

**SISTEM PENGENDALIAN KECEPATAN PADA PROTOTIPE
CRANE MAGNET DENGAN KONTROL LOGIKA FUZZY**

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh:

**KAREL ANDILOLO
NIM. 0410632024**

**DEPARTEMEN PENDIDIKAN NASIONAL
UNIVERSITAS BRAWIJAYA
FAKULTAS TEKNIK
JURUSAN TEKNIK ELEKTRO
MALANG
2007**

repository.ub.ac.id

SISTEM PENGENDALIAN KECEPATAN PADA PROTOTIPE CRANE MAGNET DENGAN KONTROL LOGIKA FUZZY

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Teknik



Disusun oleh:

KAREL ANDILOLO
NIM. 0410632024

Mengetahui dan menyetujui,

Dosen Pembimbing :

Dosen Pembimbing I

Dosen Pembimbing II

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

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



repository.ub.ac.id

SISTEM PENGENDALIAN KECEPATAN PADA PROTOTIPE CRANE MAGNET DENGAN KONTROL LOGIKA FUZZY

Disusun oleh:

KAREL ANDILOLO
NIM. 0410632024

Skripsi ini telah diuji dan dinyatakan lulus pada
Tanggal 9 Februari 2007

DOSEN PENGUJI

Penguji I

Ir. Chairuzzaini
NIP. 130 682 589

Penguji III

Ir. Purwanto, MT.
NIP. 131 574 847

Penguji II

Ir. Erni Yudaningtyas, MT.
NIP. 131 879 035

Penguji IV

Rusmi Ambarwati, ST. MT.
NIP. 132 258 188

Mengetahui

Ketua Jurusan Teknik Elektro

Ir. Purwanto, MT.
NIP. 131 574 847

PENGANTAR

Assalamualaikum Wr. Wb.

Alhamdulillah Robbil ‘Alammin, puji syukur kehadirat Allah SWT yang telah memberikan pertolongannya pada hambanya yang lemah ini, sehingga penyusunan skripsi ini dapat terselesaikan. Shalawat serta salam senantiasa tercurahkan untuk nabi Muhammad SAW beserta keluarga beliau. Semoga skripsi ini dapat memberikan manfaat baik bagi penulis maupun umat ini, Amin.

Skripsi berjudul “*Sistem Pengendalian Kecepatan Pada Prototipe Crane Magnet Dengan Kontrol Logika Fuzzy*” ini disusun sebagai salah satu syarat untuk memperoleh gelar Sarjana Teknik di Jurusan Teknik Elektro Fakultas Teknik Universitas Brawijaya. Terselesaikan skripsi ini tentunya tidak lepas dari bantuan berbagai pihak. Oleh karena itu, dengan segala kerendahan hati penulis menyampaikan terima kasih kepada:

