

**SISTEM NAVIGASI PADA WAHANA BAWAH AIR
TANPA AWAK**

*NAVIGATION SYSTEM ON THE UNDERWATER ROBOT
WITHOUT CREW*

Oleh:

Muhammad Nurul Fauzi
NRP. 7106 030 037

Dosen Pembimbing :

Didik Setyo Purnomo, S.T., M.Eng
NIP. 132.134.724

Dr.Eng. Indra A. S., S.T., M.Eng.
NIP. 19670527.199401.1.001

**JURUSAN TEKNIK ELEKTRONIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2009**



PROYEK AKHIR

**SISTEM NAVIGASI PADA WAHANA BAWAH AIR
TANPA AWAK**

*NAVIGATION SYSTEM ON THE UNDERWATER ROBOT
WITHOUT CREW*

Oleh:

Muhammad Nurul Fauzi
NRP. 7106 030 037

Dosen Pembimbing :

Didik Setyo Purnomo, S.T., M.Eng.
NIP. 132.134.724

Dr.Eng. Indra Adji Sulistijono, S.T., M.Eng.
NIP. 19670527.199401.1.001

**JURUSAN TEKNIK ELEKTRONIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2009**

**SISTEM NAVIGASI PADA WAHANA BAWAH AIR
TANPA AWAK**

Oleh :

Muhammad Nurul Fauzi
NRP. 7106 030 037

**Proyek Akhir ini Diajukan Sebagai Salah Satu Syarat Untuk
Memperoleh Gelar Ahli Madya (A.Md.)
di
Politeknik Elektronika Negeri Surabaya
Institut Teknologi Sepuluh Nopember Surabaya**

Disetujui Oleh

Dosen Penguji Proyek Akhir :

Dosen Pembimbing:

1.

Ir. Ratna Adil, M.T.
NIP. 19510323.198711.2.001

2.

Eko Henfri B., S.ST., M.Sc.
NIP. 19791223.200312.1.002

3.

Endah Suryawati N., S.T., M.T.
NIP. 19750112.200012.2.001

1.

Didik Setyo P., S.T., M.Eng.
NIP. 132.134.724

2.

Dr.Eng. Indra A. S., S.T., M.Eng.
NIP. 19670527.199401.1.001

Mengetahui :

Ketua Jurusan Teknik Elektronika

Ir. Rika Rokhana, M.T.
NIP. 19690905.199802.2.001

ABSTRAK

Secara umum, berdasarkan sistem pengendaliannya robot bawah air dibagi menjadi menjadi dua jenis yaitu *Autonomous Underwater Vehicles* (AUV) dan *Remoted Operated Vehicles* (ROV). AUV adalah kendaraan bawah air yang mampu bergerak didalam air secara otomatis tanpa adanya kontrol langsung dari manusia. Sedangkan ROV adalah kendaraan bawah air yang gerakannya dikendalikan secara langsung oleh manusia melalui *remote kontrol* dari atas permukaan air.

Hal yang paling penting pada robot bawah air adalah sistem pengendalian dan bahan kedap air yang akan digunakan. Oleh sebab itu pada proyek akhir yang berjudul “ *Sistem Navigasi Pada Wahana Bawah Air Tanpa Awak* ” ini digunakan sistem kontrol navigasi dengan menggunakan kabel untuk memberikan instruksi pada robot bawah air. Untuk penggerak robot, digunakan baling-baling motor DC sebagai alat untuk bermanuver. Sedangkan bahan penyusun kerangka robot ter buat dari pipa paralon untuk memanipulasi berat robot.

Dalam pengujian, untuk pemilihan bahan penyusun body juga baik, hal ini terlihat ketika robot berada pada kedalaman kurang lebih satu meter ternyata body robot masih bisa bermanuver meskipun agak lambat dikarenakan peletakan komponen elektronika dan pemberat kurang seimbang. Dengan prosentase keberhasilan bahan kedap air 100% berhasil. Penggunaan kompas magnetik sebagai sensor posisi robot mencapai keberhasilan antara 88,15% sampai 98%. Pemilihan media kabel sebagai pengirim instruksi input navigasi juga cukup berhasil, karena instruksi yang dikirim bisa cepat direspon oleh robot.

Kata kunci : Remote Operated Vehicles (ROV), Remot Kontrol, Navigasi, Motor DC

ABSTRACT

In generally, the robot control system based on the bottom of the water to be divided into two types, namely Autonomous Underwater Vehicles (AUV) and Remoted Operated Vehicles (ROV). AUV is under water vehicle capable of moving in the water automatically without direct human control. ROV while the vehicle is under water moving which is controlled directly by the people through the remote control from the top surface of the water.

The most important on the robot system is under the control of water and water-resistant material that will be used. Thus the end of the project entitled "Navigation System On The Underwater Robot Without Crew" are used with the navigation control system using a cable to provide instruction in the robot under water. To drive a robot, used rotor DC motor as a tool for maneuver. While the compiler framework of a robot pitch pipe paralon to manipulate the robot weight.

In the test, the composer for the election body is also good, this is visible when the robot is located at a depth of approximately one meter turns robot body can still maneuver although some what slow due to electronics components and laying ballast less balanced. Percentage of success with water-resistant material 100% successful. The use of magnetic compass as a position sensor robot succeed between 88.15% to 98%. With the use of media as the cable input navigation instructions also quite successful, because the instructions sent more quickly respons by robots..

Keyword : *Remote Operated Vehicles (ROV), remote control, Navigation, DC Motor*

KATA PENGANTAR

Alhamdulillah, segala puji syukur bagi Allah SWT karena berkat rahmat dan hidayah-Nya, penulis dapat menyelesaikan proyek akhir yang berjudul :

“ SISTEM NAVIGASI PADA WAHANA BAWAH AIR TANPA AWAK ”

Pembuatan dan penyusunan proyek akhir ini diajukan sebagai salah satu syarat untuk menyelesaikan studi Diploma-3(D3) dan memperoleh gelar Ahli Madya (A.Md.) di jurusan Elektronika Politeknik Elektronika Negeri Surabaya, Institut Teknologi Sepuluh Nopember Surabaya.

Penulis berusaha secara optimal dengan segala pengetahuan dan informasi yang didapatkan dalam menyusun laporan proyek akhir ini. Namun, penulis menyadari berbagai keterbatasannya, karena itu penulis memohon maaf atas keterbatasan materi laporan proyek akhir ini. Penulis sangat mengharapkan masukan berupa saran dan kritik yang membangun demi kesempurnaan laporan proyek akhir ini.

Demikian besar harapan penulis agar laporan proyek akhir ini dapat bermanfaat bagi pembaca, khususnya dalam mempelajari sistem navigasi pada robot bawah air.

Surabaya, 10 Juli 2009

Penulis

UCAPAN TERIMA KASIH

Dengan penuh rasa syukur kehadiran Allah S.W.T dan tanpa menghilangkan rasa hormat yang mendalam, kepada junjungan kita Nabi Muhammad SAW yang telah memberikan petunjuk dari jaman kegelapan menuju jaman yang terang-benderang melalui agama islam. Saya selaku penyusun dan penulis mengucapkan terima kasih yang sebesar-besarnya kepada pihak-pihak yang telah membantu penulis untuk menyelesaikan proyek akhir ini, penulis mengucapkan terima kasih kepada :

1. Kedua orang tuaku tercinta, **Bapak Abd. Kamid** dan **Ibu Kumaidah** yang selalu memberikan kasih sayang, bimbingan, dukungan dan mendo'akan sehingga proyek akhir ini dapat terlaksana sesuai jadwal yang ditentukan. Proyek akhir ini kupersembahkan untuk Beliau berdua.
2. Bapak **Dr. Ir. Dadet Pramadihanto, M.Eng** selaku direktur PENS-ITS.
3. Ibu **Ir. Rika Rokhana, M.T.** selaku ketua jurusan Teknik Elektronika PENS-ITS.
4. Bapak **Didik Setyo Purnomo, S.T., M.Eng**, selaku dosen pembimbing proyek akhir saya yang memberikan motivasi dan bimbingan kepada saya demi terselesaikannya proyek akhir ini.
5. Bapak **Dr.Eng Indra Adji Sulistijono, S.T., M.Eng**, selaku dosen yang selalu memberikan motivasi dan petuah bijak kepada penulis dalam menyelesaikan proyek akhir ini, semoga Bapak dapat menjalankan tugas dengan lancar di negeri sakura sana.
6. Ibu **Ir. Ratna Adil, M.T.**, Bapak **Eko Henfri B., S.ST, M.Sc.** , Ibu **Endah Suryawati N., S.T., M.T.**, selaku para dosen penguji proyek akhir yang telah memberikan saran dan masukan yang membangun pada penyelesaian proyek akhir ini.
7. Seluruh Bapak dan Ibu dosen yang telah membimbing dan membekali ilmu kepada penulis selama penulis menempuh pendidikan di kampus tercinta ini, **Politeknik Elektronika Negeri Surabaya-ITS (PENS-ITS)**.
8. Keluargaku tercinta, **Mas Bambang, Mba' Yuli, Mba' Ani(Alm), Mas Malik** dan Ponakan kecilku **M. Syafri "Saproll" Az-Zuhri**, kalian semua adalah inspirasiku...

9. Untuk keluarga besar **Bapak Sanali** dan **Bapak Kasim(Alm)**, terima kasih semuanya...
10. Seluruh guru MI, SMP, SMA serta Pesantren yang telah membekali ilmu dalam menjalani kehidupan.
11. Untuk Partner PA-ku Taufik Iskandar “ngelu bareng bro”, Naruto dan keluarga trims telah diperbolehkan mengacak-acak rumahnya buat ngerjain PA, Hyuga Art yang telah mengajari dan membantu dalam pengerjaan proyek akhir ini.
12. Onny dan keluarga, selaku partner kuliah bareng terima kasih banyak ya..
13. Teman-teman kos kejawan gang IV no.24, Deni konco kamar yang sering gua repotin...
14. Teman-teman EB D3'06, teman-teman lab.D3 dan lab.D4, thanks yaaaa....
15. Semua pihak yang telah membantu penulis hingga terselesainya proyek akhir ini yang tidak dapat penulis sebutkan secara satu-persatu.

Semoga Allah S.W.T selalu memberikan perlindungan, rahmat dan nikmat-Nya bagi kita semua. Amin.....

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN	ii
ABSTRAK	iii
ABSTRACT	iv
KATA PENGANTAR	v
UCAPAN TERIMA KASIH	vi
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
BAB I. PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Tujuan Proyek Akhir	2
1.3 Perumusan Masalah	2
1.4 Batasan Masalah	3
1.5 Metodologi	3
1.6 Sistematika Pembahasan	4
BAB II. TEORI PENUNJANG	
2.1 Teori Motor DC	7
2.1.1 Prinsip kerja motor DC	7
2.1.2 Bagian-bagian Motor DC	7
2.2 Mikrokontroler AT89S51	9
2.3 Pemrograman Bahasa Assembly	11
2.4 Mikrokontroler AVR ATmega 16	13
2.5 Pemrograman Bahasa C	15
2.6 Komunikasi Serial	17
2.7 CMPS03 Modul Magnetik Kompas	19
2.8 Kontrol Jarak Jauh	21
BAB III. PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK	
3.1 Konfigurasi Sistem	23
3.2 Perancangan dan Pembuatan Perangkat Keras	27
3.2.1 Perancangan dan Pembuatan Mikrokontroler	27
3.2.2 Rangkaian <i>Clock Generator</i>	27

3.2.3Perancangan Rangkaian Power Supply	28
3.2.4Perancangan I/O	29
3.2.5Perancangan dan Pembuatan Kontrol Arah	31
3.2.6Perancangan <i>Driver</i> Motor DC.....	32
3.3 Perancangan dan Pembuatan Perangkat Lunak	33
3.4 Perancangan dan Mekanik.....	34
3.4.1Perancangan dan Pembuatan Baling-baling.....	34
3.4.2Perancangan dan Pembuatan Bodi Robot.....	36
 BAB IV. PENGUJIAN DAN ANALISA	
4.1 Pengujian Perangkat Elektronik	39
4.1.1Pengujian Minimum Sistem	39
4.1.2Pengujian <i>Driver</i> Motor.....	41
4.1.3Pengujian Tombol Kontrol Arah dan Program Kontrol.....	41
4.2 Pengujian Mekanik	42
4.2.1Pengujian Baling-baling Motor DC	42
4.2.2Pengujian Kerangka dan Bodi Robot.....	42
4.3 Pengujian Setelah Terjadi Perubahan Pada Sistem Navigasi Robot Robot Bawah Air.....	45
4.3.1Pengujian Pembacaan Modul CMPS03/sensor Posisi	45
4.3.2Pengujian Tombol Kontrol Arah dan <i>Driver</i> Motor DC.....	47
4.3.3Pengujian Integrasi Sistem Secara Keseluruhan...	47
 BAB V. PENUTUP	
5.1 Kesimpulan	55
5.2 Saran	56
 DAFTAR PUSTAKA	
LAMPIRAN	
BIODATA PENULIS	

DAFTAR GAMBAR

Gambar 1.1 (a) Penyelaman secara langsung	1
Gambar 1.1 (b) Penyelaman dengan robot bawah air.....	1
Gambar 1.1 Cara penyelaman dalam air	1
Gambar 2.1 Bentuk motor DC	7
Gambar 2.2 Prinsip kerja motor dc.....	8
Gambar 2.3 Bagian-bagian motor DC	9
Gambar 2.4 Konfigurasi pin AT89S51	10
Gambar 2.5 Konfigurasi pin ATmega16	14
Gambar 2.6 Diagram pengiriman data secara <i>serial</i>	18
Gambar 2.7 Konfigurasi pin pada CMPS03.....	19
Gambar 2.8 Start bit dan stop bit pin scl dan sda	20
Gambar 3.1 Blok diagram sistem navigasi robot bawah air sebelum terjadi perubahan sistem	23
Gambar 3.2 Diagram alir sistem navigasi pada robot bawah air sebelum terjadi perubahan pada sistem.....	24
Gambar 3.3 Blok diagram sistem navigasi robot setelah terjadi Perubahan sistem.....	25
Gambar 3.4 Diagram alir system navigasi robot bawah air setelah Terjadi perubahan sistem	26
Gambar 3.5 Rangkaian oscilator.....	28
Gambar 3.6 Rangkaian Power Supply untuk mikrokontroler	29
Gambar 3.7 Rangkaian mikrokontroler, driver dan regulator	30
Gambar 3.8 Rangkaian tombol kontrol arah	32
Gambar 3.9 Rangkaian switching transistor dan driver relay	32
Gambar 3.10 Motor DC	34
Gambar 3.11 Motor DC yang sudah dibungkus bahan kedap air.....	35
Gambar 3.12 Baling-baling motor DC pada robot bawah air	35
Gambar 3.13 Mekanik robot bawah air.....	36
Gambar 3.14 Prototype robot bawah air	38
Gambar 4.1 Pembacaan nilai kompas arah utara	46
Gambar 4.2 Pembacaan nilai kompas arah timur.....	46
Gambar 4.3 Pembacaan nilai kompas arah selatan	46
Gambar 4.4 Pembacaan nilai kompas arah barat	46
Gambar 4.5 Kalibrasi kompas.....	49
Gambar 4.6 Kompas arah timur laut	50
Gambar 4.7 Kompas arah timur	50

Gambar 4.8 Kompas arah tenggara.....	51
Gambar 4.9 Kompas arah selatan.....	51
Gambar 4.10 Kompas arah barat daya	52
Gambar 4.11 Kompas arah barat.....	52
Gambar 4.12 Kompas arah barat laut.....	53
Gambar 4.13 Kompas arah utara.....	53

DAFTAR TABEL

Tabel 2.1 Ukuran memori untuk tipe data.....	16
Tabel 3.1 Data input dan output kontroler.....	27
Tabel 4.1 Hasil Pengujian Minimum Sistem	40
Tabel 4.2 Hasil pengujian driver	41
Tabel 4.3 Pengujian tombol pushbutton dan program kontrol	42
Tabel 4.4 Pengujian kekedapan baling-baling motor DC	43
Tabel 4.5 Hasil pengujian kekedapan kerangka dan bodi robot	52
Tabel 4.5 Hasil pengujian pembacaan status kompas magnetik....	54

BAB I PENDAHULUAN

1.1 LATAR BELAKANG

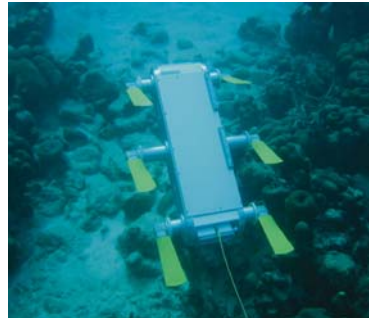
Secara umum, berdasarkan sistem pengendaliannya robot bawah air dibagi menjadi menjadi dua jenis yaitu *Autonomous Underwater Vehicles (AUV)* dan *Remoted Operated Vehicles (ROV)*. AUV adalah kendaraan bawah air yang mampu bergerak didalam air secara otomatis tanpa adanya kontrol langsung dari manusia. Sedangkan ROV adalah kendaraan bawah air yang gerakannya dikendalikan secara langsung oleh manusia melalui *remote control* dari atas permukaan air.

