

PERANCANGAN PENGENDALI LOGIKA FUZZY PADA MODEL ROBOT
PENGANGKUT BARANG MENGGUNAKAN MIKROKONTROLER AT89S52

SKRIPSI

*Diajukan untuk memenuhi persyaratan
memperoleh gelar sarjana*

UNIVERSITAS BRAWIJAYA



Disusun Oleh :

Rahmad Kuspinuji Budiantoro

0001063350-63

DEPARTEMEN PENDIDIKAN NASIONAL

UNIVERSITAS BRAWIJAYA

FAKULTAS TEKNIK

MALANG

2006

PERANCANGAN PENGENDALI LOGIKA FUZZY PADA MODEL ROBOT
PENGANGKUT BARANG BERBASIS MIKROKONTROLER AT89S52

SKRIPSI

*Diajukan untuk memenuhi persyaratan
memperoleh gelar sarjana*



Disusun Oleh :

RAHMAD KUSPINUJI BUDIANTORO

000106335063

Mengetahui dan Menyetujui :

PEMBIMBING I

PEMBIMBING II

M. Rif'an, ST. MT
NIP. 132 283 659

Dipl. Ing. Ir. Moch. Rusli
NIP. 131 653 473

KATA PENGANTAR

Puji syukur kepada Allah SWT yang telah memberikan segala nikmat dan rahmat-Nya sehingga penyusunan skripsi ini dapat diselesaikan. Karena hanya dengan pertolongan-Nya semata penulis mampu melewati segala kendala yang ada selama penyusunan skripsi ini.

Skripsi dengan judul **"Perancangan Pengendali Logika Fuzzy Pada Model Robot Pengangkut Barang Menggunakan AT89S52"** ini disusun sebagai salah satu syarat untuk mendapatkan gelar Sarjana Teknik di Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya.

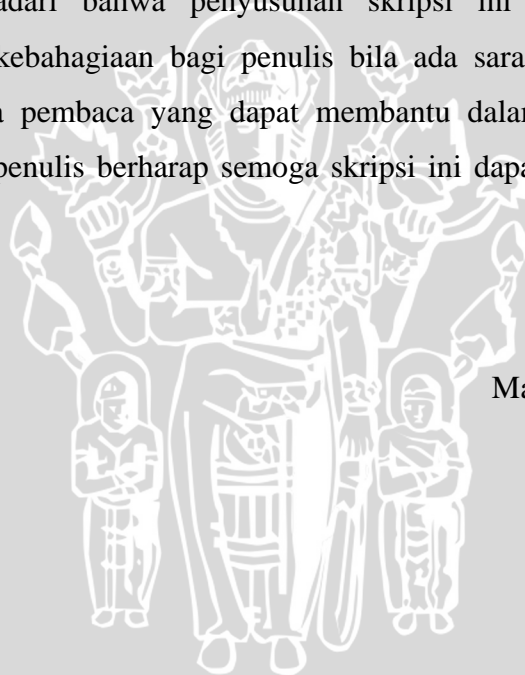
Dalam penyusunan skripsi ini, tidak lepas dari bantuan, bimbingan, dorongan dan motivasi dari berbagai pihak. Oleh karena itu dengan segala kerendahan hati penulis menyampaikan terima kasih kepada:

- ❏ Abi dan Umi tercinta yang selalu memberikan segala perhatian dan doa tulusnya.
- ❏ Ir. Purwanto, MT selaku Ketua Jurusan Teknik Elektro yang telah mengarahkan dan menentukan dosen pembimbing sehingga tercapai kesuksesan ini.
- ❏ Ir. Hery Pornomo selaku Sekretaris Jurusan Teknik Elektro.
- ❏ Moch. Rif'an, ST. MT dan Dipl. Ing. Ir. M. Rusli selaku dosen pembimbing yang telah banyak memberikan bimbingan, petunjuk, maupun saran yang sangat membantu penulis dalam menyelesaikan skripsi ini.
- ❏ Bapak dan Ibu dosen beserta staf dan karyawan Jurusan Teknik Elektro yang baik secara langsung ataupun tidak langsung telah membantu menyelesaikan skripsi ini.
- ❏ Adik-adikku Agrei, Yusuf Bahtiar dan keluarga besar Eyang Purnomo yang telah memberikan perhatian, dukungan materi dan do'anya.
- ❏ Pak Imam, Bulek Arifah dan yang aku sayangi Rahma P. Azizah, Rahma K. Hanifah, Nuya, Dinar, Dhuha, Mas Zainul, Mbak Yanti sekeluarga yang telah banyak membantu, memberi masukan dan dukungan.

- To *A.W.S* (*Git my soul...*) terimakasih untuk semua masa indah yang telah kau berikan dan itu semua sangat berarti dalam penyelesaian skripsi ini, semoga semua ini tak akan berakhir hingga kita mencapai akhir impian kita berdua yang indah.
- Bpk.Dahlan, Ibu Machmudah dan Nugroho "terima kasih atas pinjeman komputernya" dan membantu dalam memberikan fasilitas tempat tinggal.
- Rekan-rekan Jurusan Teknik Elektro Universitas Brawijaya angkatan 2000 dan 2001, "anak-anakku" di kost putri Veteran Dalam 11A, terima kasih atas kebersamaannya selama ini.

Semoga semua bantuan do'a, bimbingan, dukungan, petunjuk maupun nasehat yang telah diberikan kepada penulis mendapat balasan dari Allah SWT.

Penulis menyadari bahwa penyusunan skripsi ini masih jauh dari sempurna, untuk itu kebahagiaan bagi penulis bila ada saran dan kritik yang membangun dari para pembaca yang dapat membantu dalam penyempurnaan skripsi ini. Akhirnya penulis berharap semoga skripsi ini dapat bermanfaat bagi para pembaca.



Malang, Juli 2006

Penulis

UNIVERSITAS BRAWIJAYA



DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
ABSTRAK	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
BAB I : PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan	3
1.5 Sistematika Penulisan	3
BAB II : ROBOT PENGANGKUT BARANG	4
2.1 Pengenalan Robotika	4
2.1.1 Mobil Robot	4
2.1.1.1 <i>Operator Oriented</i>	5
2.1.1.2 <i>Self Running</i>	5
2.2 Pengertian Robot Pengangkut Barang	5
2.2.1 Elemen Pengolahan Robot Pengangkut Barang	6
BAB III : TINJAUAN PUSTAKA	7
3.1 Sensor Cahaya	7
3.1.1 <i>Light Emiting Diode (LED) Infra Merah</i>	7
3.1.2 Phototransistor	8
3.2 <i>Optocoupler</i>	8
3.2.1 Prinsip Kerja <i>Optocoupler</i>	9
3.3 Sensor Berat	10
3.4 Driver Motor	11



3.5	Motor DC	11
3.6	Logika <i>Fuzzy</i>	12
3.6.1	Himpunan <i>Fuzzy</i>	13
3.6.2	Fungsi Keanggotaan <i>Fuzzy</i>	14
3.6.3	Operasi Himpunan <i>Fuzzy</i>	16
3.6.4	Variabel Linguistik (<i>Linguistic Variable</i>)	17
3.6.5	Aturan <i>Fuzzy</i> “ <i>If-Then</i> ”	18
3.6.6	Kontrol Logika <i>Fuzzy</i>	20
3.6.1.1	Fuzzifikasi	21
3.6.1.2	Defuzzifikasi	21
3.6.7	Struktur Kontroler Logika <i>Fuzzy</i> Dasar	22
3.7	Mikrokontroler AT89S52	24
3.7.1	Konfigurasi Pin	26
3.7.2	Struktur dan Operasi Port	27
3.7.3	Register Fungsi Khusus	29
3.7.4	Osilator	30
3.7.5	Reset	31
BAB IV : METODE PENELITIAN		32
4.1	Perancangan Sistem	32
4.1.1	Perancangan Perangkat Keras (<i>Hardware</i>)	32
4.1.2	Perancangan Perangkat Lunak (<i>Software</i>)	33
4.2	Pembuatan Alat	33
4.2.1	Pembuatan Perangkat Keras (<i>Hardware</i>)	33
4.2.2	Pembuatan Perangkat Lunak (<i>Software</i>)	33
4.3	Pengujian Alat	33
4.3.1	Pengujian Perangkat Keras (<i>Hardware</i>)	33
4.3.2	Pengujian Perangkat Lunak (<i>Software</i>)	34
4.3.3	Pengujian Keseluruhan	34
4.4	Analisa Data dan Pengambilan Kesimpulan	34
BAB V : PERENCANAAN DAN PEMBUATAN ALAT		35
5.1	Gambaran Umum	35

5.2	Perancangan <i>Hardware</i>	35
5.2.1	Rangkaian Sensor Kecepatan	36
5.2.2	Rangkaian Sensor Posisi	37
5.2.2.1	Rangkaian Pemancar	37
5.2.2.2	Rangkaian Penerima	38
5.2.2.3	Rangkaian Pembanding (Komparator)	39
5.2.3	Sensor Berat	39
5.2.4	Sistem Mikrokontroler AT89S52	41
5.2.4.1	<i>Power On Reset</i>	42
5.2.5	Rangkaian Driver Motor DC	44
5.3	Perancangan <i>Software</i>	45
5.3.1	Perancangan Kontroler Logika <i>Fuzzy</i>	46
5.3.1.1	Fuzzifikasi	49
5.3.1.2	Kaidah Atur Kontroler Logika <i>Fuzzy</i>	50
5.3.1.3	Defuzzifikasi	51
BAB VI : PENGUJIAN ALAT		52
6.1	Pengujian Perangkat Keras	52
6.1.1	Pengujian Sensor Kecepatan	52
6.1.2	Pengujian Sensor Berat	53
6.1.3	Pengujian Sensor Posisi	54
6.1.4	Pengujian Driver Motor	55
6.1.5	Pengujian Pengukuran Kecepatan	56
6.1.6	Pengujian PWM	59
6.2	Pengujian Perangkat Lunak	60
6.2.1	Pengujian Sistem Mikrokontroler	60
6.3	Pengujian Sistem Secara Keseluruhan	62
BAB VII : PENUTUP		65
7.1	Kesimpulan	65
7.2	Saran	65
DAFTAR PUSTAKA		67
LAMPIRAN		

DAFTAR GAMBAR

Gambar 3.1	Simbol LED Infra Merah	7
Gambar 3.2	Simbol Phototransistor	8
Gambar 3.3	<i>Optocoupler</i> Jenis IC	9
Gambar 3.4	<i>Optocoupler</i> Jenis U dan Diagram Rangkaianya	9
Gambar 3.5	Perancangan Secara Mekanik	10
Gambar 3.6	Konfigurasi pin-pin Driver Motor L293D	11
Gambar 3.7	Motor DC	12
Gambar 3.8	<i>S-Function</i>	15
Gambar 3.9	π - <i>Function</i>	15
Gambar 3.10	<i>T-Function</i>	16
Gambar 3.11	Contoh Operasi Gabungan Himpunan <i>Fuzzy</i> A dan B	17
Gambar 3.12	Contoh Operasi Irisan Himpunan <i>Fuzzy</i> A dan B	17
Gambar 3.13	Contoh Operasi Komplemen Himpunan <i>Fuzzy</i> A	17
Gambar 3.14	Diagram Representasi dari <i>Fuzzy</i> Kecepatan	18
Gambar 3.15	Metode Penalaran <i>Fuzzy</i> MAX-MIN	20
Gambar 3.16	Metode Penalaran <i>Fuzzy</i> MAX-DOT	20
Gambar 3.17	Struktur Kontroler Logika <i>Fuzzy</i> Dasar	23
Gambar 3.18	Arsitektur Mikrokontroler	24
Gambar 3.19	Blok Diagram AT89S52	25
Gambar 3.20	Konfigurasi Pin dari AT89S52	26
Gambar 3.21	Rangkaian Osilator	30
Gambar 3.22	<i>Power On Rreset</i> dan Rangkaian Ekuivalennya	31
Gambar 5.1	Diagram Sistem Kontrol Kecepatan, Posisi dan Berat	35
Gambar 5.2	Rangkaian <i>Optocoupler</i>	37
Gambar 5.3	Rangkaian Sensor Posisi	37
Gambar 5.4	Rangkaian Pemancar	38
Gambar 5.5	Rangkaian Penerima	38
Gambar 5.6	Skema Komparator dan Fungsi Alih Pembanding	39
Gambar 5.7	Perancangan Secara Mekanik	40
Gambar 5.8	Rangkaian <i>Optocoupler</i> dari Sensor Berat	41
Gambar 5.9	Minimum Sistem Mikrokontroler AT89S52	42

Gambar 5.10	Rangkaian <i>Power On Reset</i>	43
Gambar 5.11	Rangkaian Driver Motor DC	44
Gambar 5.12	Diagram Alir Pemrograman <i>Fuzzy</i> Secara Umum	45
Gambar 5.13	Diagram Alir <i>Routine Fuzzy</i> Secara Umum	46
Gambar 5.14	Fungsi Keanggotaan <i>Crisp Input</i> Berat	47
Gambar 5.15	Fungsi Keanggotaan <i>Crisp Input</i> Kecepatan	47
Gambar 5.16	Fungsi Keanggotaan <i>Crisp Output Duty cycle</i>	47
Gambar 5.17	Fungsi Keanggotaan <i>Crisp Input</i> Berat	48
Gambar 5.18	Fungsi Keanggotaan <i>Crisp Input</i> Kecepatan	48
Gambar 5.19	Fungsi Keanggotaan <i>Crisp Output Duty cycle</i>	49
Gambar 6.1	Blok Diagram Pengujian Sensor Kecepatan	52
Gambar 6.2	Blok Diagram Pengujian Sensor Berat	53
Gambar 6.3	Blok Diagram Pengujian Sensor Posisi	55
Gambar 6.4	Blok Diagram Pengujian Driver Motor	56
Gambar 6.5	Blok Diagram Pengujian pengukuran kecepatan	57
Gambar 6.6	Grafik Pengukuran Kecepatan Dengan Menggunakan Tachometer	58
Gambar 6.7	Grafik Pengukuran Kecepatan Dari Pembacaan <i>Software</i>	58
Gambar 6.8	Grafik Tegangan Terhadap Waktu Untuk Duty Cycle 25%	59
Gambar 6.9	Grafik Tegangan Terhadap Waktu Untuk Duty Cycle 50%	60
Gambar 6.10	Grafik Tegangan Terhadap Waktu Untuk Duty Cycle 75%	60
Gambar 6.11	Blok Diagram Pengujian Mikrokontroler	61
Gambar 6.12	Blok Diagram Pengujian Sistem secara Keseluruhan	62
Gambar 6.13	Grafik Hubungan Duty Cycle Terhadap Berat Beban Untuk Kecepatan 50 rpm	63
Gambar 6.14	Grafik Hubungan Duty Cycle Terhadap Berat Beban Untuk Kecepatan 89 rpm	63
Gambar 6.15	Grafik Hubungan Duty Cycle Terhadap Berat Beban Untuk Kecepatan 160 rpm	63
Gambar 6.16	Grafik Hubungan Duty Cycle Terhadap Berat Beban Untuk Kecepatan 220 rpm	64

DAFTAR TABEL

Tabel 3.1	Tabel Kaidah Atur Dalam Bentuk Matrik	24
Tabel 3.2	Fungsi Khusus <i>Port 3</i>	28
Tabel 5.1	Kondisi <i>Optocoupler</i> Disetiap Berat	40
Tabel 5.2	Tabel Kaidah pengaturan Kecepatan dan ketika belok	50
Tabel 6.1	Hasil Pengujian Sensor Kecepatan	52
Tabel 6.2	Hasil Pengujian Sensor Berat	54
Tabel 6.3	Hasil Pengujian Sensor Posisi	55
Tabel 6.4	Hasil Pengujian Driver motor Depan	56
Tabel 6.5	Hasil Pengujian Driver motor Belakang	56
Tabel 6.6	Hasil Pengujian Kecepatan	57
Tabel 6.7	Hasil Pengujian Sistem mikrokontroler	61
Tabel 6.8	Hasil Pengukuran Nilai Keluaran Kontroler dalam Duty Cycle	63



ABSTRAK

Rahmad Kuspinuji Budiantoro, 2006, **Perancangan Pengendali Logika Fuzzy Pada Model Robot Pengangkut Barang Menggunakan AT89S52**, Jurusan Teknik Elektro, Fakultas Teknik, Universitas Brawijaya, Malang. Pembimbing: Moch. Rif'an, ST. MT dan Dipl. Ing. Ir. M. Rusli

Algoritma *Fuzzy* merupakan salah satu algoritma yang mudah, handal dan sangat baik untuk diaplikasikan pada berbagai macam sistem dengan hanya menggunakan variabel masukan dan keluaran sistem tanpa harus mengetahui fungsi alih terlebih dahulu. Keuntungan lain penggunaan algoritma *fuzzy* adalah kemudahan dalam memperbaiki kinerja sistem yaitu dengan mengubah *software*-nya tanpa harus mengubah *hardware*-nya. Pada tugas akhir ini kontrol logika *fuzzy* digunakan untuk mengendalikan prototipe robot pengangkut barang dan masukan yang diberikan untuk kontroler logika *fuzzy* adalah berat dan kecepatan. Dengan adanya dua variabel masukan ini, diharapkan variabel keluaran yang sesuai, sehingga prototipe robot pengangkut barang dapat bergerak secara otomatis. Pada sistem ini digunakan mikrokontroler AT89S52, yang dilengkapi dengan berbagai fasilitas *kernel petrafuz*. Dengan adanya alat ini, diharapkan bisa melahirkan sistem pengendali logika *fuzzy* pada model robot pengangkut barang. Setelah dilakukan perancangan dan pengujian, pengendalian sistem ini dapat berjalan sesuai yang diharapkan dengan didapatkan *error* rata-rata pembacaan rpm secara software sebesar 2,2% dan hasil pengukuran nilai keluaran kontroler dalam *duty cycle* didapatkan dari grafik hubungan *duty cycle* terhadap berat beban untuk kecepatan 50 rpm, 89 rpm, 160 rpm, dan 220 rpm.

Kata Kunci : Robot Pengangkut Barang, Logika *Fuzzy*, *Duty cycle*, PWM



BAB I PENDAHULUAN

1.1 Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi di bidang sistem kontrol sangat bermanfaat dalam mempermudah kerja dan segala aktifitas manusia. Dengan perkembangan elektronika yang demikian pesatnya sehingga membawa dampak yang luar biasa di berbagai bidang kehidupan. Pada umumnya konsumen atau manusia akan tertarik terhadap suatu produk atau rancangan yang dapat meringankan suatu pekerjaan. Misalnya alat atau rancangan yang dapat melakukan pemindahan barang yang dapat dikontrol.

Robot pengangkut barang adalah salah satu hasil dari rancangan kecanggihan teknologi, dimana fungsi dari robot pengangkut barang ini yaitu untuk meringankan atau mempermudah dalam pemindahan barang. Sesuai dengan namanya, robot pengangkut barang ini menggunakan motor elektrik sebagai penggerak roda.

Dalam perancangan prototipe robot pengangkut barang, model dari *plant* robot pengangkut barang yang akan dikendalikan haruslah diketahui lebih dulu. Berdasarkan model yang diperoleh itu, maka perancangan kendali dapat menggunakan beberapa teknik identifikasi untuk memperoleh algoritma kontrolnya.

Perancangan prototipe robot pengangkut barang ini menggunakan teknologi kontrol logika *fuzzy*. Karena pada dasarnya pengendalian logika *fuzzy* merupakan metode pengendalian yang mensimulasikan cara berpikir manusia dengan melibatkan ketidakpastian yang terdapat pada sistem fisik. Pengendalian *fuzzy* dilakukan oleh sistem pakar berbasis aturan *fuzzy* (*fuzzy rule base expert system*) sebuah model proses berpikir secara logika yang mirip dengan cara manusia mengambil keputusan.

Dalam penelitian ini akan dirancang suatu sistem pengendalian kecepatan prototipe robot pengangkut barang berbasis algoritma *fuzzy* yang disesuaikan dengan keperluan aplikasi yang akan dilakukan. Pemilihan robot pengangkut barang dilatarbelakangi oleh banyaknya kebutuhan dalam pemindahan barang.

Sedangkan algoritma *fuzzy* merupakan algoritma yang mudah dan handal untuk diaplikasikan sebagai pengendali pada berbagai macam sistem dengan hanya menggunakan variabel-variabel masukan dan keluaran sistem tanpa harus mengetahui fungsi alih atau persamaan dinamik sistem tersebut. Keuntungan lain pemakaian algoritma *fuzzy* adalah kemudahan dalam memperbaiki kinerja sistem yaitu dengan mengubah *software*-nya tanpa harus mengubah konfigurasi hardware-nya. Sehingga praktis dan ekonomis dalam penggunaannya.