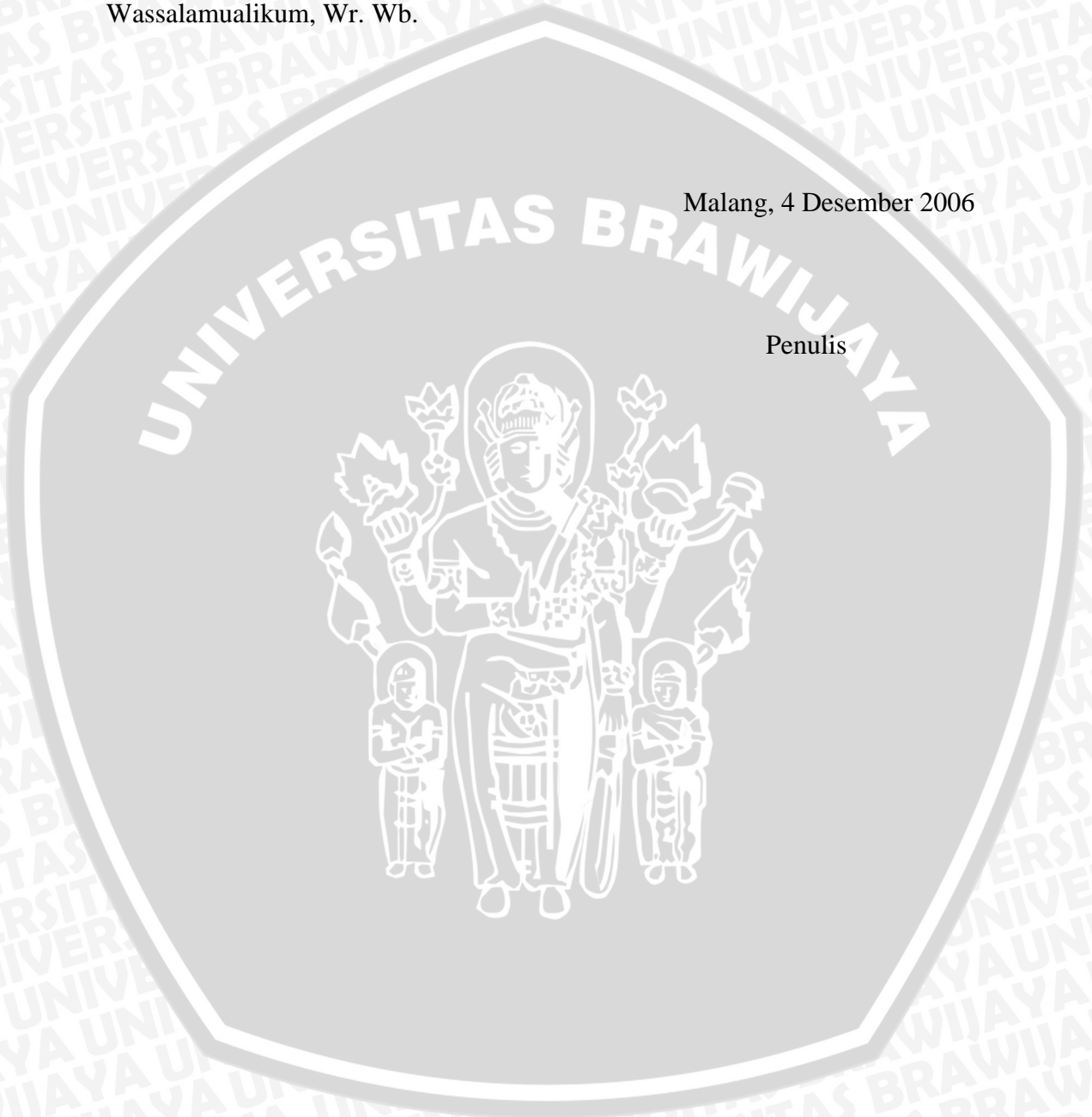
1. Bapak Ir. Purwanto, MT selaku Ketua Jurusan Teknik Elektro.
2. Bapak Ir. Hery Purnomo selaku Sekretaris Jurusan Teknik Elektro.
3. Bapak Dipl. Ing. Ir. Moch. Rusli selaku Ketua Kelompok Pengajar Keahlian Kontrol dan sekaligus dosen pembimbing yang senantiasa memberikan motivasi, koreksi, masukan, bimbingan serta nasehat hingga Tugas Akhir ini terselesaikan.
4. Bapak Moch. Rif’an, ST. MT selaku dosen pembimbing yang senantiasa memberikan koreksi, masukan, bimbingan serta nasehat hingga Tugas Akhir ini terselesaikan.
5. Seluruh dosen dan karyawan Jurusan Teknik Elektro.
6. Kedua orang tuaku Bapak Andilolo dan Mama Suciati, adikku Tri Subianto dan Masku Heriyanto dan seluruh keluarga besar yang telah memberikan do’a, kasih sayang, perhatian, serta dukungannya kepada penulis sehingga bisa menyelesaikan skripsi ini.
7. Temen-temen Elektro UB terutama angkatan 2004 alih jenjang yaitu Fathur, Syahrul, Rendra, Mas Gembong dan Ubed atas segala perhatian, bantuan, doa, serta kekompakannya.
8. Teman-teman angkatan 2000, 2001, 2002, 2003, 2004, dan 2005 atas segala dukungan, bantuan, dan doanya.