Dari kedua jenis kendaraan bawah laut tersebut pada dasarnya mempunyai tugas yang sama yaitu untuk melakukan misi/kegiatan di bawah permukaan air, akan tetapi satu dengan lainnya mempunyai kelebihan dan kekurangan dalam menjalankan misi bawah air tersebut. Untuk saat ini, pengembangan robot bawah air lebih ditekankan pada *Remote Operated Vehicle (ROV)* daripada *Autonomous Underwater Vehicle (AUV)* dikarenakan ROV memiliki kelebihan untuk menjalankan tugas-tugas yang menuntut ketelitian dan keakuratan.[7]

Akhir-akhir ini, perkembangan teknologi robot bawah air berkembang pesat dan telah banyak digunakan dalam pengembangan eksplorasi kelautan dan militer. Untuk eksplorasi kelautan, diperlukan pemantauan kondisi bawah laut secara tepat, teliti dan akurat. Gambar 1.a dan 1.b menjelaskan cara-cara pengamatan kondisi bawah laut.



Gambar 1.1(a)
Penyelaman secara langsung



Gambar 1.1(b)
Penyelaman dengan robot bawah air

Gambar 1.1 Cara penyelaman dalam air

Gambar 1.1.(a) memperlihatkan pengamatan kondisi bawah laut yang dilakukan oleh manusia dengan cara menyelam secara langsung ke dalam laut. Cara ini mengandalkan keahlian penyelam dalam mengambil gambar/video beserta keadaan bawah air lainnya. Jika keahlian dan kondisi fisik penyelam kurang optimal, maka keselamatan jiwa penyelam akan terancam dan hasil pengamatan bawah laut pun akan tidak sesuai harapan. Pada gambar 1.1.(b) pengamatan bawah laut dilakukan oleh robot bawah air (ROV) yang dikontrol oleh operator dari atas permukaan air [1]. Cara ini tidak membahayakan jiwa manusia, tetapi mengandalkan kemampuan operator dalam membuat dan mengoperasikan (ROV) robot bawah air tersebut.

Berdasarkan uraian di atas, mahasiswa termotivasi untuk membuat proyek akhir dengan judul "SISTEM NAVIGASI WAHANA BAWAH AIR TANPA AWAK" dengan harapan proyek akhir tersebut bisa bermanfaat dan membantu dalam pengamatan kondisi bawah air.

1.2 TUJUAN PROYEK AKHIR

Adapun tujuan dari pembuatan proyek akhir ini adalah:

1. Membuat suatu sistem navigasi pada ROV dengan tetap menjaga keseimbangan robot sewaktu di dalam air.
2. Membuat robot yang bermanfaat dan mampu untuk memberikan informasi-informasi penting dalam menjalankan misi di dalam air.

1.3 PERUMUSAN MASALAH

Dalam mengerjakan proyek akhir ini ada beberapa tahapan permasalahan yang harus diselesaikan, antara lain:

- Mekanik
 - 1) Bagaimana membuat body robot yang sederhana serta mampu mengapung dan tenggelam di dalam air.
 - 2) Bagaimana membuat sistem kedap air agar piranti elektronika yang ada pada kerangka robot tidak rusak.
 - 3) Bagaimana membuat keseimbangan yang baik ketika robot berada di dalam air.
- Hardware
 - 1) Bagaimana mengintegrasikan mikrokontroler dengan baling-baling.

- 2) Bagaimana membuat rangkaian sesimpel mungkin agar tidak membutuhkan ruangan yang besar sehingga memudahkan dalam penempatan.
- Software
 - 1) Membuat program yang dapat digunakan untuk mengontrol robot dari atas permukaan air.

1.4 BATASAN MASALAH

Berdasarkan permasalahan diatas dapat dibuat suatu batasan masalah dalam pembuatan proyek akhir ini. Batasan masalah yang dimaksud di antaranya :

1. Kontrol pada robot dilakukan dengan menggunakan kabel dari atas permukaan air.
2. Perubahan akibat pengaruh gaya hidrostatika dan hidrodinamika diabaikan.
3. Pengujian dilakukan pada kondisi air tenang dan dengan kedalaman kurang lebih 100 cm.

1.5 METODOLOGI PROYEK AKHIR

Adapun penjelasan tahapan metodologi dalam penyelesaian proyek akhir ini adalah:

- **Tahap studi literatur dan tinjauan lapangan**

Tahap ini merupakan tahap persiapan dalam menyelesaikan proyek akhir, dimana bertujuan untuk memperoleh teori-teori penunjang yang melandasi pemecahan masalah dilapangan, baik itu bersumber dari referensi, buku diktat, *web site*, ataupun jurnal ilmiah.

- **Perancangan Sistem**

Pada tahap inilah mahasiswa merancang suatu sistem (baik *software* maupun *hardware*) yang dalam pembuatannya dilakukan pada tahap berikutnya.

➤ **Pembuatan Hardware dan Software**

Pada tahap ini dilakukan:

- a. Pembuatan mekanik robot.
- b. Pembuatan minimum sistem.
- c. Pembuatan *driver* motor.
- d. Pembuatan pushbutton kontrol arah.
- e. Pembuatan program kontrol.

➤ **Pengujian dan Analisis Sistem**

Dalam tahap ini, dilakukan berbagai macam pengujian diantaranya yaitu menguji keseimbangan robot didalam air, kesesuaian gerakan kemudi robot dengan perintah yang diberikan, serta sistem kedap air pada body robot. Analisa sistem dilakukan setelah dilakukan pengujian, apakah sistem sudah bekerja dengan baik atau belum.

➤ **Perbaikan perangkat keras dan perangkat lunak**

Tahap ini dilakukan untuk penyempurnaan bila ada sistem yang belum bisa bekerja secara optimal sebelum pembuatan laporan.

➤ **Pembuatan Laporan Akhir**

Pembuatan laporan akhir dilaksanakan setelah semua langkah-langkah terselesaikan sehingga hasil yang diperoleh dari pembuatan alat dapat dijelaskan secara rinci sesuai dengan data-data yang diperoleh.

1.6 SISTEMATIKA PEMBAHASAN

Sistematika pembahasan dalam proyek akhir ini adalah sebagai berikut :

➤ **BAB I PENDAHULUAN**

Menguraikan tentang latar belakang, tujuan, permasalahan, batasan masalah, metodologi, dan sistematika pembahasan masalah yang digunakan dalam pembuatan proyek akhir ini.

➤ **BAB II TEORI PENUNJANG**

Berisi teori tentang pembahasan secara garis besar mikrokontroller AT89S51, mikrokontroller AT Mega16, bahasa assembly, bahasa c, dan motor DC.

➤ **BAB III PERENCANAAN DAN PEMBUATAN**

Membahas secara detail tentang perencanaan dan pembuatan sistem yang akan dibangun diantaranya adalah perancangan mekanik, dan sistem perangkat keras. Pada bagian perangkat keras akan membahas tentang pembuatan minimum system AT89S51, minimum system ATMega16, rangkaian *driver* motor DC dan rangkaian kontrol. Untuk perangkat lunak akan dijelaskan tentang pembuatan program kontrol pada robot.

➤ **BAB IV PENGUJIAN DAN ANALISA**

Membahas tentang pengujian dari sistem yang telah dibuat beserta analisisnya.

➤ **BAB V KESIMPULAN DAN SARAN**

Berisi tentang kesimpulan dan saran yang berdasarkan analisa hasil data yang diperoleh.

➤ **DAFTAR PUSTAKA**

Pada bagian ini berisi tentang referensi-referensi yang telah dipakai oleh penulis sebagai acuan dan penunjang serta parameter yang mendukung penyelesaian proyek akhir ini baik secara praktis maupun teoritis.

*****Halaman ini sengaja dikosongkan*****

BAB II

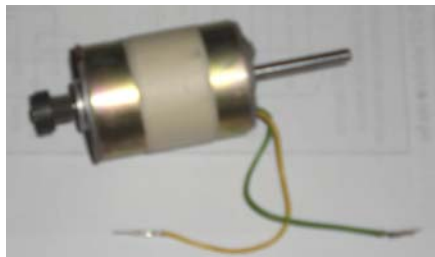
TEORI PENUNJANG

Teknologi robot bawah air (ROV) dalam aplikasinya ternyata telah menghasilkan efisiensi kinerja bila dibandingkan dengan cara konvensional, bahkan ROV mampu menjalankan tugas yang teknologi konvensional tidak mampu melakukannya. Dengan ROV, banyak sekali penemuan-penemuan besar terjadi sehingga keberadaan ROV sampai saat ini sangat dibutuhkan untuk melaksanakan misi didalam air.

Dalam proyek akhir kali ini agar ROV bisa menjalankan misi dalam air dengan baik dan optimal maka dalam merancang, membuat, dan mengimplementasikan robot bawah air, diperlukan pemahaman tentang teori-teori yang berhubungan dengan pembuatan robot bawah air tersebut.

2.1 TEORI MOTOR DC

Motor DC berfungsi mengubah tenaga listrik menjadi tenaga mekanis dimana gerak tersebut berupa putaran dari motor. Motor DC pada saat ini digunakan pada industri yang memerlukan gerakan dengan kepresisian yang tinggi untuk pengaturan kecepatan pada torsi yang konstan.

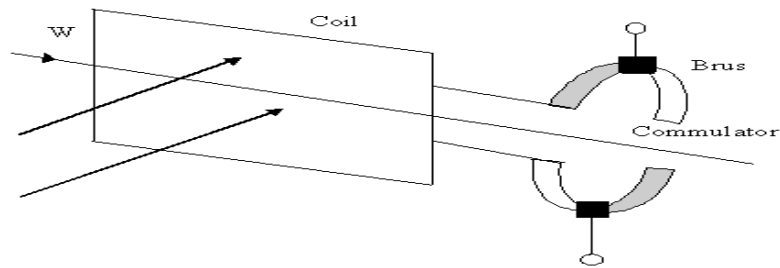


Gambar 2.1. Bentuk motor DC

2.1.1 Prinsip Kerja Motor DC

Motor DC berfungsi mengubah tenaga listrik menjadi tenaga mekanis dimana gerak tersebut berupa putaran dari motor. Prinsip dasar dari motor arus searah adalah bila sebuah kawat berarus diletakkan

diantara kutub magnet (U-S), maka pada kawat itu akan bekerja suatu gaya yang menggerakkan kawat itu. Arah gerakan kawat dapat ditentukan dengan menggunakan kaidah tangan kiri, yang berbunyi sebagai berikut :”Apabila tangan kiri terbuka diletakkan diantara kutub U dan S, sehingga garis-garis gaya yang keluar dari kutub utara menembus telapak tangan kiri dan arus di dalam kawat mengalir searah dengan arah keempat jari, maka kawat itu akan mendapat gaya yang arahnya sesuai dengan arah ibu jari”. Lebih jelasnya dapat dilihat pada gambar 2.2 berikut.



Gambar 2.2. Prinsip kerja motor DC

Pada motor arus searah medan magnet akan dihasilkan oleh medan dengan kerapatan fluks sebesar B . bila kumparan jangkar yang dilingkupi medan magnet dari kumparan medan dialiri arus sebesar I , maka akan menghasilkan suatu gaya F dengan besarnya gaya tersebut adalah:

$$F = B I L \dots\dots\dots (1)$$

Dimana :

- F = gaya pada penghantar (Newton)
- B = kepadatan fluks magnet (Tesla)
- I = arus listrik yang mengalir (Ampere)
- L = panjang penghantar (meter)

2.1.2 Bagian-bagian Motor DC

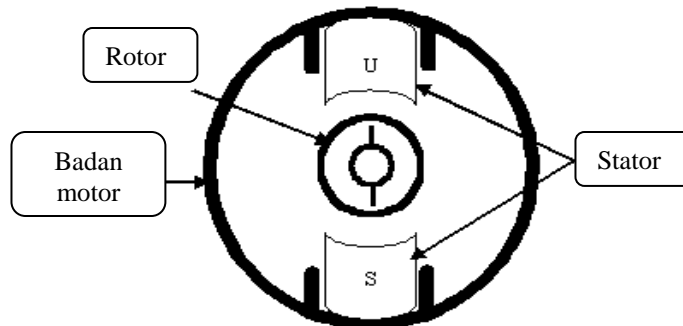
Motor DC mempunyai dua bagian dasar yaitu :

- 1) Bagian diam/tidak berputar (Stator)
Stator ini menghasilkan medan magnet, baik yang dibangkitkan dari sebuah koil (elektromagnetik) atau magnet permanen.

Bagian stator terdiri dari bodi motor yang memiliki magnet yang melekat padanya. Untuk motor kecil, magnet tersebut adalah magnet permanen sedangkan untuk motor besar menggunakan elektromagnetik. Kumputan yang dililitkan pada lempeng-magnet disebut kumputan medan.

2) Bagian berputar (Rotor)

Rotor ini berupa sebuah koil dimana arus listrik mengalir. Suatu kumputan motor akan berfungsi apabila mempunyai Kumputan medan, berfungsi sebagai penghasil medan magnet. Kumputan medan, berfungsi sebagai pembangkit GGL pada konduktor yang terletak pada laur-alur jangkar. Celah udara yang memungkinkan berputarnya jangkar dalam medan magnet.



Gambar 2.3. Bagian-bagian motor DC

2.2 Mikrokontroler AT89S51

Mikrokontroler, sesuai namanya adalah suatu alat atau komponen pengontrol atau pengendali yang berukuran mikro atau kecil. Sebelum ada mikrokontroler, telah ada terlebih dahulu muncul mikroprosesor. Bila dibandingkan dengan mikroprosesor, mikrokontroler jauh lebih unggul karena terdapat berbagai alasan, diantaranya :

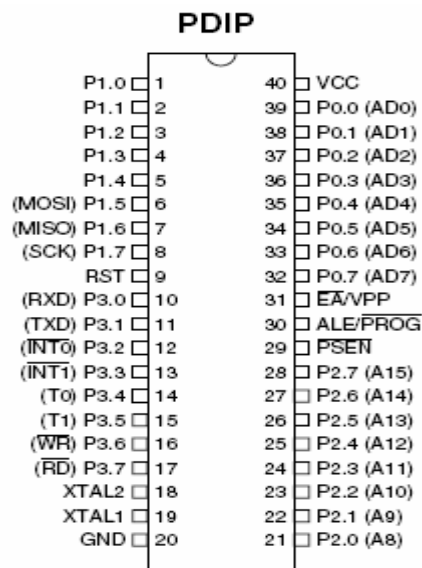
- Tersedianya I/O

I/O dalam mikrokontroler sudah tersedia, bahkan untuk AT89S51 ada 32 jalur I/O, sementara pada mikroprosesor dibutuhkan IC tambahan untuk menangani I/O tersebut. IC I/O yang dimaksud adalah PPI 8255.

- Memori Internal

Memori merupakan media untuk menyimpan program dan data sehingga mutlak harus ada. Mikroprosesor belum memiliki memori internal sehingga memerlukan IC memori eksternal.

Dengan kelebihan – kelebihan di atas, ditambah dengan harganya yang relatif murah sehingga banyak penggemar elektronika yang kemudian beralih ke mikrokontroler. Namun demikian, meski memiliki berbagai kelemahan, mikroprosesor tetap digunakan sebagai dasar dalam mempelajari mikrokontroler.. Dengan memiliki dasar pengetahuan yang cukup tentang mikroprosesor, maka pada saat belajar mikrokontroler kita akan cepat mengerti dengan lebih sempurna. Inti kerja dari keduanya adalah sama, yakni sebagai pengendali suatu sistem. Keistimewaan dari AT89S51 dapat dilihat pada lembar lampiran datasheet.



Gambar 2.4. Konfigurasi pin AT89S51

2.3 PEMROGRAMAN BAHASA ASSEMBLY

Berikut ini akan dijelaskan beberapa instruksi dasar yang umum dipakai pada pemrograman AT89S51.

- **MOV, MOVC dan MOVX**
 1. Instruksi MOV digunakan untuk menyalin data antara 2 operand
 2. Instruksi MOVC digunakan untuk menyalin data yang terdapat pada memori program memori eksternal.
 3. Instruksi MOVX digunakan untuk menyalin data yang terdapat pada memori program eksternal.

- **ADD dan SUB**
 1. Instruksi ADD digunakan untuk melakukan proses operasi penjumlahan akumulator dengan suatu operand dan hasilnya disimpan dalam akumulator.
 2. Instruksi SUB digunakan untuk melakukan operasi pengurangan akumulator dengan suatu operand dan hasilnya disimpan dalam akumulator.

- **MUL AB dan DIV AB**
 1. Instruksi MUL AB digunakan untuk melakukan operasi perkalian antara akumulator dengan register B. Hasilnya berupa data 16 bit dengan *low byte* pada A dan *high byte* pada B.
 2. Instruksi DIV AB digunakan untuk melakukan operasi pembagian antara akumulator dengan register B. Hasilnya, pembagian disimpan pada akumulator dan sisa pembagian pada register B.

- **DEC dan INC**
 1. Instruksi DEC digunakan untuk melakukan pengurangan sebesar 1 pada suatu *operand*.
 2. Instruksi INC digunakan untuk melakukan penambahan sebesar 1 pada suatu *operand*.

- **ORL, AND dan CPL**

1. Instruksi ORL digunakan untuk melakukan operasi OR antara 2 *operand*.
2. Instruksi ANL digunakan untuk melakukan operasi AND antara 2 *operand*.
3. Instruksi CPL digunakan untuk melakukan operasi komplemen suatu *operand*.

