy*, robot membutuhkan energi untuk dapat bergerak. *Intelligence*, robot dapat berfikir dan memutuskan tindakan (cerdas). Kemampuan berfikir dan tingkat kecerdasannya ini sangat ditentukan oleh pencipta robot yang membuat algoritma pemrograman. (Endy Sjaiful Alim, 2012)

*Mobile* dapat diartikan bergerak, sehingga robot ini dapat memindahkan dirinya dari satu tempat ke tempat lain. Robot ini merupakan robot yang paling populer dalam dunia penelitian robotika. Dari segi manfaat, robot ini diharapkan dapat membantu manusia dalam melakukan otomasi dalam transportasi, platform bergerak untuk robot industri, eksplorasi tanpa awak, dan masih banyak lagi. (Raden Supriyanto, 2010)

Sistem mekanika dari robot beroda adalah robot yang menggunakan beberapa roda sebagai aktuator. Penggunaan roda pada sebuah robot merupakan teknik yang paling mudah dan paling efisien untuk pergerakan robot pada permukaan datar. Pemilihan roda sering digunakan pada robot karena memberikan *traction* yang bagus, mudah diperoleh dan dipakai, dan dalam pemasangannya cukup mudah.

Robot yang seringkali dijumpai adalah robot yang bergerak dengan menggunakan roda. Roda merupakan teknik tertua, paling mudah, dan paling efisien untuk menggerakkan robot melintasi permukaan datar. Roda seringkali dipilih, karena memberikan *traction* yang bagus, mudah diperoleh dan dipakai, dan juga mudah untuk memasangnya pada robot. *Traction* merupakan variabel dari material roda dan permukaan yang dilintasi oleh roda. Material roda yang lebih lembut memiliki koefisien *traction* yang besar, dan koefisien *traction* yang besar ini member gesekan (*friction*) yang besar pula, dan memperbesar daya yang dibutuhkan untuk menggerakkan motor. Jumlah roda yang digunakan pada robot beragam, dan dipilih sesuai selera si pembuat robot. Robot dapat dibangun dengan menggunakan berbagai macam roda, misalnya beroda dua, beroda empat, beroda enam, atau beroda caterpillar (*tank-treaded*). (B Anggoro, 2013)

Robot *tank* ini digunakan agar ketika mendapatkan atau mengalami keadaan medan yang tidak rata, robot masih dalam keadaan bergerak. Penggunaan *timing belt* membantu robot dalam melakukan pergerakan dalam keadaan segala medan. Karena roda yang terpasang mengalami pergerakan sama, sehingga kemungkinan roda mengalami selip hampir tidak ada.



Gambar 2.1 Skema Roda dengan *Timing Belt* (*Tank*)

(Sumber : Paul Strauss, 2010)

## 2.2 Sensor Ultrasonic HC-SR04

Sensor ultrasonik adalah sebuah sensor yang berfungsi untuk mengubah besaran fisis (bunyi) menjadi besaran listrik dan sebaliknya. Cara kerja sensor ini didasarkan pada prinsip dari pantulan suatu gelombang suara sehingga dapat dipakai untuk menafsirkan eksistensi (jarak) suatu benda dengan frekuensi tertentu. Disebut sebagai sensor ultrasonik karena sensor ini menggunakan gelombang ultrasonik (bunyi ultrasonik).

HC-SR04 memiliki kinerja yang baik dalam mendeteksi jarak, dengan tingkat akurasi yang tinggi serta deteksi yang stabil. Sensor ini merupakan sensor siap pakai yang memiliki fungsi sebagai pengirim, penerima, dan pengontrol gelombang ultrasonik. Alat ini bisa digunakan untuk mengukur jarak benda dari 2cm – 4m dengan akurasi 3mm. Alat ini memiliki 4 pin, pin *Vcc*, *Gnd*, *Trigger*, dan *Echo*. Pin *Vcc* untuk listrik positif dan *Gnd* untuk *ground*. Pin *Trigger* untuk trigger keluarnya sinyal dari sensor dan pin *Echo* untuk menangkap sinyal pantul dari benda. (Hari Santoso, 2015)



Gambar 2.2 Sensor Ultrasonic HC-SR04

(Sumber : Hari Santoso, 2015)

### 2.2.1 Spesifikasi Sensor HC-SR04

- Catu Daya : 5V DC
- Arus pada moda siaga : < 2mA

- Konsumsi arus saat deteksi : 15mA
- Lebar sudut deteksi : 15°
- Jarak deteksi : akurat hingga 1 meter, dapat mendeteksi (namun kurang presisi) hingga jarak 4 meter
- Resolusi : 3 mm (perhitungan dari faktor kecepatan rambat suara dan kecepatan MCU pada 16 MHz)
- Dimensi : 45 x 20 x 15 mm

### 2.2.2 Cara Kerja Sensor HC-SR04

Berikut diagram waktu dari penggunaan sensor ini :



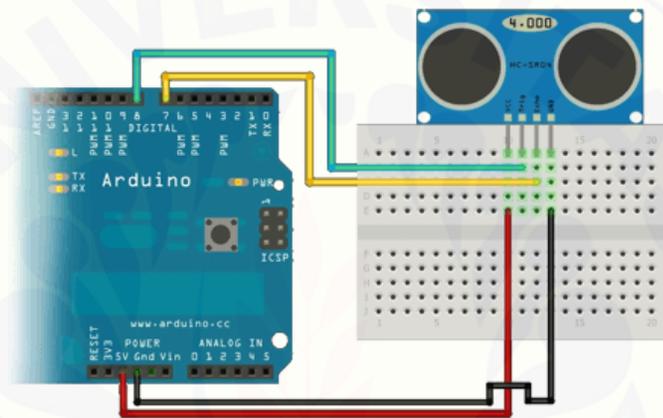
Gambar 2.3 Diagram Waktu Sensor HC-SR04

(Sumber : [www.Vcc2GND/sku/HC-SR04.com](http://www.Vcc2GND/sku/HC-SR04.com) )

Bangkitnya pulsa pendek sepanjang 10µs sebagai sinyal pemicu ke pin picu masukan dari modul untuk memulai pendeteksian, HC-SR04 akan memancarkan 8 siklus gelombang ultrasonik pada frekuensi 40 KHz. Ketika gelombang suara ini menabrak objek (benda atau dinding didepannya), gelombang akan dipantulkan balik dan diterima oleh detektor yang kemudian membangkitkan sinyal deteksi di pin keluaran modul. Lama selang waktu antara pengiriman *signal* hingga pendeteksian sinyal pantulan adalah waktu yang ditempuh gelombang suara, yaitu sepanjang dua kali jarak antara sensor dan objek yang terdeteksi karena *signal* berjalan pulang-pergi.

Dengan mengetahui selang waktu ini dan kecepatan rambat suara di udara (340 meter/detik pada udara kering, atau  $3,4 \times 10^8 \mu\text{s}$ ), jarak dapat dihitung sesuai rumus jarak = kecepatan x waktu.

Rumus matematika yang disederhanakan untuk menghitung jarak tersebut dalam sentimeter adalah lama selang waktu dalam mikrodetik (sepersejuta detik) dibagi 58,2. Berikut contoh diagram koneksi antara Arduino Uno dengan modul sensor jarak ultrasonik HC-SR04 : ([www.Vcc2GND/sku/HC-SR04.com](http://www.Vcc2GND/sku/HC-SR04.com))



Gambar 2.4 Diagram Koneksi HC-SR04 dengan Arduino UNO

(Sumber : [www.Vcc2GND/sku/HC-SR04.com](http://www.Vcc2GND/sku/HC-SR04.com))

### 2.3 LCD (*Liquid Cristal Display*)

LCD (*Liquid Cristal Display*) adalah sebuah salah satu display elektronik yang dibuat dengan teknologi CMOS logic yang bekerja tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap *front-lit* atau mentransmisikan cahaya dari *back-lit*. Display elektronik adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. LCD (*Liquid Cristal Display*) memiliki fungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik.

LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan seven-segment dan lapisan

elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. Lapisan *sandwich* memiliki *polarizer* cahaya vertikal depan dan *polarizer* cahaya horisontal belakang yang diikuti dengan lapisan reflektor. Cahaya yang dipantulkan tidak dapat melewati molekul-molekul yang telah menyesuaikan diri dan segmen yang diaktifkan terlihat menjadi gelap dan membentuk karakter data yang ingin ditampilkan.



Gambar 2.5 LCD (*Liquid Cristal Display*) 16x2

(Sumber : Agus Purnama, 2012)

### 2.3.1 Kontroler LCD (*Liquid Cristal Display*)

Dalam modul LCD (*Liquid Cristal Display*) terdapat mikrokontroler yang berfungsi sebagai pengendali tampilan karakter LCD (*Liquid Cristal Display*). Mikrokontroler pada suatu LCD (*Liquid Cristal Display*) dilengkapi dengan memori dan register. Memori yang digunakan mikrokontroler *internal* LCD adalah :

- DDRAM (*Display Data Random Access Memory*) merupakan memori tempat karakter yang akan ditampilkan berada.
- CGRAM (*Character Generator Random Access Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai dengan keinginan.

- CGROM (*Character Generator Read Only Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut merupakan karakter dasar yang sudah ditentukan secara permanen oleh pabrikan pembuat LCD (*Liquid Cristal Display*) tersebut sehingga pengguna tinggal mengambilnya sesuai alamat memorinya dan tidak dapat merubah karakter dasar yang ada dalam CGROM.

Register control yang terdapat dalam suatu LCD diantaranya adalah :

- Register perintah yaitu register yang berisi perintah-perintah dari mikrokontroler ke panel LCD (*Liquid Cristal Display*) pada saat proses penulisan data atau tempat status dari panel LCD (*Liquid Cristal Display*) dapat dibaca pada saat pembacaan data.
- Register data yaitu register untuk menuliskan atau membaca data dari atau ke DDRAM. Penulisan data pada register akan menempatkan data tersebut ke DDRAM sesuai dengan alamat yang telah diatur sebelumnya.

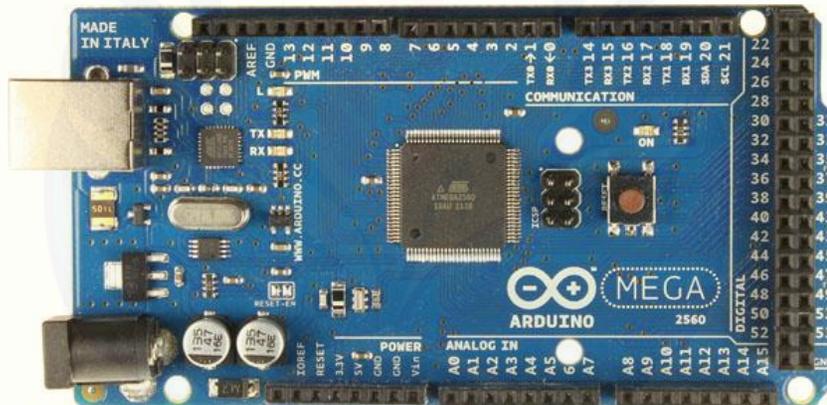
Pin, kaki atau jalur input dan kontrol dalam suatu LCD (*Liquid Cristal Display*) diantaranya adalah :

- Pin data adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan LCD (*Liquid Cristal Display*) dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit.
- Pin RS (*Register Select*) berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika *low* menunjukkan yang masuk adalah perintah, sedangkan logika *high* menunjukkan data.
- Pin R/W (*Read Write*) berfungsi sebagai instruksi pada modul jika *low* tulis data, sedangkan *high* baca data.
- Pin E (*Enable*) digunakan untuk memegang data baik masuk atau keluar.
- Pin VLCD berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 K ohm, jika tidak digunakan dihubungkan ke

*ground*, sedangkan tegangan catu daya ke LCD sebesar 5 Volt. (Agus Purnama, 2012)

#### 2.4 Arduino Mega 2560

Arduino Mega2560 adalah papan mikrokontroler berbasis ATmega2560. Arduino Mega2560 memiliki 54 pin digital *input* atau *output*, dimana 15 pin dapat digunakan sebagai *output* PWM, 16 pin sebagai *input analog*, dan 4 pin sebagai UART (*port serial hardware*), 16 MHz kristal osilator, koneksi USB, *jack power*, *header ICSP*, dan tombol *reset*. Ini semua yang diperlukan untuk mendukung mikrokontroler. Cukup dengan menghubungkannya ke komputer melalui kabel USB atau *power* dihubungkan dengan adaptor AC-DC atau baterai untuk mulai mengaktifkannya. (Dede Hendriono, 2014)



Gambar 2.6 Arduino Mega 2560

(Sumber : Zerfani Yulias, 2013)

Arduino Mega 2560 adalah *board* Arduino yang merupakan perbaikan dari *board* Arduino Mega sebelumnya. Arduino Mega awalnya memakai chip Atmega1280 dan kemudian diganti dengan chip ATmega2560, oleh karena itu namanya diganti menjadi Arduino Mega 2560. Arduino Mega 2560 sudah sampai pada revisinya yang ke 3 (R3). (Zerfani Yulias, 2013)

Arduino Mega 2560 Revisi 3 memiliki fitur-fitur baru sebagai berikut :

- 1.0 pin *out* : Ditambahkan pin SDA dan pin SCL yang dekat dengan pin AREF dan dua pin baru lainnya ditempatkan dekat dengan pin *RESET*, IOREF memungkinkan *shield* untuk beradaptasi dengan tegangan yang tersedia pada papan. Di masa depan, *shield* akan kompatibel baik dengan papan yang menggunakan AVR yang beroperasi dengan 5 Volt dan dengan Arduino Due yang beroperasi dengan tegangan 3.3 Volt. Dan ada dua pin yang tidak terhubung, yang disediakan untuk tujuan masa depan.
- Sirkuit RESET.
- Chip ATmega16U2 menggantikan chip ATmega8U2. (Dede Hendriono, 2014)

#### Deskripsi Arduino Mega 2560

Tabel 2.1 Deskripsi Arduino Mega 2560

Mikrokontroler	ATmega2560
Tegangan Operasi	5 V
<i>Input Voltage</i> (disarankan)	7-12 V
<i>Input Voltage</i> (limit)	6-20 V
Pin Digital I/O	54 (15 pin digunakan sebagai <i>Output PWM</i> )
Pin <i>Input</i> Analog	16
Arus DC per pin I/O	40 mA
Arus DC per pin 3.3 V	50 mA
<i>Flash Memory</i>	256 kb (8 kb digunakan untuk <i>bootloader</i> )
SSRAM	8 kb
EEPROM	4 kb
<i>Clock Speed</i>	16 MHz

(Sumber : Dede Hendriono, 2014)

#### 2.4.1 Sumber Daya

Arduino Mega dapat diaktifkan melalui koneksi USB atau dengan catu daya eksternal. Sumber daya dipilih secara otomatis. Sumber daya eksternal (non-USB)

dapat berasal baik dari adaptor AC-DC atau baterai. Adaptor dapat dihubungkan dengan mencolokkan steker 2,1 mm yang bagian tengahnya terminal positif ke *jack* sumber tegangan pada papan. Jika tegangan berasal dari baterai dapat langsung dihubungkan melalui *header* pin Gnd dan pin Vin dari konektor power.

Papan Arduino ATmega2560 dapat beroperasi dengan pasokan daya eksternal 6 Volt sampai 20 volt. Jika diberi tegangan kurang dari 7 Volt, maka, pin 5 Volt mungkin akan menghasilkan tegangan kurang dari 5 Volt dan ini akan membuat papan menjadi tidak stabil. Jika sumber tegangan menggunakan lebih dari 12 Volt, regulator tegangan akan mengalami panas berlebihan dan bisa merusak papan. Rentang sumber tegangan yang dianjurkan adalah 7 Volt sampai 12 Volt.

Pin tegangan yang tersedia pada papan Arduino sebagai berikut :

- VIN : Adalah *input* tegangan untuk papan Arduino ketika menggunakan sumber daya eksternal (sebagai ‘saingan’ tegangan 5 Volt dari koneksi USB atau sumber daya ter-regulator lainnya). Anda dapat memberikan tegangan melalui pin ini, atau jika memasok tegangan untuk papan melalui *jack power*, kita bisa mengakses atau mengambil tegangan melalui pin ini.
- 5V : Sebuah pin yang mengeluarkan tegangan ter-regulator 5 Volt, dari pin ini tegangan sudah diatur (ter-regulator) dari regulator yang tersedia (built-in) pada papan. Arduino dapat diaktifkan dengan sumber daya baik berasal dari jack power DC (7-12 Volt), konektor USB (5 Volt), atau pin VIN pada board (7-12 Volt). Memberikan tegangan melalui pin 5V atau 3.3V secara langsung tanpa melewati regulator dapat merusak papan Arduino.
- 3V3 : Sebuah pin yang menghasilkan tegangan 3,3 Volt. Tegangan ini dihasilkan oleh regulator yang terdapat pada papan (*on-board*). Arus maksimum yang dihasilkan adalah 50 mA.
- GND : Pin *Ground* atau *Massa*.
- IOREF : Pin ini pada papan Arduino berfungsi untuk memberikan referensi tegangan yang beroperasi pada mikrokontroler. Sebuah perisai (*shield*) dikonfigurasi dengan benar untuk dapat membaca pin tegangan IOREF dan

memilih sumber daya yang tepat atau mengaktifkan penerjemah tegangan (*voltage translator*) pada *output* untuk bekerja pada tegangan 5 Volt atau 3,3 Volt.