Penulis menyadari ketidaksempurnaan dalam penyusunan skripsi ini, sehingga penulis sangat mengharapkan kritik, saran dan masukan dari pembaca untuk dapat memperbaiki segala kekurangan di dalam skripsi ini. Akhir kata, semoga skripsi ini dapat memberikan manfaat dan berguna bagi pembaca khususnya mahasiswa Jurusan Teknik Elektro Universitas Brawijaya Malang.

Wassalamualikum, Wr. Wb.

Malang, 4 Desember 2006

Penulis



DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	iii
DAFTAR GAMBAR	vi
DAFTAR TABEL	viii
DAFTAR LAMPIRAN	ix
ABSTRAK	x
BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Ruang Lingkup	2
1.4. Tujuan	2
1.5. Sistematika Penulisan	3
BAB II MIKROKONTROLER ATmega8 DAN KONTROL LOGIKA FUZZY	
2.1. Mikrokontroler ATmega8	4
2.1.1. Fungsi Beberapa Pin ATmega8	5
2.1.2. Arsitektur AVR	6
2.1.3. <i>Timer/Counter</i>	8
2.1.4. Pemrograman AVR	11
2.2. Kontrol Logika <i>Fuzzy</i>	12
2.2.1. Himpunan <i>Fuzzy</i>	13
2.2.2. Fungsi Keanggotaan <i>Fuzzy</i>	14
2.2.3. Operasi Himpunan <i>Fuzzy</i>	16
2.2.4. Variabel Linguistik	17
2.2.5. Aturan <i>Fuzzy</i> "If - Then"	17
2.2.6. Kontrol Logika <i>Fuzzy</i>	19
2.2.6.1. Fuzzifikasi	20
2.2.6.2. Defuzzifikasi	20
2.2.7. Struktur Kontrol Logika <i>Fuzzy</i> Dasar	21

BAB III KOMPONEN PENDUKUNG

3.1.	Crane Magnet	23
3.2.	Motor DC	24
3.3.	Magnet	24
3.4.	<i>Optocoupler</i>	25
3.5.	Sensor Berat.....	26
3.6.	<i>Limit Switch</i>	27
3.7.	LCD	27
3.7.1.	Konfigurasi Pin-Pin LCD	29
3.7.2.	Fungsi-Fungsi Terminal	30

BAB IV METODE PENELITIAN

4.1.	Penentuan Spesifikasi Alat	32
4.2.	Perancangan Kontroler	33
4.3.	Perancangan Alat	34
4.3.1.	Perancangan Perangkat Keras	34
4.3.2.	Perancangan Perangkat Lunak	34
4.4.	Perealisasi Alat	34
4.4.1.	Perealisasi Perangkat Keras	34
4.4.2.	Perealisasi Perangkat Lunak	35
4.5.	Pengujian Alat	35
4.5.1.	Pengujian Perangkat Keras	35
4.5.2.	Pengujian Perangkat Lunak	35
4.6.	Pengambilan Kesimpulan	35

BAB V PERANCANGAN DAN PEMBUATAN ALAT

5.1.	Diagram Blok Sistem	36
5.2.	Prinsip Kerja Alat	37
5.3.	Perancangan Perangkat Keras (<i>Hardware</i>)	37
5.3.1.	Sub Sistem Masukan	38
5.3.1.1.	Perancangan Rangkaian Sensor Kecepatan ..	38
5.3.1.2.	Perancangan Rangkaian Sensor Berat	40
5.3.1.3.	Perancangan Rangkaian <i>Limit Switch</i>	41

5.3.2.	Sub Sistem Pengolah Data	42
5.3.2.1.	Perancangan Mikrokontroler ATmega8	42
5.3.3.	Sub Sistem Keluaran	44
5.3.3.1.	Rangkaian <i>Driver</i> Motor Magnet dan Crane..	44
5.3.3.2.	Rangkaian <i>Driver</i> Magnet	46
5.3.3.3.	<i>Liquid Crystal Display</i>	48
5.4.	Perancangan Kontroler Logika <i>Fuzzy</i>	50
5.5.	Perancangan Perangkat Lunak (<i>software</i>)	55