➤ **RR, RL dan SWAP**

1. Instruksi RR digunakan untuk melakukan operasi pergeseran ke kanan sebanyak 1 bit.
2. Instruksi RL digunakan untuk melakukan operasi pergeseran ke kiri sebanyak 1 bit.
3. Instruksi SWAP digunakan untuk melakukan operasi pertukaran data *low nibble* dan *high nibble*.

➤ **SETB dan CLR**

1. Instruksi SETB digunakan untuk memberikan logik 1 pada bit operand.
2. Instruksi CLR digunakan untuk memberikan logik 0 pada bit operand.

➤ **PUSH dan POP**

1. Instruksi PUSH digunakan untuk menyimpan operand ke dalam stack.
2. Instruksi POP digunakan untuk mengembalikan nilai operand ke dalam stack.

➤ **JMP, JB, JNB, JZ, JNZ, dan CJNE**

1. JMP (*Jump*) digunakan untuk melakukan lompatan ke suatu blok program.
2. JB (*Jump if Bit*) dan JNB (*Jump if Not Bit*) digunakan untuk melakukan lompatan ke suatu blok program jika nilai operand 1 (bit) atau 0 (Not Bit).
3. JZ (*Jump if Zero*) dan JNZ (*Jump if Not Zero*) digunakan untuk melakukan lompatan ke suatu blok program jika nilai operand 0 atau bukan 0.
4. CJNE (*Compare and Jump if Not Zero*) digunakan untuk melakukan perbandingan 2 operand dan lompat ke blok program lain jika tidak sama.

➤ **CALL dan RET**

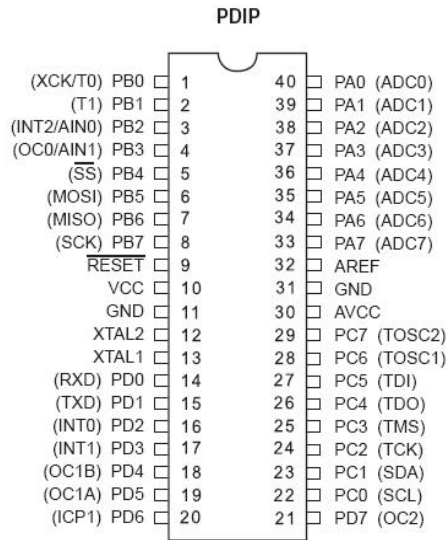
1. Instruksi CALL yang digunakan untuk memanggil prosedur tertentu dalam program (sub program).
2. Instruksi RET yang digunakan untuk mengembalikan baris program yang melakukan CALL.

2.4 Mikrokontroler AVR AT Mega16

AVR (*Advance Versatile RISC*) adalah mikrokontroler RISC (*reduce instruction set computer*) 8 bit berdasarkan arsitektur havard yang dibuat tahun 1996. Penggunaan mikrokontroler ini dikarenakan AVR memiliki keunggulan yaitu kecepatan eksekusi program yang lebih besar. Hal ini karena sebagian besar instruksi dieksekusi dalam 1 siklus *clock*. Pin-pin yang terdapat pada AVR AT MEGA 32 dapat dijelaskan sebagai berikut:

- a. VCC merupakan *pin* yang berfungsi sebagai masukan catu daya.
- b. GND merupakan *pin ground*.
- c. Port A (PA0-PA7) , Port B (PB0-PB7) , Port C (PC0-PC7) , dan Port D (PD0-PD7) merupakan *pin input/output* dua arah dan *pin* fungsi khusus.
- d. RESET merupakan *pin* yang digunakan untuk *me-reset* mikrokontroler.
- e. XTAL1 dan XTAL2 merupakan *pin* masukan *clock* eksternal.
- f. AVCC merupakan *pin* masukan tegangan untuk ADC.
- g. AREF merupakan *pin* masukan tegangana referensi ADC.

Instruksi pada memori program dieksekusi dengan *pipelining single level*. Selagi sebuah instruksi sedang dikerjakan, instruksi berikutnya diambil dari memori program. Arsitektur CPU dari AVR ditunjukkan oleh gambar. Untuk arsitektur dan keistimewaan AT Mega16 bisa dilihat pada lampiran datasheet AT Mega 16.



Gambar 2.5. Konfigurasi Pin AT Mega 16

➤ Port sebagai input/output digital

ATMega16 mempunyai empat buah port yang bernama PortA, PortB, PortC, dan PortD. Keempat port tersebut merupakan jalur bidirectional dengan pilihan internal pull-up. Tiap port mempunyai tiga buah register bit, yaitu DDxn, PORTxn, dan PINxn. Huruf 'x' mewakili nama huruf dari port sedangkan huruf 'n' mewakili nomor bit. Bit DDxn terdapat pada I/O address DDRx, bit PORTxn terdapat pada I/O address PORTx, dan bit PINxn terdapat pada I/O address PINx. Bit DDxn dalam register DDRx (Data Direction Register) menentukan arah pin. Bila DDxn diset 1 maka Px berfungsi sebagai pin output. Bila DDxn diset 0 maka Px berfungsi sebagai pin input. Bila PORTxn diset 1 pada saat pin terkonfigurasi sebagai pin input, maka resistor pull-up akan diaktifkan. Untuk mematikan resistor pull-up, PORTxn harus diset 0 atau pin dikonfigurasi sebagai pin output.

Pin port adalah tri-state setelah kondisi reset. Bila PORTxn diset 1 saat pin terkonfigurasi sebagai pin output maka pin port akan berlogika 1. Dan bila PORTxn diset 0 pada saat pin terkonfigurasi sebagai pin output maka pin port akan berlogika 0. Saat mengubah

kondisi port dari kondisi *tri-state* (DDxn=0, PORTxn=0) ke kondisi *output high* (DDxn=1, PORTxn=1) maka harus ada kondisi peralihan apakah itu kondisi *pull-up enabled* (DDxn=0, PORTxn=1) atau kondisi *output low* (DDxn=1, PORTxn=0). Biasanya, kondisi pull-up enabled dapat diterima sepenuhnya, selama lingkungan impedansi tinggi tidak memperhatikan perbedaan antara sebuah *strong high driver* dengan sebuah pull-up. Jika ini bukan suatu masalah, maka bit PUD pada register SFIOR dapat diset 1 untuk mematikan semua pull-up dalam semua port. Peralihan dari kondisi *input dengan pull-up* ke kondisi *output low* juga menimbulkan masalah yang sama. Kita harus menggunakan kondisi *tri-state* (DDxn=0, PORTxn=0) atau kondisi *output high* (DDxn=1, PORTxn=0) sebagai kondisi transisi. Bit 2 – PUD : Pull-up Disable Bila bit diset bernilai 1 maka pull-up pada port I/O akan dimatikan walaupun register DDxn dan PORTxn dikonfigurasi untuk menyalakan pull-up (DDxn=0, PORTxn=1).

2.5 PEMROGRAMAN BAHASA C

Akar bahasa C adalah bahasa BCPL yang dikembangkan oleh Martin Richards pada tahun 1967. Bahasa C adalah bahasa standart, artinya suatu program yang ditulis dengan versi bahasa C tertentu akan dapat dikompilasi dengan versi bahasa C yang lain dengan sedikit modifikasi. Beberapa alasan mengapa bahasa C banyak digunakan, diantaranya adalah sebagai berikut ini :

1. Bahasa C tersedia hampir di semua jenis komputer.
2. Kode bahasa C sifatnya portabel.
3. Bahasa C hanya menyediakan sedikit kata – kata kunci.
4. Proses *executable program* bahasa C lebih cepat.
5. Dukungan Pustaka yang banyak.
6. C adalah bahasa yang terstruktur.
7. Selain bahasa tingkat tinggi, C juga dianggap sebagai bahasa tingkat Menengah.
8. Bahasa C adalah compiler.

➤ Struktur Penulisan Program C

Struktur dari program C dapat dilihat sebagai kumpulan dari sebuah atau lebih fungsi – fungsi. Fungsi pertama yang harus ada di program C sudah ditentukan namanya, yaitu bernama main(). Suatu

fungsi di program C dibuka dengan kurung kurawal ({} dan ditutup dengan kurung kurawal tertutup (}). Diantara kurung kurawal dapat dituliskan statemen – statemen program C. Berikut ini adalah struktur dari program C

1. Tipe-tipe dasar

Data merupakan suatu nilai yang bias dinyatakan dalam bentuk konstanta atau variabel. Konstanta menyatakan nilai yang tetap, sedangkan variabel menyatakan nilai yang dapat diubah- ubah selama eksekusi berlangsung.

Data berdasarkan jenisnya dapat dibagi menjadi lima kelompok, yang dinamakan sebagai tipe data dasar. Kelima tipe data dasar adalah:

- Bilangan bulat (*integer*)
- Bilangan real presisi-tunggal
- Bilangan real-presisi ganda
- Karakter
- Tak bertipe (*void*)

Tabel di bawah memberikan informasi mengenai ukuran memori yang diperlukan dan kawasan dari masing – masing tipe data dasar.

Tabel 2.1. Ukuran memori untuk tipe data

Tipe	Total Bit	Kawasan	Keterangan
char	8	-128 s/d 127	Karakter
int	32	-2147483648 s/d 2147483647	Bilangan Interger
float	32	1.7E-38 s/d 3.4E+38	Bil. Real presisi-tunggal
double	64	2.2E-308 s/d 1.7E308	Bil real presisi-ganda

2. Operator

Operator atau tanda operasi adalah suatu tanda atau simbol yang digunakan untuk suatu operasi tertentu. Operator untuk operasi aritmatika yang tergolong sebagai operator binary adalah :

- (*) Perkalian
- (/) Pembagian
- (%) Sisa Pembagian
- (+) Penjumlahan
- (-) Pengurangan

Adapun operator yang tergolong sebagai operator unary

- (-) Tanda Minus
- (+) Tanda plus

3. Fungsi- Fungsi Umum yang Sering Digunakan

Menampilkan data

- **Fungsi *printf()*** Fungsi *printf()* merupakan fungsi yang paling umum digunakan dalam menampilkan data. Berbagai jenis data dapat ditampilkan ke layar dengan memakai fungsi ini.

- **Fungsi *putchar()***

Fungsi *putchar()* digunakan khusus untuk menampilkan sebuah karakter di layar. Penampilan karakter tidak diakhiri dengan perpindahan baris. Contoh :

Putchar ('A');

Menghasilkan keluaran yang sama dengan

Printf ("%c", 'A');

Memasukkan Data dari Keyboard

Data dapat dimasukkan lewat keyboard saat eksekusi berlangsung. Untuk keperluan ini, C menyediakan sejumlah fungsi, diantaranya adalah *scanf()* dan *getchar()*.

- **Fungsi *scanf()***

Fungsi *scanf()* merupakan fungsi yang dapat digunakan untuk memasukkan berbagai jenis data.

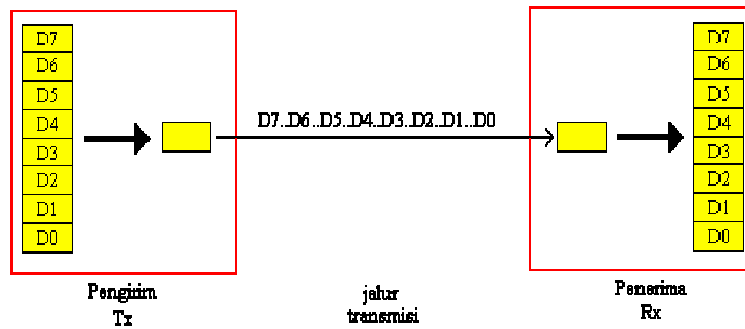
- **Fungsi *getchar()***

Fungsi *getchar()* digunakan khusus untuk menerima masukan berupa sebuah karakter dari keyboard.

2.6 Komunikasi *Serial*

Pada komunikasi serial data yang dikirimkan berbeda dengan cara pengiriman pesan secara parallel. Jika pada parallel data bit yang dikirimkan itu lebih dari satu bit dan dikeluarkan dalam waktu yang

bersamaan. Namun pada serial hanya ada satu bit data yang akan terkirim dalam satu waktu. [4]



Gambar 2.6. Diagram pengiriman data secara *serial*

Dalam pengiriman data secara serial membutuhkan sinkronisasi antara pengirim dan penerima agar data bisa dikirim dan diterima secara benar. Ada 2 mode komunikasi dalam serial yaitu mode sinkron dan mode asinkron. Pada mode sinkron data dikirim bersamaan dengan sinyal clock sehingga antara satu karakter dan yang lainnya memiliki jeda waktu yang sama. Sedangkan pada mode asinkron data dikirim tanpa sinyal clock. Hal itu menyebabkan karakter yang dikirimkan dapat sekaligus atau beberapa karakter dengan jeda waktu yang berbeda. Hal itu dikarenakan pada mode asinkron pengiriman data yang tidak melalui sinyal clock sehingga antara satu karakter dan karakter yang lainnya tidak ada waktu yang tetap. Bit bit data yang dikirim dapat diterima kapan sata oleh penerima. Namun untuk menyinkronkan data maka diberikan bit-bit penanda awal dari data tersebut dan penanda akhir di kedua sisi baik penerima maupun pengirim. Format data komunikasi serial terdiri dari parameter – parameter yang dipakai untuk menentukan bentuk data serial yang dikomunikasikan, dimana elemennya terdiri dari :

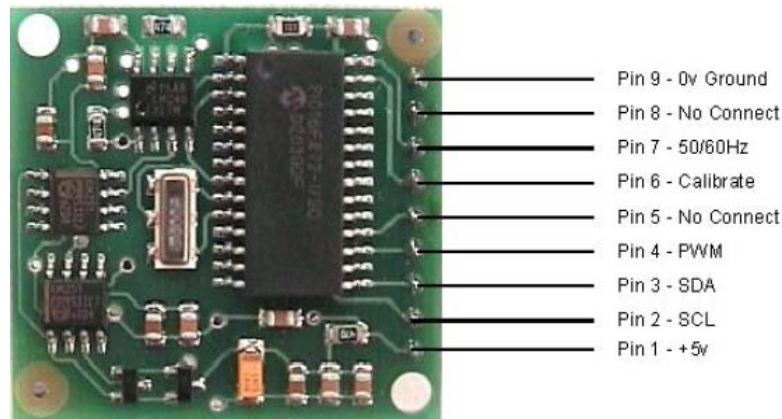
1. Kecepatan mobilisasi data per bit (*baud rate*).
2. Jumlah bit data per karakter (*data length*).
3. Parity yang digunakan untuk menjaga integritas data.
4. Jumlah *stop* bit dan *start* bit.

2.7 CMPS03 Modul Magnetic Compass

Dalam mengontrol sistem navigasi/pergerakan suatu robot bawah air, diperlukan suatu sensor posisi untuk mengetahui dimana dan kemana arah robot yang akan kita kendalikan. Jadi dengan penggunaan sensor posisi pada robot bawah air, diharapkan robot tidak hanya bergerak maju, mundur, kanan, dan kiri saja tetapi juga harus mengetahui arah dan posisi dari robot tersebut. [2]

Pada proyek akhir ini, sensor posisi yang digunakan adalah CMPS03 modul magnetik kompas, modul kompas ini didesain khusus dalam bidang robotik untuk tujuan navigasi robot. Kompas ini menggunakan dua sensor medan magnet KMZ51 buatan Philips yang cukup peka untuk mendeteksi medan magnet bumi. Dua sensor ini dipasang saling bersilangan. Pada modul kompas telah dipasang rangkaian pengkondisi sinyal dan mikrokontroler. Sehingga kita dapat mengakses berapa derajat posisi kompas secara langsung. [8]

Hubungan pin-pin pada modul kompas

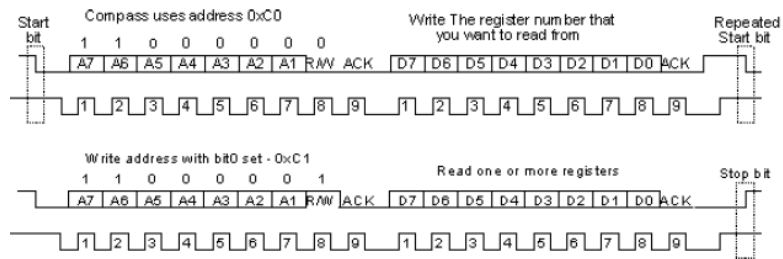


Gambar 2.7. Konfigurasi pin pada CMPS03

Modul kompas membutuhkan suplai tegangan sebesar 5VDC dengan konsumsi arus sekitar 15mA. Ada dua cara untuk membaca posisi magnet. Yaitu melalui sinyal PWM pada pin nomor 4 atau menggunakan protokol I2C pada pin nomor 2 dan 3.

Sinyal PWM yang dihasilkan oleh kompas merupakan sinyal yang lebar pulsanya dapat berubah-ubah. Pulsa berlogika 1 menyatakan derajat. Lebar pulsa berlogika 1 bervariasi antara 1mili-detik (untuk 0°) sampai 36,99mili-detik (untuk 359,9°). Dengan kata lain kompas memiliki resolusi 100μ-detik/° dengan *offset* sebesar +1mili-detik. Sinyal kemudian akan berlogika 0 selama 65mili-detik. Jadi periode sinyal PWM sebesar 65mili-detik ditambah dengan waktu sinyal yang berlogika 1, atau 66mili-detik sampai 102mili-detik. Sinyal PWM tersebut dihasilkan oleh timer 16-bit dari prosesor pada modul kompas yang menghasilkan resolusi sebesar 1μ-detik. Sehingga disarankan oleh pembuatnya untuk mendeteksi sinyal PWM dengan timer yang resolusinya lebih rendah dari yang dihasilkan oleh kompas. Yakinkan bahwa pin untuk I2C, SDA dan SCL, dihubungkan ke suplai 5VDC melalui resistor pull-up, karena pin SDA dan pin SCL tidak mempunyai pull-up.