#### 2.4.2 Memori

Arduino ATmega2560 memiliki 256 KB *flash memory* untuk menyimpan kode (yang 8 KB digunakan untuk *bootloader*), 8 KB SRAM dan 4 KB EEPROM (yang dapat dibaca dan ditulis dengan perpustakaan EEPROM).

#### 2.4.3 Input dan Output

Masing-masing dari 54 digital pin pada Arduino Mega dapat digunakan sebagai *input* atau *output*, menggunakan fungsi *pinMode()*, *digitalWrite()*, dan *digitalRead()*. Arduino Mega beroperasi pada tegangan 5 volt. Setiap pin dapat memberikan atau menerima arus maksimum 40 mA dan memiliki resistor pull-up internal (yang terputus secara default) sebesar 20-50 kOhms. Selain itu, beberapa pin memiliki fungsi khusus, antara lain :

- Serial : 0 (RX) dan 1 (TX); Serial 1 : 19 (RX) dan 18 (TX); Serial 2 : 17 (RX) dan 16 (TX); Serial 3 : 15 (RX) dan 14 (TX). Digunakan untuk menerima (RX) dan mengirimkan (TX) data serial TTL. Pins 0 dan 1 juga terhubung ke pin chip ATmega16U2 Serial USB-to-TTL.
- Eksternal Interupsi : Pin 2 (*interrupt 0*), pin 3 (*interrupt 1*), pin 18 (*interrupt 5*), pin 19 (*interrupt 4*), pin 20 (*interrupt 3*), dan pin 21 (*interrupt 2*). Pin ini dapat dikonfigurasi untuk memicu sebuah interupsi pada nilai yang rendah, meningkat atau menurun, atau perubah nilai.
- SPI : Pin 50 (MISO), pin 51 (MOSI), pin 52 (SCK), pin 53 (SS). Pin ini mendukung komunikasi SPI menggunakan perpustakaan SPI. Pin SPI juga terhubung dengan *header ICSP*, yang secara fisik kompatibel dengan Arduino Uno, Arduino Duemilanove dan Arduino Diecimila.

- LED : Pin 13. Tersedia secara *built-in* pada papan Arduino ATmega2560. LED terhubung ke pin digital 13. Ketika pin diset bernilai *HIGH*, maka LED menyala (*ON*), dan ketika pin diset bernilai *LOW*, maka LED padam (*OFF*).
- TWI : Pin 20 (SDA) dan pin 21 (SCL). Yang mendukung komunikasi TWI menggunakan perpustakaan *Wire*. Perhatikan bahwa pin ini tidak dilokasi yang sama dengan pin TWI pada Arduino Duemilanove atau Arduino Diecimila.

Arduino Mega2560 memiliki 16 pin sebagai *analog input*, yang masing-masing menyediakan resolusi 10 bit (yaitu 1024 nilai yang berbeda). Secara *default* pin ini dapat diatur maupun diukur dari mulai *ground* sampai dengan 5 Volt, juga memungkinkan untuk mengubah titik jangkauan tertinggi atau terendah mereka menggunakan pin AREF dan fungsi *analogReference()*.

Ada beberapa pin lainnya yang tersedia, antara lain :

- AREF : Referensi tegangan untuk *input analog*. Digunakan dengan fungsi *analogReference()*.
- RESET : Jalur *LOW* ini digunakan untuk *me-reset* (menghidupkan ulang) mikrokontroler. Jalur ini biasanya digunakan untuk menambahkan tombol *reset* pada *shield* yang menghalangi papan utama Arduino.

#### 2.4.4 Komunikasi

Arduino Mega2560 memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, dengan Arduino lain, atau dengan mikrokontroler lainnya. Arduino ATmega328 menyediakan 4 *hardware* komunikasi serial UART TTL (5 Volt). Sebuah chip ATmega16U2 (ATmega8U2 pada papan Revisi 1 dan Revisi 2) yang terdapat pada papan digunakan sebagai media komunikasi serial melalui USB dan muncul sebagai COM Port Virtual (pada *Device* komputer) untuk berkomunikasi dengan perangkat lunak pada komputer, untuk sistem operasi *Windows* masih tetap memerlukan file inf, tetapi untuk sistem operasi OS X dan *Linux* akan mengenali papan sebagai *port* COM secara otomatis. Perangkat lunak Arduino termasuk

didalamnya serial monitor memungkinkan data tekstual sederhana dikirim ke dan dari papan Arduino. LED RX dan TX yang tersedia pada papan akan berkedip ketika data sedang dikirim atau diterima melalui chip *USB-to-serial* yang terhubung melalui USB komputer (tetapi tidak untuk komunikasi serial seperti pada pin 0 dan 1).

#### 2.4.5 *Programming*

Arduino Mega dapat diprogram dengan perangkat lunak *software* Arduino. ATmega2560 pada Arduino Mega sudah tersedia *preburned* dengan *bootloader* yang memungkinkan untuk meng-*upload* kode baru tanpa menggunakan *programer hardware eksternal*. Hal ini karena komunikasi yang terjadi menggunakan protokol asli STK500. Juga dapat melewati (*bypass*) *bootloader* dan program mikrokontroler melalui pin *header ICSP (In-Circuit Serial Programming)*. Chip ATmega16U2 (atau 8U2 pada *board Rev. 1 dan Rev. 2*) *source code firmware* tersedia pada repositori Arduino. ATmega16U2/8U2 dapat dimuat dengan *bootloader DFU*, yang dapat diaktifkan melalui :

- Pada papan Revisi 1 : Menghubungkan jumper solder di bagian belakang papan (dekat dengan peta Italia) dan kemudian akan me-reset 8U2.
- Pada papan Revisi 2 : Ada resistor yang menghubungkan jalur HWB 8U2/16U2 ke ground, sehingga lebih mudah untuk dimasukkan ke dalam mode DFU. (Dede Hendriono, 2014)

#### 2.4.6 *Software Arduino*

Arduino Mega dapat diprogram dengan menggunakan perangkat lunak Arduino. Pada ATmega2560 sudah tersedia *preburned* dengan *bootloader* yang memungkinkan untuk meng-*upload* kode baru tanpa menggunakan *programer hardware eksternal*.

#### 2.4.7 Perangkat Lunak (*Arduino IDE*)

Arduino dikatakan sebagai sebuah *platform* dari *physical computing* yang bersifat *open source*. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Environment* (IDE) yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-*upload* ke dalam memori mikrokontroler. Ada banyak projek dan alat-alat dikembangkan oleh akademisi dan profesional dengan menggunakan Arduino, selain itu juga ada banyak modul-modul pendukung (sensor, tampilan, penggerak dan sebagainya) yang dibuat oleh pihak lain untuk bisa disambungkan dengan Arduino. (Feri Djuandi, 2011)

Untuk memprogram *board* Arduino, butuh aplikasi IDE (*Integrated Development Environment*) bawaan dari Arduino. Aplikasi ini berguna untuk membuat, membuka, dan mengedit *source code* Arduino. Source code merupakan *sketches* yang berisi logika dan algoritma yang akan diupload ke dalam IC mikrokontroler (Arduino). (Hari Santoso, 2015)

## 2.5 Transistor

Transistor adalah alat semikonduktor yang dipakai sebagai penguat, pemotong (*switching*), stabilisasi tegangan, modulasi sinyal atau fungsi lainnya. Transistor dapat berfungsi semacam kran listrik, dimana berdasarkan arus inputnya (BJT) atau tegangan inputnya (FET), memungkinkan pengaliran listrik yang sangat akurat dari sirkuit sumber listriknya.

Pada umumnya, transistor memiliki 3 terminal. Tegangan atau arus yang dipasang di satu terminalnya mengatur arus yang lebih besar yang melalui 2 terminal lainnya. Transistor adalah komponen yang sangat penting dalam dunia elektronik modern. Dalam rangkaian analog, transistor digunakan dalam *amplifier* (penguat). Rangkaian analog melingkupi pengeras suara, sumber listrik stabil, dan penguat sinyal radio. Dalam rangkaian-rangkaian digital, transistor digunakan sebagai saklar

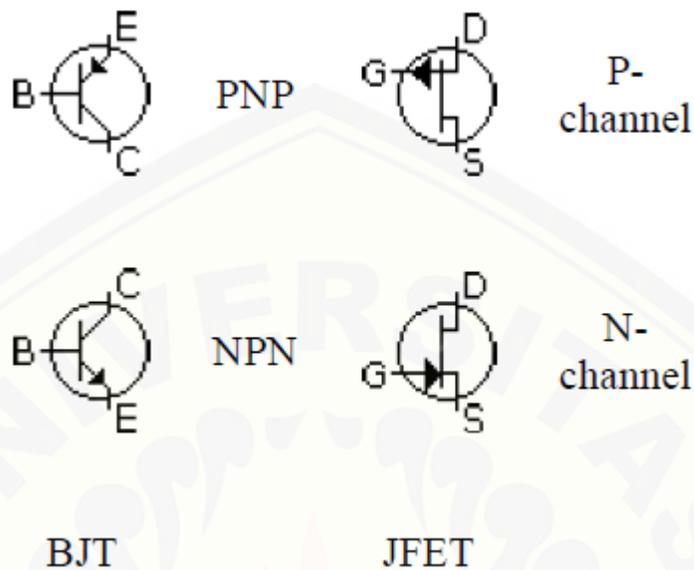
berkecepatan tinggi. Beberapa transistor juga dapat dirangkai sedemikian rupa sehingga berfungsi sebagai *logic gate*, memori, dan komponen-komponen lainnya.

### 2.5.1 Cara Kerja Transistor

Dari banyak tipe-tipe transistor modern, pada awalnya ada dua tipe dasar transistor, bipolar *junction* transistor (BJT atau transistor bipolar) dan *field-effect* transistor (FET), yang masing-masing bekerja secara berbeda. Transistor bipolar dinamakan demikian karena kanal konduksi utamanya menggunakan dua polaritas pembawa muatan elektron dan lubang, untuk membawa arus listrik. Dalam BJT, arus listrik utama harus melewati satu daerah atau lapisan pembatas dinamakan *depletion zone*, dan ketebalan lapisan ini dapat diatur dengan kecepatan tinggi dengan tujuan untuk mengatur aliran arus utama tersebut. FET (juga dinamakan transistor unipolar) hanya menggunakan satu jenis pembawa muatan (elektron atau *hole*, tergantung dari tipe FET). Dalam FET, arus listrik utama mengalir dalam satu kanal konduksi sempit dengan *depletion zone* di kedua sisinya (dibandingkan dengan transistor bipolar dimana daerah Basis memotong arah arus listrik utama). Dan ketebalan dari daerah perbatasan ini dapat dirubah dengan perubahan tegangan yang diberikan, untuk merubah ketebalan kanal konduksi tersebut.

### 2.5.2 Jenis-Jenis Transistor

BJT (Bipolar Junction Transistor) adalah salah satu dari dua jenis transistor. Cara kerja BJT dapat dibayangkan sebagai dua dioda yang terminal positif atau negatifnya berdempet, sehingga ada tiga terminal. Ketiga terminal tersebut adalah emiter (E), kolektor (C), dan basis (B). Perubahan arus listrik dalam jumlah kecil pada terminal basis dapat menghasilkan perubahan arus listrik dalam jumlah besar pada terminal kolektor. Prinsip inilah yang mendasari penggunaan transistor sebagai penguat elektronik. Rasio antara arus pada kolektor dengan arus pada basis biasanya dilambangkan dengan  $\beta$  atau  $h_{FE}$ .  $\beta$  biasanya berkisar sekitar 100 untuk transistor-transistor BJT.



Gambar 2.7 Jenis-Jenis Transistor

(Sumber : Ahmad Fali Oklilas, 2007)

FET dibagi menjadi dua keluarga, *Junction FET (JFET)* dan *Insulated Gate FET (IGFET)* atau juga dikenal sebagai *Metal Oxide Silicon* atau *Semiconductor FET (MOSFET)*. Berbeda dengan IGFET, terminal *gate* dalam JFET membentuk sebuah dioda dengan kanal (materi semikonduktor antara *Source* dan *Drain*). Secara fungsinya, ini membuat *N-channel JFET* menjadi sebuah versi *solid-state* dari tabung vakum, yang juga membentuk sebuah dioda antara antara *grid* dan *katode*. Dan juga, keduanya (JFET dan tabung vakum) bekerja di "*depletion mode*", keduanya memiliki impedansi input tinggi, dan keduanya menghantarkan arus listrik dibawah kontrol tegangan input. FET lebih jauh lagi dibagi menjadi tipe *enhancement mode* dan *depletion mode*. Mode menandakan polaritas dari tegangan *gate* dibandingkan dengan *source* saat FET menghantarkan listrik. Jika kita ambil *N-channel FET* sebagai contoh dalam *depletion mode*, *gate* adalah negatif dibandingkan dengan *source*, sedangkan dalam *enhancement mode*, *gate* adalah positif. Untuk kedua mode, jika tegangan *gate* dibuat lebih positif, aliran arus di antara *source* dan *drain* akan

meningkat. Untuk *P-channel* FET, polaritas-polaritas semua dibalik. Sebagian besar IGFET adalah tipe *enhancement mode*, dan hampir semua JFET adalah tipe *depletion mode*. (Ahmad Fali Oklilas, 2007)

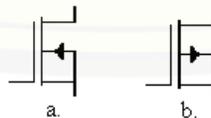
## 2.6 MOSFET

MOSFET (*Metal Oxide Semiconductor Field Effect Transistor*) adalah suatu transistor dari bahan semikonduktor (silikon) dengan tingkat konsentrasi ketidakmurnian tertentu. Tingkat dari ketidakmurnian ini akan menentukan jenis transistor tersebut, yaitu transistor MOSFET tipe-N (NMOS) dan transistor MOSFET tipe-P (PMOS). Bahan silicon digunakan sebagai landasan (*substrat*) dari penguras (*drain*), sumber (*source*), dan gerbang (*gate*). Selanjutnya transistor dibuat sedemikian rupa agar antara substrat dan gerbangnya dibatasi oleh oksida silikon yang sangat tipis. Oksida ini diendapkan di atas sisi kiri dari kanal, sehingga transistor MOSFET akan mempunyai kelebihan dibanding dengan transistor BJT (*Bipolar Junction Transistor*), yaitu menghasilkan disipasi daya yang rendah.

Bila dilihat dari cara kerjanya, transistor MOS dapat dibagi menjadi dua, yaitu :

1. Transistor Mode Pengosongan (*Transistor Mode Depletion*)

Pada transistor mode *depletion*, antara *drain* dan *source* terdapat saluran yang menghubungkan dua terminal tersebut, dimana saluran tersebut mempunyai fungsi sebagai saluran tempat mengalirnya elektron bebas. Lebar dari saluran itu sendiri dapat dikendalikan oleh tegangan gerbang. Transistor MOSFET mode pengosongan terdiri dari tipe-N dan tipe-P, simbol transistor ditunjukkan dalam Gambar 2.8.



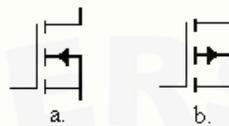
Gambar 2.8 Simbol Transistor MOSFET Mode *Depletion*

a. *N-Channel Depletion*    b. *P-Channel Depletion*

(Sumber : maulana.lecture.ub.ac.id)

## 2. Transistor Mode peningkatan (*Transistor Mode Enhancement*)

Transistor mode enhancement ini pada fisiknya tidak memiliki saluran antara drain dan sourcena karena lapisan bulk meluas dengan lapisan SiO<sub>2</sub> pada terminal gate. Transistor MOSFET mode peningkatan terdiri dari tipe-N dan tipe-P, simbol transistor ditunjukkan dalam Gambar 2.9.



Gambar 2.9 Simbol Transistor MOSFET Mode *Enhancement*

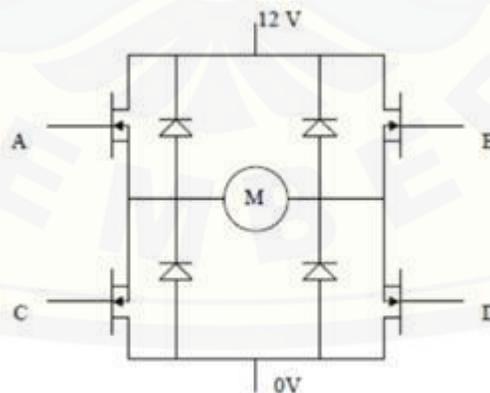
a. *N-Channel Enhancement*    b. *P-Channel Enhancement*

(Sumber : maulana.lecture.ub.ac.id)

## 2.7 *Driver Motor H-Bridge MOSFET*

### 2.7.1 Teori H-Bridge MOSFET

Dalam teori H-Bridge dijelaskan yaitu sebuah perangkat keras berupa rangkaian yang berfungsi untuk menggerakkan motor. Rangkaian tersebut dinamakan H-Bridge karena bentuk rangkaiannya yang menyerupai huruf H seperti pada Gambar 2.10.