BAB VI PENGUJIAN ALAT

6.1.	Pengujian Rangkaian Sensor	58
6.1.1.	Pengujian Rangkaian Sensor Kecepatan	58
6.1.2.	Pengujian Pengukuran Kecepatan	59
6.1.3.	Pengujian Rangkaian Sensor Berat	61
6.1.4.	Pengujian Rangkaian <i>Limit Switch</i>	63
6.2.	Pengujian Mikrokontroler	64
6.3.	Pengujian Rangkaian LCD	65
6.4.	Pengujian Rangkaian <i>Driver</i> Motor DC	66
6.4.1.	Pengujian Rangkaian <i>Driver</i> Motor DC Magnet	66
6.4.2.	Pengujian Rangkaian <i>Driver</i> Motor DC Crane	67
6.5.	Pengujian Rangkaian <i>Driver</i> Magnet	69
6.6.	Pengujian Sistem Secara Keseluruhan	70

BAB VII KESIMPULAN DAN SARAN

7.1.	Kesimpulan.....	75
7.2.	Saran.....	75

DAFTAR PUSTAKA	76
-----------------------------	----

DAFTAR GAMBAR

Gambar 2.1.	Susunan Pin ATmega8	4
Gambar 2.2.	Arsitektur AVR.....	6
Gambar 2.3.	Peta memori AVR.....	7
Gambar 2.4.	Blok diagram <i>clock timer/counter</i>	8
Gambar 2.5.	Blok diagram <i>Timer/Counter0</i>	9
Gambar 2.6.	Blok diagram <i>Timer/Counter1</i>	10
Gambar 2.7.	Pemrograman paralel (A) dan serial (B) pada AVR.....	11
Gambar 2.8.	<i>S-function</i>	14
Gambar 2.9.	π - <i>function</i>	15
Gambar 2.10.	<i>T-function</i>	15
Gambar 2.11.	Contoh Operasi Gabungan Himpunan <i>Fuzzy A</i> dan <i>B</i>	16
Gambar 2.12.	Contoh Operasi Irisan Himpunan <i>Fuzzy A</i> dan <i>B</i>	16
Gambar 2.13.	Diagram Representasi <i>Fuzzy Kecepatan</i>	17
Gambar 2.14.	Metode Penalaran <i>Fuzzy Max-Min</i>	19
Gambar 2.15.	Metode Penalaran <i>Fuzzy Max-Dot</i>	19
Gambar 2.16.	Struktur Kontroler Logika <i>Fuzzy Dasar</i>	22
Gambar 3.1.	Motor DC.....	24
Gambar 3.2.	Perancangan Magnet Secara Mekanik.....	25
Gambar 3.3.	Skematik Fungsi <i>Optocoupler</i>	26
Gambar 3.4.	<i>Optocoupler</i> Jenis <i>U</i> dan Diagram Rangkaian.....	26
Gambar 3.5.	Perancangan Sensor Berat Secara Mekanik.....	27
Gambar 3.6.	Simbol <i>Limit Switch</i>	27
Gambar 3.7.	Diagram Blok LCD Seiko Instrument M1632.....	29
Gambar 4.1.	Pengendali Kontrol Logika <i>Fuzzy</i>	33
Gambar 5.1.	Diagram Blok Sistem.....	36
Gambar 5.2.	Diagram Blok Perancangan Perangkat Keras.....	38
Gambar 5.3.	Rangkaian Sensor Kecepatan.....	38
Gambar 5.4.	Perancangan Sensor Berat Secara Mekanik.....	41
Gambar 5.5.	Rangkaian Tegangan Keluaran.....	41
Gambar 5.6.	Rangkaian <i>Limit Switch</i>	42