Pin 2 dan 3 digunakan untuk berkomunikasi dengan protokol (bahasa) I2C untuk mengambil nilai posisi kompas.



Gambar 2.8. Start bit dan stop bit pin scl dan sda

Komunikasi dengan protokol I2C pada modul kompas mempunyai cara yang sama seperti mengakses eeprom serial tipe 24C04 misalnya. Pertama kirim *start-bit*, alamat kompas (0xC0) dengan bit R/W low, kemudian nomor register yang ingin diakses. Selanjutnya diulang dengan mengirimkan *start-bit*, alamat kompas dengan bit R/W high (0xC1). Kemudian isi *register* dibaca. Fungsi register pada CMPS03 bisa dilihat pada lampiran Datasheet register CMPS03.

Pin I2C tidak mempunyai resistor pull-up pada board sehingga harus ditambahkan pada jalur komunikasi yang digunakan. Pembuat modul kompas menyarankan untuk memasang resistor 1k8 jika diinginkan bekerja pada kecepatan 400kHz dan 1k2 atau bahkan 1k bila

ingin bekerja pada kecepatan 1MHz. Modul kompas didesain untuk bekerja pada frekuensi standar (SCL) sebesar 100kHz, walaupun kecepatan sinyal clock bisa ditingkatkan sampai 1Mhz dengan beberapa tindakan yang harus diperhatikan. Pada kecepatan diatas sekitar 160kHz, CPU tidak dapat merespon dengan cepat untuk membaca data I2C. oleh karena itu delay sesaat sebesar 50 μ -detik harus ditambahkan diantara pengiriman alamat register.

Pin 7 adalah pin input untuk memilih operasi kerja 50Hz atau 60Hz. Pin ini ditambahkan setelah terlihat adanya jitter sekitar 1,5 $^{\circ}$ pada output. Penyebabnya adalah sumber listrik 50Hz pada lingkungan kerja. Dengan melakukan sinkronisasi dengan frekuensi sumber listrik dapat dikurangi sampai 0,2 $^{\circ}$. Konversi internal selesai setiap 40mili-detik (50Hz) atau setiap 33,3mili-detik (60Hz). Pin 7 mempunyai resistor pull-up pada board sehingga dapat dibiarkan tidak terhubung untuk operasi kerja 60Hz. Antara output PWM atau I2C dan proses konversi tidak ada sinkronisasi. Output PWM dan I2C mengambil pembacaan internal yang terbaru, yang dikonversi secara kontinu, apakah dipakai atau tidak.

Pin 6 digunakan untuk mengkalibrasi kompas. Pin ini memiliki resistor pull-up pada board sehingga dapat dibiarkan tak terhubung setelah melakukan kalibrasi. Pin 5 dan pin 8 tidak dihubungkan. Sebenarnya pin 8 merupakan jalur reset dan memiliki resistor pull-up pada board. Disiapkan untuk memprogram mikrokontroler yang terpasang pada board.

2.8 Remote Kontrol Arah

Untuk mengendalikan navigasi/pergerakan robot bawah air dapat digunakan bermacam-macam sistem kendali, seperti kontrol dengan kabel ataupun tanpa kabel. Beberapa peneliti sebelumnya telah mengaplikasikan kontrol tanpa kabel (*remote control*) pada robot buatannya, seperti Robinson and Keary (2000). Agar robot dapat dikendalikan dengan navigasi yang lebih baik maka robot juga dilengkapi dengan kamera. Selain untuk navigasi, kamera juga berfungsi untuk pengambilan gambar maupun untuk merekam gambar.[6]

Pada proyek akhir ini digunakan remote kontrol arah dengan memanfaatkan tombol pushbutton sebagai pengirim instruksi/perintah pada robot. Untuk mengontrol arah pergerakan/navigasi pada robot,

tombol pushbutton yang digunakan dihubungkan dengan kabel ke mikrokontroler. Kabel disini berfungsi sebagai media penghantar instruksi dari pushbutton ke mikrokontroler. [3]

Cara kerja kontrol *pushbutton* ini berdasarkan pada ada tidaknya penekanan tombol. Jika salah satu tombol ditekan maka kabel akan mengirimkan sinyal pada *input* mikrokontroler. Setelah itu perintah diolah pada mikro kemudian digunakan untuk mengaktifkan *driver* motor. *Driver* akan menggerakkan motor berdasar fungsi pada kontrol pada tombol.

BAB III PERANCANGAN DAN PEMBUATAN

3.1. KONFIGURASI SISTEM

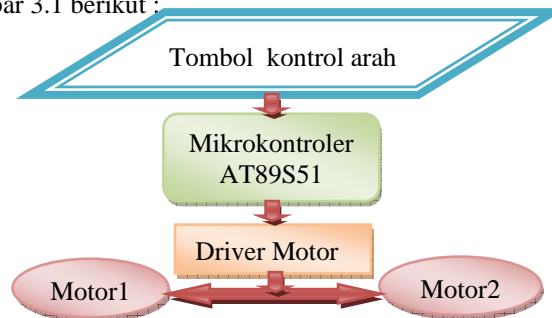
Dalam pembuatan sistem navigasi robot bawah air pada proyek akhir ini, secara umum terdiri dari tiga bagian dasar, yaitu bagian perangkat keras (*hardware*), bagian perangkat lunak (*software*), dan bagian mekanik. Dimana ketiga perangkat tersebut merupakan satu kesatuan yang saling berkaitan agar terjadi suatu harmonisasi kerja. Sistem tersebut akan menyediakan data bagi sistem kontrol untuk mengatur jalannya robot.

Pada proyek akhir ini terjadi perubahan pada sistem yang digunakan, sebab-sebab perubahan sistem antara lain :

- 1) Pada sistem sebelumnya belum terdapat sensor posisi,
- 2) Dengan adanya sensor posisi diharapkan robot tidak hanya bergerak maju, mundur, kanan, dan kiri saja, tetapi juga mengetahui dimana dan kemana posisi dan arah yang akan dituju.

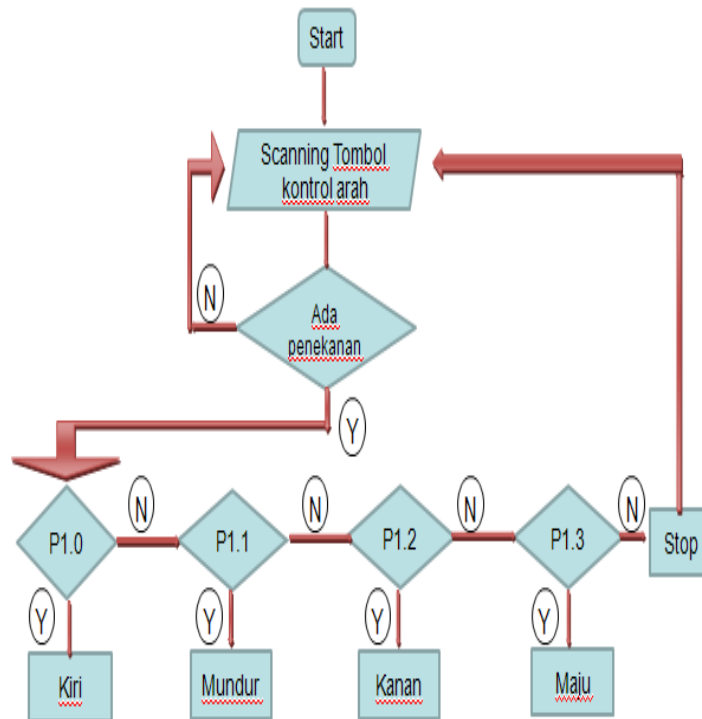
Sebelum terjadi perubahan perangkat keras yang digunakan sebagai sistem navigasi adalah rangkaian minimum sistem mikrokontroler AT89S51. Rangkaian minimum sistem ini dilengkapi dengan rangkaian pushbutton sebagai input masukan dan rangkaian driver yang berfungsi sebagai penggerak motor.

Blok diagram sistem navigasi secara keseluruhan sebelum terjadi perubahan sistem pada robot bawah air yang dibuat dapat dideskripsikan pada gambar 3.1 berikut :



Gambar 3.1. Blok diagram sistem navigasi robot bawah air sebelum terjadi perubahan sistem

Diagram alir sistem navigasi sebelum terjadi perubahan sistem pada robot bawah air



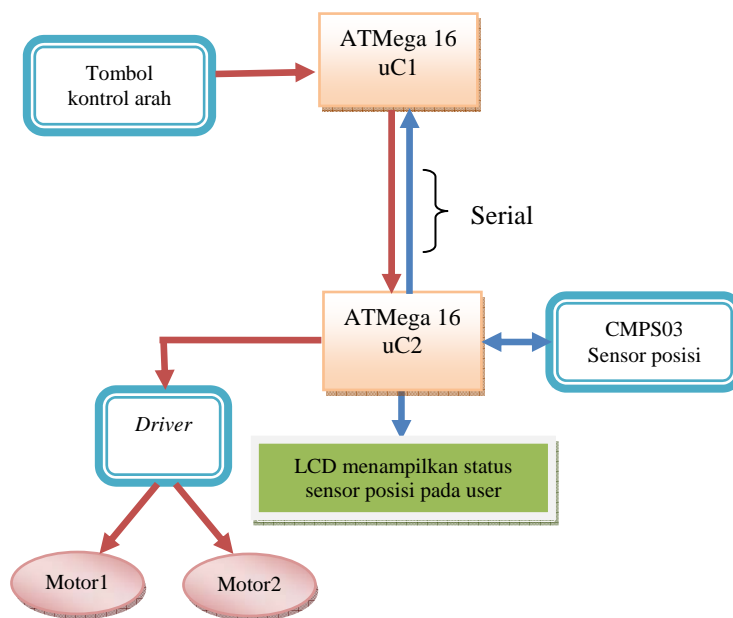
Gambar 3.2. Diagram alir sistem navigasi pada robot bawah air sebelum terjadi perubahan sistem

Adapun cara kerja dari sistem diatas adalah sebagai berikut:

1. Kapal menjalankan perintah berdasarkan scanning pada tombol kontrol arah,
2. Ketika ada penekanan pada tombol, tombol akan mengirimkan data ke input mikro untuk diolah menjadi suatu perintah,
3. Setelah mikro akan mengaktifkan driver,
4. *Driver* akan menjalankan motor sesuai dengan instruksi yang diberikan.

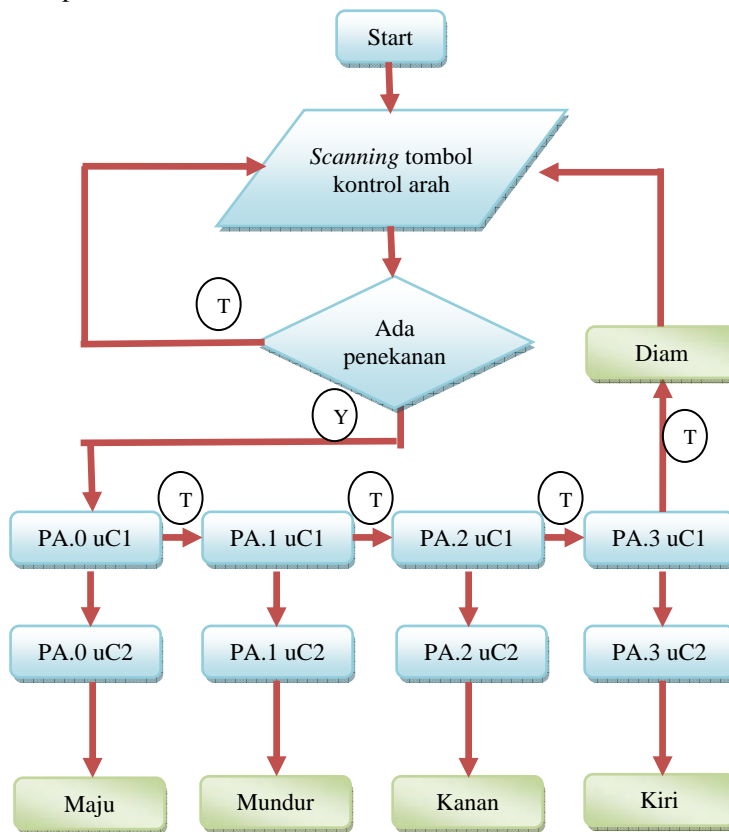
Setelah mengalami perubahan sistem, perangkat keras yang digunakan dalam proyek akhir ini adalah rangkaian minimum sistem mikrokontroler ATMEGA16. Proyek ini membutuhkan dua rangkaian minimum sistem ATMEGA16 dimana rangkaian yang pertama (uC1) berfungsi sebagai pengolah masukan data dari tombol kontrol arah dan menampilkan sudut inklinasi kompas pada LCD. Rangkaian uC1 ini terletak diluar bodi robot. Untuk rangkaian yang kedua (uC2) berfungsi menerima data dari uC1 untuk mengaktifkan *driver* motor dan juga memberikan data sudut *inklinasi* kompas pada uC1. Rangkaian uC2 diletakkan pada bodi robot. Kedua rangkaian ini berkomunikasi satu sama lain dengan cara serial (menghubungkan *pin tx-rx* serta *ground* masing-masing mikro).

Blok diagram sistem navigasi secara keseluruhan setelah mengalami perubahan sistem pada robot bawah air yang dibuat dapat dideskripsikan pada gambar 3.3 berikut :



Gambar 3.3. Blok diagram sistem navigasi robot bawah air setelah terjadi perubahan sistem

Diagram alir sistem navigasi setelah mengalami perubahan sistem pada robot bawah air



Gambar 3.4. Diagram alir sistem navigasi robot bawah air setelah terjadi perubahan sistem

Mekanisme kerja sistem navigasi setelah terjadi perubahan pada robot bawah air adalah sebagai berikut:

1. Sistem bekerja berdasarkan ada tidaknya penekanan tombol pada mikro1,
2. Jika ada penekanan tombol, mikro 1 akan memberikan perintah melalui pin serial pada mikro2 untuk menjalankan motor,

3. Jika motor bergerak dari posisi awal, otomatis kedudukan sensor posisi (kompas CMPS03) juga ikut berubah,
4. Perubahan posisi pada sensor posisi ditampilkan pada *display* LCD pada mikro2,
5. Dengan adanya sensor posisi yang nilainya ditampilkan pada LCD mikro2, maka pengontrolan sistem navigasi bisa dilakukan dengan lebih mudah.

3.2. PERANCANGAN DAN PEMBUATAN PERANGKAT KERAS

Sistem perangkat keras yang digunakan terbagi menjadi dua bagian utama, yaitu bagian kontroler dan tombol kontrol arah.

3.2.1. Perancangan dan Pembuatan Mikrokontroler

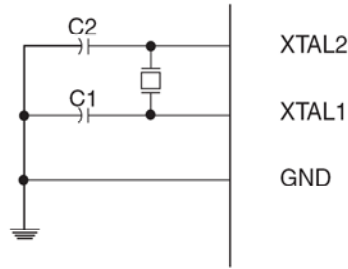
Dalam membuat rangkaian mikrokontroler memerlukan pemahaman mengenai sistem minimum dari mikrokontroler yang akan dirancang itu sendiri. Sistem rangkaian yang dirancang diusahakan menggunakan rangkaian yang sederhana mungkin dan dengan pengkabelan yang baik, karena biasanya rangkaian tersebut bekerja pada frekuensi yang relatif tinggi, sehingga peka terhadap *noise* dari luar.

Dalam pembuatan minimum sistem baik AT89S51 maupun ATmega16, diperlukan rangkaian penunjang untuk menjamin kehandalan dari minimum sistem tersebut. Rangkaian penunjang yang dibutuhkan antara lain:

- *clock generator* CPU
- *regulator* dan *noise filter*
- *interfacing* ke rangkaian luar (tergantung kebutuhan pemakai)

3.2.2. Rangkaian Clock Generator

Mikrokontroler AT89S51 dan ATmega16 memiliki *osilator internal* (*on chip oscillator*) yang dapat digunakan sebagai sumber *clock* bagi CPU. Untuk menggunakan *osilator internal* diperlukan sebuah kristal antara pin XTAL1 dan XTAL2 dan kapasitor ke *ground* seperti gambar (3.5). Untuk kristalnya digunakan kristal 12 MHz. Sedangkan untuk kapasitor dapat bernilai 27 pF sampai 33 pF.