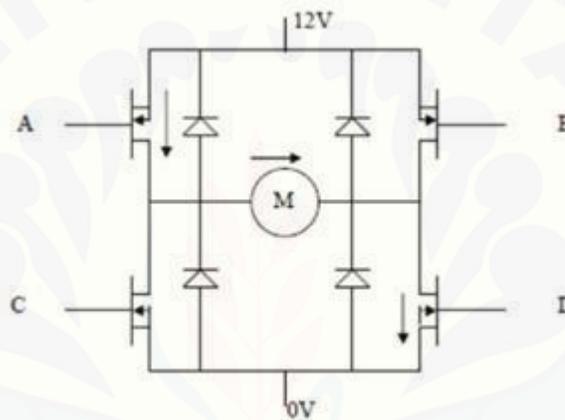


Gambar 2.10 Rangkaian H-Bridge MOSFET

(Sumber : Fahmizal, 2011)

### 2.7.2 Konfigurasi *H-Bridge* MOSFET

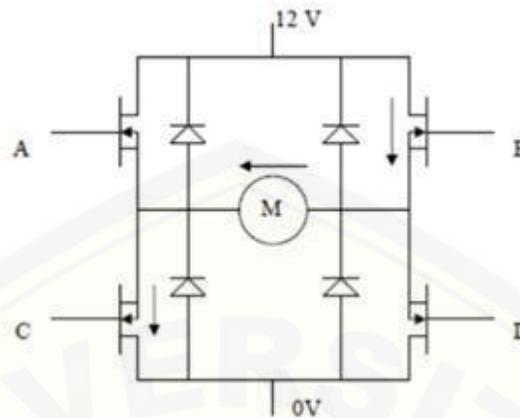
Rangkaian ini terdiri dari dua buah MOSFET kanal P dan dua buah MOSFET kanal N. Prinsip kerja rangkaian ini adalah dengan mengatur mati-hidupnya ke empat MOSFET tersebut. Huruf M pada gambar adalah motor DC yang akan dikendalikan. Pada saat MOSFET A dan MOSFET D *on* sedangkan MOSFET B dan MOSFET C *off*, maka sisi kiri dari gambar motor akan terhubung dengan kutub positif dari catu daya, sedangkan sisi sebelah kanan motor akan terhubung dengan kutub negatif dari catu daya sehingga motor akan bergerak searah jarum jam.



Gambar 2.11 *H-bridge* konfigurasi MOSFET A&D *on*, B&C *off*

(Sumber : Fahmizal, 2011)

Sebaliknya, jika MOSFET B dan MOSFET C *on* sedangkan MOSFET A dan MOSFET D *off*, maka sisi kanan motor akan terhubung dengan kutub positif dari catu daya sedangkan sisi kiri motor akan terhubung dengan kutub negatif dari catu daya. Maka motor akan bergerak berlawanan arah jarum jam. (Fahmizal, 2011)



Gambar 2.12 *H-bridge* konfigurasi MOSFET A&D *off*, B&C *on*  
(Sumber : Fahmizal, 2011)

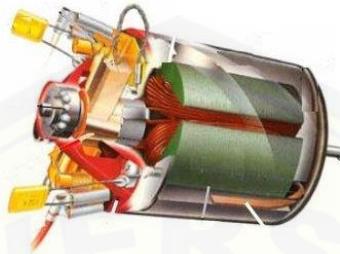
## 2.8 Motor DC

Motor DC adalah motor listrik yang memerlukan suplai tegangan arus searah pada kumparan medan untuk diubah menjadi energi gerak mekanik. Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Motor arus searah, sebagaimana namanya, menggunakan arus langsung yang tidak langsung/direct-unidirectional. Motor DC memiliki 3 bagian atau komponen utama untuk dapat berputar sebagai berikut.

Bagian atau komponen utama motor DC :

- Kutub medan. Motor DC sederhana memiliki dua kutub medan: kutub utara dan kutub selatan. Garis magnetik energi membesar melintasi ruang terbuka diantara kutub-kutub dari utara ke selatan. Untuk motor yang lebih besar atau lebih kompleks terdapat satu atau lebih elektromagnet.
- Current Elektromagnet atau Dinamo. Dinamo yang berbentuk silinder, dihubungkan ke as penggerak untuk menggerakkan beban. Untuk kasus motor DC yang kecil, dinamo berputar dalam medan magnet yang dibentuk oleh kutub-kutub, sampai kutub utara dan selatan magnet berganti lokasi.

- Commutator. Komponen ini terutama ditemukan dalam motor DC. Kegunaannya adalah untuk transmisi arus antara dinamo dan sumber daya.



Gambar 2.13 Motor DC

(Sumber : Agus Purnama, 2012)

Keuntungan utama motor DC adalah sebagai pengendali kecepatan, yang tidak mempengaruhi kualitas pasokan daya. Motor ini dapat dikendalikan dengan mengatur:

- Tegangan dinamo – meningkatkan tegangan dinamo akan meningkatkan kecepatan.
- Arus medan – menurunkan arus medan akan meningkatkan kecepatan.

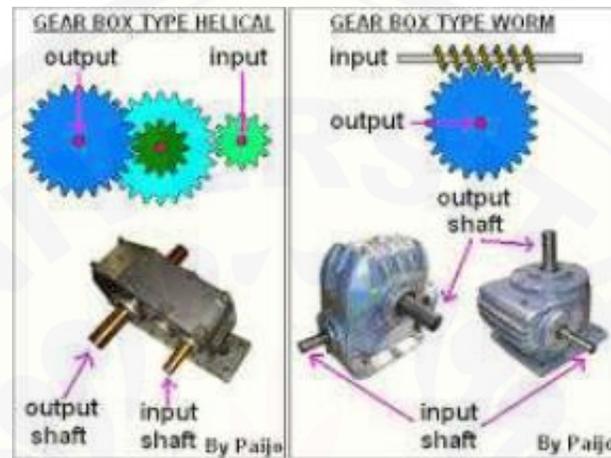
(Agus Purnama, 2012)

## 2.9 Gearbox Motor

*Gear* atau biasa disebut dengan roda gigi memiliki fungsi untuk mentransmisikan daya besar dan putaran yang tepat. Rodagigi memiliki gigi di sekelilingnya, sehingga penerusan daya dilakukan oleh gigi-gigi kedua roda yang saling berkait. Roda gigi sering di gunakan karena dapat meneruskan putaran dan daya yang lebih bervariasi dan lebih kompak daripada menggunakan alat transmisi yang lainnya, selain itu rodagigi juga memiliki beberapa kelebihan jika dibandingkan dengan alat transmisi lainnya, yaitu :

- Sistem transmisinya lebih ringkas, putaran lebih tinggi dan daya yang besar.
- Sistem yang kompak sehingga konstruksinya sederhana.
- Kemampuan menerima beban lebih tinggi.

- Efisiensi pemindahan dayanya tinggi karena faktor terjadinya slip sangat kecil.
- Kecepatan transmisi rodagigi dapat ditentukan sehingga dapat digunakan dengan pengukuran yang kecil dan daya yang besar.



Gambar 2.14 Gearbox Motor

(Sumber : PT. Delta Rekaprima Sakti)

Rodagigi harus mempunyai perbandingan kecepatan sudut tetap antara dua poros. Di samping itu terdapat pula rodagigi yang perbandingan kecepatan sudutnya dapat bervariasi. Ada pula rodagigi dengan putaran yang terputus-putus. Dalam teori, rodagigi pada umumnya dianggap sebagai benda kaku yang hampir tidak mengalami perubahan bentuk dalam jangka waktu lama. (Yefri Chan, 2007)

## BAB 3. METODOLOGI PENELITIAN

### 3.1 Tempat Penelitian

Untuk tempat pelaksanaan penelitian dilakukan beberapa tempat yang berbeda, diantaranya yaitu :

1. Workshop Jurusan Teknik Elektro Universitas jember, Jl. Slamet Riyadi 62 Patrang – jember.
2. Rumah kos penulis, Jl Jambu Gang Baru No.5 Patrang – Jember.

### 3.2 Alat dan Bahan

Alat dan bahan yang digunakan sebagai penunjang dalam penelitian ini adalah sebagai berikut.

#### 3.2.1 Pembuatan Sensor

1. HC-SR04

#### 3.2.2 Pembuatan Tampilan LCD

1. LCD 2x16
2. Variabel resistor 5K $\Omega$
3. *Header*

#### 3.2.3 Pembuatan *Driver* Motor DC

1. MOSFET IRF 530, IRF 9530
2. BD139
3. Resistor 2K2 $\Omega$ , 10K $\Omega$
4. *Header*

#### 3.2.4 Pembuatan Catu Daya

1. IC Regulator 7809
2. IC Regulator 7805
3. Dioda 1N4004
4. Resistor 1K $\Omega$
5. Kapasitor 10 uF

## 6. LED

### 3.2.5 Software

1. Proteus 7 Professional
2. Eagle PCB
3. Arduino IDE

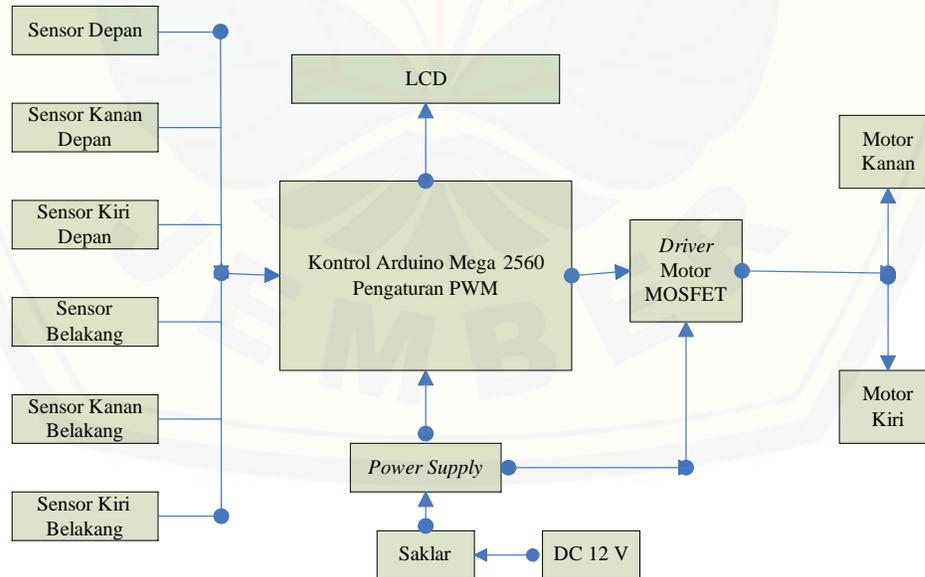
### 3.2.6 Output

1. Motor DC
2. LCD

### 3.2.7 Alat

1. Solder
2. Timah
3. Avometer
4. Tang
5. Penyedot Timah

## 3.3 Blok Diagram



Gambar 3.1 Blok Diagram

Gambar 3.1 blok diagram rancang bangun robot *avoider* segala medan berbasis Arduino Mega 2560, menjelaskan tentang bagian-bagian dari rangkaian yang tersusun menjadi suatu sistem dengan sebuah mikrokontroler Arduino yang dipusatkan sebagai kendali robot.

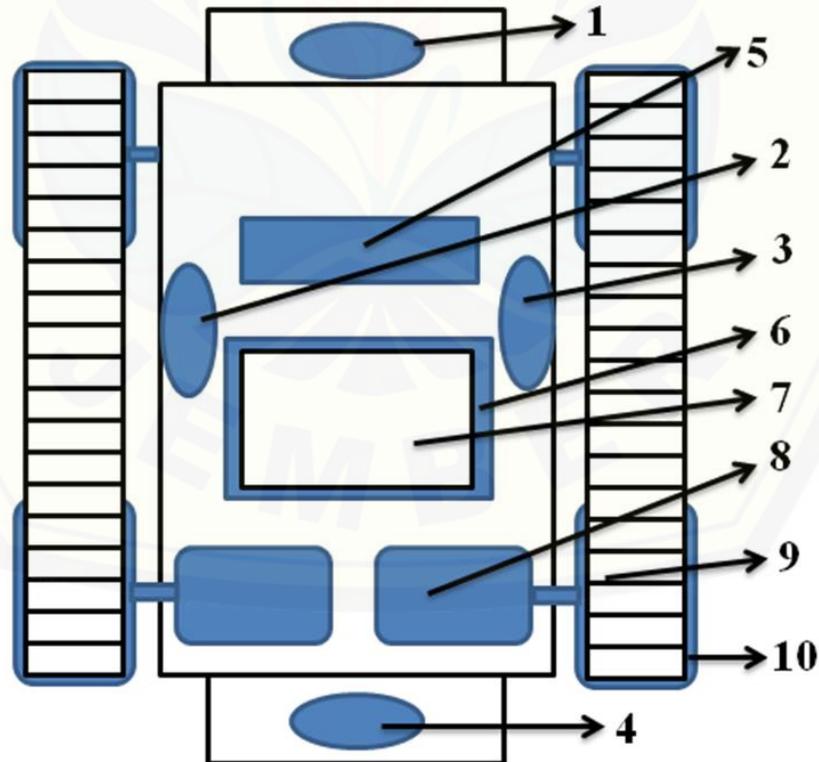
Bagian *input* pada blok diagram terdiri dari sensor jarak yaitu HC-SR04. Sedangkan bagian *output* pada blok diagram di atas yaitu motor DC yang dikontrol oleh *driver* motor MOSFET. Dari diagram blok pada gambar terlihat bahwa robot yang akan dirancang terdiri dari beberapa bagian yaitu :

1. Bagian sesor jarak menggunakan HC-SR04 yang memiliki fungsi sebagai pengukur jarak robot dari halangan atau dinding. Yang kemudian data yang dihasilkan oleh sensor jarak, selanjutnya digunakan sebagai acuan untuk melakukan instruksi atau perintah proses kerja robot.
2. Bagian catu daya ialah menggunakan *power supply* dengan satu *input* dan dua *output* tegangan, tegangan 9 Volt masuk pada Arduino Mega 2560, tegangan 5 Volt untuk *input* tegangan sensor yang digunakan dan tegangan 12 Volt masuk ke *driver* motor MOSFET.
3. Bagian kontrol ialah sebuah sistem Arduino yang berfungsi sebagai pusat kendali robot secara otomatis.
4. Bagian *output* LCD berfungsi untuk menampilkan nilai jarak antara robot dengan halangan didepan, kanan, kiri maupun belakang robot.
5. Bagian *output driver* motor digunakan untuk mengendalikan putaran motor pada robot.
6. Bagian *output* motor DC digunakan sebagai pergerakan robot untuk bergerak maju, mundur, belok kanan maupun kiri.

### 3.4 Perancangan Mekanik

#### 3.4.1 Desain Mekanik Robot

Gambar 3.2 menunjukkan perancangan mekanik robot *avoider* segala medan. Dalam robot *avoider* menggunakan 4 buah sensor HC-SR04 yang berfungsi sebagai pendeteksi jarak pusat robot dengan halangan atau dinding. Fungsi sensor tersebut sebagai pengontrol robot, yang terhubung dengan Arduino. Data yang didapat dari keempat sensor *ultrasonic* ditampilkan pada LCD. Kemudian dikontrol dalam Arduino, dengan menerima data yang diberikan sensor *ultrasonic*. Dengan menerima data dari sensor, robot dapat bergerak maju, mundur, maupun belok kanan dan kiri. Pergerakan robot tersebut dibantu dengan *driver* motor yang memiliki fungsi sebagai kontrol motor DC yang terpasang pada robot. Perancangan mekanik robot *avoider* ditambahkan roda *caterpillar* (*tank-treaded*), bertujuan agar robot dapat mempermudah bergerak disegala medan dan roda tidak terjadi selip ketika medan tidak dalam keadaan rata.