1.2 Rumusan Masalah

Tugas akhir ini diwujudkan dalam bentuk prototipe yang dapat mewakili sistem yang sesungguhnya, sehingga permasalahan dalam penelitian ini dapat dirumuskan sebagai berikut :

1. Bagaimana mengubah sinyal yang berasal dari sensor kecepatan menjadi sinyal *input* sehingga dapat diproses oleh kontrol logika *fuzzy*.
2. Bagaimana mengubah sinyal yang telah diproses oleh kontrol logika *fuzzy* menjadi sinyal keluaran sehingga dapat digunakan untuk pengendalian kecepatan gerak roda pada robot pengangkut barang.
3. Bagaimana merancang suatu sistem pengendalian menggunakan logika *fuzzy* yang dapat menghasilkan respon sesuai dengan yang direncanakan.
4. Bagaimana merancang sistem pengendalian kecepatan gerak roda dan pada prototipe robot pengangkut barang yang dapat mengatasi gangguan perubahan beban.

1.3 Batasan Masalah

Karena begitu luasnya objek kajian maka perlu dilakukan pembatasan masalah agar pembahasan lebih terfokus pada rumusan masalah. Adapun batasan masalah dalam skripsi ini adalah :

1. Menggunakan prototipe dari Robot pengangkut barang.
2. Sensor berat menggunakan kombinasi pegas dan *optocoupler*.
3. Sebagai penggerak digunakan motor DC.
4. Sensor kecepatan menggunakan rangkaian sensor potthotransistor.
5. Perancangan diutamakan pada perancangan kontrolernya serta kinerja sistem secara umum.

1.4 Tujuan

Tujuan yang ingin dicapai dari tugas akhir ini adalah terciptanya alat pengendali robot pengangkut barang sesuai dengan lintasan pergerakan secara otomatis dengan berbasis algoritma *fuzzy* untuk mempermudah orang dalam melakukan pengangkutan barang.

1.5 Sistematika Penulisan

Tugas akhir terdiri dari enam bab dengan sistematika penulisan sebagai berikut:

BAB I PENDAHULUAN

Membahas latar belakang, rumusan masalah, ruang lingkup, tujuan, dan sistematika pembahasan.

BAB II ROBOT PENGANGKUT BARANG

Membahas tentang robot pengangkut barang

BAB III TINJAUAN PUSTAKA

Menjelaskan tentang sensor yang dipakai, *Optocoupler*, Driver Motor, Motor DC, Kontrol Logika Fuzzy, dan Mikrokontroler AT89S52.

BAB IV METODE PENELITIAN

Menjelaskan tentang metode yang digunakan dalam perancangan alat.

BAB V PERENCANAAN DAN PEMBUATAN ALAT

Menjelaskan tentang diagram blok rangkaian, perinsip kerja rangkaian sistem alat tersebut dan realisasi alat.

BAB VI PENGUJIAN DAN ANALISA

Menjelaskan tentang hasil pengujian alat.

BAB VII PENUTUP

Berisikan kesimpulan dan saran.

BAB II

ROBOT PENGANGKUT BARANG

2.1 Pengenalan Robotika

Istilah robot yang biasa terdengar umumnya mengandung pengertian suatu alat yang menyerupai manusia atau bahkan bertingkah laku seperti manusia, namun struktur tubuhnya tidak seperti manusia melainkan terbuat dari bahan logam. Pada hakekatnya robot mengandung beberapa unsur-unsur pendukung :

- a. *Programable*
- b. *Automatic* (otomatis)
- c. *Manipulator* (perangkat manipulasi)
- d. *Human like* (mempunyai kemiripan dengan manusia)

Dari unsur-unsur diatas jelaslah bahwa robot bukan hanya sekedar perkakas biasa, namun merupakan mesin khusus yang dapat dikontrol oleh manusia lewat suatu *processor* atau *controller*. Terdapat dua defenisi yang dapat diterima dikalangan industri mengenai robot, yaitu :

- a. Menurut RIA (*Robotik Institute of Amerika*) robot adalah “*Manipulator* yang berfungsi jamak dan dapat diprogram ulang dan dirancang untuk mengangkut material, peralatan atau perangkat khusus melalui peubah pergerakan terprogram untuk melakukan tugas bervariasi”
- b. Robot merupakan peralatan yang melakukan fungsi-fungsi yang biasa dilakukan oleh manusia, atau peralatan yang bekerja dengan kecerdasan yang mirip dengan kecerdasan manusia.

2.1.1 Mobil Robot

Kata mobil robot mempunyai arti bergerak, yang dimaksudkan adalah sistem robot tersebut mampu memindahkan dirinya sendiri dari posisi A ke posisi B, dimana kedua posisi tersebut berada pada jarak tertentu (keseluruhan badan robot berpindah tempat), bisa dikatakan bahwa robot tersebut bergerak dinamis.

2.1.1.1 Operator Oriented

Mobil robot dengan *operator oriented* adalah pengenalan gerakan dari robot yang membutuhkan seorang operator. Jadi seluruh gerakan robot untuk memindahkan tubuhnya tergantung dari instruksi yang diberikan oleh seorang operator.

2.1.1.2 Self Running

Mobil robot dengan *Self Running* adalah pengendalian gerakan dari robot yang berdasarkan program kemudi yang diberikan sehingga seolah-olah robot tersebut bergerak sendiri. Jenis ini tidak tergantung dari kemudi seorang operator dan juga biasanya ditempatkan beberapa jenis sensor untuk mendeteksi situasi sekelilingnya (untuk mengenali medan jelajahnya). Sensor tersebut akan memberikan informasi kepada sistem robot, kemudian oleh perangkat prosesor atau kontrol informasi tersebut diolah, yang nantinya dijadikan sebagai acuan dalam melakukan pergerakan selanjutnya. *Self Running* dapat dikategorikan lagi menjadi dua jika dilihat dari tingkat kecerdasan robot tersebut, yakni robot dengan kecerdasan buatan dan tanpa kecerdasan buatan. Robot dengan kecerdasan buatan memiliki maksud bahwa robot tersebut berkemampuan secara sendiri untuk merespon atau bereaksi didalam kondisi yang tidak ditentukan sebelumnya. Selanjutnya robot dengan tanpa kecerdasan buatan secara keseluruhan tergantung kepada intstruksi yang diberikan.

2.2 Pengertian Robot Pengangkut Barang

Robot pengangkut barang bisa disebut juga seperti mobil robot, karena juga bisa bergerak atau berpindah dari posisi A ke posisi B sesuai dengan jalur yang dilewatinya, pergerakan robot tersebut tidak bisa dilepaskan dari perangkat sensor dan kecerdasan robot, dimana level yang lebih tinggi dari hierarki sebuah sensor ialah kemampuan untuk merasakan perubahan sesaat pada lingkungan tersebut yang kemudian diasosiasikan dengan antarmuka dan algoritma *artificial intelligent*.

Pada robot pengangkut barang sensor yang digunakan yaitu sebuah sensor yang dapat merasakan dan memanipulasi keadaan dengan fungsi matrik dan

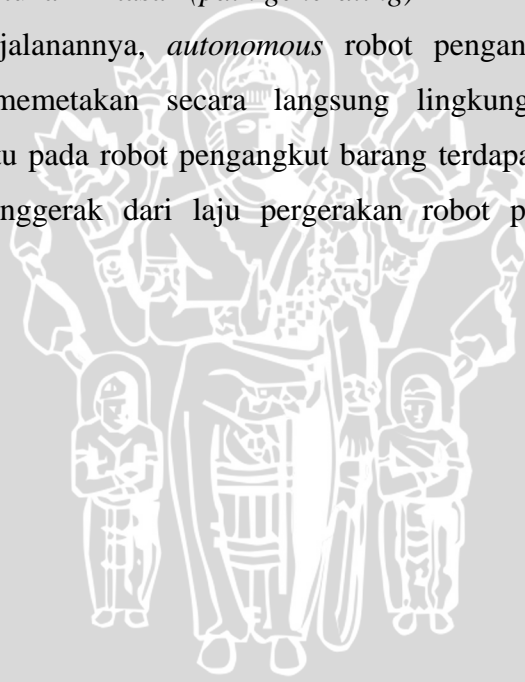
algoritma *Artificial Intellegent*. Pada robot pengangkut barang ini, banyak ditemui pemrograman perangkat keras untuk masing-masing alat yang dibutuhkan.

2.2.1 Elemen Pengolahan Robot Pengangkut Barang

Elemen pengolahan robot pengangkut barang yang akan diimplementasikan adalah berbasis visual. Dengan menggunakan sensor visual, maka robot ini diharapkan dapat bekerja sesuai dengan cara kerja manusia. Sensor visual dianalogikan seperti indra penglihatan manusia. Sistem visual yang digunakan pada robot pengangkut barang ini diharapkan dapat melakukan beberapa fungsi, yaitu :

- Pemetaan lingkungan (*environment mapping*)
- Penjejakan objek (*object tracking*)
- Pembentukan lintasan (*path generating*)

Sepanjang perjalanannya, *autonomous* robot pengangkut barang atau mobil robot akan memetakan secara langsung lingkungan yang sedang dijelajahnya. Selain itu pada robot pengangkut barang terdapat motor DC, yang fungsinya sebagai penggerak dari laju pergerakan robot pengangkut barang tersebut



BAB III

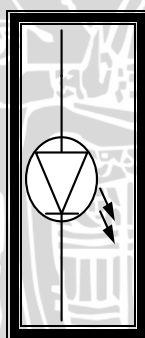
TINJAUAN PUSTAKA

3.1 Sensor Cahaya

Sensor cahaya yang digunakan terdiri atas LED sebagai *transmitter* dan *phototransistor* sebagai *receiver*. Sensor cahaya ini digunakan untuk mendeteksi suatu garis dengan warna tertentu

3.1.1 *Light Emitting Diode (LED) Infra Merah*

LED infra merah digunakan untuk menghasilkan sinar infra merah. Prinsip kerjanya adalah pada waktu LED infra merah dibias maju, elektron dari pita konduksi melewati *junction* dan jauh ke dalam *hole* pita valensi, sehingga elektron-elektron tersebut memancarkan energi. Pada dioda penyearah biasa, energi ini dikeluarkan sebagai energi panas. Tetapi pada LED, energi ini dipancarkan sebagai cahaya. Sedangkan pada LED infra merah memancarkan cahaya yang tidak kelihatan. Simbol dan bentuk fisik LED infra merah yang sering digunakan ditunjukkan dalam Gambar 3.1.



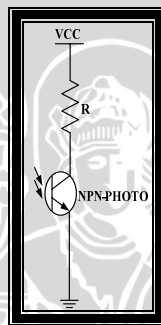
Gambar 3.1. Simbol LED Infra Merah
Sumber: Motorola Semiconductor, 1989: 4-11

LED infra merah merupakan PN *junction* yang memancarkan radiasi infra merah yang tidak terlihat oleh mata. Apabila pada anoda diberi tegangan positif dan katoda ke ground, maka LED menjadi aktif dan arus akan mengalir dari anoda ke katoda. Pada reaksi semikonduktor suatu dioda akan terjadi perpindahan elektron dari tipe N menuju tipe P dan berpindahnya *hole* dari tipe N. Proses rekombinasi antara elektron dan *hole*, mengakibatkan pelepasan energi berupa pancaran cahaya.

Efisiensi pancaran cahaya akan berkurang, dengan berkurangnya arus *input* dan kenaikan suhu. Pada LED infra merah, cahaya yang dipancarkan mempunyai panjang gelombang yang sangat kecil, sehingga pancaran gelombang tersebut tidak tertangkap oleh mata manusia.

3.1.2 Phototransistor

Phototransistor merupakan suatu transistor yang peka terhadap tenaga cahaya. Suatu sumber cahaya menghasilkan energi panas, begitu pula dengan spektrum infra merah. Karena spektrum infra merah mempunyai efek panas yang lebih besar dari cahaya tampak, maka *phototransistor* lebih peka untuk menangkap radiasi dari sinar infra merah. Simbol dari *phototransistor* seperti ditunjukkan dalam Gambar 3.2 berikut ini:



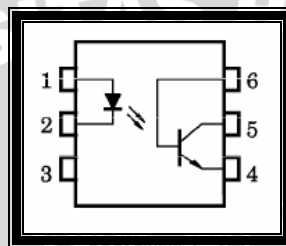
Gambar 3.2. Simbol *Phototransistor*
Sumber: Perancangan

3.2 Optocoupler

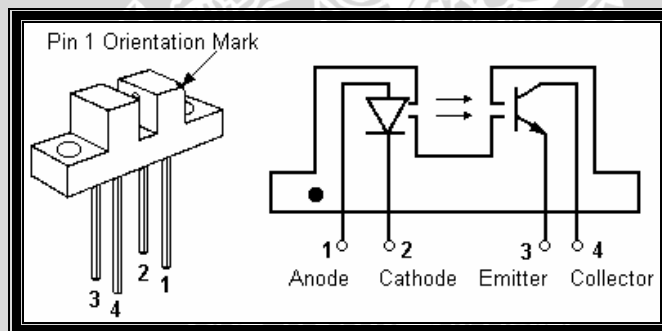
Optocoupler adalah suatu komponen penghubung (*coupling*) yang bekerja berdasarkan picu cahaya optik. *Optocoupler* terdiri dari dua bagian yaitu:

- Pada bagian *transmitter* dibangun dari sebuah LED infra merah. Jika dibandingkan dengan menggunakan LED biasa, LED infra merah memiliki ketahanan yang lebih baik terhadap sinar tampak. Cahaya yang dipancarkan oleh LED infra merah tidak terlihat oleh mata telanjang.
- Pada bagian *receiver* dibangun dengan dasar komponen *phototransistor*. *Phototransistor* merupakan suatu transistor yang peka terhadap tenaga cahaya. Suatu sumber cahaya menghasilkan energi panas, begitu pula dengan spektrum infra merah. Karena spektrum infra merah mempunyai efek panas yang lebih besar dari cahaya tampak, maka *phototransistor* lebih peka untuk menangkap radiasi dari sinar infra merah.

Ditinjau dari penggunaannya, fisik *optocoupler* dapat berbentuk bermacam-macam. Bila hanya digunakan untuk mengisolasi *level* tegangan atau data pada sisi *transmitter* dan sisi *receiver*, maka *optocoupler* ini biasanya dibuat dalam bentuk *solid* (tidak ada ruang antara *LED* dan *phototransistor*). Sehingga sinyal listrik yang ada pada *input* dan *output* akan terisolasi. Dengan kata lain *optocoupler* ini digunakan sebagai *optoisolator* antara rangkaian *input* dan *output*. Bentuk dari *optocoupler* jenis IC ditunjukkan dalam Gambar 3.3, sedangkan bentuk dan diagram dari rangkaian *optocoupler* jenis U ditunjukkan dalam Gambar 3.4.



Gambar 3.3. *Optocoupler* Jenis IC
Sumber: Toshiba, 4N25, Datasheet, 27 Februari 1998



Gambar 3.4. *Optocoupler* Jenis U dan Diagram Rangkaiannya
Sumber: Fairchild Semiconductor, H22A1, Datasheet, 2000

3.2.1 Prinsip Kerja *Optocoupler*

Optocoupler memiliki sebuah LED yang dihubungkan dengan masukan dan sebuah dioda yang dihubungkan dengan keluaran. Prinsip kerja dari *optocoupler* adalah:

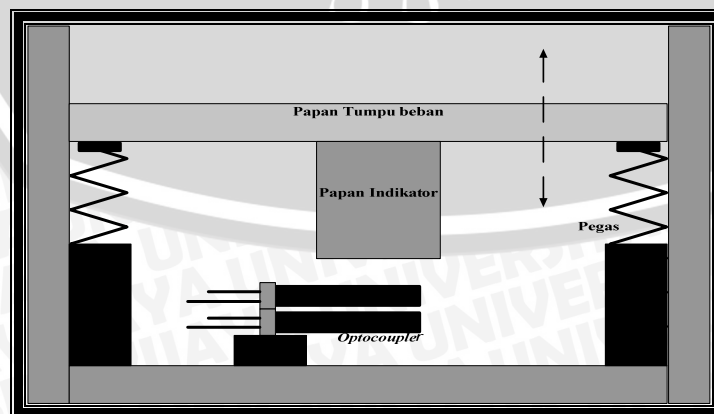
- Jika antara *phototransistor* dan LED terhalang maka *phototransistor* tersebut akan *off* sehingga *output* dari kolektor akan berlogika *high*.
- Sebaliknya jika antara *phototransistor* dan LED tidak terhalang maka *phototransistor* tersebut akan *on* sehingga *output*-nya akan berlogika *low*.

Dengan mengetahui prinsip kerja dari *optocoupler* ini maka harus dibuat penghalang antara *phototransistor* dan LED, pada penghalang itu diberi sedikit lubang. Penghalang itu harus dibuat seporos dengan roda tersebut. Sehingga ketika berputar, *output* dari *phototransistor* akan mengalami *high* dan *low* secara bergantian. Penghalang yang biasanya digunakan adalah jeruji roda. Ketika roda berputar, maka ruji-ruji pada roda akan memotong cahaya yang dipancarkan oleh LED, sehingga akan mengalir arus yang melewati resistor *output*. Karena tegangan *output* sama dengan tegangan *supply* dikurangi tegangan yang melewati resistor. Sehingga ketika masukan berubah, intensitas cahaya berubah, maka tegangan keluaran juga akan berubah.

3.3 Sensor Berat

Pada tugas akhir ini digunakan suatu perancangan sensor berat dengan menggunakan *optocoupler*. *Optocoupler* diletakkan pada dasar *chasis* dari robot pengangkut barang ini. Papan beban sebagai tempat untuk meletakkan beban dengan bertumpu pada keempat pegas yang diletakkan pada masing-masing sudutnya. Papan indikator diletakkan dibawah papan beban yang berfungsi sebagai tanda untuk menyatakan terjadinya perubahan beban. Beban yang terdeteksi dibedakan menjadi tiga kondisa yaitu ringan, sedang dan berat. Pada saat kodisi beban ringan yang terjadi papan indikator tidak melewati celah pada kedua *optocoupler*, untuk kondisi beban sedang maka papan indikator melewati celah pada *optocoupler* yang paling atas, sedangkan untuk kondisi beban yang berat papan indikator melewati celah pada kedua *optocoupler*.

Perancangan sensor berat ini secara mekanik dapat dilihat dalam Gambar 3.5.



Gambar 3.5. Perancangan Secara Mekanik
Sumber: Perancangan

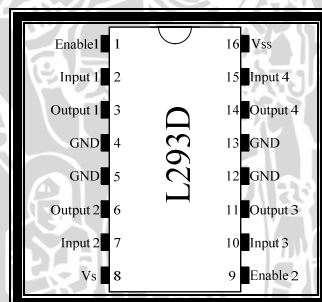
3.4 Driver Motor

Driver motor yang digunakan pada tugas akhir ini adalah L293D. Dimana driver motor L293D ialah sebuah IC yang mempunyai suatu tegangan tinggi yang terintegrasi monolitik, dan mempunyai empat saluran driver Driver motor ini baik untuk digunakan dalam menswitch aplikasi pada frekwensi sampai ke 5 KHZ.

Karakteristik pada driver motor L293D adalah sebagai berikut :

- Arus keluaran persaluran 600mA
- Arus keluaran puncak persaluran 1.2A
- Mempunyai keluaran dioda *clamp* untuk beban induktif transient
- Mempunyai suplai tegangan dalam antara 4.5V sampai 36V
- Perlindungan terhadap temperatur tinggi
- Logika 0 tegangan masukan sampai pada 1.5V (*High Noise Immunity*)

konfigurasi pin driver motor L293D dan diagram bloknya dapat dilihat dalam Gambar 3.6 dan.

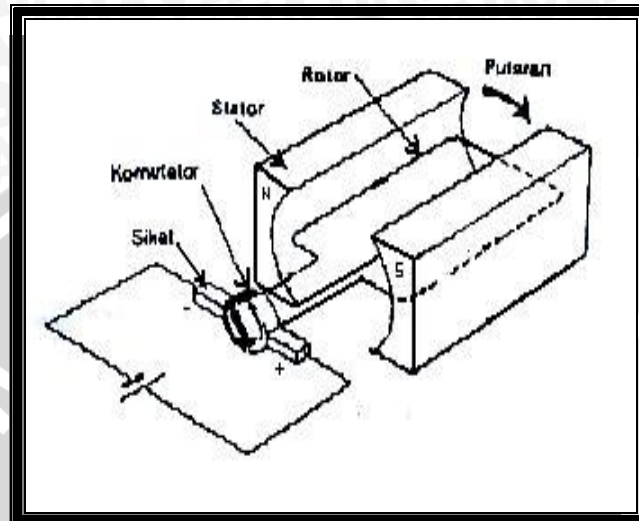


Gambar 3.6. Konfigurasi pin-pin Driver Motor L293D
Sumber : SGS-THOMSON Microelectronics, 1994

3.5 Motor DC

Motor DC hampir sama konstruksinya dengan motor AC, perbedaannya terletak pada sikat dan cincin belah (komutator). Saat siklus pertama, arus mengalir dari kutub positif ke negatif. Aliran arus yang melewati bagian kabel yang berada didekat kutub N magnet akan menimbulkan gaya Lorentz ke bawah. Sementara itu aliran arus yang melewati kabel yang berada di dekat kutub S magnet akan menyebabkan gaya Lorentz ke atas. Kedua perpaduan gaya Lorentz tersebut akan menyebabkan kawat berputar. Pada siklus berikutnya terjadi hal yang serupa seperti pada siklus sebelumnya. Apabila arus terus-menerus dialirkan,

maka kawat akan berputar secara terus menerus pula. Pada aplikasi sesungguhnya, kawat adalah sebuah rotor yang akan dikopel dengan sebuah as dan akan memutar as tersebut terus menerus seiring perputaran motor. Motor DC ditunjukkan dalam Gambar 3.7.