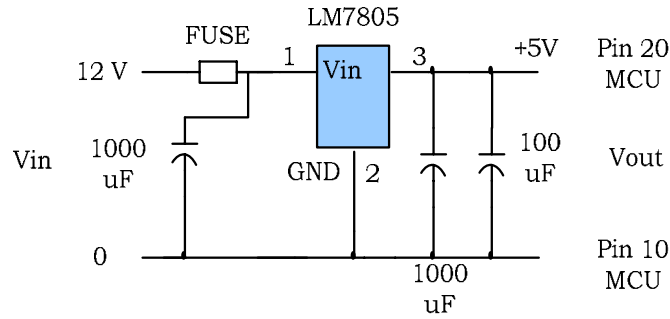
Gambar 3.5. Rangkaian *Oscillator*

3.2.3. Perancangan Rangkaian *Power Supply*

Rangkaian *power supply* yang digunakan untuk memberi *supply* tegangan mikrokontroler harus stabil, dan mempunyai arus yang cukup untuk mensupply mikrokontroler sehingga tidak terjadi *drop* tegangan saat mikrokontroler dioperasikan.

Mikrokontroler membutuhkan sebuah tegangan *supply* tunggal sebesar +5 Volt. Sumber tegangan yang digunakan untuk mensupply robot secara keseluruhan adalah menggunakan 2 buah *batere* kering 12V 1,2A yang diseri. Pemilihan *batere* ini karena bentuk fisiknya yang relatif kecil sehingga memudahkan peletakkannya pada bodi robot, disamping itu *batere* ini mempunyai kapasitas daya yang cukup untuk mensupply rangkaian secara keseluruhan.

Tegangan yang digunakan untuk mensupply mikrokontroler diambilkan dari *batere* yang terpasang pada badan robot. Supaya tegangan dari *batere* tersebut sesuai dengan tegangan kerja dari mikrokontroler, maka perlu diberikan rangkaian regulator tegangan yang berfungsi menurunkan tegangan dari *batere* dari 12 Volt menjadi 5 Volt. *Regulator* tegangan yang digunakan disini adalah dengan menggunakan IC 7805, seperti yang terlihat pada gambar 3.6 :



Gambar 3.6. Rangkaian *Power Supply* untuk Mikrokontroler

IC 7805 diatas mempunyai arus keluaran maksimal sampai 1 Ampere sehingga cukup untuk memberi *supply* pada mikrokontroler tanpa diberi rangkaian *buffer* arus lagi. Pemasangan kapasitor *filter* juga perlu dilakukan, karena biasanya *supply* yang berasal dari *batere* mendapatkan *noise* dari rangkaian motor. Dalam perancangannya grounding dan pengawatan serta *filter* dari rangkaian eksternal dari mikrokontroler harus baik untuk menghindari *noise* yang masuk ke kaki-kaki mikrokontroler terutama kaki mikrokontroler yang digunakan sebagai *clock*. *Noise* yang disebabkan oleh *ripple* tegangan power supply akan sangat mengganggu kestabilan pembangkitan frekwensi *clock*, karena ketidakstabilan dari rangkaian ini akan menurunkan performa mikrokontroler dan rangkaian secara keseluruhan, yang mana imbasnya akan juga mengurangi performa dari robot.

3.2.4. Perancangan Interfacing I/O

Rangkaian I/O dari mikrokontroler mempunyai kontrol direksi yang tiap bitnya dapat dikonfigurasi secara *individual*, maka dalam pengkonfigurasi I/O yang digunakan ada yang berupa operasi *port* ada pula yang dikonfigurasi tiap *bit* I/O.

Berikut ini akan diberikan konfigurasi dari I/O mikrokontroler tiap *bit* yang digunakan pada rangkaian kontroler AT89S51 :

◇ Port 3

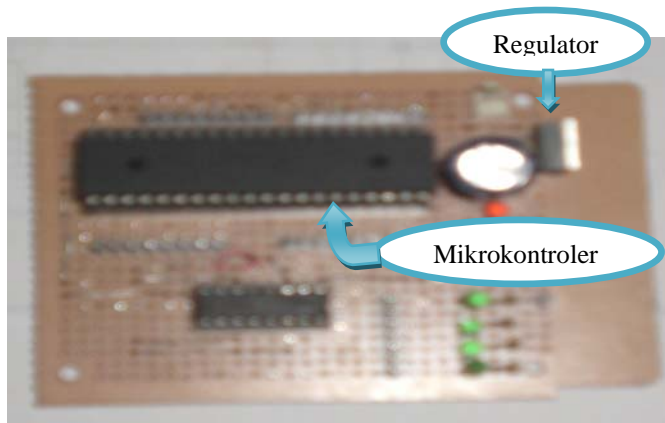
Port 3 digunakan sebagai input dengan konfigurasi sebagai berikut :

- Port 3.0 sebagai input tombol kontrol arah
- Port 3.1 sebagai input tombol kontrol arah
- Port 3.2 sebagai input tombol kontrol arah
- Port 3.3 sebagai input tombol kontrol arah

◇ **Port 0**

Port 0 digunakan sebagai keluaran data dari mikrokontroler ke *driver* motor, dengan konfigurasi sebagai berikut :

- Port 0.4 sebagai *output* data *biner*
- Port 0.5 sebagai *output* data *biner*
- Port 0.6 sebagai *output* data *biner*
- Port 0.7 sebagai *output* data *biner*



Gambar 3.7. Rangkaian mikrokontroler AT89S51 dan *regulator*

Berikut ini akan diberikan konfigurasi dari I/O mikrokontroler tiap bit yang digunakan pada rangkaian kontroler ATmega16 :

Untuk mikro1 konfigurasi I/O sebagai berikut :

◇ **PortA**

PortA digunakan sebagai input dengan konfigurasi sebagai berikut:

- PortA.0 sebagai input tombol kontrol arah
- PortA.1 sebagai input tombol kontrol arah

- PortA.2 sebagai input tombol kontrol arah
- PortA.3 sebagai input tombol kontrol arah
- ◇ PortD
 - PortD.0 digunakan sebagai komunikasi *serial*
 - PortD.1 digunakan sebagai komunikasi *serial*

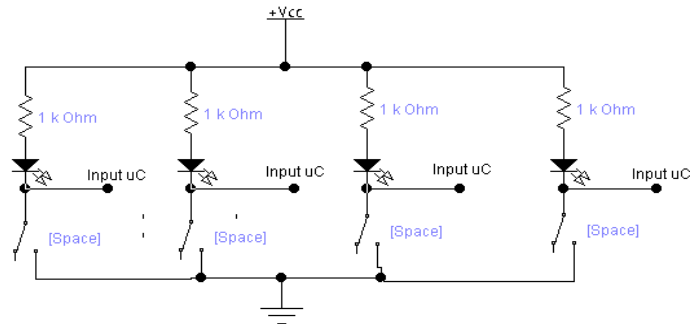
Untuk mikro2 konfigurasi I/O sebagai berikut:
- ◇ PortA
 - PortA.3 digunakan untuk komunikasi dengan *pin sda* pada modul CMPS03
 - PortA.4 digunakan untuk komunikasi dengan *pin scl* pada modul CMPS03
- ◇ PortB
 - PortB.0 digunakan sebagai *output data biner*
 - PortB.1 digunakan sebagai *output data biner*
 - PortB.2 digunakan sebagai *output data biner*
 - PortB.3 digunakan sebagai *output data biner*
- ◇ PortC

Pin pada portC digunakan untuk interfacing dengan LCD
- ◇ PortD
 - PortD.0 digunakan sebagai komunikasi *serial*
 - PortD.1 digunakan sebagai komunikasi *serial*

3.2.5. Perancangan dan Pembuatan Kontrol Arah

Dalam perancangan tombol kontrol arah ini, tombol yang dibuat dihubungkan dengan kabel untuk mengontrol pergerakan navigasi robot bawah air. Kabel disini berfungsi sebagai media penghantar instruksi dari tombol kontrol arah ke mikrokontroller.

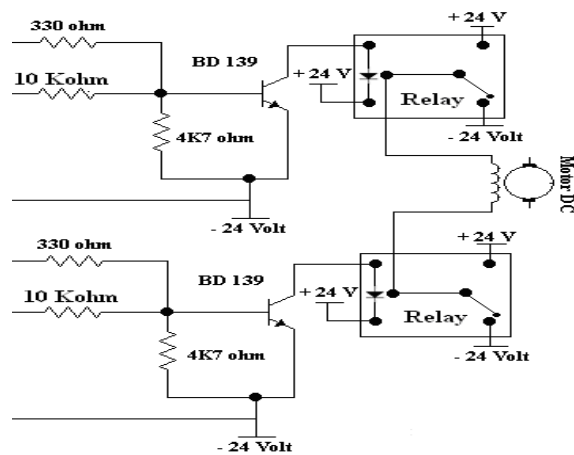
Cara kerja tombol kontrol arah ini berdasarkan pada ada tidaknya penekanan tombol. Jika salah satu tombol ditekan maka kabel akan mengirimkan sinyal pada input mikrokontroller. Setelah itu perintah diolah pada mikro kemudian digunakan untuk mengaktifkan *driver* motor. *Driver* akan menggerakkan motor berdasar fungsi kontrol pada tombol.



Gambar 3.8. Rangkaian tombol kontrol arah

3.2.6. Perancangan *Driver* Motor DC

Rangkaian *switching transistor* dan *driver relay* tampak pada Gambar 3.9 :



Gambar 3.9. Rangkaian *Switching Transistor* Dan *Driver Relay*

Prinsip kerja dari rangkaian ini adalah sama seperti sakelar *on – off*. Tipe transistor NPN yang digunakan adalah BD 139 karena mampu dilewati arus besar dengan tegangan 12 V. Pada *relay* juga dipasang dioda pemblokir arus balik akibat induksi magnet dari *solenoid relay*.

Untuk masing – aktuator terdiri dari dua *relay* dan dua *switching*. Kerena gerakan aktuator terdiri dari dua macam gerakan yaitu CW dan CCW. Apabila salah satu tombol ditekan maka motor akan berputar dan apabila tidak ada penekanan motor akan berhenti.

3.3. PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK

Perancangan program *software* sistem navigasi robot bawah air pada proyek akhir kali ini dibuat sesederhana mungkin dengan maksud agar mudah dipahami oleh pembaca. Meskipun sederhana, program ini cukup handal untuk bisa mengendalikan navigasi pada robot.

Program ini berjalan berdasarkan *scanning* pada tombol port1. Berikut ini tabel untuk memudahkan pembuatan program.

Tabel 3.1. Data *input* dan *output* kontroler

No	Input Tombol				Output				Ket.
	F	B	L	R	P0.7	P0.6	P0.5	P0.4	
1	1	0	0	0	L	H	H	L	Maju
2	0	1	0	0	H	L	L	H	Mundur
3	0	0	1	0	x	x	L	H	Kiri
4	0	0	0	1	H	L	x	x	Kanan
5	x	x	x	x	x	x	x	x	Stop

F = *forward* ; B = *backward* ; L = *left* ; R = *right*
 x = *don'tcare*

Program kontrol sistem navigasi bisa dilihat pada lembar lampiran *program kontrol sistem navigasi AT89S51*.

3.4. PERANCANGAN DAN PEMBUATAN MEKANIK

Pada proyek akhir kali ini, desain perancangan mekanik merupakan salah satu hal terpenting dalam membuat robot bawah air. Usahakan desain bodi agar tidak terlalu besar dan juga tidak terlalu kecil, hal ini dimaksudkan agar mempermudah dalam peletakan pemberat dan komponen. Perancangan dan pembuatan mekanik terbagi menjadi dua yaitu perancangan baling-baling serta perancangan bodi robot. perlu diperhatikan agar memudahkan dalam mengatur letak komponen dan pemberat.

3.4.1 Perancangan dan Pembuatan Baling-Baling

Untuk pembuatan baling-baling pada robot bawah air, hal yang harus diperhatikan adalah faktor kedap air. Disini baling-baling harus *waterproof* karena tersusun atas motor DC, jadi perencanaan pembuatannya harus dipikirkan secara matang. Hal-hal yang dilakukan dalam pembuatan baling-baling robot bawah air antara lain:

- 1) Membungkus motor DC kedalam pipa/paralon
- 2) Menutup bagian poros motor DC dengan sheell/karet kemudian ditutup dengan *acrilyc*
- 3) Menutup bagian belakang motor dengan karet
- 4) Mengintegrasikan motor dengan diameter luar baling-baling menggunakan klem
- 5) Melapisi bagian-bagian yang rawan kemasukan air dengan *gluegun*

Berikut ini adalah urutan gambar dalam pembuatan baling-baling motor DC

Spesifikasi motor DC :

Diameter = 35 mm

Panjang bodi motor = 55 mm

Diameter poros = 5 mm

Panjang poros (As) = 110 mm

Tegangan *supply* = 12 V

Arus = 2 A



Gambar 3.10. Motor DC

Spesifikasi pembungkus :

➤ Pipa

Diameter dalam pipa = 38 mm

Diameter luar pipa = 42 mm

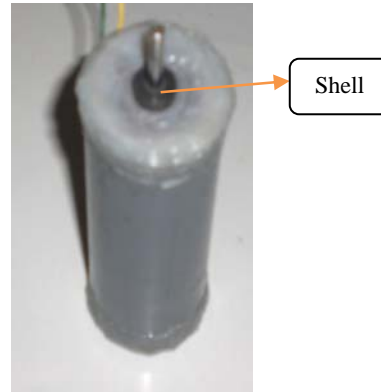
Panjang pipa = 90 mm

➤ Karet/shell

Diameter dalam = 5 mm

Diameter luar = 15 mm

Tebal = 4 mm



Gambar 3.11.

Motor DC yang sudah dibungkus bahan kedap air

Spesifikasi baling-baling :

➤ Pipa

Diameter dalam = 120 mm

Diameter luar = 125 mm

Panjang = 100 mm

➤ Baling-baling

Diameter = 110 mm



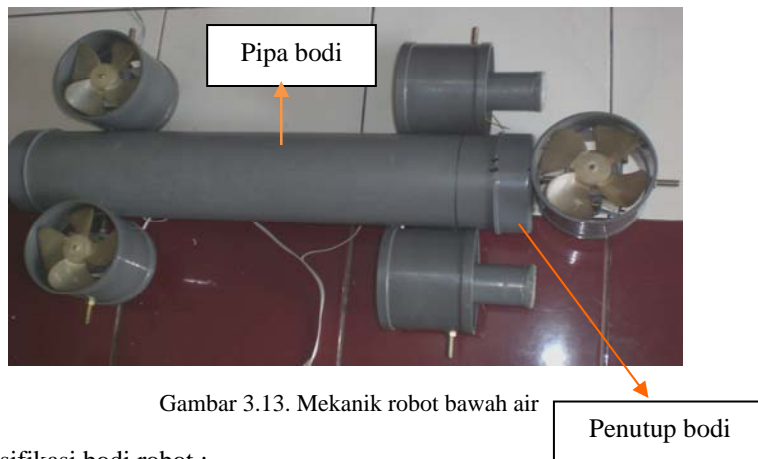
Gambar 3.12.

Baling-baling motor DC pada robot bawah air

3.4.2 Perancangan dan Pembuatan Bodi Robot

Disini bodi robot dibuat sesederhana mungkin dengan maksud untuk memudahkan dalam pengendalian keseimbangan pada saat didalam air. Bodi robot terbuat dari pipa/paralon tunggal, ukuran diameternya adalah 4 dim (satuan diameter pipa) atau sekitar 4 inci. Bodi ini dibuat tidak terlalu kecil dan tidak terlalu besar dengan maksud agar memudahkan dalam peletakan komponen dan batere. Disamping itu jika bodi terlalu besar nantinya juga akan membutuhkan tenaga yang besar untuk dapat melakukan kondisi tenggelam. Jadi bodi robot dibuat dengan ukuran sedang disesuaikan dengan berat baling-baling dan batere.

Bodi robot yang sudah diintegrasikan dengan baling-baling bisa dilihat pada gambar 3.13 dibawah ini.



Gambar 3.13. Mekanik robot bawah air

Spesifikasi bodi robot :

➤ Pipa bodi

Diameter dalam = 120 mm

Diameter luar = 125 mm

Panjang = 600 mm

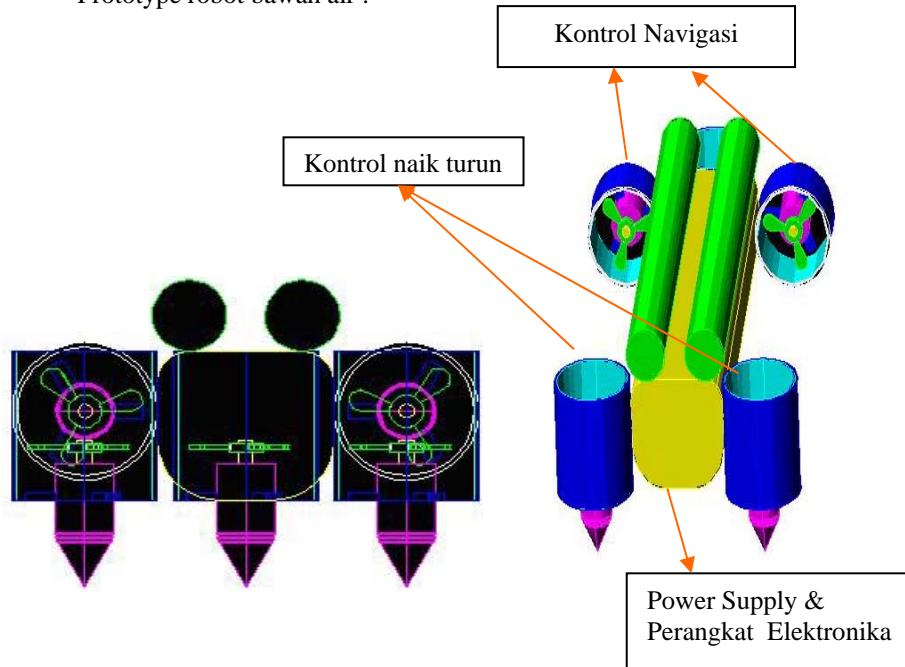
➤ Penutup bodi

Diameter dalam = 125 mm

Diameter luar = 130 mm

Panjang = 5 cm

Prototype robot bawah air :



Gambar 3.14. Prototype robot bawah air

*** Halaman ini sengaja dikosongkan ***

BAB IV

PENGUJIAN DAN ANALISA

Dalam Bab ini akan dibahas tentang pengujian berdasarkan perencanaan dari sistem yang dibuat. Pengujian ini dilaksanakan untuk mengetahui kehandalan dari sistem dan untuk mengetahui apakah sudah sesuai dengan perencanaan atau belum. Pengujian pertama-tama dilakukan secara terpisah, dan kemudian dilakukan ke dalam sistem yang telah terintegrasi.