Gambar 3.2 Perancangan Mekanik Robot *Avoider* Segala Medan

Keterangan :

1. Sensor HC-SR04 depan
2. Sensor HC-SR04 kiri
3. Sensor HC-SR04 kanan
4. Sensor HC-SR04 belakang
5. LCD
6. Arduino Mega 2560
7. *Driver* Motor
8. Motor DC
9. *Tank-treaded*
10. Roda

#### 3.4.2 Bahan dan Bentuk Robot

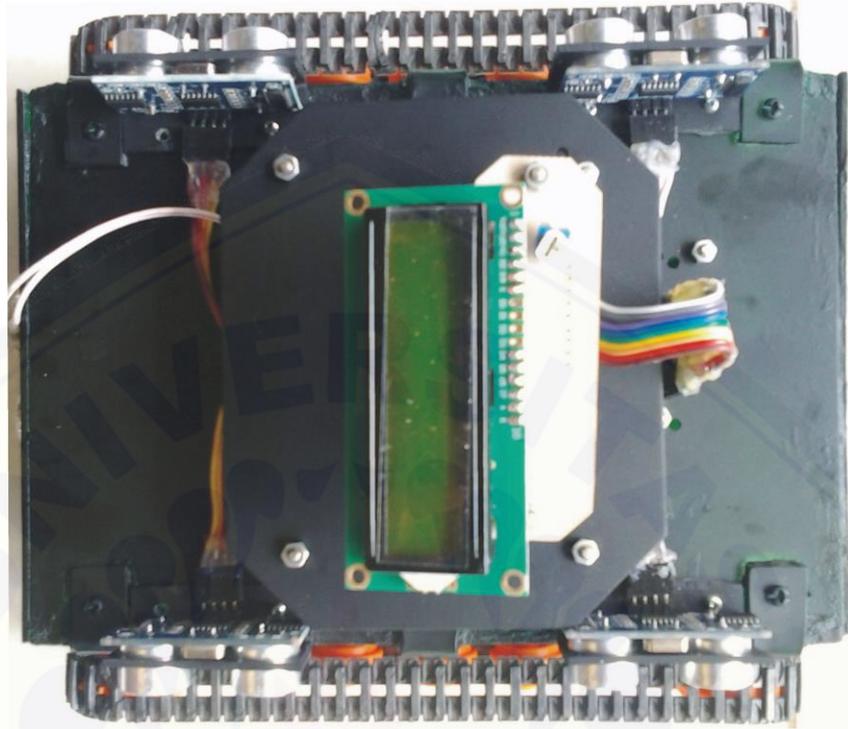
➤ Bahan Mekanik Robot

1. Akrilik
2. Ass Roda Alumunium
3. Roda *Caterpillar (tank-treaded)*

➤ Bentuk Mekanik Robot

Bentuk robot berbentuk seperti balok, bentuk tersebut dibuat semirip mungkin dengan bentuk Tank pada umumnya. Bentuk robot dapat dilihat pada Gambar

3.3.



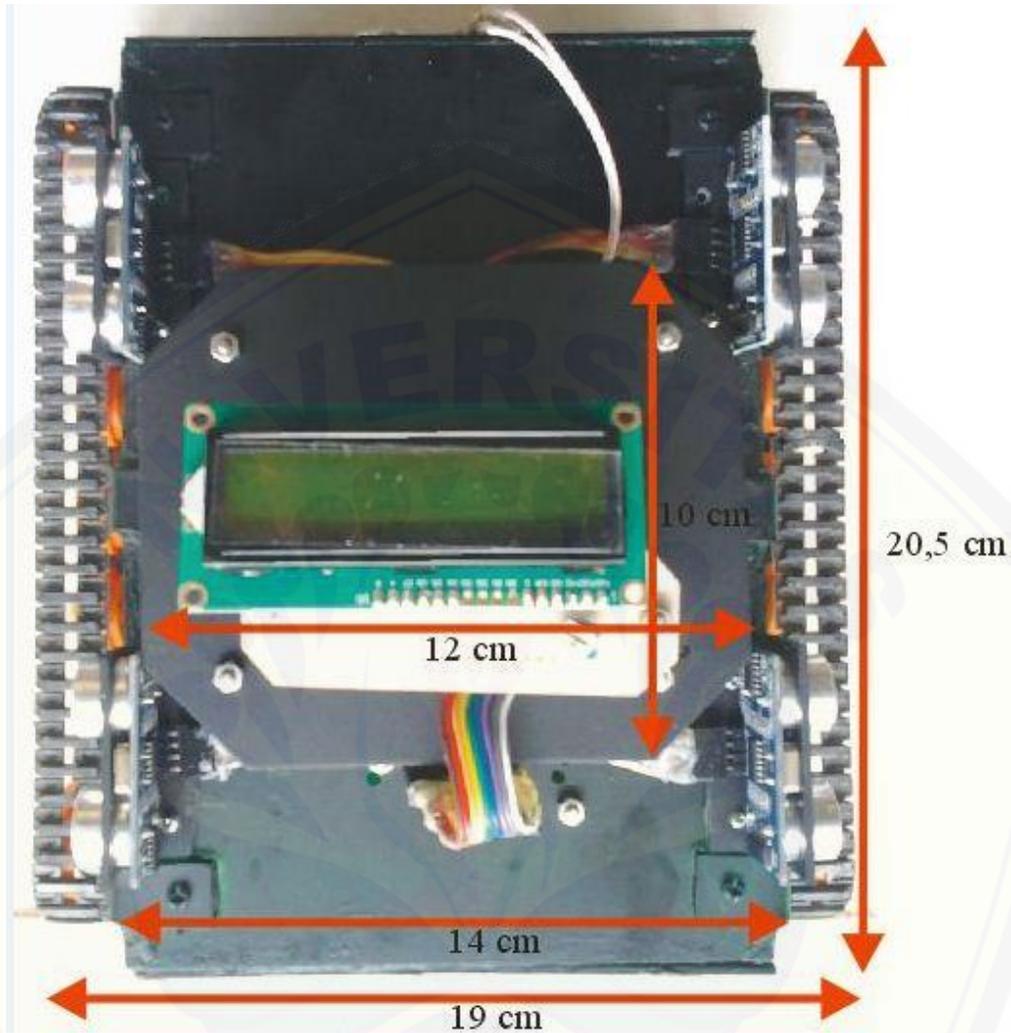
Gambar 3.3 Bentuk Robot *Avoider*

#### 3.4.3 Hasil Perancangan Mekanik Robot *Avoider*

Dalam desain mekanik yang digambarkan awal, memiliki perbedaan. Yaitu sensor yang digunakan terdapat empat, namun ketika robot sudah jadi dan mengalami uji coba singkat. Robot dengan empat sensor tersebut tidak bisa melakukan belok kanan maupun kiri dengan baik. Sehingga penambahan sensor diperlukan dalam hal ini. Penambahan yang dilakukan adalah dengan menambah dua buah sensor pada bagian yang diperlukan. Peletakan sensor juga dirubah, kedua sensor kanan-kiri diletakkan agak kedepan. Dan juga dua sensor kanan-kiri diletakkan agak kebelakang pada badan robot. Hal ini dilakukan agar robot dapat bekerja maksimal ketika maju belok kanan maupun kiri ataupun mundur kanan atau kiri.

##### ➤ Bodi Robot dari Atas

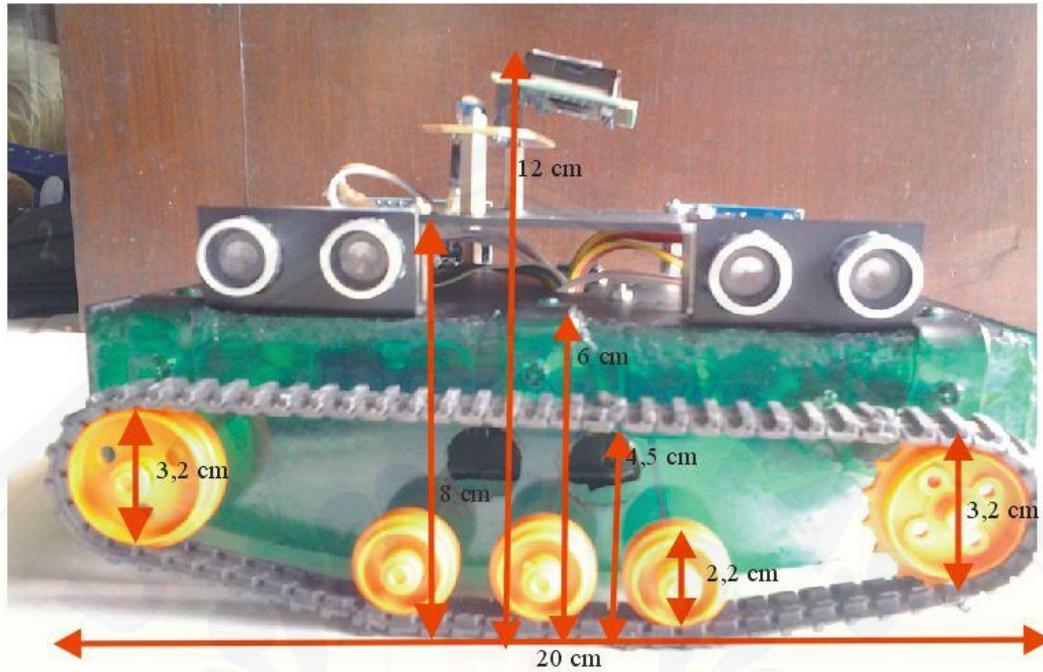
Bodi robot ini merupakan bentuk robot secara keseluruhan jika dilihat dari sisi atas dengan spesifik ukuran yang ditunjukkan pada gambar dibawah ini.



Gambar 3.4 Bodi Robot *Avoider* Nampak Atas

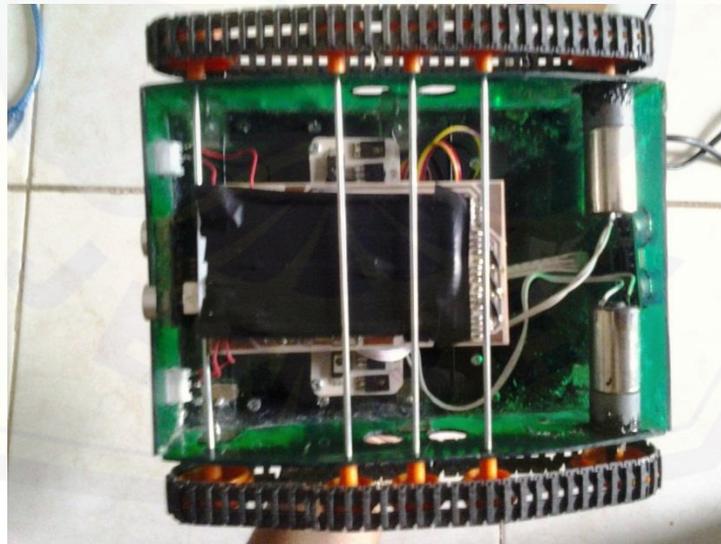
➤ Bodi Robot dari Samping

Gambar dibawah menunjukkan robot dilihat dari samping dengan tinggi tiap-tiap bagian robot sudah di jelaskan pada gambar. Roda yang digunakan ialah roda yang biasa digunakan pada *tank*.



Gambar 3.5 Bodi Robot *Avoider* Nampak Samping

➤ Bodi Robot Bagian Bawah



Gambar 3.6 Bodi Robot Bagian Bawah

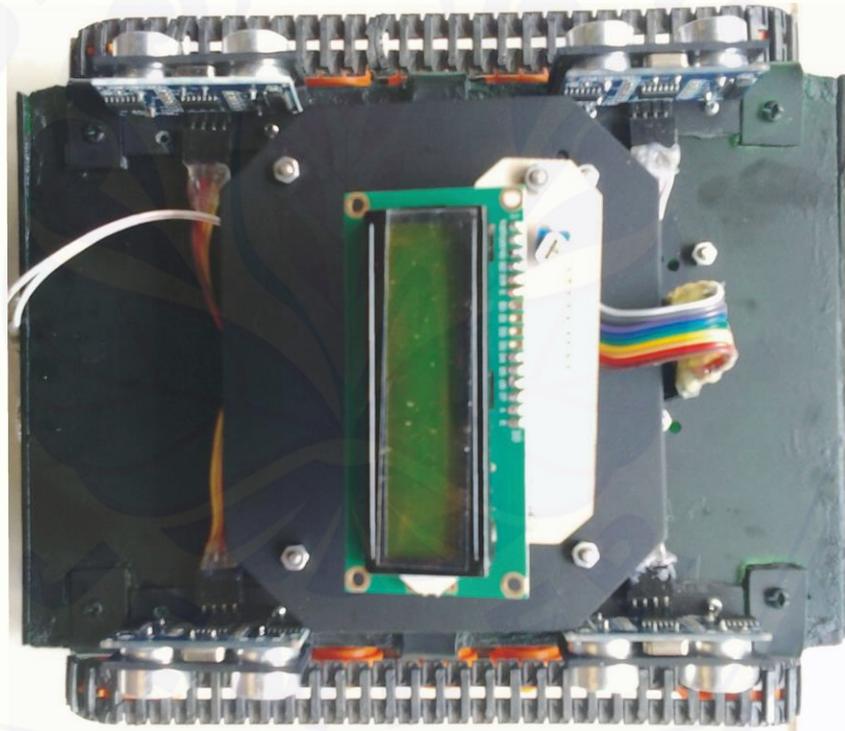
Di dalam terdapat beberapa komponen yang meliputi *shield* Arduino, Arduino Mega 2560, *gearbox* motor dc, serta instalasi kabel pada robot yang ditunjukkan pada gambar diatas.

➤ Bagian-Bagian Letak Sensor

Sensor yang digunakan ialah HC-SR04, sensor ini berfungsi sebagai alat pendeteksi jarak robot dari dinding. Nilai yang dihasilkan sensor kemudian dikirim pada Arduino Mega 2560 selanjutnya melakukan perintah-perintah berikutnya.

1. Sensor Kanan dan Kiri

Sensor diletakkan pada kanan dan kiri posisi robot memiliki fungsi sebagai penanda dinding yang terletak pada kanan dan kiri robot.



Gambar 3.7 Sensor Kanan dan Kiri

## 2. Sensor Depan

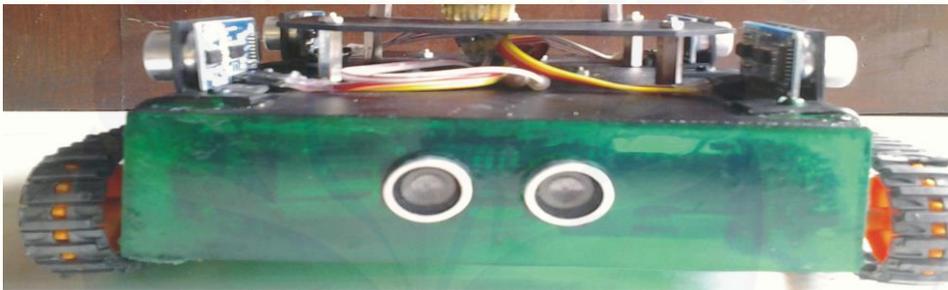
Sensor depan memiliki fungsi untuk mendeteksi dinding bagian depan.



Gambar 3.8 Sensor Depan

## 3. Sensor Belakang

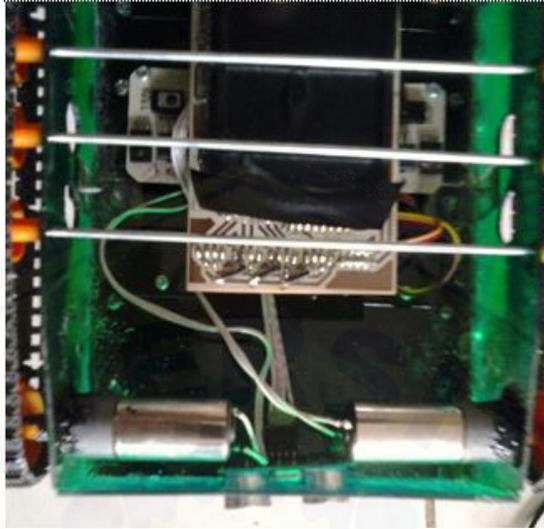
Sensor belakang memiliki fungsi untuk mendeteksi dinding bagian belakang.



Gambar 3.9 Sensor Belakang

### ➤ Bagian-Bagian Aktuator

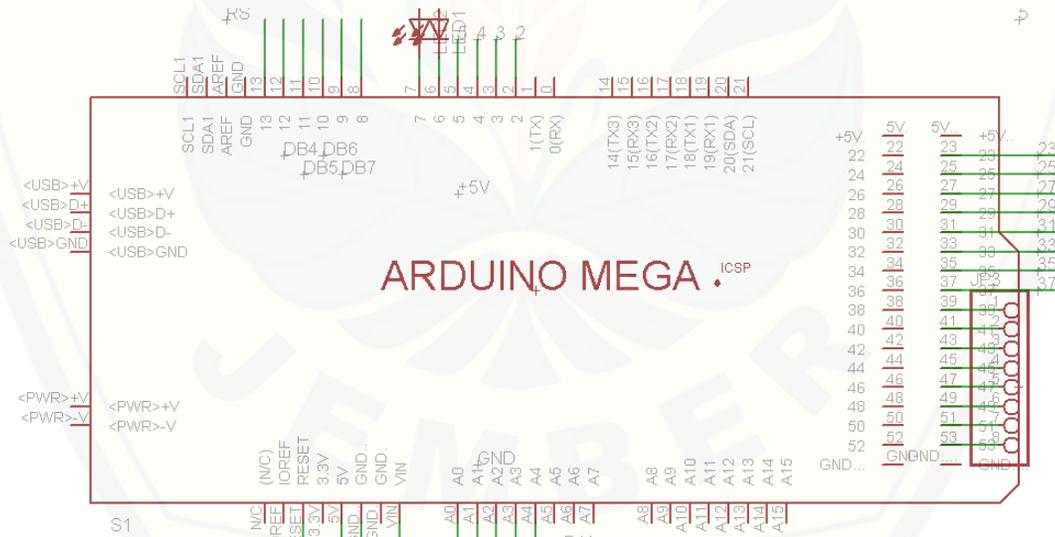
Pergerakan robot tersebut dibantu dengan *driver* motor MOSFET yang memiliki fungsi sebagai kontrol *gearbox* motor DC yang terpasang pada robot. Perancangan mekanik robot *avoider* ditambahkan *timing belt* pada roda penggerak robot, bertujuan agar robot dapat mempermudah bergerak disegala medan dan roda tidak terjadi selip ketika medan tidak dalam keadaan rata yang ditunjukkan pada Gambar 3.10.