Gambar 3.7. Motor DC
Sumber: Kismet F [1994:98]

3.6 Logika Fuzzy

Lotfi A. Zadeh pada tahun 1965 mengemukakan suatu himpunan yang disebut himpunan *fuzzy* (*fuzzy set*) untuk menerangkan suatu logika bertingkat. Logika ini kemudian dikenal dengan logika *fuzzy*. Logika ini menjadi dasar dari semua logika, dengan mengabaikan banyaknya tingkat kebenaran yang diasumsikan. Zadeh memilih kata “*fuzz*” untuk merepresentasikan suatu nilai logika kontinu antara 0 (pasti salah) dan 1 (pasti benar). [Jamshidi, 1993: 12].

Ciri utama dari logika *fuzzy* adalah adanya peluang untuk memanfaatkan toleransi ketidakpastian (*inexactness*) dan ketidaktepatan (*imprecision*). Perbedaan antara logika *fuzzy* dengan logika klasik adalah pada manipulasi simbolik dan numerik. Logika *fuzzy* dalam hal ini dapat menangani baik manipulasi simbolik maupun manipulasi numerik. Sedangkan logika klasik hanya dapat menangani manipulasi simbolik.

Zadeh mengatakan bahwa logika *fuzzy* diarahkan untuk menyediakan sebuah sistem formal berorientasi komputasi yang berisi konsep dan teknik untuk menghubungkan mode penalaran aproksimasi (*approximate reasoning*) dengan

penalaran eksak (*exact reasoning*). Logika *fuzzy* menjadi alat matematik yang membuat komputer mampu memodelkan sistem-sistem yang tidak terdefinisi secara tepat. Logika ini menjadikan komputer dapat membuat keputusan dari data-data yang tersedia walaupun tingkat keakuratannya sangat rendah.

Sebuah sistem berbasis logika *fuzzy* dapat mengurangi ketidakakuratan yang terjadi pada sistem klasik. Himpunan klasik (*crisp set*) memiliki persyaratan keanggotaan yang membatasi nilai anggota-anggota himpunannya hanya pada satu atau nol saja. Batasan dari logika ini terdefinisi dengan tajam pada angka nol dan satu, sehingga tingkat keakuratannya menjadi rendah. Sedangkan pada himpunan *fuzzy*, sebuah himpunan unsur-unsur atau anggota himpunan dapat menjadi anggota himpunan dengan tingkat keanggotaan antara nol dan satu. Ada transisi yang bertahap dari unsur yang menjadi anggota atau bukan anggota himpunan dalam sebuah himpunan *fuzzy*.

3.6.1 Himpunan Fuzzy

Himpunan didefinisikan sebagai kelompok objek-objek tertentu yang disebut elemen [Pebles, 1993: 3]. Himpunan *fuzzy* didefinisikan dalam semesta himpunan X , himpunan *fuzzy* ditentukan oleh fungsi keanggotaan yang memetakan anggota X ke daerah keanggotaan yang selalu berada dalam interval $[0,1]$. [Jun Yan, 1994: 15].

Jika X merupakan kumpulan objek, $X=\{x\}$ dengan X disebut semesta pembicaraan dan x merupakan anggota dari X , sebuah himpunan fuzzy A pada X digambarkan sebagai sekumpulan pasangan yang diurutkan:

$$A = \{(x, \mu_A(x)) | x \in X\} \quad (3.1)$$

dengan μ_A adalah fungsi keanggotaan yang memetakan X ke interval $[0,1]$, yaitu

$$\mu_A : X \rightarrow [0,1]$$

Jika penyokong himpunan *fuzzy* (x) hanya sebuah titik tunggal, maka $A = \mu_1 / x$, disebut sebuah *fuzzy tunggal* (*singleton fuzzy*).

Himpunan *fuzzy* yang dinyatakan dengan gabungan *fuzzy tunggal* diekspresikan sebagai

$$A = \mu_1 / x_1 + \mu_2 / x_2 + \dots + \mu_n / x_n \quad (3.2)$$

$$A = \sum \mu_A(x_i) / x_i$$

Bila X bersifat kontinu dan tak terbatas:

$$A = \int_x \mu_A(x) / x \tag{3.3}$$

3.6.2 Fungsi Keanggotaan Fuzzy

Fungsi keanggotaan menotasikan nilai kebenaran anggota himpunan. Interval nilai yang digunakan untuk menentukan fungsi keanggotaan, yaitu nol dan satu. Tiap fungsi keanggotaan memetakan elemen himpunan *crisp* ke semesta himpunan *fuzzy* [G. Klir dan Bo Yuan, 1995: 11].

Ada dua cara untuk menentukan fungsi keanggotaan himpunan *fuzzy*, yaitu secara numerik dan secara fungsional [Jun Yuan, 1994: 17].

- Pendefinisian secara numerik, mengekspresikan fungsi keanggotaan himpunan *fuzzy* sebagai suatu vektor angka yang tergantung pada tingkat diskritisasi.
- Pendefinisian secara fungsional, mendefinisikan fungsi keanggotaan himpunan *fuzzy* dalam ekspresi analitik yang mengikuti tingkatan tiap elemen dalam pendefinisian semesta himpunan yang dihitung.

Pada himpunan *fuzzy* semua informasi terkandung dalam fungsi keanggotaannya. Oleh karena itu penting untuk mengetahui bagian-bagian pada sebuah fungsi keanggotaan *fuzzy*. Ada beberapa fungsi keanggotaan dalam logika *fuzzy*, tetapi yang sering digunakan adalah bentuk fungsi-S, fungsi- π , dan fungsi T (*triangular*).

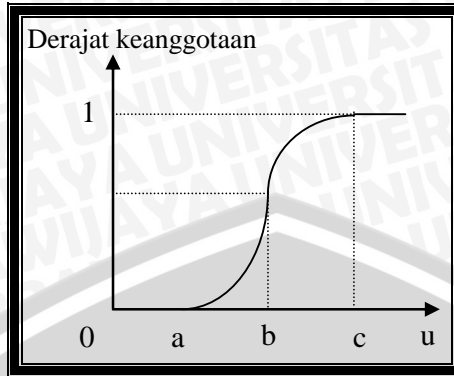
a. S-function

Definisi S-function adalah sebagai berikut:

$$S(u,a,b,c) = \begin{cases} 0 & u < a \\ 2\left(\frac{u-a}{c-a}\right) & a \leq u \leq b \\ 1-2\left(\frac{u-a}{c-a}\right) & b \leq u \leq c \\ 1 & u > c \end{cases} \tag{3.4}$$



Bentuk diagramatik *S-function* ditunjukkan dalam Gambar 3.8.



Gambar 3.8. *S-Function*

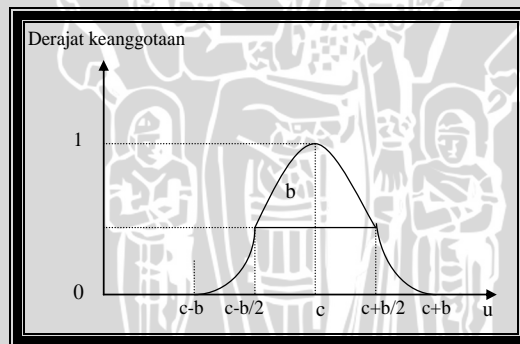
Sumber: Jun Yan, 1994: 29

b. π -function

Definisi π -Function adalah sebagai berikut:

$$\pi(u; b, c) = \begin{cases} S(u; c - b, c - \frac{b}{2}, c) & u \leq c \\ 1 - S(u; c, c + \frac{b}{2}, c + b) & u \geq c \end{cases} \quad (3.5)$$

Bentuk diagramatik π -Function ditunjukkan dalam Gambar 3.9.



Gambar 3.9. π -Function

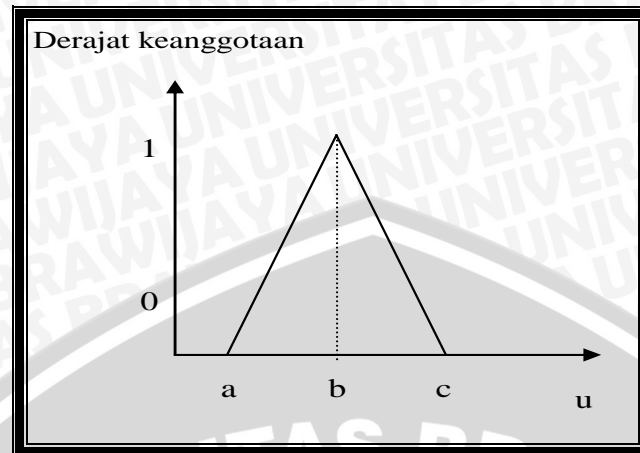
Sumber: Jun Yan, 1994: 29

c. Triangular-function

Triangular-function didefinisikan sebagai berikut:

$$T(u; a, b, c) = \begin{cases} 0 & u < a \\ \frac{u - a}{b - a} & a \leq u \leq b \\ \frac{c - u}{c - b} & b \leq u \leq c \\ 0 & u > c \end{cases} \quad (3.6)$$

Bentuk diagramatik *Triangular-function* ditunjukkan dalam gambar 3.10.



Gambar 3.10. T-Function
Sumber: Jun Yan, 1994: 30

3.6.3 Operasi Himpunan Fuzzy

Operasi dasar yang digunakan dalam himpunan *fuzzy* A dan B pada semesta himpunan X dengan fungsi keanggotaan μ_A dan μ_B terdiri dari gabungan, irisan, dan komplemen. Operasi dasar yang berlaku didefinisikan sebagai berikut:

a. Gabungan (*union*)

Gabungan himpunan *fuzzy* A dan B didefinisikan sebagai berikut:

$$\mu_{A \cup B} = \mu_A(x) \vee \mu_B(x) = \text{maks}\{\mu_A(x), \mu_B(x)\} \quad (3.7)$$

Operasi gabungan diperlihatkan dalam Gambar 3.11.

b. Irisan (*intersection*)

Irisan dua himpunan *fuzzy* A dan B didefinisikan sebagai berikut:

$$\mu_{A \cap B} = \mu_A(x) \wedge \mu_B(x) = \min\{\mu_A(x), \mu_B(x)\} \quad (3.8)$$

Operasi irisan diperlihatkan dalam Gambar 3.12.

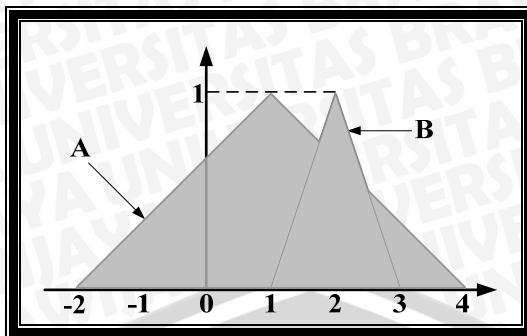
c. Komplemen (*complement*)

Komplemen himpunan *fuzzy* A dinotasikan sebagai \bar{A} . Operasi komplemen didefinisikan sebagai berikut:

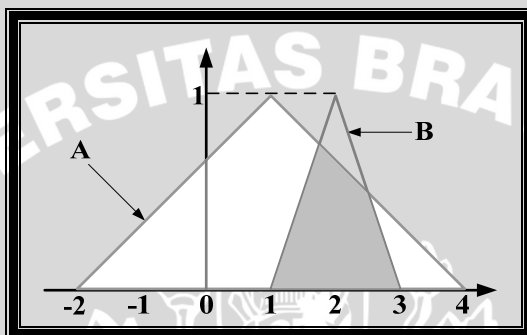
$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (3.9)$$

Operasi komplemen himpunan *fuzzy* A diperlihatkan dalam Gambar 3.13.

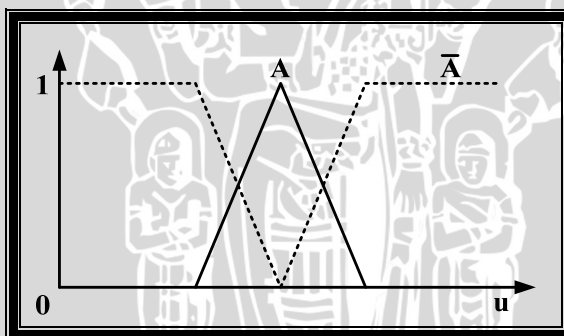




Gambar 3.11. Contoh Operasi Gabungan Himpunan *Fuzzy* A dan B
 Sumber: G. Klir dan Bo Yuan, 1995: 111



Gambar 3.12. Contoh Operasi Irisan Himpunan *Fuzzy* A dan B
 Sumber: G. Klir dan Bo Yuan, 1995: 111



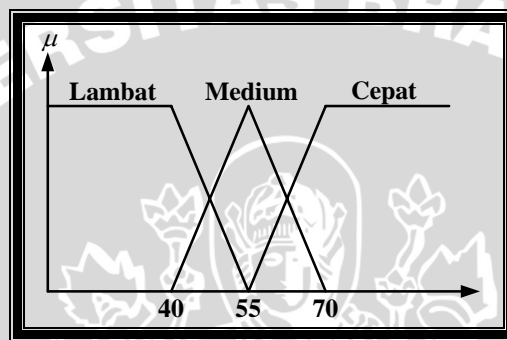
Gambar 3.13. Contoh Operasi Komplemen Himpunan *Fuzzy* A
 Sumber: Jamshidi, dkk., 1993: 19

3.6.4 Variabel Linguistik (*Linguistic Variable*)

Pendekatan dengan himpunan *fuzzy* untuk mempresentasikan cara berpikir manusia memerlukan variabel linguistik sebagai pengganti dari variabel yang biasa digunakan dalam pendekatan kuantitatif. Variabel linguistik ini berupa ungkapan-ungkapan yang bersifat kualitatif dan berfungsi untuk menyatakan himpunan *fuzzy*. Dalam berpikir manusia menggunakan ungkapan yang bersifat kualitatif terhadap informasi-informasi yang diinderanya.



Himpunan *fuzzy* memiliki nilai kebenaran yang dinyatakan dalam bentuk informasi linguistik, seperti sangat lambat, lambat, sedang, agak cepat, cepat dan sebagainya sebagai variabel, yang disebut variabel *fuzzy*. Variabel linguistik dapat dipandang sebagai variabel yang nilainya merupakan bilangan *fuzzy* atau sebagai variabel yang nilai-nilainya didefinisikan dalam bentuk linguistik. Sebagai contoh, lambat didefinisikan sebagai kecepatan dibawah 40 m/jam, medium sebagai kecepatan 55 m/jam dan cepat sebagai kecepatan diatas 70 m/jam, maka fungsi keanggotaan dengan kecepatan sebagai variabel linguistik dengan tiga variabel *fuzzy*: lambat, medium dan cepat yang ditunjukkan dalam Gambar 3.14.



Gambar 3.14. Diagram Representasi dari Fuzzy Kecepatan
Sumber: Lee, 1990: 406

3.6.5 Aturan *Fuzzy* “If-Then”

Kaidah atur yang digunakan dalam logika *fuzzy* berupa kaidah atur “If-Then”. Aturan “If” menunjukkan situasi sedangkan aturan “Then” menunjukkan aksi. Aturan “If” disebut antiseden sedangkan “Then” disebut konsekuen. Antiseden ini menunjukkan daerah masukan yang dikarakteristikan dalam semesta himpunan variabel masukan. Konsekuen mendefinisikan aksi kontrol yang diinginkan [Smith dan Corner, 1991: 19]. Bentuk umum dari aturan logika *fuzzy* ini sebagai berikut,

$$\text{If (kondisi), Then (aksi)} \quad (3.10)$$

Untuk menghubungkan suatu pernyataan dengan pernyataan lain digunakan operator AND, OR, dan ALSO. Operator AND dan OR digunakan pada bagian antiseden, sedangkan ALSO digunakan pada bagian konsekuen. Operator AND menyatakan operasi irisan, OR menyatakan operasi gabungan, dan ALSO digunakan untuk mengindikasikan adanya keluaran majemuk dalam aturan *fuzzy*. (Jun Yuan, 1994: 31)

Penggunaan operator ini dicontohkan sebagai berikut:

If x_1 adalah A_{k1} OR x_2 adalah A_{k2} AND x_3 adalah A_{k3} Then y_1 adalah B_{k1}

ALSO y_2 adalah B_{k2}

secara sederhana ditulis

If A_{k1} OR A_{k2} AND A_{k3} Then B_{k1} ALSO B_{k2}

Untuk sistem MISO (*multi input single output*) dengan n masukan, aturan *fuzzy* ke- k ditunjukkan sebagai berikut:

$$\text{If } A_{k1} \text{ AND } \dots \text{ AND } A_{ki} \text{ AND } \dots \text{ AND } A_{kn} \text{ Then } B_k \quad (3.11)$$

dengan: $i = 1, 2, \dots, n$

A_{ki} = Himpunan *fuzzy* variabel masukan x_i dalam aturan *fuzzy* ke- k

B_k = Himpunan *fuzzy* variabel keluaran y_j dalam aturan *fuzzy* ke- k

Diasumsikan terdapat dua basis kaidah atur untuk *fuzzy*, yaitu:

Kaidah atur 1 : If x adalah A_1 AND y adalah B_1 Then z adalah C_1

Kaidah atur 2 : If x adalah A_2 AND y adalah B_2 Then z adalah C_2

Jika kaidah atur ke- i dinyatakan dengan α_i maka dua kaidah atur di atas dapat dinyatakan kembali sebagai berikut:

$$\begin{aligned} \alpha_1 &= \mu_{A_1}(x_0) \wedge \mu_{B_1}(y_0) \\ \alpha_2 &= \mu_{A_2}(x_0) \wedge \mu_{B_2}(y_0) \end{aligned} \quad (3.12)$$

dengan x_0 dan y_0 menyatakan masukan *crisp*.

Metode-metode penalaran untuk memperoleh keputusan *fuzzy* ada dua, yaitu:

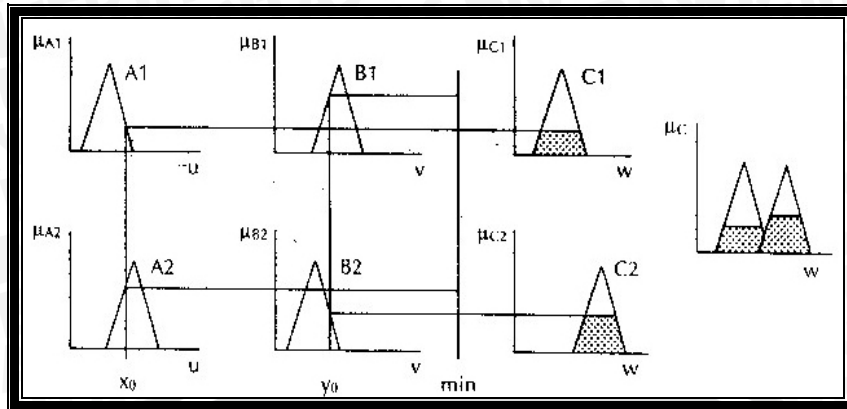
a. Metode penalaran MAX-MIN

Pada metode penalaran MAX-MIN fungsi keanggotaan konsekuen C dinyatakan dengan:

$$\mu_c(w) = (\alpha_1 \wedge \mu_{c1}(w)) \vee (\alpha_2 \wedge \mu_{c2}(w)) \quad (3.13)$$

metode penalaran MAX-MIN diperlihatkan dalam Gambar 3.15.