Pengujian yang dilakukan pada bab ini antara lain:

1. Pengujian perangkat elektronik
2. Pengujian mekanik

Pengujian sistem dilakukan dua kali, pertama pengujian sebelum terjadi perubahan sistem dan yang kedua yaitu pengujian terhadap sistem yang sudah mengalami perubahan.

Berikut adalah pengujian sebelum terjadi perubahan pada sistem navigasi robot bawah air.

4.1 PENGUJIAN PERANGKAT ELEKTRONIK

Pada pengujian ini meliputi :

- 1) Pengujian minimum system
- 2) Pengujian driver
- 3) Pengujian tombol pushbutton

4.1.1 Pengujian Minimum Sistem

Pengujian minimum sistem AT89S51 dilakukan dengan menghubungkan masing-masing port mikrokontroler AT89S51 pada LED (*Light Emiting Diode*). Jika program telah didownloadkan pada mikrokontroler tersebut, maka masing-masing LED yang terhubung pada port mikrokontroler akan menyala secara bergantian. Dengan demikian rangkaian minimum sistem AT89S51 telah berfungsi dengan baik.

Tujuan :

1. Memastikan sistem minimum dapat berfungsi dengan baik
2. Memastikan *downloader serial* berfungsi dengan baik

Peralatan :

1. Rangkaian Sistem Minimum AT89S51
2. DC *power supply* +5V.
3. PC dilengkapi ISP Program v1.4.

Persiapan :

1. Memasang rangkaian pada downloader serial
2. Menulis program pengujian pada notepad kemudian dijadikan *file hex*.
3. Men-download program dengan menggunakan perangkat ISP Program.
4. Menjalankan program pada system minimum.

Listing program pengujian port input output pada AT89S51 dapat dilihat pada lampiran program *input output*.

Tabel 4.1. Hasil Pengujian Minimum Sistem

Detik ke	Mikrokontroler AT89s51							
	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
1	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	1	0
3	0	0	0	0	0	1	0	0
4	0	0	0	0	1	0	0	0
5	0	0	0	1	0	0	0	0
6	0	0	1	0	0	0	0	0
7	0	1	0	0	0	0	0	0
8	1	0	0	0	0	0	0	0

Analisa:

Setelah detik ke-8 nyala led kembali ke detik-1, untuk menguji port input output yang lain tinggal mengubah program *mov P1,A* menjadi (*P0,A* atau *P2,A* atau *P3,A*) dan hasil yang didapat sesuai dengan tabel diatas. Hal ini menunjukkan bahwa minimum sistem bias bekerja dengan baik

4.1.2 Pengujian Driver Motor

Pada proyek sistem navigasi robot bawah air ini proses pergerakannya dilakukan oleh motor. Jadi hal yang perlu dilakukan adalah menguji kehandalan *driver* motor. Disini digunakan *switching transistor* dan *driver relay*, hasil yang didapat seperti pada tabel 4.2 berikut :

Tabel 4.2. Hasil pengujian driver

Input PortB				Aksi
PortB.3	PortB.2	PortB.1	PortB.0	
H	L	L	H	Maju
L	H	H	L	Mundur
H	H	L	H	Kiri
H	L	H	H	Kanan
H	H	H	H	Diam

Catatan : L = Low H = High

Analisa:

Pada tabel hasil pengujian diatas, robot akan bergerak maju jika portB.3-PortB.0 mendapat logika H-L-L-H secara berturut-turut, akan bergerak mundur jika portB.3-PortB.0 mendapat logika L-H-H-L secara berturut-turut, akan bergerak ke kiri jika portB.3-PortB.0 mendapat logika H-H-L-H secara berturut-turut, akan bergerak ke kanan jika portB.3-PortB.0 mendapat logika H-L-H-H secara berturut-turut,dan robot akan berhenti/diam jika portB.3-PortB.0 mendapat logika H-H-H-H secara berturut-turut, hal ini dikarenakan transistor switching akan aktif jika mendapat logika *low* (aktif *low*). Berdasarkan tabel hasil pengujian diatas, dapat disimpulkan driver yang digunakan pada sistem navigasi robot bawah air telah bekerja dengan baik.

4.1.3 Pengujian Tombol Kontrol Arah dan Program Kontrol

Pengujian ini dimaksudkan untuk menyesuaikan perintah antara tombol kontrol arah dengan pergerakan navigasi robot. Dalam pengujian tombol kontrol arah terlebih dahulu harus diintegrasikan dengan minimum sistem dan driver. Untuk program

kontrolnya bisa dilihat pada lampiran *program kontrol sistem navigasi AT89S51*.

Tabel 4.3 adalah hasil pengujian terhadap tombol pushbutton dan program kontrol.

Tabel 4.3. Pengujian tombol Pushbutton dan program kontrol

No	InputPushbutton				Output				Ket.
	F	R	B	L	P0.7	P0.6	P0.5	P0.4	
1	1	0	0	0	L	H	L	H	Maju
2	0	1	0	0	L	H	H	L	Kanan
3	0	0	1	0	H	L	H	L	Mundur
4	0	0	0	1	H	L	H	L	Kiri
5	x	x	x	x	x	x	x	x	Stop

Analisa :

Dengan melihat tabel diatas, ternyata hasil penekanan tombol kontrol arah sudah sesuai dengan program yang telah dibuat, sehingga pergerakan navigasi robot sesuai dengan instruksi yang diberikan. Hal ini menunjukkan bahwa tombol kontrol arah, program kontrol serta driver bisa bekerja dengan baik.

4.2 PENGUJIAN MEKANIK

Pengujian ini meliputi:

- 1) Pengujian Baling-baling motor DC
- 2) Pengujian kerangka dan bodi robot

4.2.1 Pengujian Baling-baling Motor DC

Tujuan :

- 1) Menguji kekedapan pembungkus motor DC
- 2) Memastikan tidak adanya kebocoran pada bahan pembungkus sehingga baling-baling bias bekerja dengan baik

Perlengkapan :

- 1) Baling-baling motor DC
- 2) Power Supply +12V
- 3) Kolam air ukuran 80cm x 60cm x 120cm

Persiapan :

- 1) Nyalakan baling-baling dengan tegangan +12V
- 2) Masukkan baling-baling pada kolam yang sudah tersedia
- 3) Amati apakah baling-baling masih bias berputar pada saat berada didalam air

Tabel 4.4. Pengujian kededapan Baling-baling motor DC

Celah yang dimungkinkan bocor	Hasil	
	Bocor	Tidak Bocor
Sambungan penutup motor – poros motor		x
Lubang kabel polaritas motor		x
Lubang penutup belakang motor		x

x = hasil pengujian yang didapatkan

Analisa :

Pada saat baling-baling dimasukkan kedalam air, baling-baling masih bisa berputar. Hal ini menunjukkan bahwa tidak ada kebocoran pada pembungkus motor DC sehingga dapat disimpulkan bahwa baling-baling motor DC bias bekerja dengan baik.

4.2.2 Pengujian Kerangka dan Bodi Robot

Tujuan :

- 1) Menguji kededapan bahan penyusun bodi robot
- 2) Memastikan tidak adanya kebocoran pada bodi robot sehingga tidak merusak perngkat elektronika yang ada didalamnya
- 3) Menguji ketahanan bahan kerangka robot ketika dimasukkan pada kedalaman tertentu

Perlengkapan :

- 1) Body dan kerangka robot
- 2) Power supply +12V
- 3) Kolam ukuran 80cm x 60cm x 120cm

Persiapan :

- 1) Integrasikan bodi dengan baling-baling motor DC

- 2) Berikan tegangan sebesar +12V
- 3) Amati apakah robot bisa bekerja pada saat dimasukkan dalam air
- 4) Tenggelamkan robot pada kedalaman tertentu secara bertahap, apakah bodi robot masih bisa bertahan dan tidak ada kebocoran

Tabel 4.5. Hasil pengujian kededapan kerangka dan bodi robot

Celah yang dimungkinkan bocor	Hasil	
	Bocor	Tidak bocor
Sambungan penutup motor-poros motor		x
Lubang kabel polaritas motor		x
Lapisan pembungkus motor		x
Sambungan baling-baling motor – bodi robot		x
Penutup bodi robot		x
Lubang kabel pushbutton		x

x = hasil pengujian yang didapatkan

$$\begin{aligned}
 \% \text{ Kededapan bahan} &= \frac{(\text{Hasil tidak bocor} - \text{Hasil bocor})}{\text{Kemungkinan bocor}} \times 100\% \\
 &= \frac{(6 - 0)}{6} \times 100\% \\
 &= \frac{6}{6} \times 100\%
 \end{aligned}$$

$$\% \text{ Kededapan bahan} = 100\%$$

Analisa :

Berdasarkan hasil pengujian diatas, dapat disimpulkan bahwa bahan penyusun mekanik pada robot bawah air tidak ada celah yang bocor. Dengan demikian dapat disimpulkan bahwa kededapan mekanik kerangka dan bodi robot telah berhasil.

4.3 Pengujian Setelah Terjadi Perubahan Pada Sistem Navigasi Robot Bawah Air

Pengujian sistem navigasi setelah terjadi perubahan *hanya* dilakukan dengan cara mengaplikasikan kerja sistem pada robot *line tracer*. hal ini dilakukan dengan pertimbangan bahwa peletakan sensor posisi didalam air cukup mengkhawatirkan baik dari *segi keamanan* sensor maupun segi *kehandalan kerja sensor*.

Pengujian yang dilakukan setelah terjadi perubahan pada sistem navigasi robot bawah air adalah sebagai berikut:

4.3.1 Pengujian Pembacaan Modul CMPS03/Sensor Posisi

Tujuan :

- 1) Memastikan modul kompas CMPS03 bisa bekerja dengan baik
- 2) Memastikan kehandalan program untuk pembacaan sensor posisi

Peralatan :

- 1) Modul kompas CMPS03
- 2) Minimum sistem ATmega16
- 3) LCD 16x2
- 4) Supply tegangan +5v
- 5) Kabel penghubung

Persiapan :

- 1) Rangkai modul CMPS03 pada minimum sistem ATmega16 dengan cara menghubungkan pin scl dan sca pada kompas dengan pin pada ATmega16
- 2) Rangkai modul LCD 16x2 pada minimum sistem dengan cara menghubungkan pin-pin LCD dengan pin pada minimum sistem ATmega16
- 3) Berikan suply +5v pada minimum sistem, modul LCD, dan modul kompas CMPS03.
- 4) Pada minimum sistem ATmega 16, downloadlah pada *program pembacaan kompas* pada lampiran.
- 5) Amati apa yang terjadi pada LCD

Hasil Pengujian :

- Pada saat kompas menghadap utara



Gambar 4.1. Pembacaan nilai kompas arah utara

- Pada saat kompas diputar 90 derajat arah jarum jam/menghadap timur



Gambar 4.2. Pembacaan nilai kompas arah timur

- Pada saat kompas diputar 180 derajat arah jarum jam/menghadap selatan



Gambar 4.3. Pembacaan nilai kompas arah selatan

- Pada saat kompas diputar 270 derajat arah jarum jam/menghadap barat



Gambar 4.4. Pembacaan nilai kompas arah barat

Analisa :

Berdasarkan hasil pengujian diatas, terlihat bahwa pembacaan nilai kompas sudah berhasil. Hal ini menunjukkan bahwa modul kompas CMPS03, LCD 16x2, minimum sistem ATmega16, dan program pembacaan kompas pada sistem navigasi robot bawah air sudah berjalan dengan baik.

4.3.2 Pengujian Tombol Kontrol Arah dan Driver Motor DC

Tujuan :

- 1) Memastikan tombol kontrol arah bisa bekerja dengan baik
- 2) Memastikan driver bekerja dengan baik

Peralatan :

- 1) Rangkaian tombol kontrol arah
- 2) Rangkaian Driver Motor DC
- 3) Minimum sistem ATmega16
- 4) Supply +5v dan +12v

Persiapan :

- 1) Rangkai tombol kontrol arah pada minimum sistem ATmega16
- 2) Rangkai driver motor DC pada minimum sistem ATmega16
- 3) Berikan tegangan +5v pada minimum sistem ATmega16
- 4) Berikan tegangan +12v pada driver motor DC

Analisa :

Tombol kontrol arah dan driver motor DC yang digunakan pada sistem navigasi robot bawah air untuk sistem yang tidak mengalami perubahan dan yang mengalami perubahan adalah sama. Jadi dapat disimpulkan bahwa tombol kontrol arah dan driver motor DC yang dipakai bisa bekerja dengan baik.

4.3.3 Pengujian Integrasi Sistem Secara Keseluruhan

Setelah berhasil melakukan pengujian sistem secara perbagian, maka hal yang harus dilakukan adalah mengintegrasikan sistem tersebut.

Tujuan :

- 1) Melakukan integrasi antar sistem agar bisa digunakan untuk mengontrol navigasi pada robot bawah air (line tracer)
- 2) Membandingkan antara nilai kompas magnetic dengan analog

Peralatan :

- 1) Dua buah minimum system ATMega16
- 2) Modul kompas CMPS03, modul kompas analog
- 3) Modul LCD 16x2
- 4) Rangkaian tombol kontrol arah
- 5) Rangkaian driver motor DC
- 6) Supply +5v dan +12v

Persiapan :

- 1) Rangkai tombol kontrol arah pada mikro1
- 2) Rangkai modul kompas CMPS03, LCD 16x2, driver motor DC pada mikro2
- 3) Berikan tegangan +5v pada mikro1, mikro2, LCD 16x2, modul kompas CMPS03. Untuk driver motor DC berikan tegangan +12v
- 4) Hubungkan pin serial (Tx, Rx) mikro1 dengan mikro2 dengan cara bersilangan.
- 5) Pada mikro 1, downloadlah program dengan nama *program kontrol arah mikro1* pada lampiran.
- 6) Pada mikro 2, downloadlah program dengan nama *program kontrol sensorposisi mikro2* pada lampiran.
- 7) Amati apa yang terjadi pada robot bawah air/line tracer beserta tampilan LCDnya

Cara kerja program :

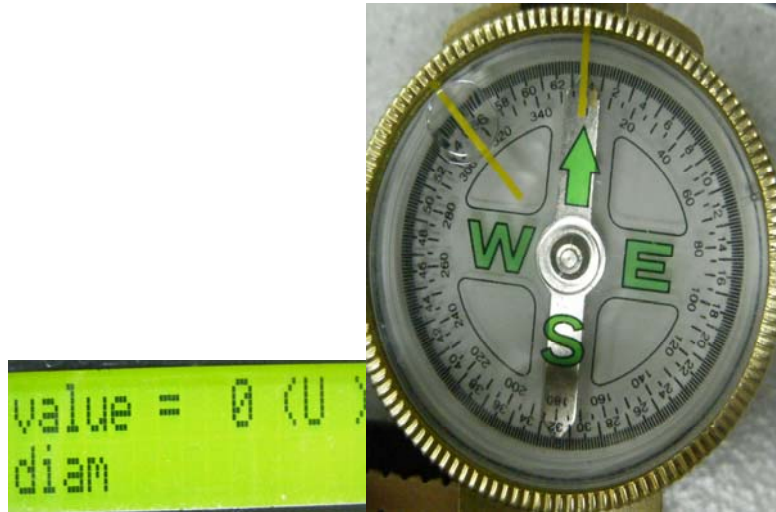
- Pada saat program telah selesai didownload, LCD langsung menampilkan pembacaan kompas beserta status (diam atau bergerak) dan juga arah yang ditunjukkan kompas.
- Ketika ada penekanan tombol kontrol arah, line tracer akan bergerak sesuai dengan tombol yang ditekan apakah maju, mundur, kanan, atau kiri
- Ketika ada penekanan otomatis nilai kompas akan berubah, perubahan nilai kompas ini akan terus diupdate arah beserta status pergerakan robot untuk ditampilkan ke LCD.

Catatan:

Kompas digital bernilai 0 sampai 255, jadi untuk mencapai satu putaran penuh (360 derajat) maka tiap nilai kompas magnetic yang keluar/ditampilkan pada LCD harus dikalikan dengan 1,4.