Gambar 5.7.	Rangkaian Minimum Sistem	43
Gambar 5.8.	Rangkaian <i>Driver</i> Motor.....	44
Gambar 5.9.	Rangkaian <i>Driver</i> Magnet	46
Gambar 5.10.	Rangkaian LCD	49
Gambar 5.11.	Fungsi Keanggotaan <i>Crisp Input</i> Berat	51
Gambar 5.12.	Fungsi Keanggotaan <i>Crisp Input</i> Kecepatan	51
Gambar 5.13.	Fungsi Keanggotaan <i>Crisp Output</i> Kecepatan Motor Magnet .	52
Gambar 5.14.	Fungsi Keanggotaan <i>Crisp Input</i> Berat	52
Gambar 5.15.	Fungsi Keanggotaan <i>Crisp Input</i> Kecepatan	53
Gambar 5.16.	Fungsi Keanggotaan <i>Crisp Output</i> Kecepatan Motor Crane....	53
Gambar 5.17a.	Diagram Alir Perangkat Lunak Sistem.....	55
Gambar 5.17b.	Lanjutan Diagram Alir Perangkat Lunak Sistem.....	56
Gambar 5.17c.	Lanjutan Diagram Alir Perangkat Lunak Sistem.....	57
Gambar 6.1.	Diagram Blok Pengujian Rangkaian Sensor Kecepatan.....	59
Gambar 6.2.	Diagram Blok Pengujian Pengukuran Kecepatan.....	60
Gambar 6.3.	Blok Diagram Pengujian Rangkaian Sensor Berat.....	62
Gambar 6.4.	Diagram Blok Pengujian Rangkaian <i>Limit Switch</i>	63
Gambar 6.5.	Diagram Blok Pengujian Mikrokontroler.....	64
Gambar 6.6.	Diagram Blok Pengujian LCD.....	65
Gambar 6.7.	Hasil Pengujian LCD.....	65
Gambar 6.8.	Diagram Blok Pengujian <i>Driver</i> Motor DC Magnet	67
Gambar 6.9.	Diagram Blok Pengujian <i>Driver</i> Motor DC Crane.....	69
Gambar 6.10.	Diagram Blok Pengujian <i>Driver</i> Magnet.....	70
Gambar 6.11.	Grafik Keluaran Motor Magnet Naik 60 RPM.....	71
Gambar 6.12.	Grafik Keluaran Motor Magnet Turun 60 RPM.....	73
Gambar 6.13.	Grafik Keluaran Motor Crane (geser) 80 RPM.	74

DAFTAR TABEL

Tabel 2.1.	Pemilihan <i>clock timer/counter</i> pada AVR.....	9
Tabel 2.2.	Mode <i>compare</i> pada Timer/Counter1.....	10
Tabel 2.3.	Tabel Kaidah Atur Dalam Bentuk Matrik	22
Tabel 3.1.	Konfigurasi pin-pin LCD.....	29
Tabel 3.2.	Fungsi-fungsi Terminal.....	30
Tabel 3.3.	Register Seleksi.....	31
Tabel 5.1.	Tabel Terminal I/O pada LCD.....	48
Tabel 5.2.	Tabel Kaidah Atur Pengaturan Kecepatan Motor Magnet	54
Tabel 5.3.	Tabel Kaidah Atur Pengaturan Kecepatan Motor Crane	54
Tabel 6.1.	Hasil Pengujian Rangkaian Sensor Kecepatan	59
Tabel 6.2.	Hasil Pengujian Kecepatan	61
Tabel 6.3.	Hasil Pengujian Rangkaian Sensor Berat	62
Tabel 6.4.	Hasil Pengujian Rangkaian <i>Limit Switch</i>	63
Tabel 6.5.	Hasil Pengujian Mikrokontroler	64
Tabel 6.6.	Data <i>Input Driver</i> Motor DC Magnet.....	66
Tabel 6.7.	Hasil Pengujian <i>Driver</i> Motor DC Magnet.....	67
Tabel 6.8.	Data <i>Input Driver</i> Motor DC	68
Tabel 6.9.	Hasil Pengujian <i>Driver</i> Motor DC Magnet.....	69
Tabel 6.10.	Data <i>Input Driver</i> Magnet.....	69
Tabel 6.11.	Hasil Pengujian <i>Driver</i> Magnet	70
Tabel 6.12.	Hasil Pengujian Motor Magnet Naik dengan Set Point Kecepatan 60 rpm	71
Tabel 6.13.	Hasil Pengujian Motor Magnet Naik dengan Set Point Kecepatan 60 rpm	72
Tabel 6.14.	Hasil Pengujian Motor Crane (geser) dengan Set Point Kecepatan 80 rpm	73
Tabel 7.1.	Tabel Respon transien kecepatan Naik (beban 1 kg).....	75
Tabel 7.2.	Tabel Respon transien kecepatan Turun (beban 1 kg).....	75
Tabel 7.3.	Tabel Respon transien kecepatan Geser (beban 1 kg)	75