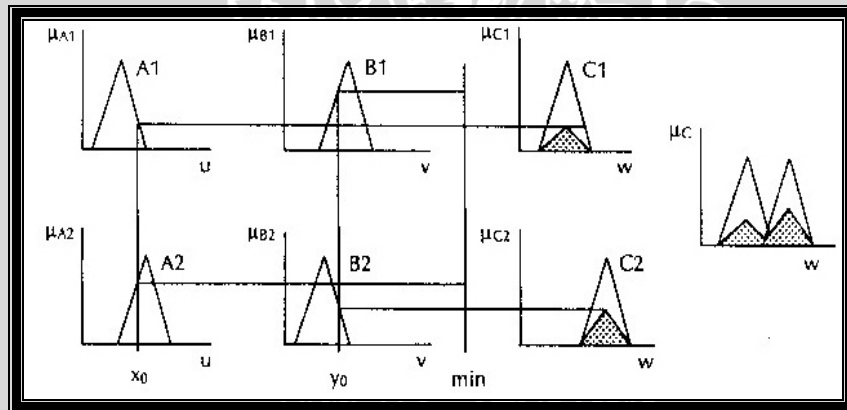
Gambar 3.15. Metode Penalaran *Fuzzy* MAX-MIN
 Sumber: Jun Yan, dkk., 1994: 55

b. Metode penalaran MAX-DOT

Pada metode penalaran MAX-DOT fungsi keanggotaan konsekuen C dinyatakan dengan:

$$\mu_c(w) = (\alpha_1 \cdot \mu_{c1}(w)) \vee (\alpha_2 \cdot \mu_{c2}(w)) \tag{3.14}$$

metode penalaran MAX-DOT diperlihatkan dalam Gambar 3.16.



Gambar 3.16. Metode Penalaran *Fuzzy* MAX-DOT
 Sumber: Jun Yan, dkk., 1994: 55

3.6.6 Kontrol Logika *Fuzzy*

Kontrol logika *fuzzy* adalah sistem berbasis aturan (*rule based system*) yang di dalamnya terdapat himpunan aturan *fuzzy* yang mempresentasikan mekanisme pengambilan keputusan. Aturan yang dibuat digunakan untuk memetakan variabel *input* ke variabel *output* dengan pernyataan *If-Then*.

Pengontrol ini akan menggunakan data tertentu (*crisp*) dari sejumlah sensor kemudian mengubahnya menjadi bentuk linguistik atau fungsi keanggotaan melalui proses fuzzifikasi. Lalu dengan aturan *fuzzy*, *inference engine* yang akan

menentukan hasil keluaran *fuzzy*. Setelah itu hasil ini akan diubah kembali menjadi bentuk numerik, proses ini disebut defuzzifikasi.

3.6.6.1 Fuzzifikasi

Fuzzifikasi merupakan proses pemetaan dari himpunan *crisp* ke himpunan *fuzzy*. Fuzzifikasi berfungsi untuk mengubah hasil pengukuran ke dalam pernyataan linguistik. Operasi fuzzifikasi melibatkan tranformasi sebuah himpunan *non fuzzy* ke dalam himpunan *fuzzy*. Proses fuzzifikasi dilakukan dengan menentukan fungsi keanggotaan. Proses fuzzifikasi diekspresikan sebagai berikut:

$$x = \text{fuzzifier}(x_0) \tag{3.15}$$

dengan: x_0 = vektor nilai *crisp* variabel masukan

x = vektor himpunan *fuzzy* variabel yang terdefinisi

fuzzifier = operator fuzzifikasi yang memetakan himpunan *crisp* ke himpunan *fuzzy*

Pedoman memilih fungsi keanggotaan untuk proses fuzzifikasi, menurut Jun Yan, menggunakan:

1. Himpunan *fuzzy* dengan distribusi simetris.
2. Gunakan himpunan *fuzzy* dengan jumlah ganjil, pemilihan sebaiknya 3, 5, atau 7.
3. Mengatur himpunan *fuzzy* agar saling menumpuk.
4. Menggunakan fungsi keanggotaan bentuk segitiga atau trapesioda.

3.6.6.2 Defuzzifikasi

Defuzzifikasi merupakan proses pemetaan himpunan *fuzzy* ke himpunan *crisp*. Proses ini merupakan kebalikan dari proses fuzzifikasi berfungsi untuk mengubah keluaran linguistik ke dalam keluaran *crisp*. Strategi defuzzifikasi diekspresikan sebagai berikut:

$$y_0 = \text{defuzzifier}(y) \tag{3.16}$$

dengan: y = aksi kontrol *fuzzy*

y_0 = aksi kontrol *crisp*

defuzzifier = operator defuzzifikasi



Metode defuzzifikasi ada dua, yaitu: Metode rata-rata maksimum dan metode pusat luasan [Jun Yang ,dkk.,1994: 56].

a. Metode Rata-Rata Maksimum (*Mean of Maximum*) MOM

Metode MOM disebut pula sebagai defuzzifikasi tinggi, aksi kontrol mewakilkan aksi kontrol yang fungsi keanggotaan mencapai maksimum. Nilai *crisp* yang dihasilkan oleh operasi MOM adalah:

$$W = \frac{\sum_{i=1}^n \alpha_i H_i W_i}{\sum_{i=1}^n \alpha_i H_i} \tag{3.17}$$

- dengan: n = jumlah aturan
- H_i = tinggi maksimum fungsi keanggotaan himpunan *fuzzy*
- W_i = kontrol keluaran dengan nilai *crisp*
- α_i = *fire strength* aturan ke-i

b. Metode Pusat Luasan (*Center of Area*) COA

Metode COA menghasilkan aksi kontrol pada pusat berat distribusi.

$$W = \frac{\sum_{i=1}^n \alpha_i M_i}{\sum_{i=1}^n \alpha_i A_i} \tag{3.18}$$

- dengan: n = jumlah aturan
- A_i = luas himpunan *fuzzy*
- M_i = momen fungsi keanggotaan himpunan *fuzzy* aturan keluaran kontrol ke-i
- α_i = *fire strength* aturan ke-i

3.6.7 Struktur Kontroler Logika *Fuzzy* Dasar

Kontroler logika *fuzzy* menggunakan himpunan aturan kontrol dan mekanisme pengambil keputusan (*inference*) untuk menentukan aksi kontrol pada *plant*. Aturan kontrol merupakan ekspresi *fuzzy* yang merelasikan logika *fuzzy* masukan kontroler ke nilai keluaran kontroler. Mekanisme pengambil keputusan mengevaluasi kaidah atur kemudian menentukan aksi kontrol yang sesuai.

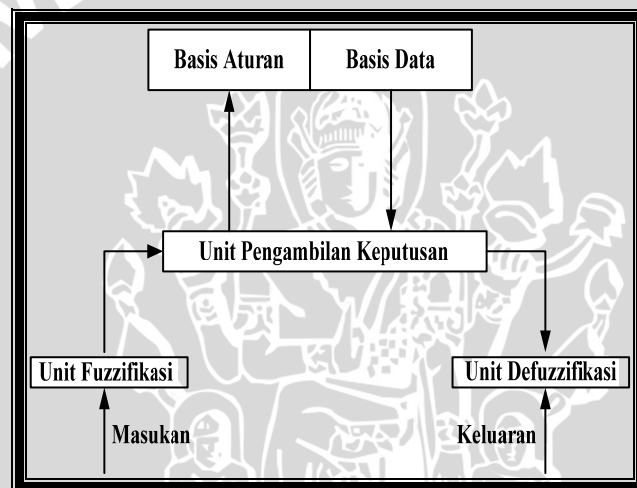
Elemen utama kontroler logika *fuzzy* terdiri dari unit fuzzifikasi, unit mekanisme pengambil keputusan, unit defuzzifikasi, dan unit basis pengetahuan. Basis pengetahuan terbagi dua yaitu basis data untuk menentukan fungsi



keanggotaan himpunan *fuzzy* dan basis aturan untuk memetakan nilai *fuzzy* dari masukan ke nilai keluaran. Variabel sistem yang digunakan ada dua yaitu variabel masukan (E dan CE) dan variabel keluaran (U). Struktur kontroler logika *fuzzy* dasar diperlihatkan dalam Gambar 3.17.

Jun Yan menentukan variabel masukan kontroler *fuzzy* akan diseleksi sebagai kesalahan (*error*, E) dan perubahan kesalahan (*change of error*, CE), keluaran merupakan variasi perintah yang disusun sebagai berikut:

- Jika keluaran sama dengan nilai yang diinginkan dan perubahan kesalahan sama dengan nol, maka keluaran yang sebelumnya dipertahankan.
- Jika keluaran tidak sesuai dengan yang diinginkan maka aksi kontrol tergantung pada nilai *error* dan perubahan *error*.



Gambar 3.17. Struktur Kontroler Logika *Fuzzy* Dasar
Sumber: Jun Yan, dkk., 1994: 47

Suatu pengontrolan pada dasarnya bertujuan untuk mengontrol *plant* agar keluaran sesuai dengan nilai referensi atau titik acuan sehingga *error* antara referensi dan nilai keluaran *plant* sama dengan nol.

Jika variabel E dan CE dibagi menjadi tujuh variabel yaitu Negatif Besar (NB), Negatif Sedang (NS), Negatif Kecil (NK), Sekitar Nol (SN), Positif Kecil (PK), Positif Sedang (PS), dan Positif Besar (PB), maka kaidah atur secara lengkap diperlihatkan dalam Tabel 2.1.

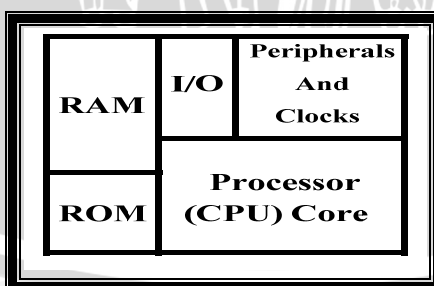
Tabel 3.1. Tabel Kaidah Atur Dalam Bentuk Matrik

	NB	NS	NK	SN	PK	PS	PB
PB	SN	PK	PS	PB	PB	PB	PB
PS	NK	SN	PK	PS	PB	PB	PB
PK	NS	NK	SN	PK	PS	PB	PB
SN	NB	BS	NK	SN	PK	PS	PB
NK	NB	NB	NS	NK	SN	PK	PS
NS	NB	NB	NB	NS	NK	SN	PK
NB	NB	NB	NB	NB	NS	NK	SN

Sumber: Reznik, Leonid, 1997: 82

3.7 Mikrokontroler AT89S52

Mikrokontroler merupakan komponen utama, karena komponen inilah yang akan mengontrol keseluruhan sistem agar dapat bekerja dengan baik dan optimal. Mikrokontroler terdiri atas tiga blok utama yaitu blok prosesor (CPU), memori, dan unit I/O. Selain tiga blok utama tersebut terdapat blok perangkat (*peripheral*) lainnya. Ketersediaan *peripheral-peripheral* ini di dalam MK, dapat mengurangi adanya perangkat eksternal sehingga memperkecil ukuran alat elektronik secara keseluruhan. Adapun arsitektur mikrokontroler secara keseluruhan dapat dilihat dalam Gambar 3.18.



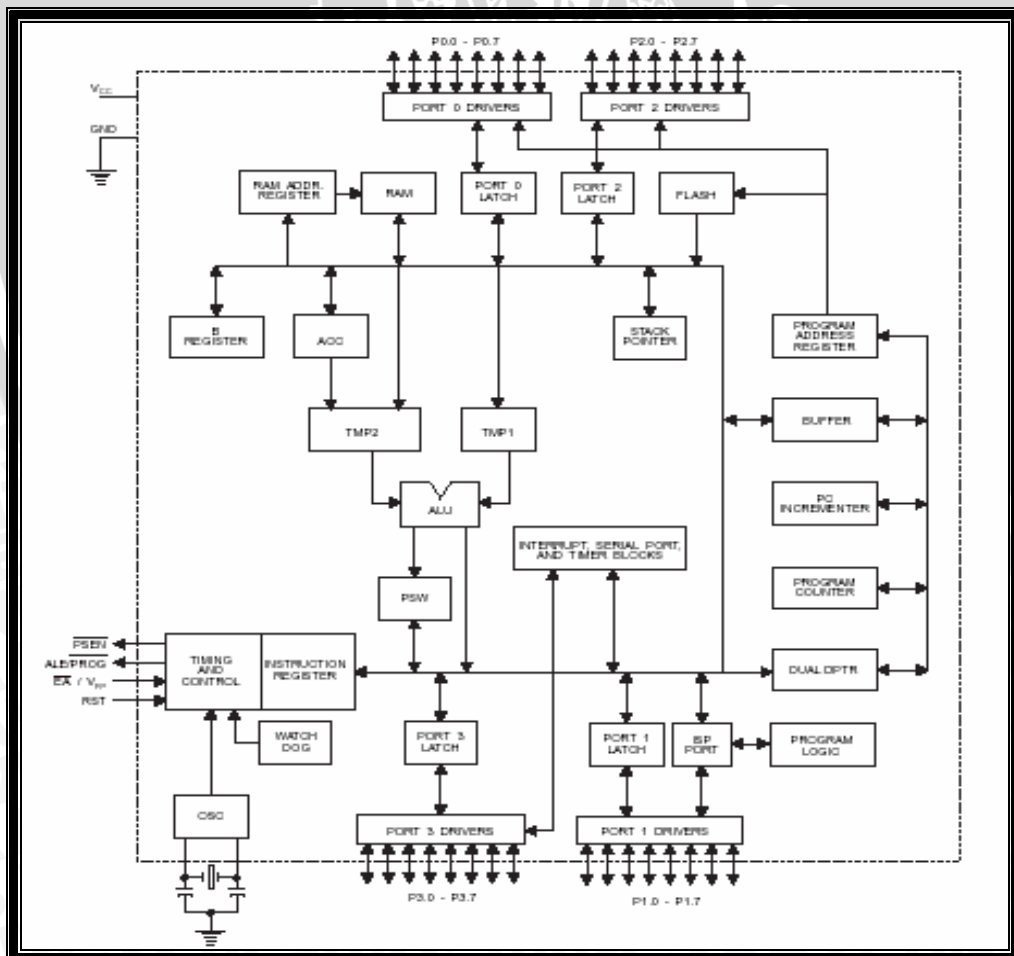
Gambar 3.18. Arsitektur Mikrokontroler
 Sumber: <http://www.atmel.com/89S52.pdf>

Secara umum arsitektur yang dimiliki oleh mikrokontroler AT89S52 adalah sebagai berikut:

- 8 bit CPU dengan *register A (accumulator)* dan *match register (register B)*.

- 16 bit *Program Counter (PC)* dan *Data Pointer (DPTR)* register.
- 8 bit *Program Status Word (PSW)* register, 8 bit *Stack Pointer*.
- Internal ROM dan EPROM dengan kapasitas 8 kbyte.
- Internal RAM dengan kapasitas 256 byte yang digunakan untuk 4 buah *register bank*, yang masing-masing terdiri dari 8 *register*, 16 *byte*, yang mana dapat dieksekusi pada masing-masing bit secara *independent (Bit Addesable)* dan sebagai memori *variable* 8 bit.
- 32 input/output yang disusun pada 4 *port (port 0 – port 3)*.
- 2 buah 16 bit *timer/counter*: T0 dan T1.
- *Full Duplex Serial Data Communication*: SBUF.
- *Control Register* :TCON, TMOD, PCON, IP dan IE.
- 2 eksternal interrupt dan 3 internal *interrupt*.
- Oscilator dan *Clock Circuit*.

Blok diagram dari mikrokontroller AT89S52 ditunjukkan dalam Gambar 3.19.

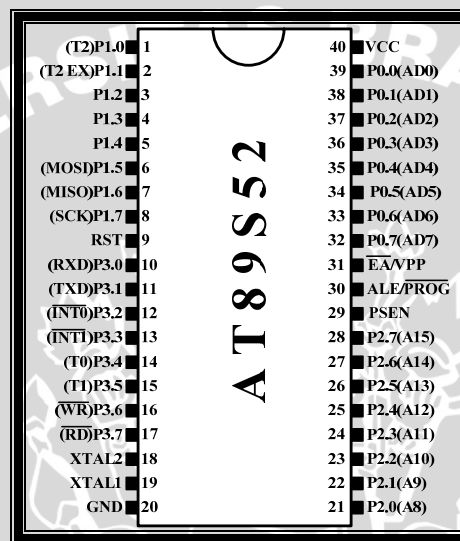


Gambar 3.19. Blok Diagram AT89S52

Sumber: ATMEL, AT89S52, Datasheet, 2005

3.7.1 Konfigurasi Pin

Masing-masing kaki atau pin dalam mikrokontroler AT89S52 mempunyai fungsi tersendiri. Dengan mengetahui fungsi dari masing-masing pin tersebut, akan lebih mudah dalam merencanakan dan membuat aplikasi sistem dengan mikrokontroler ini. Susunan pin dari mikrokontroler AT89S52 ditunjukkan dalam Gambar 3.20.

**Gambar 3.20.** Konfigurasi Pin dari AT89S52

Sumber: ATMEL, AT89S52, Datasheet, 2005

Adapun fungsi dari masing-masing pin dari AT89S52 adalah:

- Port 1 (Pin 1-8)

Port 1 merupakan sebuah *port I/O bidirectional* yang mempunyai *internal pull-up*.

- RST (Pin 9)

Pin ini dihubungkan dengan rangkaian *power on reset*, dan digunakan untuk mereset mikrokontroler AT89S52.

- Port 3 (Pin 10-17)

Port paralel 8 bit dua arah yang memiliki fungsi pengganti. Apabila fungsi pengganti tidak digunakan, maka pin ini dapat digunakan sebagai *port I/O* biasa. Fungsi pengganti ini dapat dilihat dalam Tabel 2.2.

- XTAL1 (Pin 18)

Merupakan masukan ke inverting amplifier osilator dan masukan pada operasi internal clock.

❑ XTAL2 (Pin 19)

Merupakan keluaran dari inverting amplifier osilator.

❑ Ground (Pin 20)

Merupakan pin yang dihubungkan dengan *ground* rangkaian.

❑ Port 2 (Pin 21-28)

Merupakan *port* I/O paralel 8 bit *bidirectional*, dapat digunakan sebagai *port* I/O biasa atau untuk mengirim *byte* alamat bila digunakan untuk mengakses memori eksternal.

❑ PSEN/Program Store Enable (Pin 29)

Program Store Enable adalah merupakan *strobe* keluaran yang dipergunakan untuk membaca program memori eksternal.

❑ ALE/PROG (Pin 30)

Pin *ALE* (aktif tinggi) digunakan untuk menahan alamat memori eksternal selama pelaksanaan instruksi. Sementara untuk *PROG* (aktif rendah) digunakan sebagai masukan pulsa program selama proses pemrograman EPROM berlangsung.

❑ EA/VPP (Pin 31)

Jika diberi logika tinggi, maka mikrokontroler akan melaksanakan instruksi dari ROM/EPROM. Bila diberikan logika rendah, mikrokontroler akan melaksanakan instruksi dari memori program luar.

❑ Port 0 (Pin 32-39)

Merupakan *port* I/O 8 bit *bidirectional* yang tidak mempunyai *internal pull-up*. *Port* ini dapat digunakan sebagai *port* I/O biasa dan dapat juga digunakan untuk memultiplek alamat dengan data ketika mengakses memori eksternal.

❑ V_{CC} (Pin 40)

Merupakan pin yang dihubungkan dengan sumber tegangan.

3.7.2 Struktur dan Operasi Port

Mikrokontroler AT89S52 memiliki 3 buah *port*. Setiap *port* memiliki 8 buah jalur I/O yang bersifat *bidirectional*. Beberapa karakteristik *port* mikrokontroler AT89S52 dijelaskan secara singkat berikut ini:

- ❑ *Port 0* dapat digunakan sebagai *port I/O* biasa, pada fungsi sebagai *input* maka dapat dilakukan dengan memberikan logika 1 pada *port* tersebut. Pada *port* ini berlaku ketentuan yang berbeda dengan *port* lainnya, yaitu bila digunakan sebagai keluaran harus diberikan tambahan resistor *pull-up*.
- ❑ *Port 1* merupakan *port I/O* biasa. *Port* ini mempunyai internal *pull-up* dan berfungsi sebagai *output* dengan memberikan logika 1. Sebagai *output port* ini dapat memberikan *output sink* ke empat buah *input* TTL. *Port 1* memiliki kemampuan untuk menyangga beban sampai dengan 20 mA dan dapat mengemudikan *LED* secara langsung.
- ❑ *Port 2* merupakan *port I/O* biasa. *Port* ini mempunyai internal *pull-up* dan berfungsi sebagai *output* dengan memberikan logika 1. Sebagai *output port* ini dapat memberikan *output sink* ke empat buah *input* TTL.
- ❑ *Port 3*, sebagai *I/O* biasa *port 3* mempunyai sifat yang sama dengan *port 1* maupun *port 2*. Sedangkan sebagai fungsi khusus *port-port* ini mempunyai penjelasan seperti yang terlihat dalam Tabel 2.2.