Gambar 3.10 Aktuator atau Roda Robot

### 3.5 Perancangan Elektronik

#### 3.5.1 Rangkaian *Shield* Arduino Mega 2560



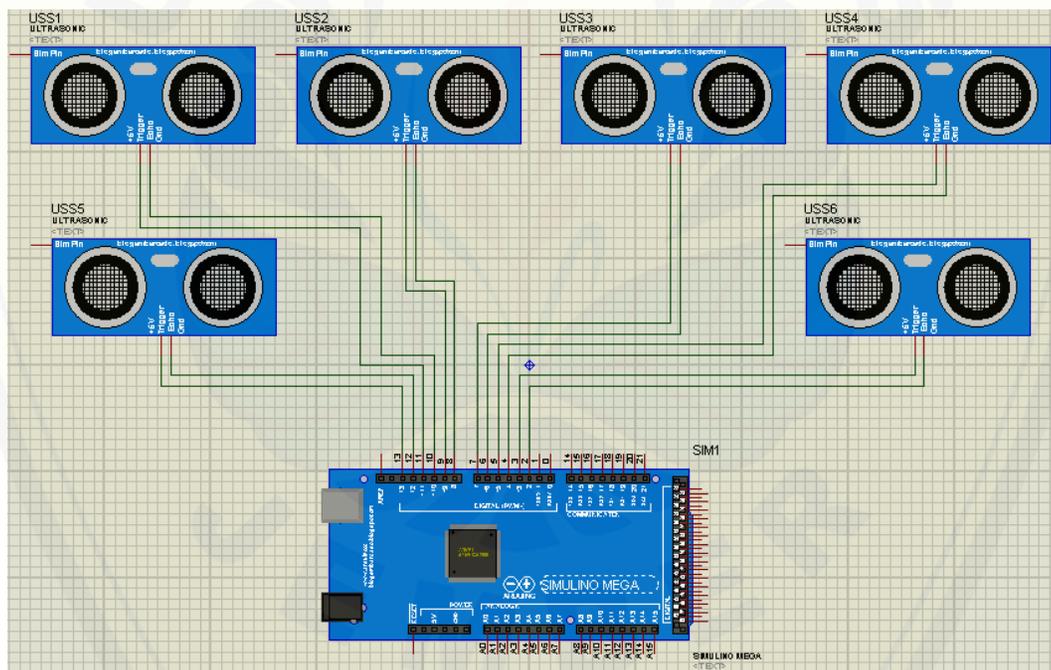
Gambar 3.11 Rangkaian *Shield* Arduino Mega 2560

Gambar 3.11 menunjukkan rangkaian *shield* Arduino Mega 2560, rangkaian tersebut berisikan pin-pin Arduino yang digunakan dalam kontrol robot. Pin 8,9,10,11,12,13 digital pada Arduino disambungkan pada pin LCD. Untuk keenam sensor menggunakan pin 23,25,27,29,31,33,,35,37,39,41,43,45 digital disambungkan

pada pin-pin sensor *ultrasonic* HC-SR04. Untuk kontrol motor digunakan pin 2,3,4,5 digital PWM yang disambungkan pada pin *driver* MOSFET. Pemilihan Arduino Mega 2560 dikarenakan pin yang tersedia banyak sehingga memudahkan dalam pemasangan setiap komponen pendukung alat.

### 3.5.2 Rangkaian Sensor *Ultrasonic* HC-SR04

HC-SR04 merupakan sensor jarak dengan melakukan pantulan gelombang suara yang mengenai benda di depan kemudian ditangkap oleh bagian *reciever*. Dengan mengetahui lamanya waktu antara dipancarkannya gelombang suara sampai ditangkap kembali, dapat diketahui jarak benda yang ada didepan modul. Kemudian akan dikirimkan pada Arduino sebagai pemberi perintah sistem kontrol robot *avoider*. Dapat diketahui rangkaian sensor *ultrasonic* dibawah ini.

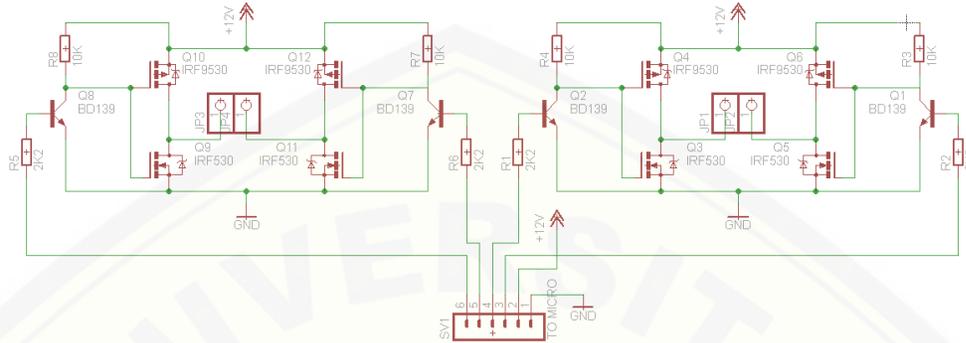


Gambar 3.12 Rangkaian Sensor *Ultrasonic* HC-SR04

### 3.5.3 Rangkaian *Driver* Motor MOSFET

*Driver* motor MOSFET ini merupakan sebuah kontrol *output* yang menggerakkan motor DC. Penggunaan MOSFET ini memiliki keuntungan yaitu arus

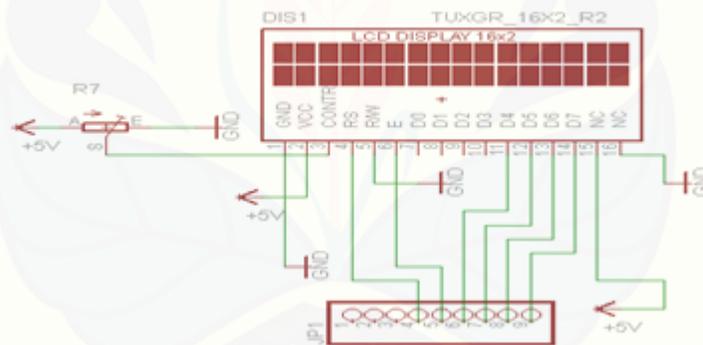
yang dihasilkan lebih besar serta tegangan kerja cukup besar dari *driver* motor yang menggunakan IC. Transistor yang digunakan yaitu IRF530 untuk *switching* PWM.



Gambar 3.13 Rangkaian *Driver* MOSFET

### 3.5.4 Rangkaian LCD

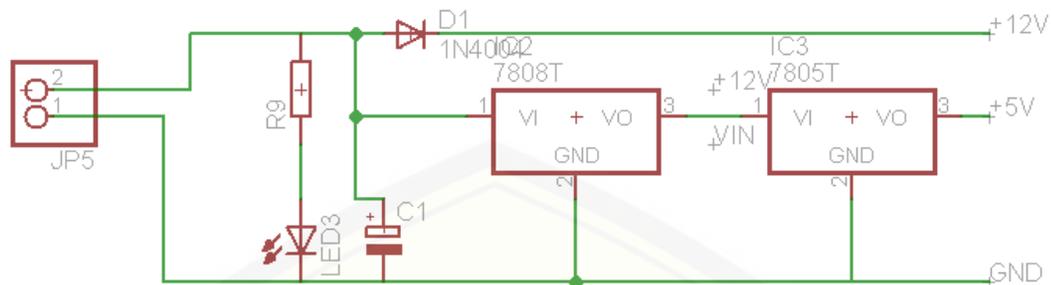
Rangkaian LCD pada sistem ini dipakai sebagai penampil hasil dari pembacaan sensor *ultrasonic* HC-SR04.



Gamabr 3.14 Rangkaian LCD

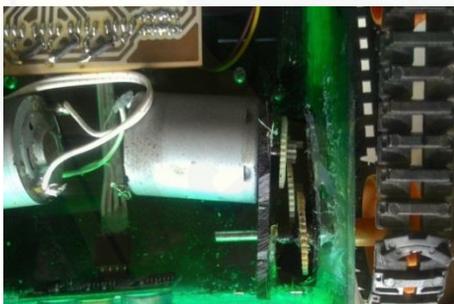
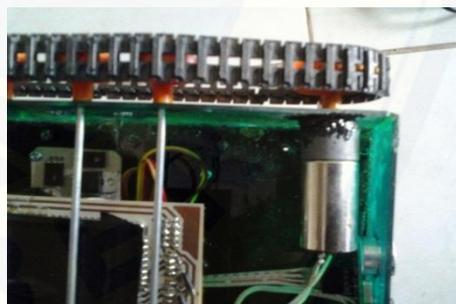
### 3.5.5 Rangkaian *Power Supply*

Rangkaian *power supply* ini merupakan sumber tegangan atau *Vcc*, yang digunakan untuk memberikan tegangan pada mikrokontroler Arduino Mega 2560 dan *driver* motor maupun pada sistem rangkaian robot. Rangkaian ini memiliki tiga *output* tegangan, yaitu tegangan 5 Volt disambungkan pada seluruh sensor yang terpasang, tegangan 9 Volt disambungkan pada Arduino Mega 2560 dan tegangan 12 Volt disambungkan pada *driver* motor untuk mengatur kontrol motor DC.

Gambar 3.15 Rangkaian *Power Supply*

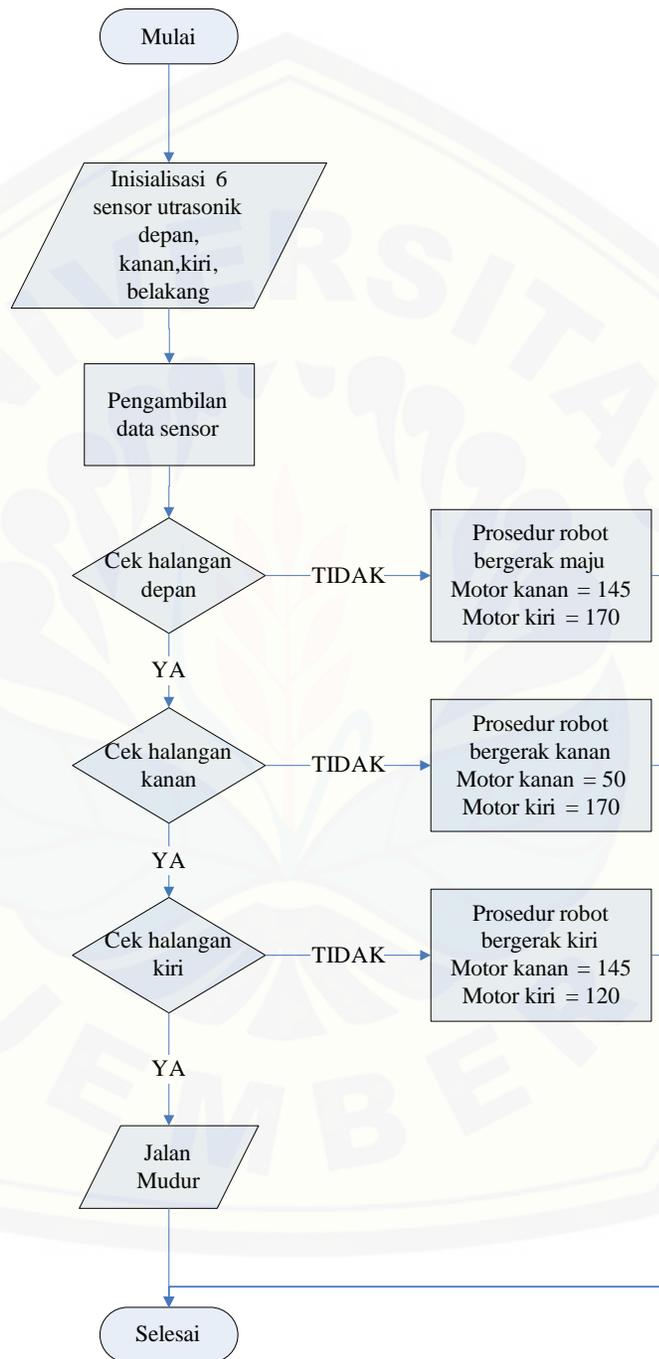
### 3.5.6 *Gearbox* Motor

Fungsi dari *gearbox* sendiri sebagai penggerak atau dapat dikatakan untuk menambah daya beban agar robot dapat bergerak. Pada gambar 3.16 merupakan gambar motor DC dengan *gearbox* buatan sendiri. Kekurangan *gearbox* jenis ini tidak tahan lama dan juga sering mengalami kerusakan ketika pengambilan data. Maka dari itu diganti dengan *gearbox* motor DC buatan pabrik, yang ditunjukkan pada gambar 3.17. Kelebihan *gearbox* ini tidak cepat aus atau rusak, mudah dalam pemasangan dengan roda robot. Kekurangannya setiap jenis motor terdapat perbedaan pada kecepatan putarannya, walaupun diproduksi pada pabrik yang sama. Tegangan *gearbox* motor DC sebesar 12 Volt dengan kecepatan 400 Rpm.

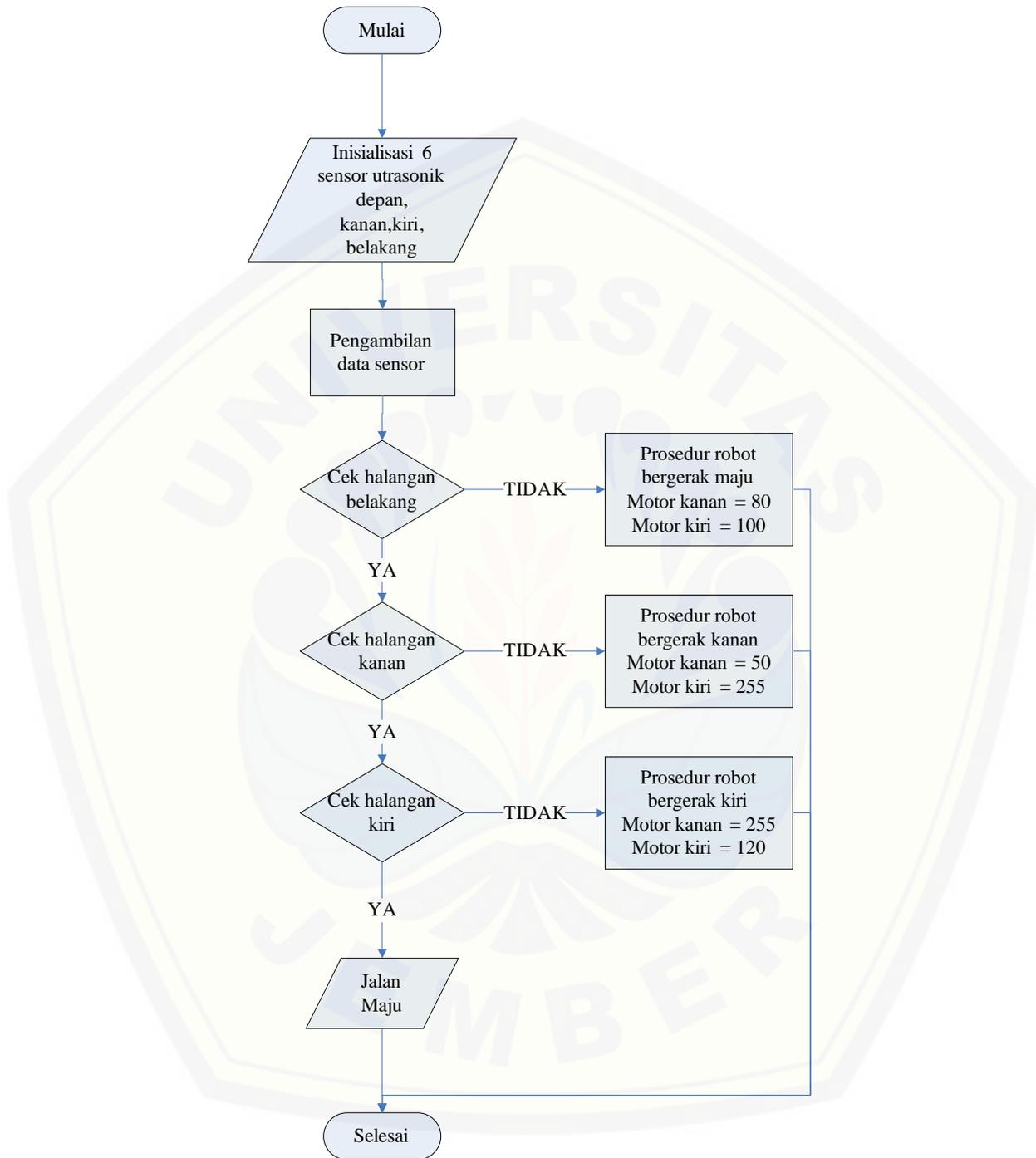
Gambar 3.16 *Gearbox* Motor 1Gambar 3.17 *Gearbox* Motor 2

### 3.6 Perancangan Software

#### 3.6.1 Flowchart



Gambar 3.18 Flowchart Robot Maju



Gambar 3.19 Flowchart Robot Mundur

Pada gambar 3.18 dan gambar 3.19 menunjukkan proses jalanya robot *avoider*, dari ketika *start* maka langkah awal ialah menginisialisai sensor dan jika sensor mendeteksi benda maka akan ditampilkan pada LCD “Jarak Depan, kanan, kiri, dan belakang”. Kemudian robot akan mengalami pergerakan ketika di depan tidak ada halangan, maka robot bergerak maju. Ketika di depan ada halangan sebelah kanan tidak ada maka robot bergerak ke kanan. Kemudian ketika di depan ada halangan sebelah kiri tidak ada halangan maka robot bergerak ke kiri. Namun ketika di depan ada halangan, sebelah kanan kiri ada halangan pula maka robot bergerak mundur sampai tidak ada halangan disekeliling robot *avoider*. Arduino akan memerintahkan pembacaan ulang setiap proses tersebut selama robot bekerja.