Kalibrasi :

Pada saat kompas dikalibrasi menghadap utara/sudut nol derajat



Gambar 4.5. Kalibrasi kompas

Untuk mengetahui besarnya error pembacaan pada kompas magnetik, diperlukan adanya kalibrator sebagai acuan kebenarannya. Pada pengujian ini digunakan kompas analog sebagai kalibratornya. Perhitungan nilai error pembacaan kompas magnetik bisa dihitung dengan rumus dalam kotak dibawah ini:

$$\% \text{ error} = \frac{|\text{nilai kompas digital} * 1,4 - \text{nilai derajat mata angin}|}{\text{nilai derajat mata angin}} \times 100\%$$

Hasil Pengujian :

- Pada saat kompas menghadap arah timur laut/ sudut 45 derajat



Gambar 4.6. Kompas arah timur laut

$$\begin{aligned} \% \text{ error} &= \frac{|34 \cdot 1,4 - 45|}{45} \times 100\% \\ &= \frac{2,6}{45} \times 100\% \\ &= 5,7\% \end{aligned}$$

- Pada saat kompas menghadap arah timur / sudut 90 derajat



Gambar 4.7. Kompas arah timur

$$\begin{aligned} \% \text{ error} &= \frac{|63 \cdot 1,4 - 90|}{90} \times 100\% \\ &= \frac{1,8}{90} \times 100\% \\ &= 2\% \end{aligned}$$

- Pada saat kompas menghadap arah tenggara / sudut 135 derajat



Gambar 4.8. Kompas arah tenggara

$$\begin{aligned} \% \text{ error} &= \frac{|85 \cdot 1,4 - 135|}{135} \times 100\% \\ &= \frac{16}{135} \times 100\% \\ &= 11,8\% \end{aligned}$$

- Pada saat kompas menghadap arah selatan/ sudut 180 derajat



Gambar 4.9. Kompas arah selatan

$$\begin{aligned} \% \text{ error} &= \frac{|117 \cdot 1,4 - 180|}{180} \times 100\% \\ &= \frac{16,2}{180} \times 100\% \\ &= 9\% \end{aligned}$$

- Pada saat kompas menghadap arah barat daya/ sudut 225 derajat



Gambar 4.10. Kompas arah barat daya

$$\begin{aligned} \% \text{ error} &= \frac{|147 \cdot 1,4 - 225|}{225} \times 100\% \\ &= \frac{19,2}{225} \times 100\% \\ &= 8,53\% \end{aligned}$$

- Pada saat kompas menghadap arah barat/ sudut 270 derajat



Gambar 4.11. Kompas arah barat

$$\begin{aligned} \% \text{ error} &= \frac{|170 \cdot 1,4 - 270|}{270} \times 100\% \\ &= \frac{32}{270} \times 100\% \\ &= 11,85\% \end{aligned}$$

- Pada saat kompas menghadap arah barat laut/ sudut 315 derajat



Gambar 4.12. Kompas arah barat laut

$$\begin{aligned} \% \text{ error} &= \frac{|202 \cdot 1,4 - 315|}{315} \times 100\% \\ &= \frac{32,2}{315} \times 100\% \\ &= 10,2\% \end{aligned}$$

- Pada saat kompas menghadap arah utara/ sudut 360 derajat



Gambar 4.13. Kompas arah utara

$$\begin{aligned} \% \text{ error} &= \frac{|245 \cdot 1,4 - 360|}{360} \times 100\% \\ &= \frac{17}{360} \times 100\% \\ &= 6,53\% \end{aligned}$$

Dari data pengujian diatas, dapat dibuat tabel hasil pengujian seperti pada tabel 4.6 dibawah ini.

Tabel 4.6. Hasil pengujian pembacaan status kompas magnetik

No.	Arah mata angin	Pembacaan sudut kompas		% Error
		CMPS03	Analog	
1.	Utara	343	0 atau 360	6,53
2.	Timur Laut	47,6	45	5,7
3.	Timur	88,2	90	2
4.	Tenggara	119	135	11,8
5.	Selatan	163,8	180	9
6.	Barat Daya	205,8	225	8,53
7.	Barat	238	270	11,85
8.	Barat Laut	282,8	315	10,2

Analisa :

Berdasarkan hasil pengujian yang didapatkan, untuk pembacaan antara kompas magnetic dengan kompas analog masih terdapat sedikit perbedaan tetapi masih dalam batas kewajaran, hal ini bisa disebabkan oleh kesalahan pengujian dalam melakukan pembacaan kompas analog. Untuk kerja dari sistem dapat dikatakan bahwa sistem komunikasi serial antar mikro bisa berjalan dengan baik, program pada mikro1 dan mikro2 bisa diaplikasikan pada robot bawah air, gerakan robot beserta status pembacaan pada LCD sudah sesuai dengan instruksi input tombol kontrol arah yang diberikan. Dengan kata lain sistem yang telah dibuat pada proyek akhir ini bisa berjalan dengan lancar dan sesuai dengan yang diinginkan.

BAB V

PENUTUP

5.1 Kesimpulan

Setelah melakukan perencanaan dan pembuatan sistem kemudian dilakukan pengujian dan analisisnya, maka dapat diambil beberapa kesimpulan tentang kerja dari sistem navigasi pada robot bawah air antara lain :

1. Dalam proyek akhir ini terjadi perubahan sistem. Sebelum mengalami perubahan, dalam melakukan pergerakan/navigasi dalam air robot hanya mengandalkan aktuator. Setelah terjadi perubahan sistem, pengujian dilakukan pada robot line tracer. Dalam melakukan pergerakan robot sudah dilengkapi dengan sensor posisi (kompas magnetik) sehingga pergerakan robot lebih mudah dikendalikan.
2. Kelemahan dari sistem yang dibuat antara lain :
 - a. Penentuan letak komponen dan pemberat masih belum ideal, sehingga robot kesulitan untuk mencapai kondisi seimbang.
 - b. Pembacaan status kompas magnetik masih ada sedikit perbedaan bila dibandingkan dengan kalibrator kompas analog.
3. Kelebihan dari sistem yang dibuat antara lain :
 - a. Data input yang dikirim melalui kabel lebih cepat direspon daripada menggunakan wireless
 - b. Dengan menggunakan bahan paralon yang dilapisi shell, membuat sistem kedap air cukup terjamin
 - c. Penggunaan baling-baling motor DC lebih efisien daripada menggunakan tuas kemudi.
 - d. Dengan adanya kompas yang nilainya selalu diupdate memudahkan pengguna dalam melakukan kontrol pada sistem navigasi robot bawah air.

5.2 Saran

Untuk mendapatkan performa yang lebih baik dari sistem navigasi robot bawah air ini, dapat digunakan motor DC dengan torsi yang lebih besar sehingga robot bisa bermanuver lebih cepat. Desain baling-baling motor DC untuk manuver bisa diletakkan ditengah bodi robot, sehingga kestabilan lebih mudah terjaga. Pada saat pengujian selain kondisi arus air, volume kolam juga perlu diperhitungkan agar tidak mengganggu kestabilan robot.

Semoga apa yang telah disampaikan dalam buku ini dapat berguna untuk para pembaca sekalian, terutama dalam pembuatan robot bawah air yang menggunakan metode baling-baling. Segala saran, kritik dan masukan yang bersifat membangun sangat diharapkan untuk kesempurnaan proyek ini nantinya.

DAFTAR PUSTAKA

- [1] Associate Professor Gerald Seet. (2007) : *Underwater Imaging. URV and applications. Multi-robot coordination.* Nanyang Technology University.
- [2] G. Dudek, P. Giguere, and J. Sattar, (2006) : *Sensor-Based Behavior Control for an Autonomous Underwater Vehicle.* Experimental Robotics, Springer-Verlag,.
- [3] Hermawan. (2007) : *Pemanfaatan Remote Operated Vehicle Untuk Penelitian Laut Dalam.* html
- [4] Oktavianto, Hary. (2004) : *Komunikasi Serial.* hary@eepis-its.edu. Indonesia.
- [5] Pratama, A. Dan Dwika, Indra. (2008) : *Pengembangan Sistem Kemudi (steering) dan Sistem Ballast Pada Robot Bawah Air.* EEPIS, Indonesia.
- [6] Robinson, H. and Keary, A. (2000) : *Remote Control of Unmanned Undersea Vehicle.* International Unmanned Undersea Vehicle Symposium.
- [7] Salman, Abu. (2007) : *ROV : Robot Pintar di Bawah Laut.* Jejak Hydrographer.
- [8] Tim Workshop KRI, KRCI (2007) : *CMPS Modul Magnetik Kompas.* EEPIS, Indonesia.
- [9] www.alldatasheet.com Datasheet AT89S51 dan ATMega16.

Program input output.

```
                PROGRAM LED BERJALAN DARI P1.0 S/D P1.7 KEMUDIAN
                KEMBALI LAGI
org             0h
mov             A,0feh ; data led nyala dikirim melalui
akumulator
                ; agar P1.0 pertama kali menyala
mulai:
mov            P1,A   ; kirimkan data di akumulator ke port
1
acall         delay ; lakukan penundaan sesaat
rl            A     ; rotasikan isi akumulator ke kiri
sjmp         mulai ; ulangi lagi dari mulai
delay:
mov           R0,#0  ; isi register r1 dg 0 (256x ulang)
delay1:
mov           R1,#0  ; isi register r1 dg 0 (256x ulang)
djnz         R1,$
djnz         R0,delay1
ret
end
```

Program pembacaan kompas.

```
unsigned char compas_readb(unsigned char addr)
{
    unsigned char posisi;
    i2c_start();
    i2c_write(0xC0);
    i2c_write(addr);
    i2c_start();
    i2c_write(0xC1);
    posisi=i2c_read(0);
    i2c_stop();
    return posisi;
}
while (1)
{
    data=compas_readb(1);
    sprintf(kata,"value = %3d ",data);
    lcd_gotoxy(0,1);
    lcd_puts(kata);
    delay_ms(500);
}
```


Program kontrol arah mikro1

```
#include <mega16.h>
#include <stdio.h>
#include <delay.h>

unsigned char kata[16];
unsigned char terima[5];
unsigned int data;

#asm
    equ __lcd_port=0x15 ;PORTC
while (1)
    {

        start:
            if (PINA.0==0)
            {
                printf("B");
                delay_ms(100);
                goto start;
            }
            if (PINA.1==0)
            {
                printf("D");
                delay_ms(100);
                goto start;
            }
            if (PINA.2==0)
            {
                printf("A");
                delay_ms(100);
                goto start;
            }
            if (PINA.3==0)
            {
                printf("C");
                delay_ms(100);
                goto start;
            }
            else
            {
                printf("E");
                delay_ms(100);
                goto start;
            }
        };
    }
}
```

Program kontrol sensor posisi mikro2

```
#include <mega16.h>
#include <stdio.h>
#include <delay.h>
#include <lcd.h>

unsigned char kata[17];
unsigned char status[17];
unsigned int data;
unsigned char terima;

// I2C Bus functions
#asm
.equ __lcd_port=0x15 ;PORTC
.equ __i2c_port=0x1B ;PORTA
.equ __sda_bit=3
.equ __scl_bit=4

// Declare your global variables here

unsigned char compas_readb(unsigned char addr)
{
    unsigned char posisi;
    i2c_start();
    i2c_write(0xC0);
    i2c_write(addr);
    i2c_start();
    i2c_write(0xC1);
    posisi=i2c_read(0);
    i2c_stop();
    return posisi;
}

while (1)
{

next:
    PORTB=0x00;
    data=compas_readb(1);

    if((data>=240)&&(data<=255))
    {
        sprintf(kata,"value =%3d (U )",data);
        //printf("%d\n",data);
        lcd_gotoxy(0,0);
        lcd_puts(kata);
        //delay_ms(100);
        goto next2;
    }
}
```

```

}
if((data>=0)&&(data<=15))
{
sprintf(kata,"value =%3d (U )",data);
//printf("%d\n",data);
lcd_gotoxy(0,0);
lcd_puts(kata);
//delay_ms(100);
goto next2;
}
if((data>=16)&&(data<=47))
{
sprintf(kata,"value =%3d (TL)",data);
//printf("%d\n",data);
lcd_gotoxy(0,0);
lcd_puts(kata);
//delay_ms(100);
goto next2;
}
if((data>=48)&&(data<=79))
{
sprintf(kata,"value =%3d (T )",data);
//printf("%d\n",data);
lcd_gotoxy(0,0);
lcd_puts(kata);
//delay_ms(100);
goto next2;
}
if((data>=80)&&(data<=101))
{
sprintf(kata,"value =%3d (TG)",data);
//printf("%d\n",data);
lcd_gotoxy(0,0);
lcd_puts(kata);
//delay_ms(100);
goto next2;
}
if((data>=102)&&(data<=143))
{
sprintf(kata,"value =%3d (S )",data);
//printf("%d\n",data);
lcd_gotoxy(0,0);
lcd_puts(kata);
//delay_ms(100);
goto next2;
}
if((data>=144)&&(data<=175))
{
sprintf(kata,"value =%3d (BD)",data);
//printf("%d\n",data);

```

```

    lcd_gotoxy(0,0);
    lcd_puts(kata);
    //delay_ms(100);
    goto next2;
}
if((data>=176)&&(data<=207))
{
    sprintf(kata,"value =%3d (B )",data);
    //printf("%d\n",data);
    lcd_gotoxy(0,0);
    lcd_puts(kata);
    //delay_ms(100);
    goto next2;
}
if((data>=208)&&(data<=239))
{
    sprintf(kata,"value =%3d (BL)",data);
    //printf("%d\n",data);
    lcd_gotoxy(0,0);
    lcd_puts(kata);
    //delay_ms(100);
    goto next2;
}

next2:

terima=getchar();

if(terima=='A')
{
    PORTB=0B00000110;
    delay_ms(100);
    sprintf(status,"maju ");
    lcd_gotoxy(0,1);
    lcd_puts(status);
    goto next;
}
if(terima=='B')
{
    PORTB=0B00001001;
    delay_ms(100);
    sprintf(status,"mundur");
    lcd_gotoxy(0,1);
    lcd_puts(status);

    goto next;
}
if(terima=='C')
{
    PORTB=0B00000010;

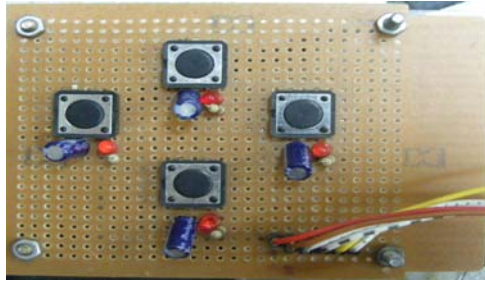
```

```
        delay_ms(100);
        sprintf(status,"kiri ");
        lcd_gotoxy(0,1);
        lcd_puts(status);

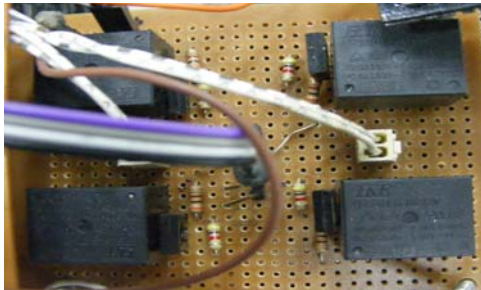
        goto next;
    }
    if(terima=='D')
    {
        PORTB=0B00000100;
        delay_ms(100);
        sprintf(status,"kanan ");
        lcd_gotoxy(0,1);
        lcd_puts(status);
        //delay_ms(100);
        goto next;
    }

    else
    {
        PORTB=0B00000000;
        sprintf(status,"diam ");
        lcd_gotoxy(0,1);
        lcd_puts(status);
        //delay_ms(100);
        goto next;
    }
};
}
```