Tabel 3.2. Fungsi Khusus *Port 3*

Jalur	Fungsi Alternatif
P3.0	<i>Input</i> data serial RXD
P3.1	<i>Output</i> data serial TXD
P3.2	<i>Input</i> Interupsi Eksternal INT0
P3.3	<i>Input</i> Interupsi Eksternal INT1
P3.4	<i>Input</i> Counter T0
P3.5	<i>Input</i> Counter T1
P3.6	<i>External Data Memory Write Strobe</i>
P3.7	<i>External Data Memory Read Strobe</i>

Sumber: ATMEL, AT89S52, Datasheet, 2005

3.7.3 Register Fungsi Khusus

Register fungsi khusus (*Special Function Register*) terletak pada 128 byte bagian atas memori data internal. Wilayah SFR ini terletak pada alamat 80_H sampai FF_H . Register-register ini hanya dapat diakses dengan pengalamatan langsung, baik per bit maupun per byte.

SFR berisi register-register dengan fungsi tertentu. Masing-masing register seperti ditunjukkan dalam Tabel 2.2 yang meliputi simbol, nama, dan alamatnya serta keadaannya dalam nilai biner pada saat terjadi *power-on reset*. Beberapa kegunaan register fungsi khusus yang penting dijelaskan sebagai berikut:

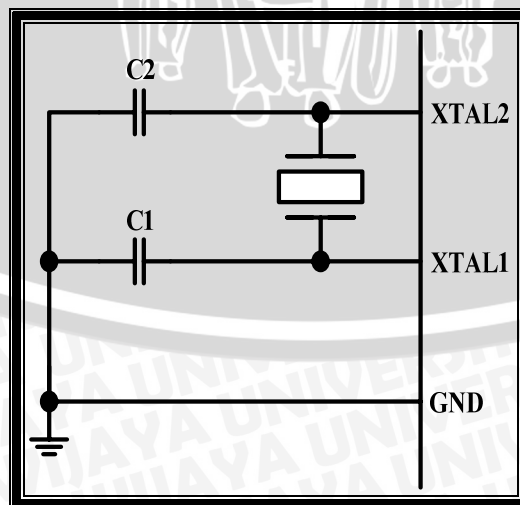
- ❑ *Accumulator* (ACC); merupakan register untuk penambahan dan pengurangan. Akumulator juga disebut sebagai register A.
- ❑ *Register B*; merupakan register yang berfungsi untuk melayani operasi perkalian dan pembagian.
- ❑ *PSW*; terdiri dari beberapa bit status yang menggambarkan kejadian di akumulator sebelumnya, yang pengkondisian keadaan akumulator tersebut melalui *flag register* yang terdiri dari *carry flag*, *auxiliary carry flag*, *parity flag*, *overflow flag*, dua bit pemilih bank, dan dua flag yang dapat didefinisikan sendiri oleh pemakai.
- ❑ *Control Register*; terdiri atas register yang mempunyai fungsi kontrol. Terdapat dua register khusus untuk mengontrol sistem interupsi, yaitu register *IP* (*Interrupt Priority*) dan register *IE* (*Interrupt Enable*). Untuk mengontrol pelayanan timer/counter terdapat register khusus, yaitu register *TMOD* (*Timer/counter Mode Control*) dan register *TCON* (*Timer/counter Control*).
- ❑ *Stack Pointer* (SP); merupakan register 8 bit yang diletakkan di alamat manapun pada RAM internal. Isi register ini ditambah sebelum data disimpan, selama instruksi *PUSH* dan *CALL*. Pada saat reset, register SP diinisialisasi pada alamat 07_H , sehingga stack akan dimulai pada lokasi 08_H .
- ❑ *Data Pointer* (DPTR); terdiri dari dua register, yaitu untuk byte tinggi (*Data Pointer High*) dan untuk byte rendah (*Data Pointer Low*). Fungsinya untuk

menahan alamat 16-bit. DPTR dapat dimanipulasi sebagai *register* 16-bit atau sebagai dua buah *register* 8-bit.

- ❑ *Port* 0 sampai *Port* 3 merupakan *register* yang berfungsi untuk membaca dan mengeluarkan data pada *port* 0, 1, 2, dan 3. Masing-masing *register* ini dapat dialamati per-bit maupun per-byte.
- ❑ *Serial Data Buffer* seharusnya merupakan dua buah *register* yang terdiri dari *transmitt buffer register* dan *receive buffer register*. Pada saat data dipindahkan ke SBUF, *register* tersebut akan menjadi *transmit buffer register* sedangkan pada saat data dipindahkan dari SBUF maka *register* tersebut akan berubah menjadi *receive buffer register*.
- ❑ *Timer Register*; terdiri dari pasangan *register* TH0, TL0 dan TH1, TL1 merupakan *register* 16 bit yang berfungsi sebagai *register counter* untuk *timer/counter* 0 dan *timer/counter* 1.

3.7.4 Osilator

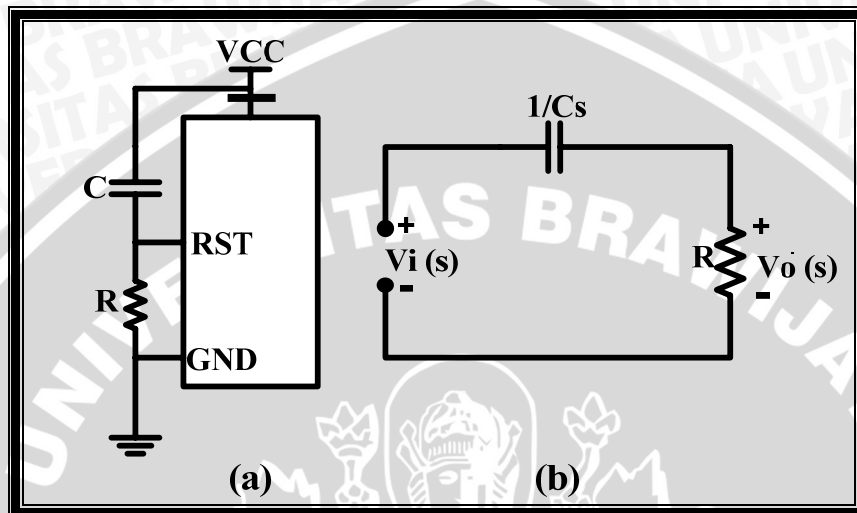
Mikrokontroler AT89S52 memiliki osilator internal (*on chip oscillator*) yang dapat digunakan sebagai sumber pewaktuan (*clock*) bagi CPU. Jika menggunakan osilator internal maka diperlukan sebuah kristal atau resonator keramik antara pin XTAL₁ dan pin XTAL₂ serta 2 buah kapasitor. Untuk kristalnya dapat digunakan dengan frekuensi antara 6 sampai 12 MHz, sedangkan kapasitor dapat bernilai dari 27 pF hingga 33 pF. Gambar 3.21 menunjukkan rangkaian osilator yang digunakan.



Gambar 3.21. Rangkaian Osilator
Sumber: ATMEL, AT89S52, Datasheet, 2005

3.7.5 Reset

Rangkaian *power on reset* diperlukan untuk mereset mikrokontroler secara otomatis setiap catu daya *on*. Gambar 3.22 menunjukkan rangkaian *power on reset*. Ketika catu daya diaktifkan, rangkaian *reset* menahan logika tinggi pin RST dengan jangkang waktu yang ditentukan oleh besarnya pengisian muatan C.



Gambar 3.22. (a) Rangkaian *power on reset* (b) Rangkaian ekivalen *power on reset*
Sumber: Perancangan

BAB IV

METODE PENELITIAN

Penyusunan skripsi ini diwujudkan dalam bentuk prototipe, yaitu perencanaan dan perealisasiannya agar dapat menampilkan unjuk kerja sesuai dengan yang direncanakan dengan mengacu pada rumusan masalah. Data dan spesifikasi komponen yang digunakan dalam perencanaan merupakan data sekunder yang diambil dari buku data komponen elektronika. Pemilihan komponen berdasarkan perencanaan dan disesuaikan dengan komponen yang ada di pasaran.

Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang akan dibuat secara umum adalah sebagai berikut:

- Perancangan sistem
- Pembuatan alat
- Pengujian alat
- Analisa data dan pengambilan kesimpulan

4.1 Perancangan Sistem

Perancangan sistem dilakukan sebagai langkah awal sebelum terbentuknya suatu sistem beserta rangkaian elektronik pendukungnya yang siap untuk direalisasikan. Hal ini dilakukan agar sistem yang dibuat dapat berjalan sebagaimana mestinya. Perancangan sistem yang dilakukan meliputi:

4.1.1 Perancangan Perangkat Keras (*Hardware*)

- Penentuan spesifikasi alat yang akan digunakan.
- Pembuatan blok diagram sistem secara lengkap, dengan tujuan untuk mempermudah pemahaman mengenai cara kerja alat yang akan dibuat.
- Perancangan dan pembuatan skema rangkaian.
- Penentuan dan perhitungan komponen perangkat keras yang akan digunakan. Adapun dalam pemilihan komponen tersebut berdasarkan pada komponen yang mudah didapatkan di pasaran lokal.

4.1.2 Perancangan Perangkat Lunak (*Software*)

Setelah perangkat keras dirancang, maka langkah selanjutnya adalah perancangan perangkat lunak. Perangkat lunak ini berfungsi untuk mengatur kinerja keseluruhan dari sistem yang terdiri dari beberapa perangkat keras sehingga sistem ini dapat bekerja dengan baik. Maka perancangan ini dimulai dengan membuat diagram alir dari subrutin program dan program utama.

4.2 Pembuatan Alat

Setelah perancangan sistem dilakukan, maka langkah selanjutnya adalah pembuatan alat yang meliputi:

4.2.1 Pembuatan Perangkat Keras (*Hardware*)

Adapun tahap pembuatan alat dimulai dengan pembuatan unit rangkaian per blok pada PCB. Selanjutnya tiap blok rangkaian dilakukan pengecekan apakah pemasangannya sudah sesuai. Kemudian dilakukan penggabungan dari tiap – tiap blok tadi menjadi satu blok rangkaian. Sebagai tahap terakhir dari pembuatan alat adalah perakitan komponen elektronika pada PCB yang telah jadi. Untuk pemasangan komponen diawali dengan memasang komponen pasif seperti jumper, resistor, kapasitor, dan soket IC. Kemudian dilanjutkan dengan pemasangan komponen aktif seperti IC.

4.2.2 Pembuatan Perangkat Lunak (*Software*)

Pembuatan perangkat lunak dilakukan dengan mengimplementasikan diagram alir dari program yang telah direncanakan, penulisan program dalam bahasa Assembler untuk kemudian dikonversi ke dalam bahasa mesin.

4.3 Pengujian Alat

Untuk memastikan bahwa sistem ini berjalan sesuai yang direncanakan maka perlu dilakukan pengujian alat meliputi perangkat keras (*hardware*) maupun perangkat lunak (*software*) yang dilakukan baik secara per blok rangkaian maupun keseluruhan sistem. Pengujian ini meliputi:

4.3.1 Pengujian Perangkat Keras (*Hardware*)

Pengujian perangkat keras dilakukan dengan tujuan untuk menyesuaikan nilai tegangan dan arus pada alat dengan cara pengukuran yang dibandingkan

dengan nilai tegangan dan arus yang diijinkan bekerja dalam komponen alat berdasarkan data sekunder komponen.

4.3.2 Pengujian Perangkat Lunak (*Software*)

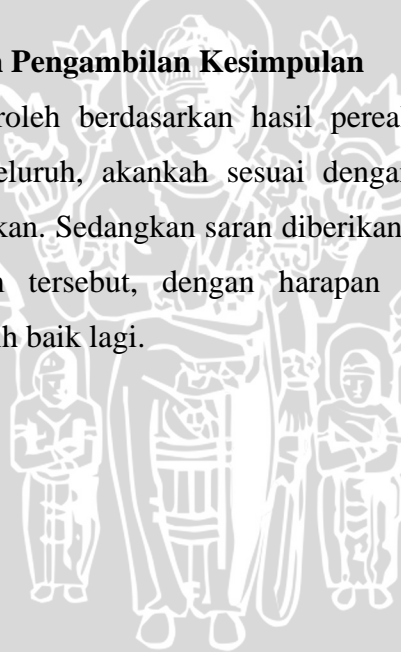
Pengujian perangkat lunak dilakukan dengan melakukan pengujian pada subrutin-subrutin program dengan memberi masukan dan mengamati keluarannya, maupun pada rangkaian yang telah dirakit secara lengkap.

4.3.3 Pengujian Keseluruhan Sistem

Setelah perangkat keras (*hardware*) dan perangkat lunak (*software*) diintegrasikan menjadi satu, kemudian dilakukan pengujian sistem secara keseluruhan dengan beban maupun tanpa beban. Tujuan pengujian sistem secara keseluruhan adalah untuk mengetahui unjuk kerja alat setelah perangkat keras dan perangkat lunak diintegrasikan bersama.

4.4 Analisa Data dan Pengambilan Kesimpulan

Kesimpulan diperoleh berdasarkan hasil perealisasiian dan pengujian pada sistem secara menyeluruh, akankah sesuai dengan tujuan dan rumusan masalah yang telah ditentukan. Sedangkan saran diberikan setelah melihat adanya kekurangan dalam sistem tersebut, dengan harapan sistem tersebut dapat dikembangkan menjadi lebih baik lagi.



ABAB V

PERENCANAAN DAN PEMBUATAN ALAT

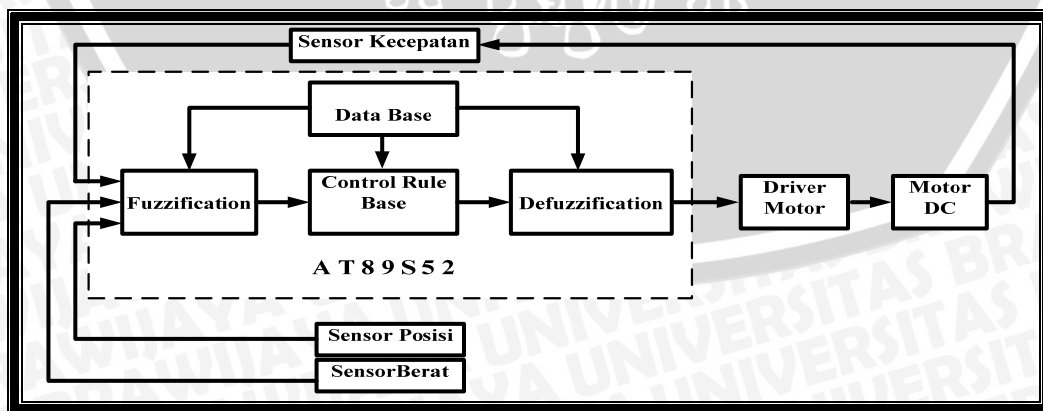
5.1 Gambaran Umum

Bab ini menjelaskan tentang proses perancangan dan pembuatan sistem pengaturan kecepatan dan berat pada prototipe robot pengangkut barang, dimana prototipe robot pengangkut barang tersebut mampu bergerak sesuai jalur pergerakannya. Sensor cahaya, sensor berat dan sensor kecepatan akan membaca keadaan pada lingkungan dan memberi masukan pada mikrokontroler AT89S52. Perancangan dilakukan dengan pembuatan blok diagram sistem agar dapat bekerja sesuai dengan perancangan sistem secara keseluruhan.

Perancangan sistem pengaturan kecepatan dan berat pada prototipe robot pengangkut barang ini terdiri dari perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*). Perancangan perangkat keras meliputi rangkaian sensor kecepatan, rangkaian sensor posisi, rangkaian sensor berat, rangkaian *driver* motor DC, sistem mikrokontroler AT89S52. Sedangkan perangkat lunaknya diperlukan untuk mengendalikan sistem mikrokontroler AT89S52 sehingga *hardware* bisa bekerja sesuai fungsinya.

Pada dasarnya perancangan pengaturan kecepatan dan berat pada prototipe robot pengangkut barang ini merupakan pengaturan kerja dari Motor DC untuk menghasilkan perubahan kecepatan sesuai dengan yang diinginkan.

5.2 Perancangan Hardware



Gambar 5. 1. Blok Diagram Sistem Kontrol Kecepatan, Posisi dan Berat

Sumber : Perancangan



Dari diagram blok diatas dilihat sistem ini memiliki bagian-bagian sebagai berikut:

1. Rangkaian sensor kecepatan.
2. Rangkaian sensor posisi
3. Rangkaian sensor berat.
4. Sistem Mikrokontroler.
5. Rangkaian driver motor.
6. Motor DC

5. 2. 1 Rangkaian Sensor Kecepatan

Pengendalian kecepatan pada prototipe robot pengangkut barang ini terdapat dua pengendalian, yaitu pengendalian kecepatan awal dan pengendalian kecepatan saat berbeban. Sensor kecepatan menggunakan *optocoupler*.

Sensor kecepatan terdiri *optocoupler* dan sebuah *plate* yang dipasang pada roda. Rangkaian *optocoupler* sebagai pembangkit dan penerima cahaya, sedangkan *plate* sebagai penghalang atau tidaknya cahaya dari LED. Rangkaian *optocoupler* juga berfungsi untuk mengkonversi putaran roda prototipe robot pengangkut barang menjadi pulsa-pulsa listrik. Kemudian pulsa-pulsa listrik ini akan diolah oleh mikrokontroler AT89S52 menjadi data kecepatan prototipe robot pengangkut barang.

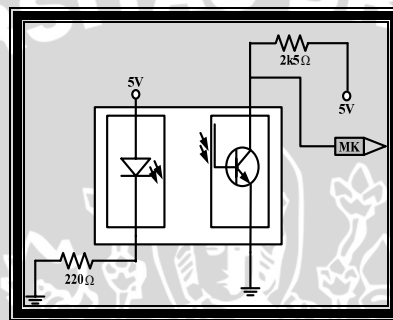
LED berfungsi sebagai pembangkit cahaya, sedangkan *phototransistor* sebagai sensor cahaya. Agar LED dapat menghasilkan cahaya yang diinginkan, maka perlu dilakukan pengaturan arus. Pengaturan arus dilakukan dengan menggunakan sebuah resistor yang terhubung seri dengan LED dan sumber tegangan. Berdasarkan dari *datasheet* Motorola H22A2, menunjukkan bahwa pada saat LED memancarkan cahaya tegangan antara anoda dan katoda sebesar 1,2 volt. Diinginkan arus yang mengalir pada LED sebesar 20 mA, maka dengan menggunakan persamaan 5.1, nilai resistansi R_S yang dibutuhkan adalah sebagai berikut:

$$R_S = \frac{V_{CC} - V_{LED}}{I_{LED}} = \frac{5 - 1,2}{20 \cdot 10^{-3}} = 190\Omega, R_S \approx 220\Omega \quad (5.1)$$

Untuk menghitung besarnya R_C dapat ditentukan melalui persamaan (5.2) sebagai berikut:

$$R_C = \frac{V_{CC} - V_{CE}}{I_C} = \frac{5 - 0,4}{0,0018} = 2555,55\Omega, R_C \approx 2k5\Omega \quad (5.2)$$

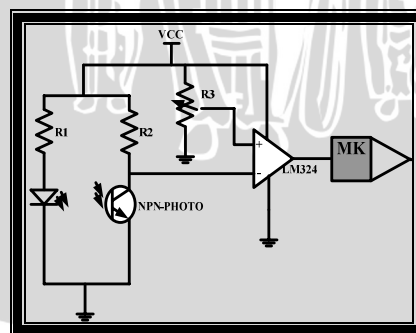
Dalam perancangan ini besarnya I_C tergantung pada besarnya I_f (dioda). Dengan I_f sebesar 20 mA, mak besarnya I_C adalah 1,8 mA dan besarnya V_{CE} adalah 0,4 mA. Sehingga pada saat cahaya dari LED diterima oleh *phototransistor*, diharapkan tegangan keluaran 5 volt. Saat cahaya dari LED tidak diterima oleh *phototransistor* karena terhalang oleh bagian yang tidak tembus cahaya dari *plate*, maka diharapkan tegangan keluaran 0 volt. Rangkaian *optocoupler* ditunjukkan dalam gambar 5.2.



Gambar 5.2. Rangkaian *Optocoupler*
Sumber: Perancangan

5.2.2 Rangkaian Sensor Posisi

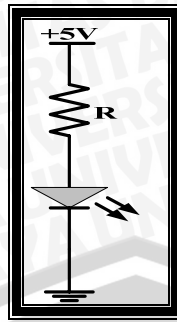
Pada tugas akhir ini sensor posisi adalah suatu penerapan dari rangkaian sensor cahaya, yang terdiri dari rangkaian pemancar cahaya dan rangkaian penerimanya. Pemancar yang digunakan adalah LED inframerah.



Gambar 5.3. Rangkaian Sensor Posisi
Sumber: Perancangan

5.2.2.1 Rangkaian Pemancar

Rangkaian pemancar (*transmitter*) berfungsi untuk memancarkan pulsa-pulsa listrik yang nantinya akan diterima oleh rangkaian penerima. Gambar 5.4 menunjukkan rangkaian pemancar inframerah.