### 3.6.2 Program Arduino

Pemrograman yang digunakan untuk memberi perintah pada robot menggunakan *software* ArduinoIDE. Berikut *source code* program inti dari robot *avoider* :

```
void loop(){
```

```
Main:
```

```
    sensordepan = cm;
```

```
    sensorkiri = cm1;
```

```
    sensorkanan = cm2;
```

```
    sensorbelakang = cm3;
```

```
    sensorkiriBLK = cm4;
```

```
    sensorkananBLK = cm5;
```

```
    if (sensordepan <= 5){
```

```
        sensordepan = 5;
```

```
    }
```

```
    if (sensorkiri <=5){
```

```
        sensorkiri = 5;
```

```
}  
if (sensorkanan <= 5){  
  sensorkanan = 5;  
}  
if (sensorbelakang <= 5){  
  sensorbelakang = 5;  
}  
if (sensorkiriBLK <= 5){  
  sensorkiriBLK = 5;  
}  
if (sensorkananBLK <= 4){  
  sensorkananBLK = 5;  
}  
susurkanan();  
delay(100);  
//aktif();  
sensor1();  
sensor2();  
sensor3();  
sensor4();  
sensor5();  
sensor6();  
}
```

Program diatas sudah diringkas, jadi hanya melakukan program pemanggilan pada program yang sudah dibuat sebelumnya. Program tersebut akan selalu dibaca secara berulang-ulang sesuai urutan pemanggilan program yang lain.

### 3.7 Desain Lintasan



Gambar 3.20 Lintasan Labirin

Lintasan robot *avoider* dibuat semirip mungkin seperti reruntuhan gedung ketika mengalami bencana. Dinding-dinding yang dipakai ialah dinding biasa (triplek), dinding putih, dinding hitam, dinding kaca, dan dinding tidak rata. Dari semua jenis dinding tersebut dapat mewakili jenis dinding-dinding yang dipakai dalam bangunan gedung. Kemudian untuk medannya ditambahkan bidang rata, bidang tidak rata yang ditambahkan dengan batu kerikil, dan terdapat dua tanjakan-turunan yang dipakai pada lintasan. Luas lintasan yang digunakan berukuran 240 x 240 cm, sedangkan untuk jarak antar dinding sebesar 30 cm.

## BAB 4. HASIL DAN PEMBAHASAN

Pengujian alat ini memiliki tujuan untuk mengetahui sejauh mana kinerja dan fungsi dari masing-masing rangkaian yang digunakan. Sehingga diperoleh hasil pengujian pada masing-masing rangkaian yang digunakan dalam sistem.

### 4.1 Pengujian Sensor Jarak *Ultrasonic* HC-SR04

Pada bagian ini dilakukan pengujian terhadap sensor jarak atau *ultrasonic* sebagai komponen utama sistem kendali sebuah robot *avoider*. Kinerja sensor ini berpengaruh terhadap kinerja robot itu sendiri. Sensor *ultrasonic* yang digunakan sebanyak enam buah, tetapi dalam pengambilan data hanya menggunakan satu sensor saja. Untuk hasil pengujiannya dapat dilihat seperti pada tabel 4.1 dibawah ini.

Tabel 4.1 Pengujian Sensor *Ultrasonic* HC-SR04

Sensor <i>Ultrasonic</i> HC-SR04					
No.	Jarak	Nilai ADC	Hasil Sensor	Penggaris	E%
1	2 cm	134	2 cm	2 cm	0%
2	4 cm	244	4 cm	4 cm	0%
3	6 cm	350	6 cm	6 cm	0%
4	8 cm	479	8 cm	8 cm	0%
5	10 cm	600	10 cm	10 cm	0%
6	12 cm	707	12 cm	12 cm	0%
7	14 cm	818	14 cm	14 cm	0%
8	16 cm	930	16 cm	16 cm	0%
9	18 cm	1050	18 cm	18 cm	0%
10	20 cm	1162	20 cm	20 cm	0%

Dari hasil pengujian dan pengukuran pada sensor *ultrasonic* HC-SR04 menggunakan perbandingan penggaris dengan nilai sensor *ultrasonic* HC-SR04. Serta pengujian dilakukan 10 kali dengan jarak 2 cm – 20 cm. Dari hasil pengujian tersebut mempunyai nilai *error percent* yang berbeda dan ada pula yang sama, yang dapat dihitung dengan persamaan dibawah ini.

$$Error \% = \frac{|HT-HP|}{HT} \times 100\% \dots\dots\dots (4.1)$$

Dimana : HT = Hasil Teori (Nilai Berdasarkan Teori)

: HP = Hasil Praktek (Nilai Hasil Pengukuran)

Perhitungsn *error %* sensor *ultrasonic* HC-SR04 :

1. Percobaan 1

$$Error \% = \frac{|2 - 2|}{2} \times 100\%$$

$$Error \% = 0 \%$$

2. Percobaan 2

$$Error \% = \frac{|4 - 4|}{4} \times 100\%$$

$$Error \% = 0 \%$$

3. Percobaan 3

$$Error \% = \frac{|6 - 6|}{6} \times 100\%$$

$$Error \% = 0 \%$$

4. Percobaan 4

$$Error \% = \frac{|8 - 8|}{8} \times 100\%$$

$$Error \% = 0 \%$$

5. Percobaan 5

$$Error \% = \frac{|10 - 10|}{10} \times 100\%$$

$$Error \% = 0 \%$$

6. Percobaan 6

$$Error \% = \frac{|12 - 12|}{12} \times 100\%$$

$$Error \% = 0 \%$$

7. Percobaan 7

$$Error \% = \frac{|14 - 14|}{14} \times 100\%$$

$$Error \% = 0 \%$$

8. Percobaan 8

$$Error \% = \frac{|16 - 16|}{16} \times 100\%$$

$$Error \% = 0 \%$$

9. Percobaan 9

$$Error \% = \frac{|18 - 18|}{18} \times 100\%$$

$$Error \% = 0 \%$$

10. Percobaan 10

$$Error \% = \frac{|20 - 20|}{20} \times 100\%$$

$$Error \% = 0 \%$$

Dari hasil pengujian diperoleh nilai sensor *ultrasonic* HC-SR04 dengan melakukan percobaan sebanyak 10 kali. Percobaan yang dilakukan mulai dengan jarak 2 cm hingga 20 cm dari dinding triplek. Nilai tegangan sensor ketika dalam keadaan aktif sebesar 5 V dan untuk nilai arus ketika dalam keadaan aktif sebesar 15 mA. Pertama ketika jarak 2 cm dari dinding, maka yang dibaca sensor sebesar 2 cm dan nilai penggaris juga sebesar 2 cm sehingga memiliki nilai *error* 0 %. Kedua ketika jarak ditambah menjadi 4 cm dari dinding, maka yang dibaca sensor sebesar 4 cm dan nilai penggaris juga sebesar 4 cm sehingga memiliki nilai *error* 0 %. Ketiga nilai jarak yang digunakan ditambah lagi menjadi 6 cm. Maka yang dibaca sensor sebesar 6 cm dan nilai penggaris juga 6 cm, sehingga nilai *error* 0 %. Keempat nilai jarak ditambah lagi menjadi 8 cm. Maka yang dibaca sensor sebesar 8 cm dan nilai penggaris juga sebesar 8 cm, sehingga nilai *error* 0 %. Kelima nilai jarak ditambah lagi

menjadi 10 cm. Maka yang dibaca sensor sebesar 10 cm dan nilai penggaris sebesar 10 cm pula, sehingga nilai *error* 0 %.

Kemudian pada percobaan yang keenam, nilai jarak yang digunakan ditambah lagi menjadi 12 cm. Maka nilai yang dibaca sensor sebesar 12 cm dan nilai penggaris sebesar 12 cm, sehingga nilai *error* 0 %. Ketujuh nilai jarak yang digunakan ditambah lagi menjadi 14 cm. Maka nilai yang dibaca sensor sebesar 14 cm dan nilai penggaris sebesar 14 cm pula, sehingga nilai *error* 0 %. Percobaan yang ke-8 nilai jarak yang digunakan 16 cm. Maka nilai yang dibaca sensor sebesar 16 cm dan nilai penggaris sebesar 16 cm, sehingga nilai *error* 0 %. Percobaan yang ke-9 nilai jarak yang digunakan sebesar 18 cm. Maka nilai yang dibaca sensor sebesar 18 cm dan nilai penggaris sebesar 18 cm, sehingga nilai *error* 0 %. Percobaan yang terakhir atau ke-10 nilai jarak yang digunakan 20 cm. Maka nilai yang dibaca sensor sebesar 20 cm dan nilai penggaris sebesar 20 cm, sehingga nilai *error* 0 %.

Dari ke-10 percobaan yang dilakukan, diketahui sensor *ultrasonic* HC-SR04 bekerja dengan baik. Tidak mengalami *error* dalam pengujian yang sudah dilakukan sebanyak 10 kali. Pengujian yang dilakukan, sudah cukup untuk menyatakan bahwa sensor yang digunakan bekerja dalam keadaan baik.

#### **4.2 Pengujian Sensor *Ultrasonic* HC-SR04 pada Jenis Halangan**

Pada pengujian yang kedua ini, masih membahas sensor. Namun pengujian ini berdasarkan jenis dinding yang akan dibaca oleh sensor *ultrasonic* HC-SR04. Jenis halangan atau dinding yang digunakan ada lima jenis, yaitu dinding biasa, dinding putih, dinding hitam, dinding kaca dan dinding tidak rata. Pemilihan jenis dinding halangan tersebut dikarekan untuk disamakan atau dibuat semirip mungkin dengan keadaan gedung yang sudah runtuh. Berikut pengujiannya dapat dilihat pada Tabel 4.2.

Tabel 4.2 Pengujian Sensor *Ultrasonic* HC-SR04 pada Jenis Halangan

Sensor <i>Ultrasonic</i> HC-SR04					
No.	Halangan	Nilai ADC	Hasil Sensor	Penggaris	E%
1	Dinding Biasa	600	10 cm	10 cm	0%
2	Dinding Putih	600	10 cm	10 cm	0%
3	Dinding Hitam	600	10 cm	10 cm	0%
4	Dinding Kaca	600	10 cm	10 cm	0%
5	Dinding Tidak Rata (Luar)	600	10 cm	12 cm	20%
6	Dinding Tidak Rata (Dalam)	600	10 cm	10 cm	0%

Dari hasil percobaan diatas diketahui nilai *error* dengan menggunakan perhitungan menggunakan persamaan (1).

Perhitungan *error* % sensor *utrasonic* HC-SR04 :

- Percobaan 1

$$Error \% = \frac{|10 - 10|}{10} \times 100\%$$

$$Error \% = 0 \%$$

- Percobaan 2

$$Error \% = \frac{|10 - 10|}{10} \times 100\%$$

$$Error \% = 0 \%$$

- Percobaan 3

$$Error \% = \frac{|10 - 10|}{10} \times 100\%$$

$$Error \% = 0 \%$$

- Percobaan 4

$$Error \% = \frac{|10 - 10|}{10} \times 100\%$$

$$Error \% = 0 \%$$

## 5. Percobaan 5

$$Error \% = \frac{|10 - 12|}{10} \times 100\%$$

$$Error \% = 20 \%$$

## 6. Percobaan 6

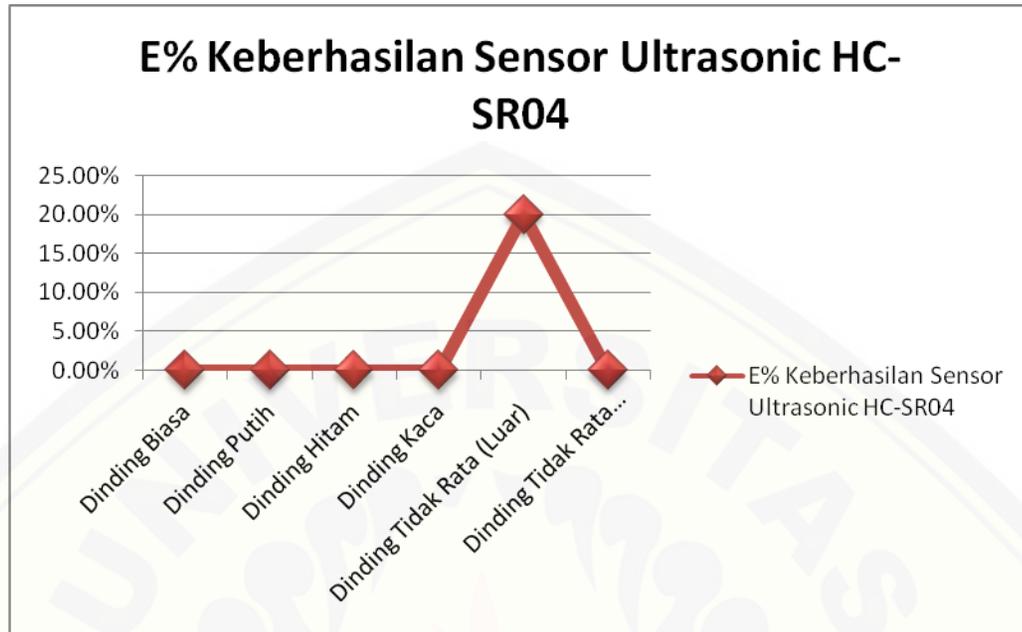
$$Error \% = \frac{|10 - 10|}{10} \times 100\%$$

$$Error \% = 0 \%$$

Dari hasil pengujian diperoleh nilai sensor *ultrasonic* HC-SR04 dengan berbagai kondisi. Dari setiap kondisi ditarik jarak yang sama dengan penggaris sebesar 10 cm, sehingga nilai yang akan terbaca oleh sensor akan sama yang ditunjukkan penggaris atau tidak sama sekali. Pertama ketika kondisi dinding biasa yang diperoleh nilai jarak dari penggaris sebesar 10 cm, maka nilai jarak yang dibaca sensor sebesar 10 cm pula serta memiliki E% sebesar 0 %. Kedua ketika kondisi dinding putih yang diperoleh nilai jarak dari penggaris sebesar 10 cm, nilai jarak yang terbaca oleh sensor sebesar 10 cm, sehingga diperoleh nilai E% sebesar 0 %. Kemudian yang ketiga ketika kondisi dinding hitam diperoleh nilai jarak dari penggaris sebesar 10 cm, nilai jarak yang terbaca oleh sensor diperoleh sebesar 10 cm, sehingga nilai E% sebesar 0 %. Pada percobaan yang keempat ketika kondisi dinding kaca yang diperoleh nilai jarak dengan penggaris sebesar 10 cm, nilai yang terbaca oleh sensor menunjukkan jarak 10 cm pula, sehingga nilai E% dapat diketahui sebesar 0 %. Kemudian percobaan yang kelima ini ketika kondisi dinding tidak rata, dalam kondisi tersebut terdapat dua bagian yaitu dinding terluar dan dinding terdalam. Sebab sensor *ultrasonic* HC-SR04 akan menembak titik terjauh dari sebuah benda atau dinding. Ketika kondisi dinding tidak rata, yang terbaca sensor ialah dinding terdalamnya maka diperoleh nilai jarak dari sensor 10 cm. Nilai jarak yang diperoleh oleh penggaris ada dua yaitu, dinding luar diperoleh nilai 10 cm sedangkan untuk dinding dalam diperoleh 10 cm pula. Nilai jarak yang sama tersebut

dikarenakan pengukuran yang dilakukan ketika dinding terluar diukur 10 cm menuju titik sensor dan dinding terdalam diukur 10 cm pula. Untuk nilai E% yang diperoleh ada dua pula, yaitu ketika dibandingkan dengan dinding luar diperoleh nilai E% sebesar 20 %. Sedangkan ketika dibandingkan dengan dinding bagian dalam diperoleh nilai E% sebesar 0 %. Untuk nilai E% mengalami perbedaan dikarenakan ruang tembak sensor lebih luas ketika dinding terdalam sebab dinding agak menjorok kedalam atau cekung. Sedangkan dinding terluar memiliki ruang tembak yang sempit sebab dinding menjorok kedepan atau cembung, ini menyebabkan sensor menembak tidak ditengah-tengah dinding.