DAFTAR LAMPIRAN

LAMPIRAN A. Foto Alat

- Gambar Keseluruhan Alat
- Gambar Proses Pemindahan Barang
- Gambar Box Rangkaian
- Gambar Barang Yang Dipindahkan

LAMPIRAN B. Gambar Rangkaian

- Gambar Rangkaian Keseluruhan

LAMPIRAN C. Listing Program

- Listing Program Keseluruhan

LAMPIRAN D. *Datasheet* Komponen

- Datasheet* ATMega8
- Datasheet* H22A
- Datasheet* IRFZ44N
- Datasheet* SS9014



ABSTRAK

KAREL ANDILOLO, Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya Malang, Februari 2007. *Sistem Pengendalian Kecepatan Pada Prototipe Crane Magnet Dengan Kontrol Logika Fuzzy.*

Dosen Pembimbing: Dipl. Ing. Ir. Moch. Rusli dan Moch. Rif'an, ST. MT.

Crane magnet merupakan suatu alat yang digunakan untuk memindahkan barang jenis logam dari satu tempat ke tempat lain. Crane magnet dibuat untuk dapat mempercepat proses operasi pemindahan barang jenis logam, karena dalam pemindahan barang tidak memerlukan proses pengikatan dan pelepasan pada barang secara manual. Untuk mencapai tujuan tersebut maka harus ditunjang oleh aspek lain yaitu dengan mengendalikan kecepatannya dengan adanya perubahan berat barang yang dipindahkan, sehingga proses operasi pemindahan barang dapat stabil. Pada sistem ini piranti pendeteksi kecepatan menggunakan rangkaian *optocoupler* dan pendeteksi berat menggunakan kombinasi potensiometer dengan pegas, metode yang digunakan untuk mengendalikan motor DC adalah menggunakan PWM (*Pulsa With Modulation*) dan mikrokontroler sebagai pengendali prototipe. Masukan FLC (*Fuzzy Logic Controler*) adalah kecepatan dan berat barang, selanjutnya diproses untuk mendapatkan nilai *duty cycle* sinyal PWM sebagai acuan pembangkit PWM. Dengan metode FLC ini kecepatan motor akan dikendalikan sampai memenuhi set point kecepatan yang diinginkan.

Dari hasil pengujian yang dilakukan, untuk beban 1 Kg crane magnet dapat bekerja sesuai dengan perancangan. Untuk bergerak naik terjadi kesalahan sebesar 3.33% dengan pemberian gangguan berupa penambahan beban pada magnet dengan berat 0.25 Kg. Untuk bergerak geser terjadi kesalahan sebesar 1.25% dengan pemberian gangguan berupa penambahan beban pada crane dengan berat 0.5 Kg. Untuk bergerak turun terjadi kesalahan sebesar 1.67% dengan pemberian gangguan berupa penambahan beban pada magnet dengan berat 0.25 Kg. Dengan adanya prototipe ini diharapkan dapat membantu penelitian di bidang kontrol dan merealisasikan ke industri.

Kata Kunci: Fuzzy, Kecepatan, Crane Magnet.