Gambar 5.4. Rangkaian Pemancar
Sumber: Perancangan

Dalam perancangan pemancar inframerah, berdasarkan data *book* MLED 81, untuk membias maju LED inframerah dibutuhkan arus forward sebesar 100 mA (max). Dalam perencanaan digunakan $I_{LED} = 15 \text{ mA}$ dan tegangan LED (V_{LED}) = 1,35 volt, sehingga nilai resistor yang dipasang adalah:

$$R = \frac{V_{CC} - V_{LED}}{I_f \text{ (dioda)}}$$

$$R = \frac{(5 - 1,35)\text{volt}}{15\text{mA}} \quad (5.3)$$

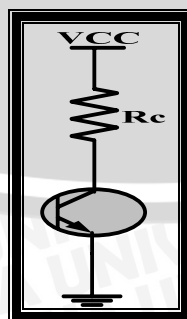
$$R = 243,33\Omega$$

$$R \approx 270\Omega$$

Jadi resistor yang dipasang adalah 270 ohm.

5.2.2.2 Rangkaian Penerima

Rangkaian penerima berfungsi untuk menerima pulsa-pulsa listrik dari pemancar inframerah yang nantinya diterima oleh komponen phototransistor dan mengubahnya ke dalam bentuk keluaran berupa tegangan. Gambar 5.5 menunjukkan rangkaian penerima inframerah.



Gambar 5.5. Rangkaian Penerima
Sumber: Perancangan

Dalam perancangan penerima inframerah digunakan phototransistor dengan data sebagai berikut:

$I_{C\ on}$ = minimal 4 mA dan maksimal 16 mA.

V_{ce} (inframerah) on > V_{ce} saturasi \geq 0,4 volt

Dalam perancangan digunakan $I_{C\ on}$ = 5 mA dan $V_{ce\ on}$ = 0,3 volt.

Sehingga resistor yang dipasang di kolektor (R_C inframerah) adalah:

$$R_C = \frac{V_{cc} - V_{ce\ on}}{I_C}$$

$$R_C = \frac{(5 - 0,3)V}{0,5\ mA}$$

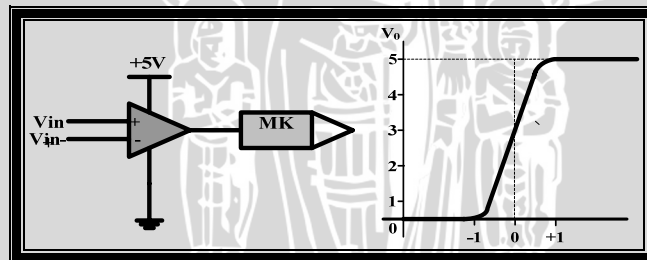
$$R_C = 9,4\ k\Omega$$

$$R_C \approx 10\ k\Omega$$
(5.4)

Sehingga resistor yang dipasang adalah 10k Ω .

5.2.2.3 Rangkaian Pembanding (Komparator)

Rangkaian pembanding atau komparator berfungsi sebagai penguat, oleh karena itu keluaran tidak berbanding lurus dengan masukan. Keluaran komparator hanya mempunyai dua nilai, misalnya 0V dan 5V saja, selain itu komparator mempunyai dua masukan, yaitu masukan membalik (-) dan tak membalik (+). Skema komparator dan fungsi alih pembanding ditunjukkan pada Gambar 5.6.

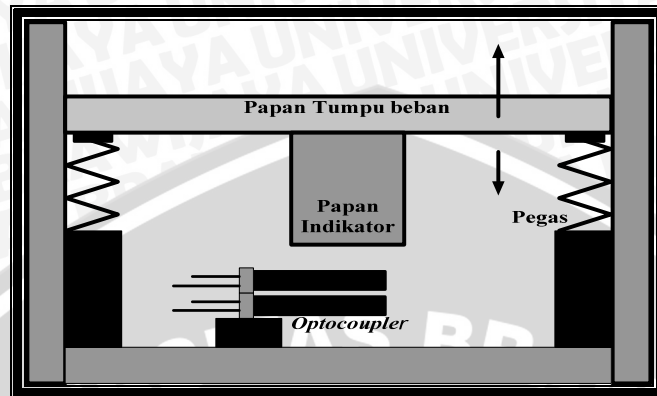


Gambar 5.6. Skema Komparator dan Grafik Keluaran Komparator
Sumber: Perancangan

5.2.3 Sensor Berat

Sensor berat yang digunakan dalam perancangan ini adalah *optocoupler* dan papan beban sebagai tempat tumpuan beban, kemudian dibawah papan tempat tumpuan beban dipasang *plate* yang berfungsi sebagai indikator dari berat. Pada setiap sudut papan beban dipasang empat pegas yang berfungsi sebagai naik turunnya atau sebagai pengatur perubahan massa beban. Perubahan massa beban akan menggeser posisi papan beban naik turun sesuai massa beban. Perubahan posisi naik turun dari papan beban akan menggeser posisi *plate* indikator. Dengan

adanya perubahan posisi *plate* indikator ini, maka nilai berat beban akan berubah sesuai dengan perubahan logika 0 dan 1. Perancangan mekanik sensor dapat dilihat dalam Gambar 5.7.



Gambar 5.7. Perancangan Secara Mekanik
Sumber: Perancangan

Tabel 5.1 Kondisi *Optocoupler* Disetiap Berat

Berat	Kondisi Optocoupler 1	Kondisi Optocoupler 2
Ringan	Tidak Terhalangi	Tidak Terhalangi
Sedang	Terhalangi	Tidak Terhalangi
Berat	Terhalangi	Terhalangi

Pada perancangan prototipe robot pengangkut barang, untuk rangkaian sensor berat, model rangkaian dan nilai pada setiap komponen sama dengan model rangkaian pada sensor kecepatan, yang membedakan adalah fungsi daripada masing-masing sensor tersebut.

Sensor berat terdiri *optocoupler* dan sebuah *plate* yang dipasang dibawah papan tempat tumpuan beban. Rangkaian *optocoupler* sebagai pembangkit dan penerima cahaya, sedangkan *plate* sebagai penghalang atau tidaknya cahaya dari LED.

LED berfungsi sebagai pembangkit cahaya, sedangkan *phototransistor* sebagai sensor cahaya. Agar LED dapat menghasilkan cahaya yang diinginkan, maka perlu dilakukan pengaturan arus. Pengaturan arus dilakukan dengan menggunakan sebuah resistor yang terhubung seri dengan LED dan sumber tegangan. Berdasarkan dari *datasheet* Motorola H22A2, menunjukkan bahwa pada saat LED memancarkan cahaya tegangan antara anoda dan katoda sebesar 1,2 volt. Diinginkan arus yang mengalir pada LED sebesar 20 mA, maka dengan

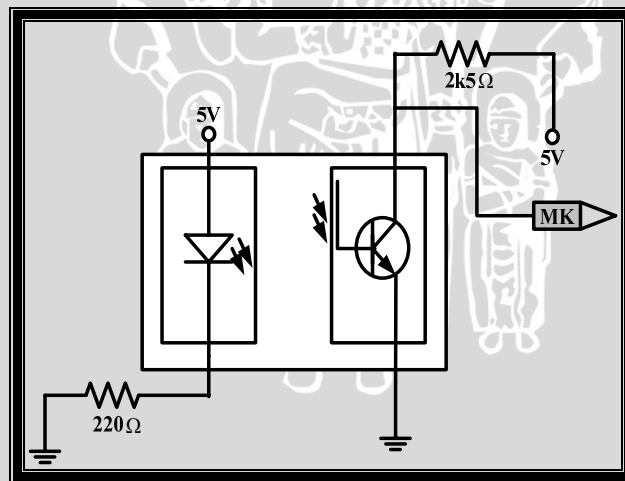
menggunakan persamaan 5.5, nilai resistansi R_S yang dibutuhkan adalah sebagai berikut:

$$R_S = \frac{V_{CC} - V_{LED}}{I_{LED}} = \frac{5 - 1,2}{20 \cdot 10^{-3}} = 190\Omega, R_S \approx 220\Omega \quad (5.5)$$

Untuk menghitung besarnya R_C dapat ditentukan melalui persamaan (5.6) sebagai berikut:

$$R_C = \frac{V_{CC} - V_{CE}}{I_C} = \frac{5 - 0,4}{0,0018} = 2555,55\Omega, R_C \approx 2k5\Omega \quad (5.6)$$

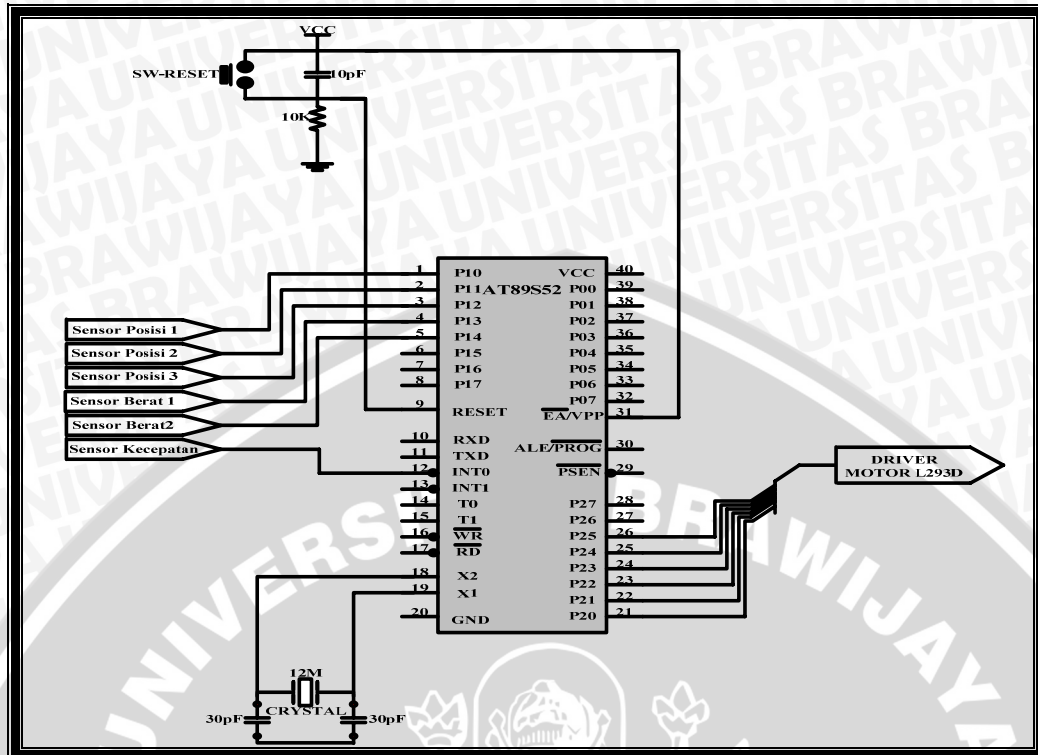
Dalam perancangan ini besarnya I_C tergantung pada besarnya I_f (dioda). Dengan I_f sebesar 20 mA, mak besarnya I_C adalah 1,8 mA dan besarnya V_{CE} adalah 0,4 mA. Sehingga pada saat cahaya dari LED diterima oleh *phototransistor*, diharapkan tegangan keluaran 5 volt. Saat cahaya dari LED tidak diterima oleh *phototransistor* karena terhalang oleh bagian yang tidak tembus cahaya dari *plate*, maka diharapkan tegangan keluaran 0 volt. Rangkaian *optocoupler* untuk sensor berat ditunjukkan dalam gambar 5.8.



Gambar 5.8. Rangkaian *Optocoupler* dari Sensor Berat
Sumber: Perancangan

5.2.4 Sistem Mikrokontroler AT89S52

Pada alat ini digunakan mikrokontroler sebagai pusat pengolahan data dan pengendali peralatan lainnya. Gambar 5.9 adalah minimum sistem mikrokontroler AT89S52.



Gambar 5.9. Minimum Sistem Mikrokontroler AT89S52
Sumber : Perancangan

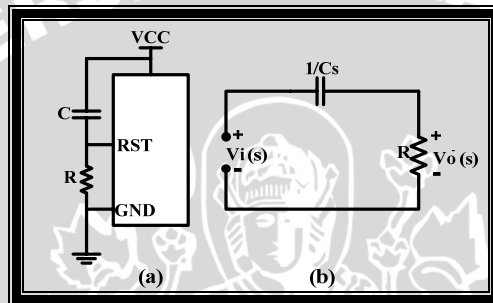
Mikrokontroler AT89S52 mempunyai 4 port, 32 jalur yang dapat diprogramkan menjadi masukan atau keluaran, pada perancangan ini pin-pin yang digunakan adalah :

- P0.0 : digunakan untuk indikator kecepatan
- P1.0 : dihubungkan dengan sensor posisi 1
- P1.1 : dihubungkan dengan sensor posisi 2
- P1.2 : dihubungkan dengan sensor posisi 3
- P1.3 : dihubungkan dengan sensor berat 1
- P1.4 : dihubungkan dengan sensor berat 2
- INT0 : dihubungkan dengan sensor kecepatan
- Port 2 : digunakan untuk jalur data keluaran dari mikrokontroler menuju driver motor

5.2.4.1 Power On Reset

Rangkaian *power on reset* diperlukan untuk mereset mikrokontroler secara otomatis setiap kali catu daya dinyalakan, hal ini akan me-*reset program counter* sehingga perintah program yang dieksekusi dimulai pada alamat 0. Ketika catu

daya diaktifkan, rangkaian *power on reset* menahan logika tinggi pin RST dengan jangka waktu yang ditentukan oleh lamanya pengisian muatan C. Jika pin RST diberi logika *high* selama dua siklus mesin maka mikrokontroler akan direset. Satu siklus mesin ($\frac{12}{f.osc}$) membutuhkan waktu 1,085 μs , sehingga untuk mereset mikrokontroler dibutuhkan waktu minimal 2,17 μs . Tegangan RST yang diijinkan sebesar $0,7 V_{cc} - V_{cc}+0,5$. Jika tegangan RST minimal 3,5 V maka tegangan pada kapasitor maksimal 1,5 V untuk dapat mereset mikrokontroler. Rangkaian *power on reset* dibentuk dengan rangkaian RC yang dapat dilihat dalam Gambar 5.10.



Gambar 5.10. Rangkaian Power On Reset
Sumber : Perancangan

Dari rangkaian setara diperoleh:

$$V_o = \frac{R}{R + \frac{1}{sC}} \cdot V_i \tag{5.7}$$

$$V_o = \frac{sCR}{sCR + 1} \cdot V_i \tag{5.8}$$

dengan tegangan V_i adalah V_{cc} yaitu 5V, dalam fungsi Laplace adalah $\frac{5}{s}$ sehingga:

$$V_o = \frac{5}{s} \cdot \frac{sCR}{sCR + 1} = 5 \cdot \frac{CR}{sCR + 1} = 5 \cdot \left[\frac{1}{s + \frac{1}{RC}} \right] \tag{5.9}$$

$$V_o = 5 \cdot e^{-\frac{t}{RC}} \tag{5.10}$$

$$\frac{5}{V_o} = e^{\frac{t}{RC}} \tag{5.11}$$

$$\ln \frac{5}{V_o} = \frac{t}{RC} \tag{5.12}$$

$$t = RC \left[\ln \frac{5}{V_o} \right] \tag{5.13}$$

dengan V_o adalah tegangan logika nominal yang diijinkan pin RST dimana

$$V_o = 0,7 \times V_{cc} = 0,7 \times 5 \text{ volt} = 3,5 \text{ volt}$$

sehingga

$$t = RC \left[\ln \frac{5}{3,5} \right] \tag{5.14}$$

$$t = 0,357.R.C \tag{5.15}$$

dengan menggunakan pemisalan $R = 100K$ dan t minimum adalah $2,17 \mu s$ maka untuk amannya dimisalkan $= 4 \mu s$ maka:

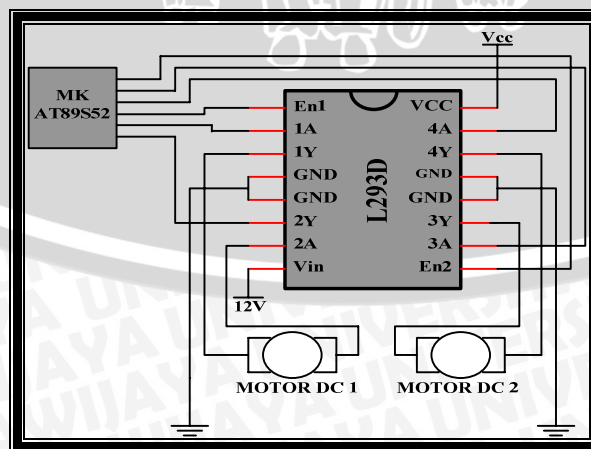
$$4.10^{-6} = 0,357 \times 100 K \times C$$

$$C = 0.1 \text{ nF}$$

telah memenuhi syarat minimal untuk dapat mereset mikrokontroler AT89S52.

5.2.5 Rangkaian Driver Motor DC

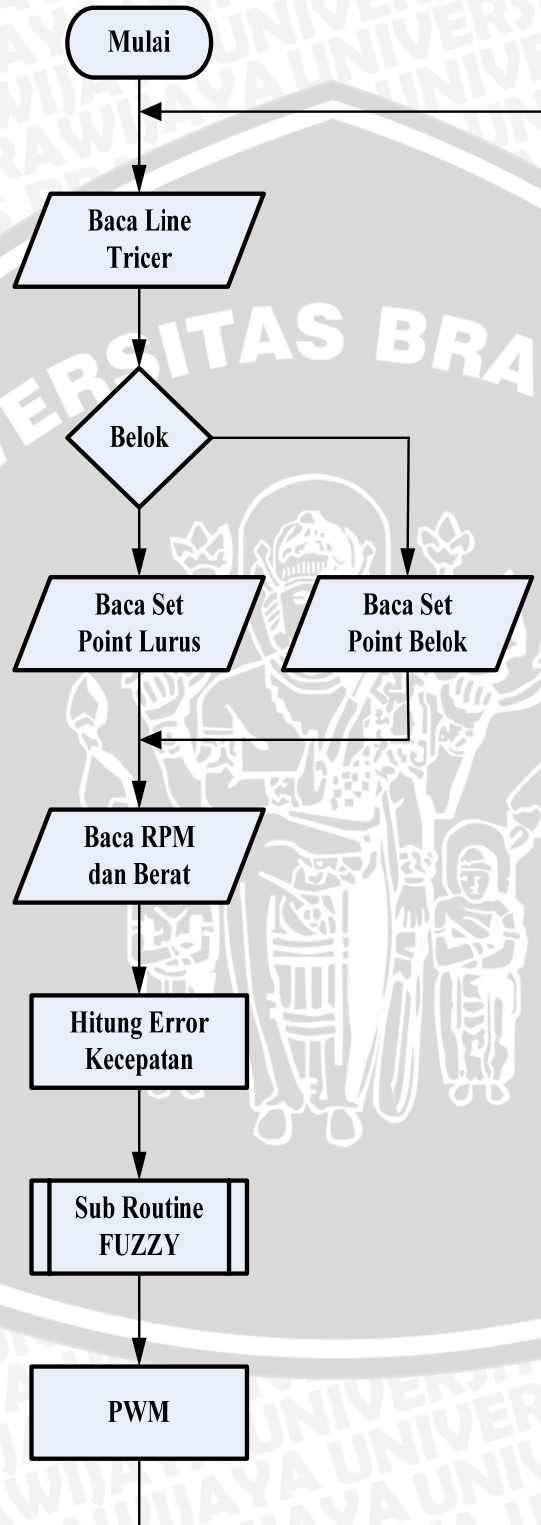
Rangkaian driver motor DC diperlukan untuk menggerakkan motor DC sesuai dengan perencanaan. Rangkaian driver motor DC ini hanya sebuah IC L293D yang mempunyai tegangan tinggi yang terintegrasi monolitis, dan mempunyai empat saluran *vriver*. Driver motor ini baik digunakan untuk menswitch aplikasi pada frekuensi sampai 5 KHZ. Gambar 5.11 menunjukkan rangkaian driver motor DC.



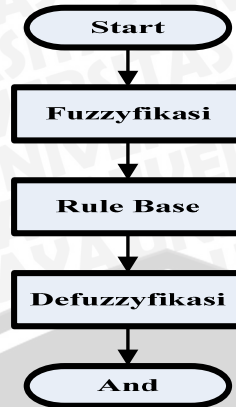
Gambar 5.11. Rangkaian Driver Motor DC
Sumber : Perancangan

5.3 Perancangan Software

Diagram alir perancangan software secara garis besar dapat dilihat pada Gambar 5.12 dan 5.13.