Setelah pengujian telah dilakukan, dari beberapa data yang diperoleh menyatakan bahwa sensor *ultrasonic* HC-SR04 akan membaca titik terjauh pada sebuah bidang yang ditembaknya. Dari percobaan pertama sampai keempat memiliki tingkat *error percent* rata-rata 0 %. Sehingga dapat dikatakan sensor dapat mengirim dan menerima data dengan baik. Sebab dinding yang digunakan masih dalam bidang datar. Meskipun dari keempat dinding tersebut menggunakan kaca, namun tidak terdapat *error*. Tetapi pada percobaan kelima menggunakan dinding yang tidak rata. Dalam percobaan tersebut sensor *ultrasonic* HC-SR04 membaca titik terjauh atau bagian dinding yang terdalam. Untuk permukaan dinding yang menjorok kedepan tidak terbaca oleh sensor *ultrasonic* HC-SR04. Ini menimbulkan nilai *error* ketika dibandingkan dengan permukaan terluar dinding, namun ketika dibandingkan dengan titik terdalam dinding dapat dikatakan tidak bernilai *error*. Dapat dikatakan sensor *ultrasonic* HC-SR04 memiliki tingkat keberhasilan sebesar 83,34 %. Sebab ketika kondisi dinding yang tidak rata memiliki nilai *error* yang cukup besar yaitu 20 %. Dan juga sensor mengalami *error* hanya pada dinding yang tidak rata. Untuk mengetahui grafik dari pengujian *error percent* sensor *ultrasonic* HC-SR04 dapat diketahui pada Grafik 4.1.



Grafik 4.1 Pengujian Sensor *Ultrasonic* HC-SR04

### 4.3 Pengujian Robot Dalam Keadaan Bekerja

Pengujian tersebut meliputi ketika robot bergerak maju, berbelok kanan, belok kiri, mundur, terdapat tanjakan-turunan, lintasan tidak rata dan ketika terdapat pertigaan dalam lintasan. Pengujian yang dilakukan sebanyak lima kali pada setiap keadaan yang dilakukan. Pertama robot bekerja dalam keadaan bergerak maju, dapat dilihat pada tabel 4.3 hasil pengujian.

Dalam pengujian kali ini, ketika robot bergerak dalam keadan maju. Pengujian yang dilakukan sebanyak lima kali percobaan dari titik awal hingga akhir jalan yang lurus. Ketika robot bergerak maju, robot tidak bergerak maju lurus secara mulus. Tetapi robot bergerak agak serong kiri maupun kanan namun robot tidak bisa menabrak dinding. Hal tersebut dikarenakan dalam program diberi nilai maksimum samping dan minimum pada sensor kanan robot. Untuk nilai dari maksimum samping yang digunakan sebesar 7 cm. Sedangkan untuk nilai minimum digunakan nilai 6 cm. Sehingga robot dapat bergerak maju dengan tidak menabrak dinding, namun robot bergerak tidak bisa maju lurus.

Tabel 4.3 Pengujian Robot Bergerak Maju

No	Hasil Pengujian	
	Berhasil	Tidak
1	√	-
2	√	-
3	√	-
4	√	-
5	√	-

Tabel 4.4 Pengujian Robot Bergerak Belok Kanan

No	Hasil Pengujian	
	Berhasil	Tidak
1	√	-
2	√	-
3	√	-
4	√	-
5	√	-

Pengujian yang kedua ialah robot bergerak belok kanan. Pengujian ini dilakukan sebanyak lima kali juga. Hasil pengujian dari kelima percobaan dinyatakan berhasil tidak terdapat kondisi robot tidak bisa melakuakn belok kanan. Hal ini dikarenakan program yang dipakai menggunakan prinsip sisir kanan atau *wall following*. Dalam prinsip tersebut menggunakan dua sensor saja yaitu sensor depan dan sensor kanan. Sensor kanan ini memiliki fungsi sebagai pengatur Bergeraknya robot untuk mengikuti dinding bagian kanan. Robot akan mengikuti dinding bagian kanan, sehingga robot ketika berbelok kanan bisa mngikuti dinding tersebut. Untuk gerak motor ketika berbelok juga diperhitungkan yaitu ketika berbelok ke kanan motor sebelah kiri diberi nilai PWM tinggi dan motor kiri diberi PWM rendah juga motor bergerak mundur untuk motor kiri. Sehingga robot tidak mengalami gangguan apapun, hasil pengujian dapat dilihat pada Tabel 4.4.

Tabel 4.5 Pengujian Robot Bergerak Belok Kiri

No	Hasil Pengujian	
	Berhasil	Tidak
1	√	-
2	√	-
3	-	√
4	√	-
5	√	-

Pengujian yang ketiga ialah robot berbelok ke kiri. Pengujian ini dilakukan sebanyak lima kali juga. Dari kelima pengujian yang dilakukan, pada percobaan yang ketiga robot mengalami gangguan atau *error* yaitu robot bisa melakukan belok kiri. Namun ketika sudah berbelok, haluan belok dari robot agak terlalu keluar atau kesisi kanan robot. Sehingga ketika melakukan pergerakan maju robot menabrak dinding kanan atau sisi bagian bodi robot. Sedangkan untuk keempat percobaan robot dapat berbelok kiri dengan baik. Pengaturan belok kiri ini tidak sama dengan belok kanan. Sebab sensor yang bekerja sebagai acuan yaitu sensor depan. Ketika jarak yang diketahui kurang dari atau lebih dari *range* sensor depan yang digunakan, robot berbelok kiri. Pergerakan motor juga diperhitungkan, robot tidak langsung berbelok. Tetapi robot berbelok kiri hampir 90° setelah itu berhenti untuk menempatkan robot lurus dengan lintasan berikutnya. Kemudian robot bergerak maju, hal itu dilakukan untuk menghindari robot menabrak dinding sebelah kanan ketika berbelok. Hasil pengujian dapat dilihat pada tabel 4.5.

Tabel 4.6 Pengujian Robot Bergerak Mundur

No	Hasil Pengujian	
	Berhasil	Tidak
1	√	-
2	√	-
3	-	√
4	√	-
5	√	-

Pengujian yang keempat ini ialah robot bergerak mundur, hasil pengujian dapat dilihat pada tabel 4.6 diatas. Pengujian yang dilakukan sebanyak lima kali percobaan. Arena yang digunakan ketika kondisi lurus dari titik awal hingga akhir. Robot bergerak mundur sesuai dengan keinginan, walaupun ketika mundur agak bergerak serong untuk menjaga *range* sensor kanan agar tidak menabrak dinding kanan. Prinsip gerak mundur ini sama dengan gerak maju pada robot. Dari kelima percobaan yang dilakukan robot dapat bergerak sesuai dengan program. Yaitu robot bergerak mundur ketika sensor depan kurang dari 15 cm, sensor kiri kurang dari 30 cm, dan sensor kanan kurang dari 20 cm.

Tabel 4.7 Pengujian Robot Bergerak dalam Tanjakan-Turunan

No	Hasil Pengujian			
	Satu Tanjakan		Dua Tanjakan	
	Berhasil	Tidak	Berhasil	Tidak
1	√	-	√	-
2	√	-	√	-
3	√	-	-	√
4	√	-	√	-
5	√	-	√	-

Percobaan yang kelima ini melakukan pengujian ketika robot bergerak dalam medan tanjakan-turunan, hasil pengujian dapat dilihat pada tabel 4.7 diatas. Tanjakan-turunan yang digunakan terdapat dua tipe yaitu pertama menggunakan satu tanjakan yang kedua menggunakan dua tanjakan. Untuk tinggi tanjakan pada masing-masing tipe memiliki tinggi 5 cm. Dalam percobaan ketika dalam keadaan satu tanjakan-turunan robot tidak mengalami masalah. Robot bergerak dengan baik tidak mengalami gangguan ketika naik maupun turun dari kelima percobaan yang dilakukan. Sedangkan ketika dua tanjakan-turunan yang digunakan sebagai misi robot. Robot mengalami *error* ketika pada percobaan yang ketiga, yaitu robot berhenti disela-sela tanjakan-turunan. Robot tidak bisa naik kembali, yang disebabkan oleh medan yang semakin susah, kecepatan robot yang tidak konstan dan mekanik robot bagian bawah yang terlalu menyentuh bidang lintasan yang digunakan. Namun

percobaan yang lainnya berhasil melalui dua tanjakan-turunan dengan baik tidak mengalami gangguan sama sekali. Sehingga dapat dikatakan robot dapat melalui tanjakan-turunan dengan baik. Hanya terdapat satu *error* saja yaitu ketika medan yang digunakan memiliki dua tanjakan-turunan.

Tabel 4.8 Pengujian Robot dalam Lintasan Tidak Rata

No	Hasil Pengujian	
	Berhasil	Tidak
1	√	-
2	√	-
3	√	-
4	√	-
5	√	-

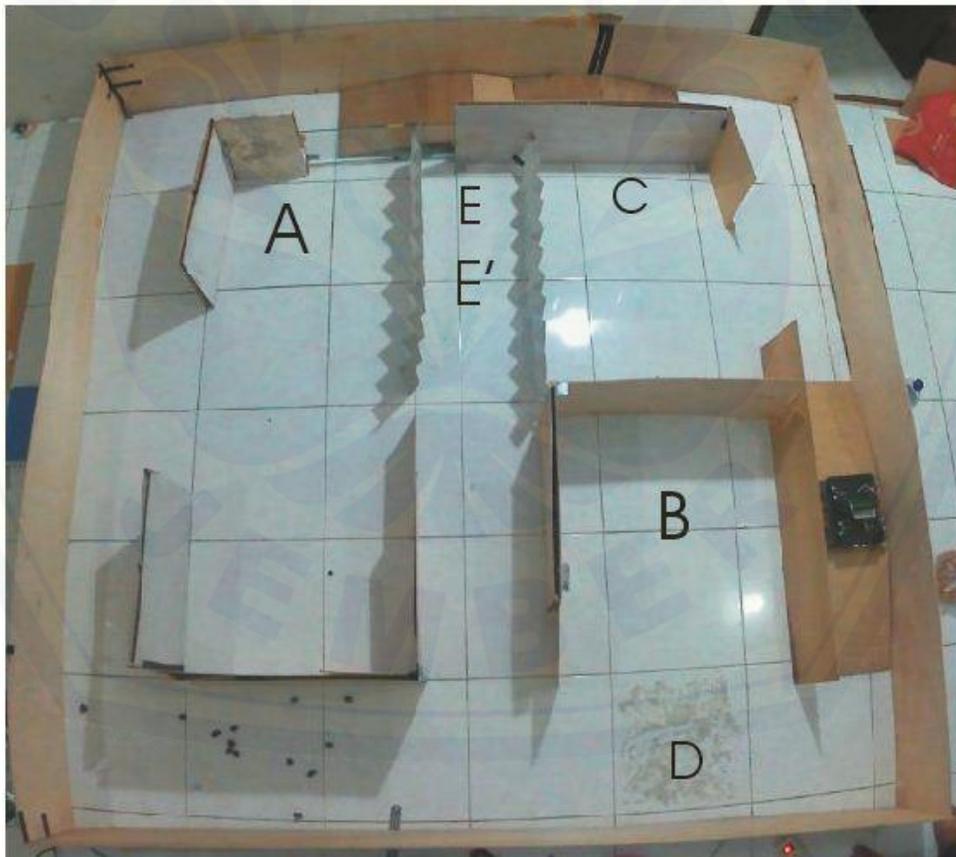
Pengujian yang keenam ini dilakukan pada lintasan lurus tetapi untuk medan ditambahkan batu kerikil. Penambahan batu kerikil tersebut menyusahkan robot untuk bergerak. Dari percobaan yang dilakukan sebanyak lima kali robot dapat bergerak maju, tetapi dalam pergerakan robot berhenti di ujung jalan setelah batu kerikil. Berhentinya robot tidak lama, namun mengganggu jalannya robot. Hal tersebut disebabkan karena beberapa hal, desain mekanik robot bagian bawah terlalu mendekati tanah atau medan. Sehingga bodi robot ditahan oleh batu kerikil yang menyebabkan robot susah untuk bergerak.

Tabel 4.9 Pengujian Robot Ketika Terdapat Pertigaan

No	Hasil Pengujian					
	Kanan-Kiri Tanpa Dinding		Depan-Kanan Tanpa Dinding		Depan-Kiri Tanpa Dinding	
	Kanan	Kiri	Depan	Kanan	Depan	Kiri
1	√	-	-	√	-	√
2	√	-	-	√	-	√
3	√	-	-	√	-	√
4	√	-	-	√	-	√
5	√	-	-	√	-	√

Dalam percobaan yang ketujuh ini dilakukan pada tiga kondisi pertigaan. Pertama, ketika kondisi kanan-kiri tidak ada dinding. Dalam pengujian kondisi tersebut dilakukan sebanyak lima kali. Kelima percobaan yang dilakukan robot memilih berbelok ke kanan. Kedua, ketika depan-kanan tidak ada dinding. Dalam pengujian tersebut dilakukan sebanyak lima kali juga. Namun robot juga memilih berbelok kanan terlebih dahulu dari pada bergerak maju. Ketiga, ketika depan-kiri tidak ada dinding. Dari kelima percobaan yang sudah dilakukan, robot bergerak maju dari pada berbelok kiri. Semua itu terjadikarena proram perintah robot menggunakan prinsip sisir kanan atau mengikuti dinding sebelah kanan.

#### 4.4 Pengujian Robot dalam Menyusuri Lintasan



Gambar 4.1 Denah Rute Lintasan Robot

Tabel 4.10 Pengujian Robot dalam Menyusuri Lintasan

No	Hasil Pengujian					
	Rute A ke B		Rute C ke D		Rute E ke E'	
	Percobaan	Waktu	Percobaan	Waktu	Percobaan	Waktu
1	A ke B	41 detk	C ke D	2 menit 47 detik	E ke E'	1 menit 41 detik
2	A ke B	40 detik	C ke D	1 menit 30 detik	E ke E'	1 menit 32 detik (tidak berhasil)
3	A ke B	42 detik	C ke D	1 menit 18 detik (tidak berhasil)	E ke E'	1 menit 48 detik

Pengujian robot menyusuri lintasan dilakukan sebanyak tiga kali setiap tiga rute yang berbeda. Pertama ketika rute A ke B memerlukan waktu 41 detik. Pergerakan robot cukup baik, sebab ketika sudah berjalan robot kadang-kadang masih menabrak dinding didepan atau samping robot. Namun meskipun robot menabrak dinding kiri, kanan, maupun belakang robot masih dapat berjalan cukup baik. Kemudian dalam rute A ke B percobaan kedua robot menyusuri labirin selama 40 detik. Robot tidak mengalami kendala apapun, baik menabrak dinding kiri dan kanan. Selanjutnya dalam rute A ke B percobaan ketiga menyusuri labirin selama 42 detik. Dalam percobaan tersebut robot tidak mengalami gangguan dan robot bekerja sesuai harapan.

Kedua ketika rute C ke D dilakukan percobaan sebanyak tiga kali juga. Dalam percobaan pertama di rute C ke D diperoleh lama waktu penyusuran 2 menit 47 detik. Robot bergerak sesuai harapan tetapi dalam menyusuri labirin ruang yang seharusnya robot masuk, robot tidak masuk ruangan. Hal itu terjadi dikarenakan ketika berbelok kanan haluan robot untuk belok terlalu besar sehingga sensor yang memberi perintah ialah sensor depan kemudian sensor kanan membaca dinding kanan. Robot kemudian menyusuri dengan pergerakan tidak teratur, namun robot masih bisa menuju *finish*. Pada percobaan yang kedua dengan rute C ke D membutuhkan penyusuran waktu selama 1 menit 30 detik. Robot bergerak sesuai

dengan harapan. Ketika berbelok kiri robot berbelok secara perlahan-lahan. Robot tidak mengalami gangguan dalam bergerak. Kemudian yang ketika robot seharusnya masuk dalam jalan yang seharusnya dimasuki, robot tidak memasuki ruangan yang seharusnya. Selanjutnya percobaan yang ketiga ketika rute C ke D. Robot bergerak sesuai harapan robot bergerak dengan baik dan memasuki ruangan yang seharusnya dimasuki. Namun tidak menyelesaikan misinya ke titik D.