Lampiran Gambar



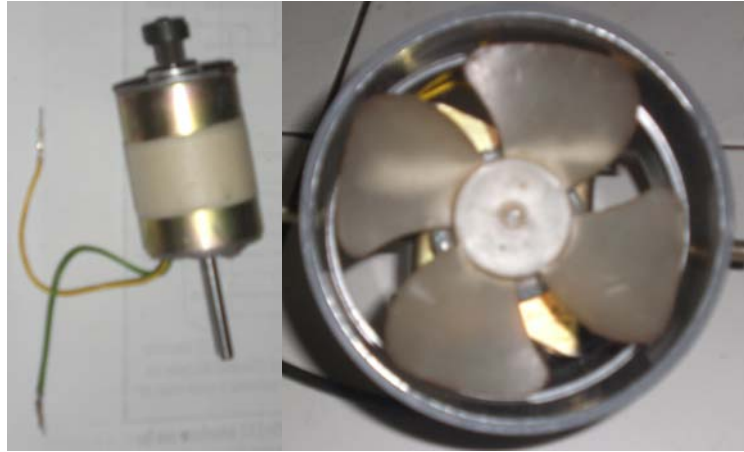
Tombol kontrol arah



Driver relay motor DC



LCD karakter 16x2



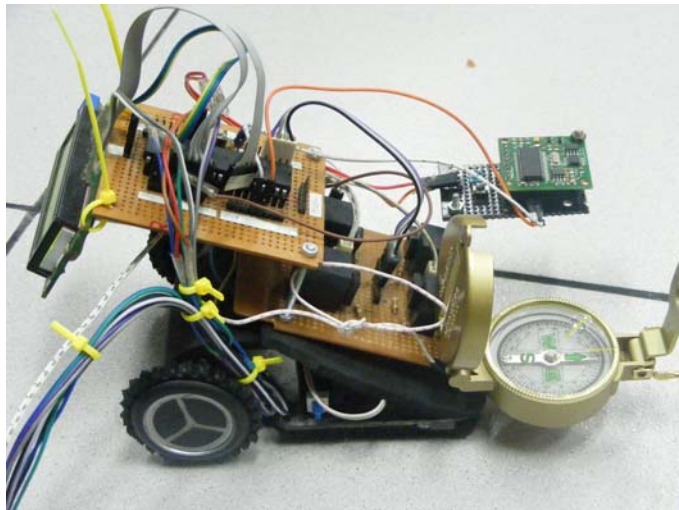
Baling-baling motor DC



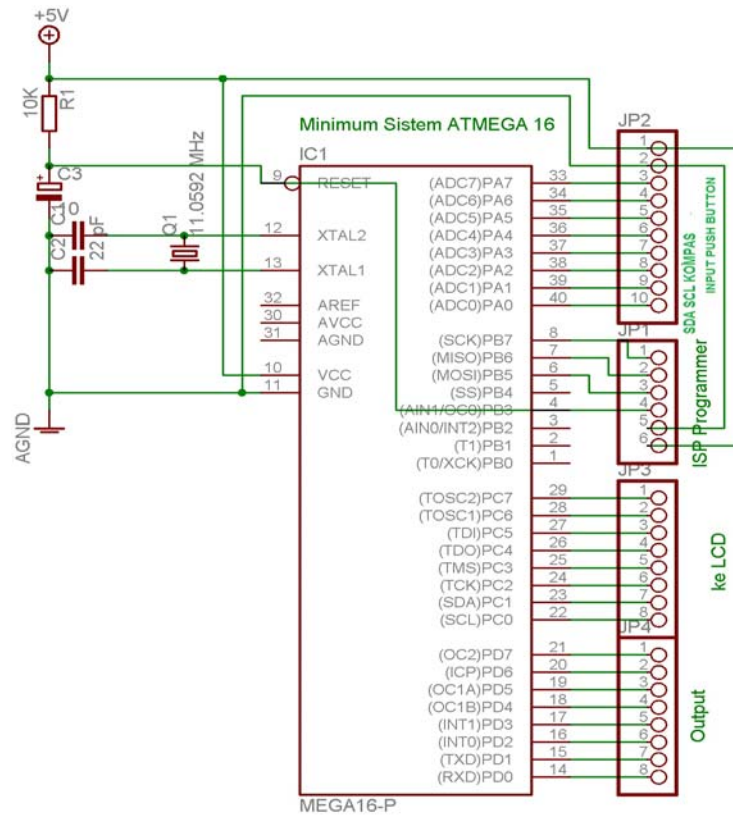
Mekanik robot bawah air



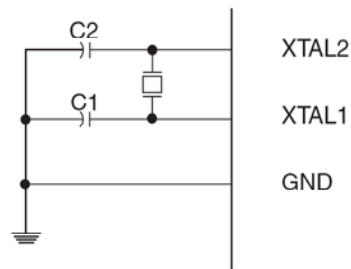
Kompas Analog



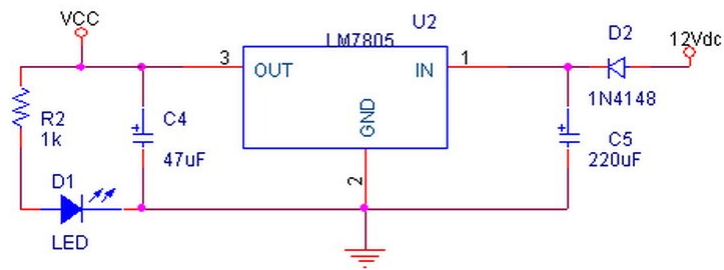
Integrasi sntem navigasi dengan line tracer



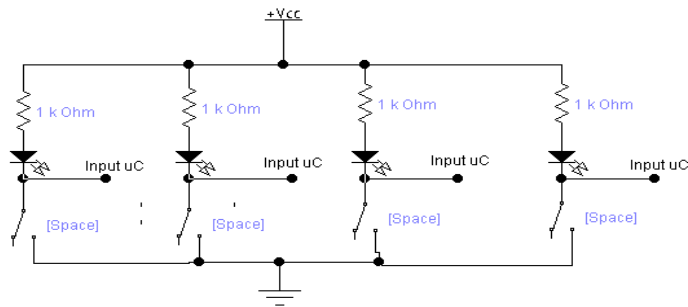
Minimum sistem ATMEGA16



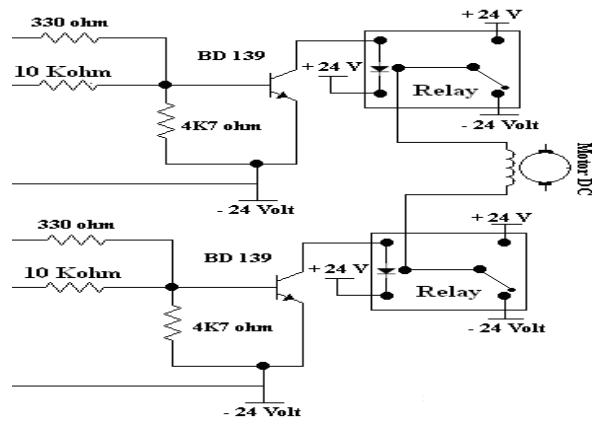
Clok Generator



Rangkaian power supply untuk mikrokontroler



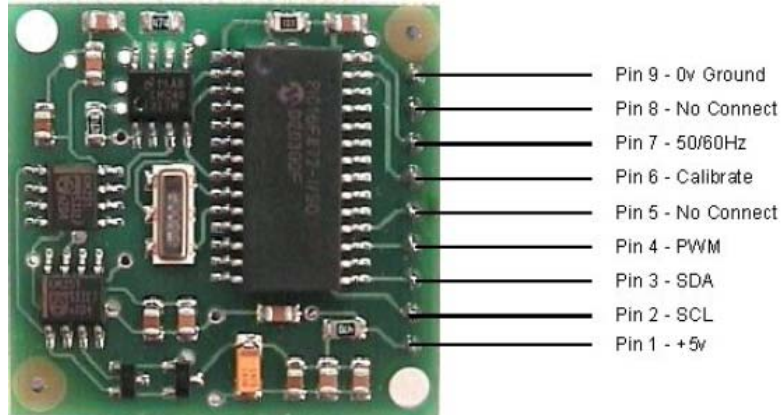
Rangkaian tombol kontrol arah



Rangkaian Switching Transistor Dan Driver Relay

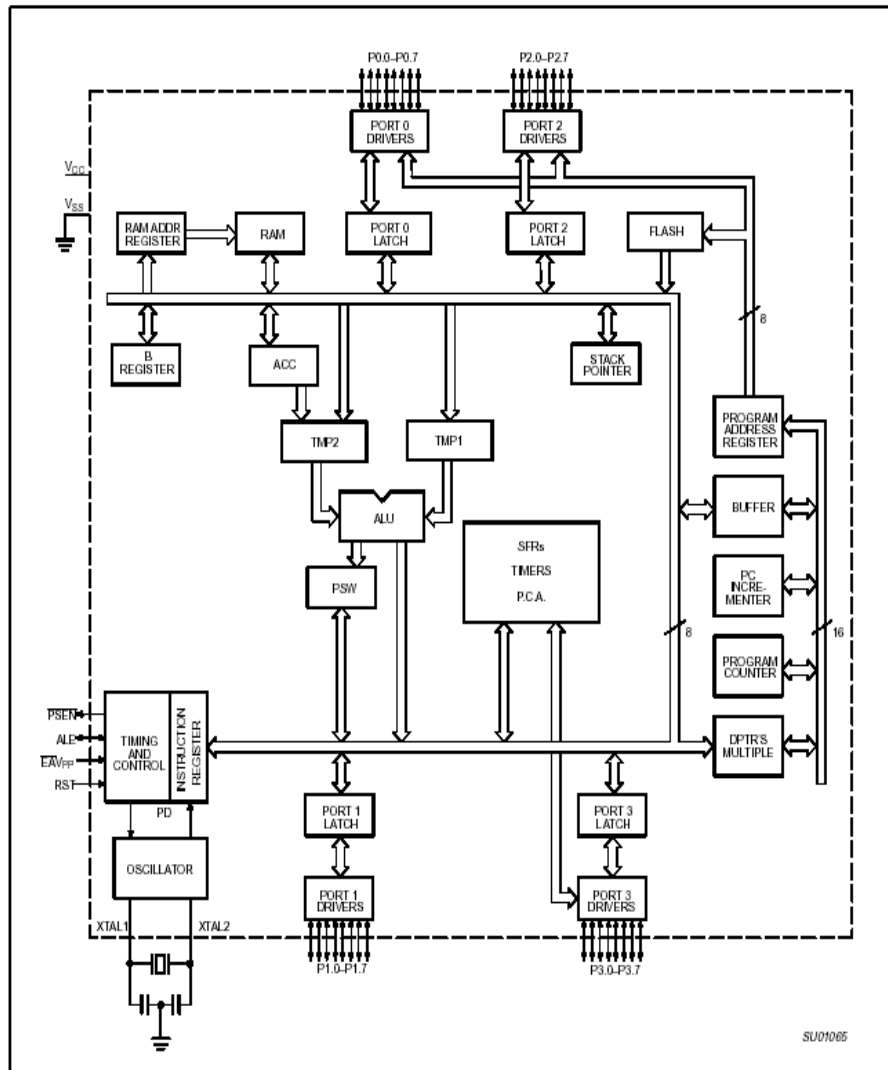
Lampiran Datasheet

Datasheet register CMPS03



Register	Function
0	Software Revision Number
1	Compass Bearing as a byte, i.e. 0-255 for a full circle
2,3	Compass Bearing as a word, i.e. 0-3599 for a full circle, representing 0-359.9 degrees.
4,5	Internal Test - Sensor1 difference signal - 16 bit signed word
6,7	Internal Test - Sensor2 difference signal - 16 bit signed word
8,9	Internal Test - Calibration value 1 - 16 bit signed word
10,11	Internal Test - Calibration value 2 - 16 bit signed word
12	Unused - Read as Zero
13	Unused - Read as Zero
14	Unused - Read as Undefined
15	Calibrate Command - Write 255 to perform calibration step. See text.

Datasheet AT89S51

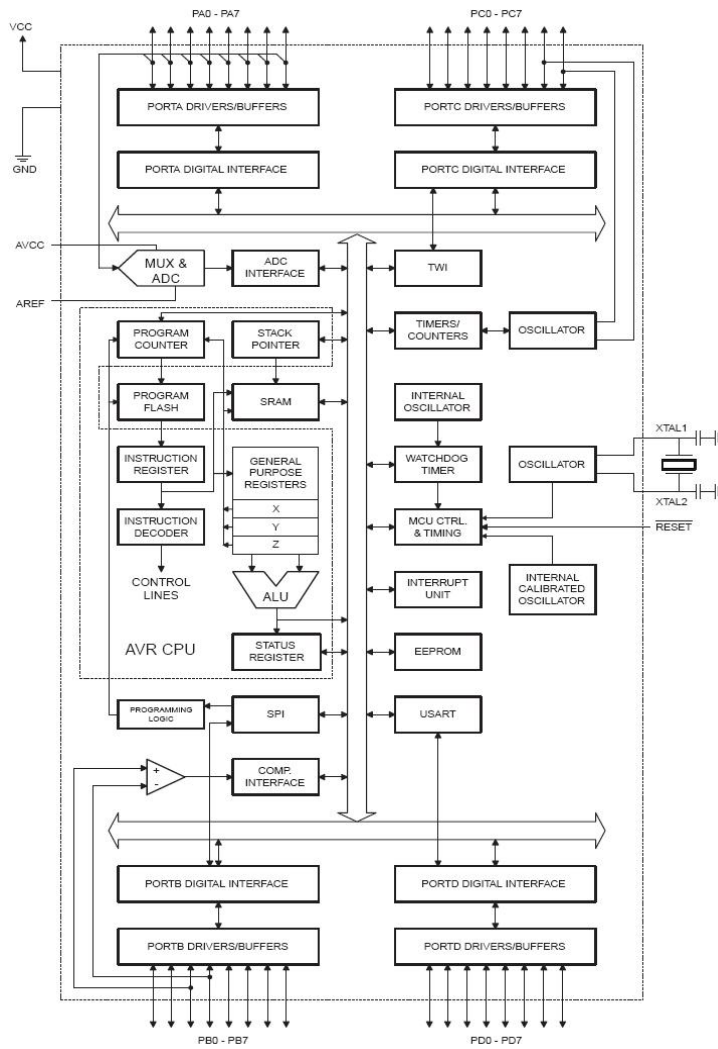


Arsitektur internal AT89S51

Arstitektur AT89S51 terdiri dari :

- Delapan bit CPU dengan register A (accumulator) dan register B
- 16 bit program counter (PC) dan 16 bit data pointer (DPTR)
- 8 bit program status word (PSW)
- 8 bit stack pointer (SP)
- Internal RAM sebanyak 128 byte terdiri dari
 - 4 bank register, dengan setiap bank terdiri dari 8 register
 - 16 byte memori yang dapat dialamati perbit
 - 8 byte data memori untuk keperluan umum
- 32 pin input/output yang tiap-tiap port terdiri dari 8 bit, port P0-P3
- 16 bit timer/counter; T0 dan T1
- Full duplex serial data receiver/transmitter menggunakan register SBUF
- Sebanyak 5 register control yaitu TCON, TMOD, SCON, PCON, IP dan IE
- Dua eksternal interrupt dan tiga sumber interrupt internal
- Osilator dan rangkaian clock .

Datasheet AT Mega16



Arsitektur CPU ATmega 16

Keistimewaan AVR ATmega16

1. Advanced RISC Architecture

- 131 Powerful Instructions – Most Single Clock Cycle Execution
- 32 x 8 General Purpose Working Registers
- Fully Static Operation
- Up to 16 MIPS Throughput at 16 MHz
- On-chip 2-cycle Multiplier

2. Nonvolatile Program and Data Memories

- 32K Bytes of In-System Self-Programmable Flash
 - Endurance: 10,000 Write/Erase Cycles
- Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
- 1024 Bytes EEPROM
 - Endurance: 100,000 Write/Erase Cycles
- 2K Bytes Internal SRAM
- Programming Lock for Software Security

3. Peripheral Features

- Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
- One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
- Real Time Counter with Separate Oscillator
- Four PWM Channels
- 8-channel, 10-bit ADC
 - 8 Single-ended Channels
 - 7 Differential Channels for TQFP Package Only
 - 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x
- Byte-oriented Two-wire Serial Interface
- Programmable Serial USART
- Master/Slave SPI Serial Interface
- Programmable Watchdog Timer with Separate On-chip Oscillator
- On-chip Analog Comparator

4. Special Microcontroller Features

- Power-on Reset and Programmable Brown-out Detection

- Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
5. I/O and Packages
- 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, 44-lead PLCC, and 44-pad MLF
6. Operating Voltages
- 2.7 – 5.5V for ATmega16L
 - 4.5 - 5.5V for ATmega16
7. Speed Grades
- 0 - 8 MHz for ATmega16L
 - 0 - 16 MHz for ATmega16

Datasheet LCD karakter 16x2

PIN	SIGNAL	LEVEL	DIRECTION	DESCRIPTION
1	V _{SS}	0v		Ground
2	V _{DD}	+5.0v		Supply voltage for logic
3	V _O	variable		Supply voltage for driving LCD is V _O = +1v typical at V _{DD} = +5v which gives a V _{LCD} = (V _{DD} - V _O) = +4v
4	RS	H/L	I	Register selection input. H: Data register (for read and write) L: Instruction code (for write)
5	R \bar{W}	H/L	I	H: Read (MPU←Module) L: Write (MPU→Module)
6	E	H,H→L	I	Read/write enable signal. H: Read data is enabled by a high level. H→L: Write data is latched on the falling edge.
7	DB0	H/L	I/O	Data bit 0
8	DB1	H/L	I/O	Data bit 1
9	DB2	H/L	I/O	Data bit 2
10	DB3	H/L	I/O	Data bit 3
11	DB4	H/L	I/O	Data bit 4
12	DB5	H/L	I/O	Data bit 5
13	DB6	H/L	I/O	Data bit 6
14	DB7	H/L	I/O	Data bit 7
15	NC			No Connection
16	NC			No Connection

Halaman ini sengaja dikosongkan

A-1 BIODATA PENULIS



Nama : M. Nurul Fauzi
Tempat/Tanggal Lahir : Gresik / 07 Maret 1987
Alamat : Jl. Tanjungan No. 167 RT/05 RW/01
Driyorejo - Gresik
Telepon/Hp : (031) 7-526280
Hobi : Olahraga, mancing plus dengerin muzik
Motto : Think before u ing....!!!

Riwayat Pendidikan :

- MI Tarbiyatul Islamiyah Driyorejo Tahun 1993 – 1999
- SMP YPM 5 Driyorejo Tahun 1999 – 2002
- SMA Negeri 1 Driyorejo Tahun 2002 – 2005
- Politeknik Elektronika Negeri Surabaya-ITS Tahun 2006 – 2009

Penulis merupakan anak terakhir dari empat bersaudara. Penulis telah mengikuti seminar Proyek Akhir pada tanggal 21 Juli 2009, sebagai salah satu persyaratan untuk memperoleh gelar Ahli Madya (A.Md).