Gambar 5.12. Diagram Alir Pemrograman *Fuzzy* Secara Umum
 Sumber : Perancangan



Gambar 5.13. Diagram Alir Routine *Fuzzy* Secara Umum
Sumber : Perancangan

Program ini dirancang agar mikrokontroler mampu membaca *input* (masukan) dari sensor posisi, sensor kecepatan dan sensor berat yang kemudian akan dibentuk himpunan keanggotaan dari ketiga masukan tersebut atau yang sering disebut dengan *Fuzzyfikasi* untuk data dari sensor posisi, sensor kecepatan dan sensor berat ini. Setelah proses *Fuzzyfikasi* terlaksana, mikrokontroler akan melakukan *inferensi rule*. Aturan-aturan yang digunakan disimpan sebagai *data base*. Hasil dari *inferensi rules* akan di-*defuzzyfikasi* sebelum dikeluarkan sebagai keputusan.

5.3.1 Perancangan Kontroler Logika *Fuzzy*

Tahapan-tahapan dalam kontrol logika *Fuzzy* adalah sebagai berikut:

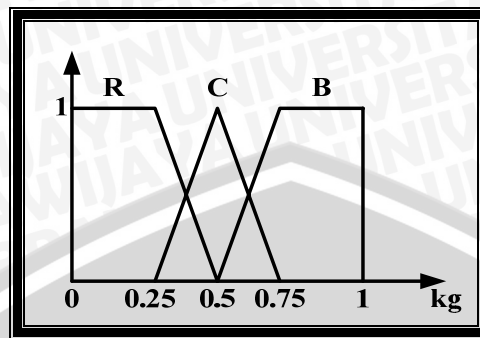
- Menentukan variabel masukan dan variabel keluaran .
Variabel masukan untuk kontroler logika *Fuzzy* adalah berat dan kecepatan, sedangkan variabel kelurannya adalah sinyal kontrol yang akan mempengaruhi besarnya arus pada motor DC.
- Menentukan jenis fungsi keanggotaan (*membership function*) masukan dan keluaran serta nilai label.

Untuk pengendalian kecepatan prototipe robot pengangkut barang ini terdapat dua pengendalian, yaitu pengendalian kecepatan awal dengan beban dan pengendalian kecepatan saat belok dengan beban. Fungsi keanggotaan untuk masukan berbentuk *trapezoid* dan *triangle* sedangkan untuk fungsi keanggotaan keluaran berbentuk *singleton*.

- Fungsi kenggotaan untuk pengendalian kecepatan awal.

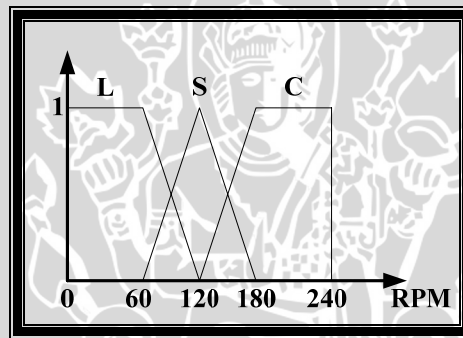
Fungsi kenggotaan masukan untuk pengendalian kecepatan awal adalah berat dan kecepatan. Fungsi keanggotaan untuk berat mempunyai tiga

label, yaitu ringan (R), cukup (C), dan berat (B). Fungsi keanggotaan berat ditunjukkan dalam Gambar 5.14



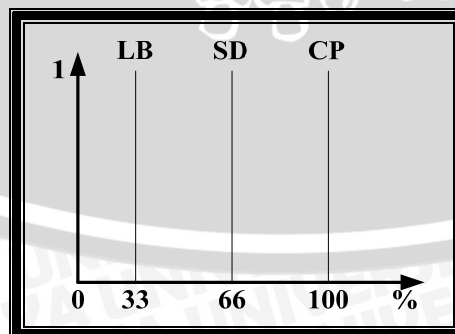
Gambar 5.14. Fungsi Keanggotaan *Crisp Input* Berat
 Sumber : Perancangan

Fungsi keanggotaan untuk kecepatan mempunyai tiga label, yaitu lambat (L), sedang (S), dan cepat (C). Fungsi keanggotaan kecepatan ditunjukkan dalam Gambar 5.15



Gambar 5.15. Fungsi Keanggotaan *Crisp Input* Kecepatan
 Sumber : Perancangan

Fungsi keanggotaan keluaran mempunyai 3 label, yaitu lambat (LB), sedang (SD), dan cepat (CP). Fungsi keanggotaan keluaran ditunjukkan dalam Gambar 5.16

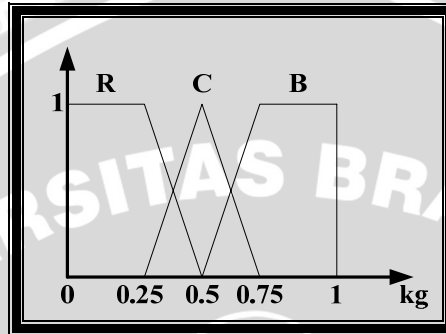


Gambar 5.16. Fungsi Keanggotaan *Crisp Output* Duty cycle
 Sumber : Perancangan

2. Fungsi keanggotaan untuk pengendalian saat berbelok.

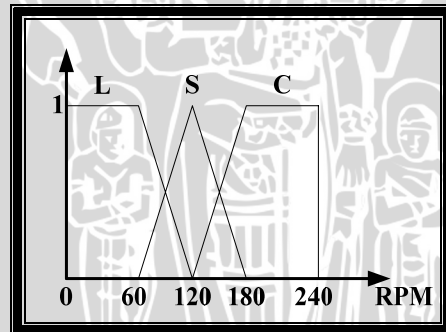
Fungsi keanggotaan masukan untuk pengendalian pada waktu belokan adalah berat dan kecepatan.

Fungsi keanggotaan untuk berat mempunyai tiga label, yaitu ringan (R), cukup (C), dan berat (B). Fungsi keanggotaan berat ditunjukkan dalam Gambar 5.17.



Gambar 5.17. Fungsi Keanggotaan *Crisp Input* Berat
 Sumber : Perancangan

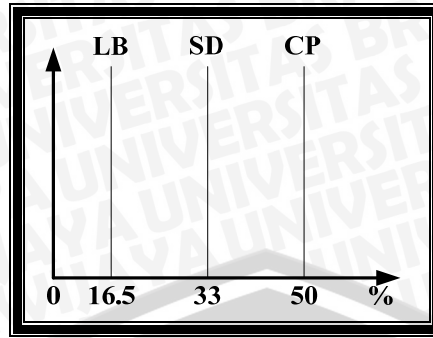
Fungsi keanggotaan untuk kecepatan mempunyai tiga label, yaitu lambat (L), sedang (S), dan cepat (C). Fungsi keanggotaan kecepatan ditunjukkan dalam Gambar 5.18.



Gambar 5.18. Fungsi Keanggotaan *Crisp Input* Kecepatan
 Sumber : Perancangan

Fungsi keanggotaan keluaran adalah *duty cycle* yang mempunyai 3 label, yaitu lambat (LB), sedang (S), dan cepat (C). Fungsi keanggotaan *duty cycle* ditunjukkan dalam Gambar 5.19.





Gambar 5.19. Fungsi Keanggotaan *Crisp Output Duty cycle*
 Sumber : Perancangan

Langkah pertama yang dilakukan penentuan batas-batas tiap anggota himpunan untuk kecepatan, seperti berikut ini :

- $0 \text{ rpm} \leq 120 \text{ rpm}$ masuk dalam himpunan lambat (L)
 - $60 < \text{kecepatan} \leq 180 \text{ rpm}$ masuk dalam himpunan sedang (S)
 - $120 < \text{kecepatan} \leq 240 \text{ rpm}$ masuk dalam himpunan cepat (C)
- set point* yang diinginkan adalah 120 rpm, karena itu nilai 120 rpm diletakkan ditengah himpunan (S).
- $0 \text{ kg} \leq 0.5 \text{ kg}$ masuk dalam himpunan ringan (R)
 - $0.25 \text{ kg} < \text{berat} \leq 0.75 \text{ kg}$ masuk dalam himpunan cukup (C)
 - $0.5 \text{ kg} < \text{berat} \leq 1 \text{ kg}$ masuk dalam himpunan berat (B)

5.3.1.1 Fuzzifikasi

Fuzzifikasi adalah proses mengubah nilai masukan (*crisp input*) menjadi input *fuzzi*. Pada masing-masing label terbagi menjadi empat bagian yaitu *Point 1*, *Slope 1*, *Point 2*, *Slope 2* seperti ditunjukkan dalam Gambar. Apabila *crisp input* terletak pada segment 1 dari suatu label ($Point 1 \leq Input < Point 2$), maka derajat keanggotaan *crisp input* pada label tersebut dinyatakan sebagai berikut:

$$\mu = (Input - Point1) \times Slope1 \tag{5.16}$$

Sedangkan apabila *crisp input* terletak pada segment 2 dari suatu label ($input \geq Point 2$), maka derajat keanggotaan *crisp input* pada label tersebut dinyatakan sebagai berikut :

$$\mu = FF - (Input - Point2) \times Slope2 \tag{5.17}$$

5.3.1.2 Kaidah atur kontroler logika *Fuzzy*

Kaidah atur kontroler logika *Fuzzy* didasarkan pada pengalaman dan dalam bentuk Jika-Maka. Setelah *crisp input* diubah menjadi *input Fuzzy*, selanjutnya diolah sesuai dengan kaidah aturnya. Karena masing-masing masukkan dibagi menjadi tiga keanggotaan maka *rules* disusun sebanyak 9 *rule* seperti berikut :

Tabel 5. 2 Tabel Kaidah pengaturan Kecepatan dan ketika belok

KECEPATAN	LAMBAT	SEDANG	CEPAT
BERAT			
RINGAN	SEDANG	LAMBAT	LAMBAT
CUKUP	CEPAT	SEDANG	LAMBAT
BERAT	CEPAT	CEPAT	SEDANG

If kecepatan is L and berat is R then duty cycle is sedang (S) (rule 1)

If kecepatan is S and berat is R then duty cycle is lambat (L) (rule 2)

If kecepatan is C and berat is R then duty cycle is lambat (S) (rule 3)

If kecepatan is L and berat is C then duty cycle is cepat (C) (rule 4)

If kecepatan is S and berat is C then duty cycle is sedang (S) (rule 5)

If kecepatan is C and berat is C then duty cycle is lambat (L) (rule 6)

If kecepatan is L and berat is B then duty cycle is cepat (C) (rule 7)

If kecepatan is S and berat is B then duty cycle is cepat (C) (rule 8)

If kecepatan is C and berat is B then duty cycle is Sedang (S) (rule 9)

Pada proses *reasoning*, nilai-nilai derajat keanggotaan (μ) yang diperoleh akan dibandingkan satu dengan yang lainnya. Karena proses ini menggunakan metode *MAX-MIN* jadi nilai derajat keanggotaan (μ) untuk aksi yang dipilih adalah nilai μ yang paling kecil.

5.3.1.3 Defuzzifikasi

Defuzzifikasi adalah proses untuk merubah *output fuzzy* menjadi *crisp output*. Hasil *defuzzifikasi* inilah yang digunakan untuk mengatur besarnya tegangan pada motor DC. Metode *defuzzifikasi* yang digunakan adalah COA (*center of Area*).

$$v_o = \frac{\sum_{k=1}^m v_k \mu_v(v_k)}{\sum_{k=1}^m \mu_v(v_k)}$$

v_o = nilai keluaran

m = tingkat kuantisasi

v_k = elemen ke- k

$\mu_v(v_k)$ = derajat keanggotaan elemen-elemen pada fuzzy set v

V = semesta pembicaraan



BAB VI PENGUJIAN ALAT

6.1 Pengujian Perangkat Keras

6.1.1 Pengujian Sensor Kecepatan

a. Tujuan

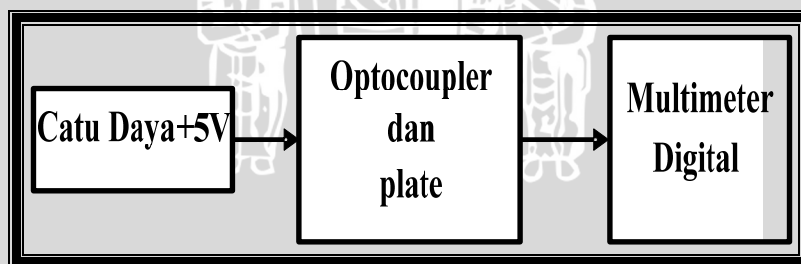
Mengetahui level tegangan sinyal listrik oleh rangkaian *optocoupler* pada saat cahaya yang dipancarkan LED terhalang atau tidak terhalang oleh bagian *plate*.

b. Peralatan

- rangkaian *optocoupler*
- catu daya + 5 volt
- multimeter digital

c. Langkah Pengujian

- merangkai peralatan seperti dalam Gambar 6.1. Keluaran rangkaian terhubung dengan multimeter digital.
- mengaktifkan catu daya.
- memutar roda penggerak prototipe robot pengangkut barang.
- mencatat tegangan yang tampak pada multimeter digital saat cahaya yang dipancarkan LED terhalang atau tidak terhalang.



Gambar 6.1. Blok Diagram Pengujian Sensor Kecepatan

d. Hasil Pengujian dan Analisis

Tabel 6.1 Hasil Pengujian Sensor Kecepatan

Cahaya LED	Tegangan pada multimeter digital (volt)	Logika
Terhalang	0,14	0
Tidak Terhalang	4,33	1



Dari Tabel 6.1 terlihat bahwa pada saat terhalang sensor kecepatan akan memberi logika “1” sedangkan pada saat tidak terhalang akan memberi logika “0”. Sensor kecepatan ini dapat digunakan dengan baik sampai pada kecepatan 900 rpm. Sehingga sensor kecepatan ini bisa dikatakan berjalan dengan baik pada aplikasi yang direncanakan.

6.1.2 Pengujian Sensor Berat

a. Tujuan

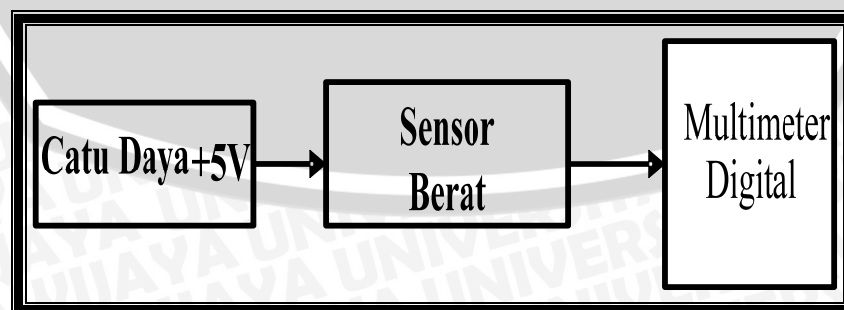
Mengetahui tegangan yang dihasilkan oleh rangkaian sensor berat pada saat diberikan beban dari 0 kg sampai 1 kg.

b. Peralatan

- rangkaian sensor berat.
- catu daya +5 volt
- multimeter digital
- timbel timbangan

c. Langkah Pengujian.

- merangkai peralatan seperti dalam Gambar 6.2. *output optocoupler* 1dihubungkan pada multimeter digital, sedangkan input dengan catu daya 5V
- mengaktifkan catu daya
- menaruh timbel timbangan di atas sensor berat, mulai dari 0 kg sampai 1 kg dengan kenaikan beban 0,25 kg.
- mencatat tegangan yang tampak pada multimeter digital untuk pengukuran tiap-tiap level beban. Pengukuran tiap-tiap level beban dilakukan sebanyak enam kali.



Gambar 6.2. Blok Diagram Pengujian Sensor Berat

d. Hasil Pengujian dan Analisis

Dari tabel 6.2 dapat disimpulkan bahwa sensor berat ini dapat bekerja dengan baik.

Tabel 6.2 Hasil Pengujian Sensor Berat

Berat (kg)	Optocoupler 1		Optocoupler 2	
	Tegangan (Volt)	Logika	Tegangan (Volt)	Logika
0	4,33	1	4,33	1
0,25	0,14	0	4,33	1
0,5	0,14	0	4,33	1
0,75	0,14	0	0,14	0
1	0,14	0	0,14	0

6.1.3 Pengujian Sensor Posisi

a. Tujuan

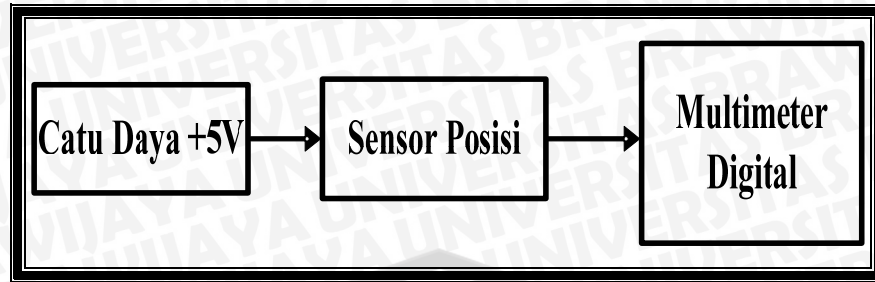
Mengetahui level tegangan sinyal listrik oleh rangkaian sensor posisi pada saat rangkaian penerima menerima atau tidak menerima sinar infrared yang dipancarkan oleh rangkaian pemancar.

b. Peralatan

- rangkaian sensor posisi, yang terdiri dari rangkaian pemancar dan penerima
- catu daya + 5 volt
- multimeter digital

c. Langkah Pengujian

- merangkai peralatan seperti dalam Gambar 6.3. Keluaran rangkaian terhubung dengan multimeter digital.
- mengaktifkan catu daya.
- mencatat tegangan yang tampak pada multimeter digital saat rangkaian penerima menerima cahaya infrared dan pada saat rangkaian penerima tidak menerima cahaya infrared.



Gambar 6.3 Blok Diagram Pengujian Sensor Posisi

d. Hasil Pengujian dan Analisis

Tabel 6.3 Hasil Pengujian Sensor Posisi

Permukaan Lintasan	Tegangan pada multimeter digital (volt)	Logika
Gelap	0	0
Terang	4,33	1

Dari tabel 6.3 terlihat bahwa pada saat lintasan terang sensor posisi akan memberikan logika “1” sedangkan pada saat lintasan gelap akan memberikan logika “0”. Sehingga sensor kecepatan ini bisa dikatakan berjalan dengan baik pada aplikasi yang direncanakan.

6.1.4 Pengujian Driver Motor

a. Tujuan

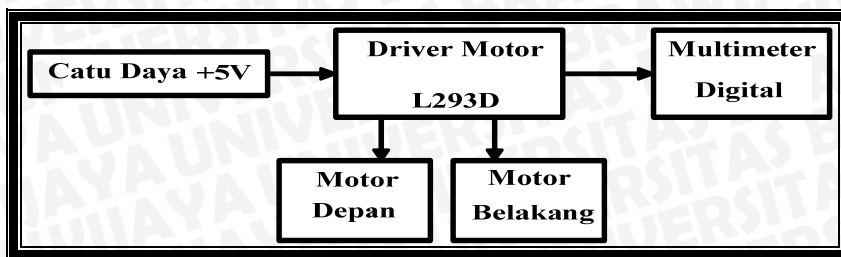
Mengetahui fungsi kerja driver motor L293D dalam mengatur putaran motor sesuai dengan tegangan yang diberikan pada driver motor.

b. Peralatan

- rangkaian driver motor L293D, yang berbentuk IC
- catu daya + 5 volt
- multimeter digital

c. Langkah Pengujian

- merangkai peralatan seperti dalam Gambar 6.4. Keluaran rangkaian terhubung dengan multimeter digital.
- mengaktifkan catu daya.
- mencatat tegangan yang tampak pada multimeter digital dan mengamati keadaan motor pada setiap perubahan tegangan



Gambar 6.4. Blok Diagram Pengujian Driver Motor

d. Hasil Pengujian dan Analisis

Tabel 6.4 Hasil Pengujian Driver motor Depan

Input A1	Input A2	Output B1	Output B2	Roda Depan
		Vcc = 9V	Vcc = 9V	
0 V	0 V	0,16 V	0,16 V	Diam
4,9V	4,9 V	8,8 V	8,8 V	Diam
0 V	4,9 V	0,16 V	8,8 V	Kanan
4,9 V	0 V	8,8 V	0,16 V	Kiri

Tabel 6.5 Hasil Pengujian Driver motor Belakang

Input A1	Input A2	Output B1	Output B2	Roda Belakang
		Vcc = 9V	Vcc = 9V	
0 V	0 V	0,16 V	0,16 V	Diam
4,9 V	4,9 V	8,8 V	8,8 V	Diam
0 V	4,9 V	0,16 V	8,8 V	Maju
4,9 V	0 V	8,8 V	0,16 V	Mundur

Dari hasil pengujian diatas dapat disimpulkan, bahwa putaran motor untuk roda depan dan belakang dapat berubah sesuai dengan perubahan tegangan yang diberikan melalui driver motor, dan driver motor dapat bekerja dengan baik.