Ketiga rute E ke E' dilakukan percobaan sebanyak 3 kali setiap sesi rute yang dilakukan. Percobaan pertama rute E ke e' dapat diselesaikan selama waktu 1 menit 41 detik. Rute berawal pada dinding yang tidak rata dan berakhir pada dinding tidak rata pula. ketika robot melewati batu kerikil robot masih dapat berjalan namun terkadang robot membawa batu kerikil. Sebab desain mekanik robot bagian bawah terlalu pendek. Hambatan tersebut tidak mengganggu kinerja robot dalam menelusuri labirin hingga mencapai *finish*. Kemudian dalam percobaan yang kedua dalam menelusuri labirin memerlukan waktu selama 1 menit 32 detik. Pertama-tama robot bergerak dengan baik dan sesuai dengan program yang diberikan. Namun kemudian robot mengalami gangguan, yaitu robot melakukan berbalik arah bukan belok kanan ketika lintasan berbelok kanan. Selanjutnya ketiga ketika percobaan sudah dilakukan dengan lama waktu 1 menit 48 detik. Robot bekerja dengan baik, namun ada sedikit menabrak dinding bagian kanan maupun kiri dari lintasan. Meskipun sedikit terkendala dengan hal tersebut robot dapat menyelesaikan misinya dengan baik hingga *finish* di titik E'. Dari percobaan yang sudah dilakukan dapat diketahui nilai waktu rata-rata robot dalam menelusuri lintasan labirin tersebut. Untuk rute pertama yaitu rute A ke B diperoleh waktu rata-rata sebesar 41 detik. Untuk rute yang kedua C ke D diperoleh waktu rata-rata sebesar 2 menit 24 detik. Sedangkan rute yang ketiga E ke E' diperoleh waktu rata-rata sebesar 1 menit 37 detik.

Dilihat dari percobaan yang sudah dilakukan terdapat ruangan yang tidak dimasuki sama sekali. Hal tersebut dikarenakan dalam program robot menggunakan prinsip susur kanan. Susur kanan ialah sensor hanya mendeteksi dinding bagian kanan atau mengikuti dinding bagian kanan. Dari hasil percobaan diketahui terdapat 2

percobaan yang gagal mencapai *finish*. Sehingga tingkat keberhasilan robot dapat dihitung dengan persamaan dibawah ini.

$$\text{Kegagalan \%} = \frac{\text{Kegagalan}}{\text{Percobaan}} \times 100\% \dots\dots\dots (4.2)$$

$$\text{Kegagalan \%} = \frac{2}{9} \times 100\%$$

$$\text{Kegagalan \%} = 22,22 \%$$

Keberhasilan :

$$\text{Keberhasilan \%} = 100 - 22,22$$

$$\text{Keberhasilan \%} = 77,78 \%$$

Sehingga robot memiliki tingkat kegagalan sebesar 22,22 % dalam menyusuri lintasan labirin. Dan memiliki tingkat keberhasilan sebesar 77,78 % dalam menyusuri lintasan labirin. Nilai tersebut diperoleh pada percobaan robot menyusuri lintasan, dengan tiga rute dan tiga kali percobaan dalam setiap rute.

## BAB 5. PENUTUP

### 5.1 Kesimpulan

Setelah melakukan perencanan dan pembuatan sisitem kemudian melakukan pengujian dapat diambil beberapa kesimpulan yaitu,

1. Ketika bidang datar yang dideteksi sensor *ultrasonic* HC-SR04 nilai E% yang didapat 0% atau sensor dapat membaca dengan baik seperti yang terlihat pada tabel 4.1. Namun sensor mengalami *error* ketika membaca dinding tidak rata dengan nilai *error* sebesar 20 % seperti pada tabel 4.2.
2. Robot dapat menyelesaikan misi dalam waktu tercepat ketika melalui rute A ke B dengan waktu 40 detik, dan dengan waktu terlama sebesar 2 menit 47 detik dari rute C ke D seperti yang terlihat pada tabel 4.10. Sehingga kinerja robot memiliki tingkat keberhasilan sebesar 77,78%.

### 5.2 Saran

Setelah melakukan penelitian tentang Rancang Bangun Robot *Avoider* Segala Medan Berbasis Arduino Mega 2560 memiliki kekurangan atau kendala berikut ini merupakan saran untuk pengembangan lebih lanjut :

1. Menggunakan sensor yang lebih peka lagi, atau yang dapat membaca ketika dinding tidak rata.
2. Menggunakan motor dc *gearbox* yang RPM rendah namun memiliki torsi yang tinggi, agar robot berjalan dengan baik sesuai harapan.
3. Desain mekanik robot perlu ada perbaikan, agar robot berjalan sesuai dengan harapan.

**DAFTAR PUSTAKA**

- Santoso, Hari. 2015. *Panduan Praktis Arduino untuk Pemula*  
Buletin Kamadhis UGM. *Bencana Alam*. Eka-citta No. XXVII/September/2007
- BAKORNAS PB, 2007. *Pengenalan Karakteristik Bencana dan Upaya Mitigasinya di Indonesia Edisi II*. Jakarta
- Hadijaya, Erik., dan Tjetje, 2012. *Akuisisi Data Kinerja Sensor Ultrasonik Berbasis Sistem Komunikasi Serial Menggunakan Mikrokontroler ATmega 32*, Jurusan Pendidikan Teknik Elektro FPTK UPI, *Electrans*, Vol.11, No.2, September 2012 , 36-43
- Xue, Yang, Dong dkk, 2014. *A New Approach for Autonomous Robot Obstacle Avoidance Using PSD Infrared Sensor Combined with Digital Compass*, Department of Automation Engineering, Shanghai University of Electric Power, Shanghai, 200090, China, *Applied Mechanics and Materials* Vols. 511-512 (2014) pp 101-104
- Endy, Kasogi N, 2012 *Implementasi Algoritma Dan Sensor Infra Red Dalam Perancangan Robot Pemadam Api Kategori Beroda Pada Kontes Robot Cerdas Indonesia 2012*, Prodi Elektro UHAMKA, Volume1, Nomor 1, Oktober 2012
- Iwan, Endah, Fernando. *Implementasi Metode Vector- Distance Function, sebagai Metode Penghindaran Halangan untuk Autonomous Mobile Robot*, Jurusan Elektronika, Politeknik Elektronika Negeri Surabaya
- Nanang S, Elang D.M, F. Trias P.W, 2014. *Rancang Bangun Robot Tank Yang Dilengkapi Kamera Terintegrasi Dengan Smartphone Android Via Wifi*, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Tanjungpura
- Nauriana, 2009. *Rancang Bangun Robot Beroda Penghindar Halangan*, Departemen Teknik Elektro, Fakultas Teknik, Universitas Indonesia, Depok, Desember 2009

Mujahidin, Maulana, 2008. *Robot Pemadam Api Menggunakan DT-Basic Mini System*, Fakultas Ilmu Komputer, Universitas Gunadarma



LAMPIRAN

**Program dari Rancang Bangun Robot *Avoider* Segala Medan Berbasis Arduino  
Mega 2560**

```
/******  
****  
#include <LiquidCrystal.h>  
const int  
led1 = 6,  
led2 = 7,  
trig = 23, //cm  
echo = 25, //cm  
trig1 = 27, //cm1  
echo1 = 29, //cm1  
trig2 = 31, //cm2  
echo2 = 33, //cm2  
trig3 = 35, //cm3  
echo3 = 37, //cm3  
trig4 = 39, //cm4  
echo4 = 41, //cm4  
trig5 = 43, //cm5  
echo5 = 45; //cm5  
int error;  
int sensordepan ;  
int sensorkiri ;  
int sensorkanan ;  
int sensorbelakang ;  
int sensorkiriBLK ;
```

```
int sensorkananBLK ;
int pinkiri1 = 2 ;
int pinkiri2 = 3 ;
int pinkanan1 = 4 ;
int pinkanan2 = 5 ;
long duration, cm, duration1, cm1, duration2, cm2, duration3, cm3, duration4, cm4,
duration5, cm5;

LiquidCrystal lcd(13, 12, 11, 10, 9, 8);

void setup(){
  pinMode(6, OUTPUT);
  pinMode(7, OUTPUT);

  lcd.begin(16,2);
  lcd.setCursor(2,0);
  lcd.print("Robot Avider");
  lcd.setCursor(4,1);
  lcd.print("Tank_W01");

  pinMode(pinkanan1, OUTPUT);
  pinMode(pinkanan2, OUTPUT);
  pinMode(pinkiri1, OUTPUT);
  pinMode(pinkiri2, OUTPUT);
}

int maksimum = 30;
int maksimumsamping = 7;
int minimum = 6;
int minimumdepan = 15;
```

```
int minimumbelakang = 15;
```

```
void loop(){
```

```
  Main:
```

```
    sensordepan = cm;
```

```
    sensorkiri = cm1;
```

```
    sensorkanan = cm2;
```

```
    sensorbelakang = cm3;
```

```
    sensorkiriBLK = cm4;
```

```
    sensorkananBLK = cm5;
```

```
    if (sensordepan <= 5){
```

```
      sensordepan = 5;
```

```
    }
```

```
    if (sensorkiri <=5){
```

```
      sensorkiri = 5;
```

```
    }
```

```
    if (sensorkanan <= 5){
```

```
      sensorkanan = 5;
```

```
    }
```

```
    if (sensorbelakang <= 5){
```

```
      sensorbelakang = 5;
```

```
    }
```

```
    if (sensorkiriBLK <= 5){
```

```
      sensorkiriBLK = 5;
```

```
    }
```

```
    if (sensorkananBLK <= 4){
```

```
      sensorkananBLK = 5;
```

```
    }
```

```
susurkanan());
delay(100);
sensor1();
sensor2();
sensor3();
sensor4();
sensor5();
sensor6();}
void susurkanan()
{
if (sensordepan >= minimumdepan && sensorkanan >= minimum && sensorkanan
<= maksimumsamping )
{
maju();
}
if (sensordepan < minimumdepan && sensorkiri >= 20)
{
maki();
delay(650);
berhenti();
delay(500);
}
if (sensordepan >= minimumdepan && sensorkanan < minimum )
{
maju1();
}
if (sensordepan >= minimumdepan && sensorkanan > maksimumsamping &&
sensorkanan < maksimum)
{
```

```
maju2();
}
if (sensordepan >= minimumdepan && sensorkanan >= maksimum)
{
maka();
}
if (sensordepan < minimumdepan && sensorkiri < 30 && sensorkanan < 30)
{
berhenti();
delay(500);
while(1)
{
if(sensorbelakang > 15)
{
if(sensorkananBLK <= 7)
{mundur1();}
if((sensorkananBLK > 7 && sensorkananBLK < 10)||sensorbelakang<25)
{mundur();}
if(sensorkananBLK >= 10)
{mundur2();}
}
else
{
berhenti();
delay(500);
while(sensorkanan>15)
{ bacasensor();
muki();}
break;
}
```

```
    berhenti();
    delay(500);
  }
  bacasensor();
  delay(200);
}
}
}
//Sensor HC-SR04
void sensor1(){
  pinMode(trig,OUTPUT);
  digitalWrite(trig,LOW);
  delayMicroseconds(2);
  digitalWrite(trig, HIGH);
  delayMicroseconds(5);
  digitalWrite(trig,LOW);
  pinMode(echo,INPUT);
  duration = pulseIn(echo, HIGH);
  cm = microsecondsToCentimeters(duration);
  lcd.clear();
  lcd.setCursor(6,0);
  lcd.print(cm);
  lcd.setCursor(9,0);
  lcd.print("cm");
}
void sensor2(){
  pinMode(trig1,OUTPUT);
  digitalWrite(trig1,LOW);
  delayMicroseconds(2);
```

```
digitalWrite(trig1, HIGH);
delayMicroseconds(5);
digitalWrite(trig1,LOW);
pinMode(echo1,INPUT);
duration1 = pulseIn(echo1, HIGH);
cm1 = microsecondsToCentimeters(duration1);
  lcd.setCursor(0,0);
  lcd.print(cm1);
  lcd.setCursor(3,0);
  lcd.print("cm");
}
void sensor3(){
pinMode(trig2,OUTPUT);
digitalWrite(trig2,LOW);
delayMicroseconds(2);
digitalWrite(trig2, HIGH);
delayMicroseconds(5);
digitalWrite(trig2,LOW);
pinMode(echo2,INPUT);
duration2 = pulseIn(echo2, HIGH);
cm2 = microsecondsToCentimeters(duration2);
  lcd.setCursor(11,0);
  lcd.print(cm2);
  lcd.setCursor(14,0);
  lcd.print("cm");
}
void sensor4(){
pinMode(trig3,OUTPUT);
digitalWrite(trig3,LOW);
```

```
delayMicroseconds(2);
digitalWrite(trig3, HIGH);
delayMicroseconds(5);
digitalWrite(trig3,LOW);
pinMode(echo3,INPUT);
duration3 = pulseIn(echo3, HIGH);
cm3 = microsecondsToCentimeters(duration3);
  lcd.setCursor(6,1);
  lcd.print(cm3);
  lcd.setCursor(9,1);
  lcd.print("cm");
}
void sensor5(){
pinMode(trig4,OUTPUT);
digitalWrite(trig4,LOW);
delayMicroseconds(2);
digitalWrite(trig4, HIGH);
delayMicroseconds(5);
digitalWrite(trig4,LOW);
pinMode(echo4,INPUT);
duration4 = pulseIn(echo4, HIGH);
cm4 = microsecondsToCentimeters(duration4);
  lcd.setCursor(0,1);
  lcd.print(cm4);
  lcd.setCursor(3,1);
  lcd.print("cm");
}
void sensor6(){
pinMode(trig5,OUTPUT);
```

```
digitalWrite(trig5,LOW);
delayMicroseconds(2);
digitalWrite(trig5, HIGH);
delayMicroseconds(5);
digitalWrite(trig5,LOW);
pinMode(echo5,INPUT);
duration5 = pulseIn(echo5, HIGH);
cm5 = microsecondsToCentimeters(duration5);
  lcd.setCursor(11,1);
  lcd.print(cm5);
  lcd.setCursor(14,1);
  lcd.print("cm");
}
long microsecondsToCentimeters(long microseconds)
{
return microseconds / 29 / 2;
}

//Driver Motor
void maju()
{
analogWrite(pinkanan1, 145);
analogWrite(pinkanan2, 0);
analogWrite(pinkiri1, 170);
analogWrite(pinkiri2, 0);
}
void maju1()
{
analogWrite(pinkanan1, 140);
```

```
analogWrite(pinkanan2, 0);
analogWrite(pinkiri1, 140);
analogWrite(pinkiri2, 0);
}

void maju2(){
analogWrite(pinkanan1, 110);
analogWrite(pinkanan2, 0);
analogWrite(pinkiri1, 170);
analogWrite(pinkiri2, 0);
}
void maka()
{
analogWrite(pinkanan1, 0);
analogWrite(pinkanan2, 50);
analogWrite(pinkiri1, 170);
analogWrite(pinkiri2, 0);
}
void maki()
{
analogWrite(pinkanan1, 145);
analogWrite(pinkanan2, 0);
analogWrite(pinkiri1, 0);
analogWrite(pinkiri2, 120);
}
void mundur2()
{
analogWrite(pinkanan1, 0);
analogWrite(pinkanan2, 100);
```

```
analogWrite(pinkiri1, 0);
analogWrite(pinkiri2, 140);
}
void mundur1()
{
analogWrite(pinkanan1, 0);
analogWrite(pinkanan2, 80);
analogWrite(pinkiri1, 0);
analogWrite(pinkiri2, 100);
}
void mundur()
{
analogWrite(pinkanan1, 0);
analogWrite(pinkanan2, 100);
analogWrite(pinkiri1, 0);
analogWrite(pinkiri2, 130);
}
void muka()
{
analogWrite(pinkanan1, 50);
analogWrite(pinkanan2, 0);
analogWrite(pinkiri1, 0);
analogWrite(pinkiri2, 255);
}
void muki()
{
analogWrite(pinkanan1, 0);
analogWrite(pinkanan2, 255);
analogWrite(pinkiri1, 120);
```

```
analogWrite(pinkiri2, 0);
}
void berhenti()
{
analogWrite(pinkanan1, 0);
analogWrite(pinkanan2, 0);
analogWrite(pinkiri1, 0);
analogWrite(pinkiri2, 0);
}
void bacasensor()
{
  sensordepan = cm;
  sensorkiri = cm1;
  sensorkanan = cm2;
  sensorbelakang = cm3;
  sensorkiriBLK = cm4;
  sensorkananBLK = cm5;

  if (sensordepan <= 4){
  sensordepan = 30;
  }
  if (sensorkiri <=4){
  sensorkiri = 30;
  }
  if (sensorkanan <= 4){
  sensorkanan = 30;
  }
  if (sensorbelakang <= 4){
  sensorbelakang = 30;
```

```
}  
if (sensorkiriBLK <= 4){  
  sensorkiriBLK = 30;  
}  
if (sensorkananBLK <= 4){  
  sensorkananBLK = 30;  
}  
sensor1();  
sensor2();  
sensor3();  
sensor4();  
sensor5();  
sensor6();  
}
```