6.1.5 Pengujian Pengukuran kecepatan

a. Tujuan

Mengetahui tingkat kesalahan pengukuran kecepatan motor DC secara software dengan bantuan sensor kecepatan.

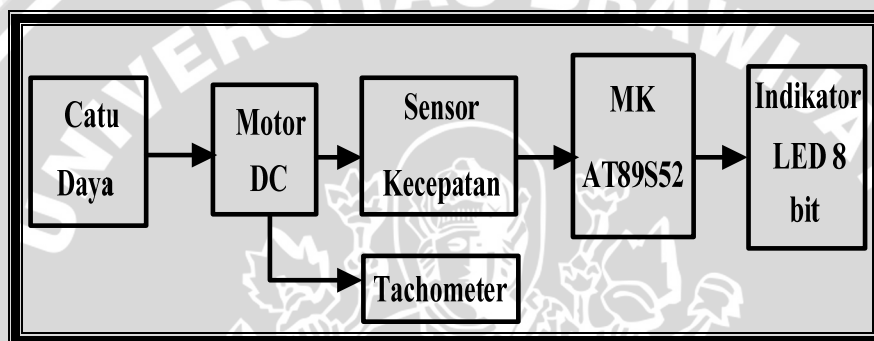
b. Peralatan

- rangkaian sensor kecepatan, yang terdiri dari rangkaian pemancar dan penerima

- catu daya + 3V, 4,5V, 6V, 7,5V, 9V, 12V, dan 13,5V
- LED indikator 8 bit
- tachometer

c. Langkah Pengujian

- merangkai peralatan seperti dalam Gambar 6.5. Keluaran rangkaian terhubung dengan multimeter digital.
- mengaktifkan catu daya seperti pada tabel 6.4.
- mencatat kondisi LED indikator 8 bit pada setiap perubahan tegangan catu daya dan mencatat hasil pembacaan tachometer.

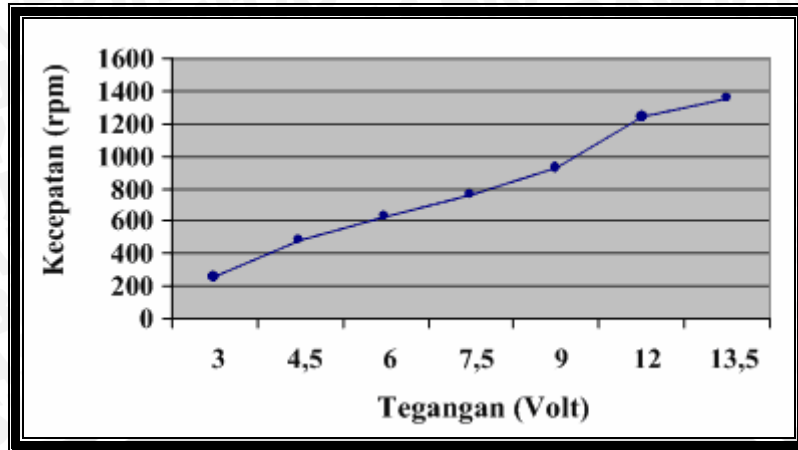


Gambar 6.5. Blok Diagram Pengujian pengukuran kecepatan

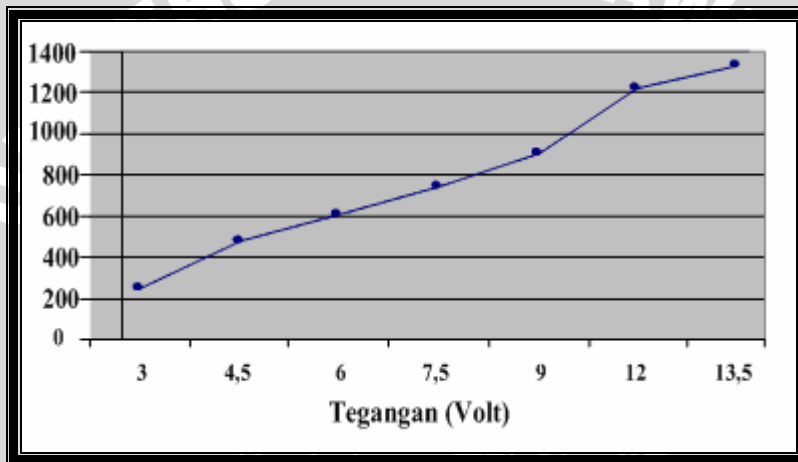
d. Hasil Pengujian dan Analisis

Tabel 6.6 Hasil Pengujian Kecepatan

Tegangan (Volt)	Pembacaan Tachometer (rpm)	LED Indikator 8 bit	Pembacaan software (rpm)	Error (%)
3	249	01010101	254	2
4,5	476	00101100	486	2,1
6	610	00100010	625	2,4
7,5	744	00011100	762	2,4
9	906	00010111	926	2,2
12	1220	00010001	1247	2,2
13,5	1328	00001111	1356	2,1



Gambar 6.6. Grafik Pengukuran Kecepatan Dengan Menggunakan Tachometer



Gambar 6.7. Grafik Pengukuran Kecepatan Dari Pembacaan Software

Untuk menghitung frekuensi putaran motor dalam menggunakan software digunakan persamaan sebagai berikut :

$$T_{clock\ timer} = \frac{12}{11,0592} \mu s$$

$$\begin{aligned} \text{Periode putaran motor} &= \text{Jumlah hitungan timer} \times 1,085 \mu s \\ &= \text{Indikator LED} \times 256 (2^8) \times 1,085 \mu s \\ &= \text{Indikator LED} \times 277,76 \mu s \end{aligned}$$

$$\begin{aligned} \text{Frekuensi putaran motor} &= \frac{1}{277,76 \mu s} \times \frac{1}{\text{Indikator LED}} \\ &= 0,0036 \times \frac{1}{\text{Indikator LED}} \text{ Mrps} \end{aligned}$$

$$= 3,6 x \frac{1}{\text{Indikator LED}} \text{ Krps}$$

$$= 21,6 x \frac{1}{\text{Indikator LED}} \text{ Krpm}$$

$$\text{Frekuensi putaran motor} = 21600 x \frac{1}{\text{Indikator LED}} \text{ rpm}$$

Dari hasil pengujian didapatkan error rata-rata pembacaan rpm secara software sebesar 2,2%.

6.1.6 Pengujian PWM

a. Tujuan

Mengetahui keluaran output mikrokontroler yang berupa PWM dengan bermacam-macam variasi *duty cycle*

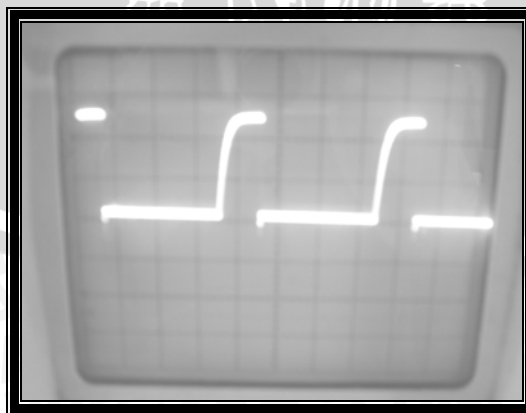
b. Peralatan

- mikrokontroler
- catu daya 5V
- osiloskop

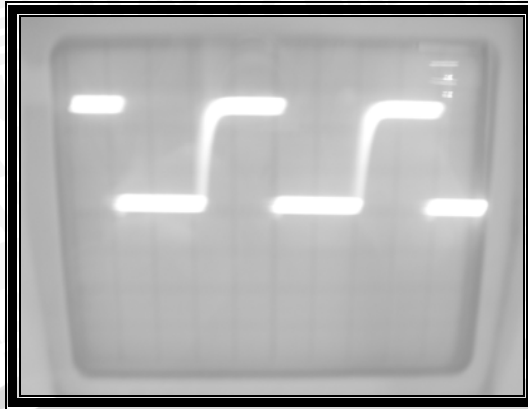
c. Langkah Pengujian

- merangkai peralatan seperti dalam Gambar 6.5. Keluaran rangkaian terhubung dengan osiloskop.
- mengaktifkan catu daya.
- menggambar keluaran hasil dari osiloskop untuk *duty cycle* 25%, 50% dan 75%.

d. Hasil Pengujian dan Analisis



Gambar 6.8. Grafik Tegangan Terhadap Waktu Untuk *Duty Cycle* 25%



Gambar 6.9. Grafik Tegangan Terhadap Waktu Untuk Duty Cycle 50%



Gambar 6.10. Grafik Tegangan Terhadap Waktu Untuk Duty Cycle 75%

Dari hasil pengujian dapat disimpulkan, bahwa PWM yang dihasilkan oleh mikrokontroler dapat bekerja dengan baik.

6.2 Pengujian Perangkat Lunak

6.2.1 Pengujian Sistem Mikrokontroler

a. Tujuan

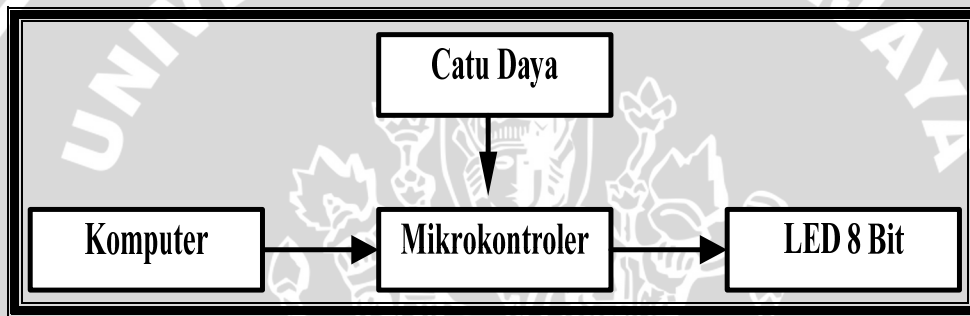
Mengetahui kondisi awal dari sistem mikrokontroler agar sesuai dengan yang diharapkan.

b. Peralatan

- komputer dengan kabel penghubungnya
- minimum sistem mikrokontroler AT89S52
- lampu LED mewakili keluaran 8 bit
- catu daya +5volt

c. Langkah Pengujian

- merangkai peralatan seperti dalam Gambar 6.11. Keluaran terhubung dengan lampu LED yang mewakili keluaran 8 bit.
- mengisi mikrokontroler dengan program sederhana yang meletakkan 0FH dan F0H pada Accumulator secara bergantian kemudian memindahkannya ke port 1. kemudian download pada mikrokontroler AT89S52.
- mengaktifkan catu daya.
- mereset program untuk mengeluarkan data mikrokontroler.
- mencatat data keluaran yang diwakili oleh lampu LED 8 bit ke dalam bentuk biner.



Gambar 6.11. Blok Diagram Pengujian Mikrokontroler

d. Hasil Pengujian dan Analisis

Tabel 6.7. Hasil Pengujian Sistem mikrokontroler

Kondisi	Keluaran pada LED Display							
	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
I	1	1	1	1	0	0	0	0
II	0	0	0	0	1	1	1	1

Dari tabel 6.7 terlihat bahwa port 1 memberikan logika 0FH dan F0H secara bergantian sesuai dengan isi program.

6.3 Pengujian Sistem Secara Keseluruhan

a. Tujuan

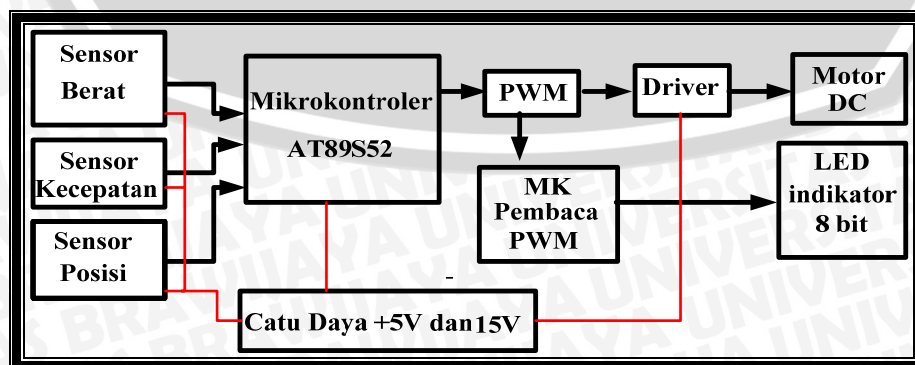
Mengetahui unjuk kerja perangkat lunak dengan perangkat keras saat diintegrasikan bersama-sama.

b. Peralatan

- sensor berat
- timbel timbangan
- sensor kecepatan
- sensor posisi
- minimum sistem mikrokontroler AT89S52
- rangkaian driver motor DC
- catu daya +5volt dan -15 volt
- multimeter digital

c. Langkah Pengujian

- merangkai peralatan seperti dalam Gambar 6.12. keluaran dari rangkaian PWM dihubungkan disimpan dalam mikrokontroler lain untuk mengetahui tegangan keluaran.
- memasukkan berat berdasarkan timbel timbangan yang sudah ada.
- memasukkan kecepatan berdasarkan tegangan yang diberikan pada motor, seperti pada saat kalibrasi.
- melakukan pengujian berturut-turut dari Kontroler I kemudian dilanjutkan dengan Kontroler II.
- mengaktifkan catu daya
- mencatat keluaran yang tampak pada LED indikator 8 bit, sebagai keluaran dari sistem.

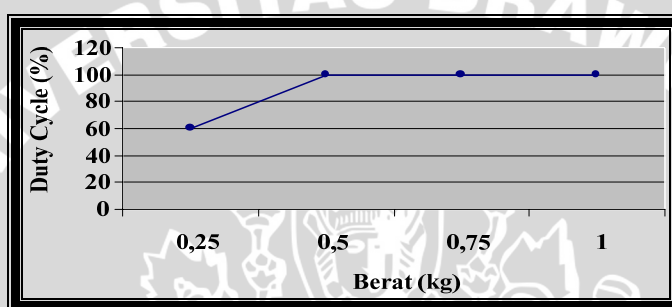


Gambar 6.12. Blok Diagram Pengujian Sistem secara Keseluruhan

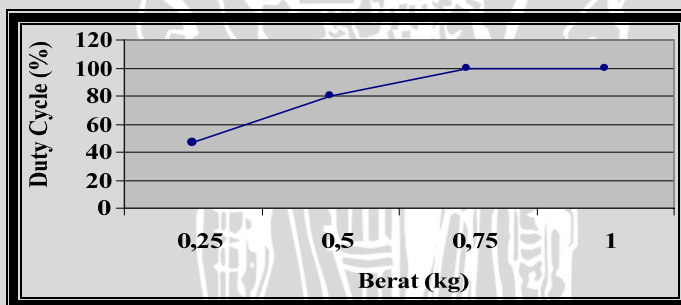
d. Hasil Pengujian dan Analisis

Tabel 6.8. Hasil Pengukuran Nilai Keluaran Kontroler dalam Duty Cycle

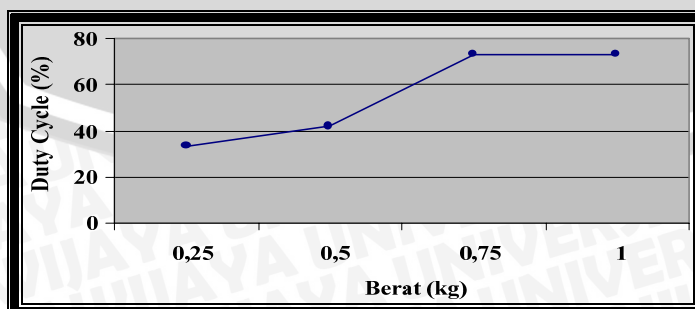
Berat (kg)	Kecepatan (rpm)			
	50	89	160	220
0.25	60%	46%	33%	33%
0.5	100%	80%	42%	33%
0.75	100%	100%	73%	60%
1	100%	100%	73%	60%



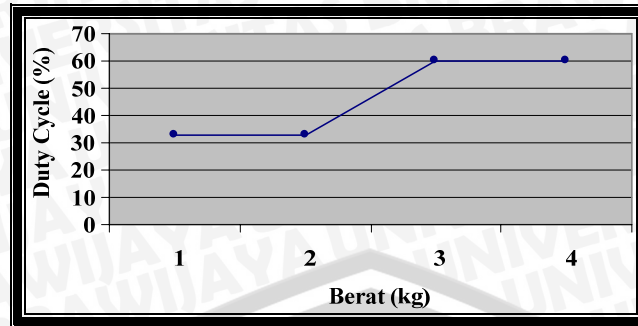
Gambar 6.13. Grafik Hubungan Duty Cycle Terhadap Berat Beban Untuk Kecepatan 50 rpm



Gambar 6.14. Grafik Hubungan Duty Cycle Terhadap Berat Beban Untuk Kecepatan 89 rpm



Gambar 6.15. Grafik Hubungan Duty Cycle Terhadap Berat Beban Untuk Kecepatan 160 rpm



Gambar 6.16. Grafik Hubungan Duty Cycle Terhadap Berat Beban Untuk Kecepatan 220 rpm

Dari Tabel 6.8 terlihat nilai keluaran kontroler yang dihasilkan sesuai dengan perencanaan, dengan ini bisa disimpulkan sistem keluaran kontroler dalam *duty cycle* berjalan sebagaimana mestinya.



BAB VII PENUTUP

7.1 Kesimpulan

Berdasarkan pada hasil pengujian dan analisis sistem pengaturan kecepatan dan berat relatif pada prototipe robot pengangkut barang, maka dapat diambil beberapa kesimpulan sebagai berikut :

- a. Sensor berat yang merupakan kombinasi *optocoupler* dengan pegas mampu mengubah beban menjadi tegangan sesuai dengan beban yang diberikan.
- b. Berat maksimum yang bisa diberikan pada sensor berat harus disesuaikan dengan keadaan dan kemampuan prototipe yang digunakan, untuk perancangan ini berat maksimum yang bisa diberikan sebesar 1 kg.
- c. Rangkaian sensor posisi dapat digunakan dengan baik pada perancangan alat ini, karena dapat mengeluarkan logika “0” dan logika “1” sebagai masukan bagi mikrokontroler.
- d. Rangkaian mikrokontroler AT89S52 dapat bekerja dengan baik sesuai dengan perancangan yang diharapkan.
- e. *Kernel Fuzzy Inference System* dapat berjalan sesuai dengan sistem yang diharapkan.

7.2 Saran

Dalam perancangan dan pembuatan alat ini masih terdapat kelemahan dan kekurangan sistem. Saran-saran untuk penyempurnaan kinerja alat dan pengembangan lebih lanjut adalah sebagai berikut:

- a. Perlu adanya penyempurnaan dalam perancangan sensor berat, sehingga berat yang dideteksi akan lebih teliti.
- b. Untuk memperbaiki kinerja robot pengangkut barang ketika belok, maka perlunya menggunakan sensor posisi yang lebih banyak beserta *gear* banding pada motor penggerak roda depan dari robot pengangkut barang.
- c. Perlu digunakan sensor yang baik untuk meningkatkan ketelitian dari besaran yang diukur. Sehingga performa sistem akan dapat ditingkatkan

sesuai dengan yang diharapkan, serta dengan menyempurnakan *kernel fuzzy inference system*.

- d. Perlu dibuat perancangan catu daya yang lebih baik, sehingga prototipe robot pengangkut barang dapat berjalan dengan baik tanpa ada hambatan lain berupa kabel yang mengikutinya.



DAFTAR PUSTAKA

- ATMEL Corp., 1996, *8-Bit Mikrokontroler with 8K bytes Flash AT89S52*, ATMEL, <http://www.atmel.com.index.html>
- Elektro Indonesia., 1995. *Teknologi Sistem fuzzy*, Elektro Indonesia. (<http://www.elektroindonesia.com>)
- Fairchild Semikonduktor, 2001, H22A1 / H22A2 / H22A3 Phototransistor Slotted Optical Switch.
- Malvino, Albert Paul, 1993. *Elektronika komputer Digital* Edisi kedua, Alih bahasa: Tjia May On, PhD, Jakarta: Penerbit Erlangga.
- Malvino, Albert Paul, 1996. *Prinsip-prinsip Elektronika*. Edisi ketiga, Alih bahasa: Hanapi gunawan, Jakarta: Penerbit Erlangga.
- Ross, Timothy J, 1997. *Fuzzy Logic With Engineering Application*. Singapore: Mcgraw-Hill, Inc.
- Texas Instruments, 1990, L293D Quadruple Half-H Driver.
- Wang, L.X, 1994. *A Course In Fuzzy Systems And Control*, New Jersey: Prentice-Hall.
- Zuhail, 1991. *Dasar Tenaga Listrik*. Bandung: Penerbit ITB.