BAB I PENDAHULUAN

1.1. Latar Belakang

Dewasa ini perkembangan dan kemajuan teknologi dunia industri sangatlah pesat khususnya perkembangan teknologi dalam bidang kontrol. Hingga kini semakin banyak sistem yang dapat dikontrol secara otomatis yang pada akhirnya membawa pengaruh pada kehidupan manusia, perkembangan teknologi yang semakin pesat tersebut membuat persaingan semakin meningkat dan bersaing untuk menjadi yang terbaik. Teknologi baru diharapkan dapat memberikan kemudahan dalam teknologi yang lain, proses-proses yang dikerjakan secara manual dapat dilakukan secara otomatis, akurat dan efisien.

Crane magnet merupakan suatu alat yang digunakan untuk memindahkan barang jenis logam dari satu tempat ke tempat lain. Crane magnet dirancang untuk dapat mempercepat proses operasi pemindahan barang jenis logam. Jenis-jenis barang logam yang dapat dipindahkan meliputi berbagai bentuk diantaranya yaitu lembaran baja, pipa, rel, scrap, kotak yang berisi benda-benda yang terbuat dari baja dan sebagainya. Keunggulan utama crane magnet ialah tidak diperlukannya pengikatan dan pelepasan pada muatan secara manual sehingga mengurangi waktu yang diperlukan untuk proses pemindahan barang secara drastis. Dimungkinkan magnet pengangkat akan dapat mengatasi muatan yang jumlahnya cukup besar dengan waktu yang minimal dan peningkatan efisiensi pengangkat yang cukup besar. Untuk mencapai tujuan tersebut maka harus ditunjang oleh aspek lain yaitu mengendalikan kecepatan dengan adanya perubahan berat barang yang dipindahkan, tujuannya adalah agar proses operasi pemindahan barang dapat stabil.

Menanggapi hal tersebut maka dalam penelitian ini akan dirancang suatu sistem pengendalian kecepatan pada crane magnet berbasis algoritma *fuzzy*. Pemilihan crane magnet dilatar belakangi oleh pentingnya peran dan keberadaan crane magnet untuk industri logam pada khususnya. 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 dari sistem tersebut. Keuntungan lain dari pemakaian algoritma *fuzzy* ini adalah kemudahan dalam memperbaiki unjuk

kerja sistem, yaitu dengan cara mengubah *software*-nya tanpa harus mengubah konfigurasi *hardware*-nya, sehingga penggunaannya sangat praktis dan ekonomis.

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 merancang dan membuat sensor berat dan sensor kecepatan pada crane magnet.
2. Bagaimana merancang sebuah sistem pengendalian menggunakan kontrol logika fuzzy
3. Bagaimana pengaruh adanya perbedaan beban yang dipindahkan terhadap sistem.

1.3. Ruang Lingkup

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

1. Crane magnet yang dibuat merupakan sebuah model atau prototipe yang telah dimodifikasi, tetapi memiliki kinerja seperti aslinya.
2. Prinsip kerja prototipe crane magnet ini dibuat sesederhana mungkin, tanpa melupakan fungsi sebenarnya.
3. Pembahasan lebih diutamakan pada proses pengendalian prototipe crane magnet.
4. Penggerak prototipe crane magnet menggunakan motor DC.
5. Tidak membahas Motor DC yang digunakan secara mendetail.
6. Tidak membahas magnet secara spesifik.
7. Barang yang dipindahkan berbentuk kotak dan terbuat dari jenis logam.
8. Berat beban maksimal yang dapat dipindahkan adalah 1kg

1.4. Tujuan

Tujuan yang ingin dicapai dari tugas akhir ini adalah terciptanya sistem pengendali kecepatan pada crane magnet berbasis algoritma *fuzzy* sebagai salah satu alternatif dari pengendalian crane magnet yang telah ada.

1.5. Sistematika Penulisan

Tugas akhir ini terdiri dari tujuh bab dengan sistematika penulisan sebagai berikut:

BAB I Pendahuluan

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

BAB II Mikrokontroler ATmega8 dan Kontrol Logika Fuzzy

Membahas teori-teori sistem pengontrol yang mendukung dalam perancangan dan pembuatan alat yang meliputi : Mikrokontroler ATmega8 dan dasar teori logika *fuzzy*, himpunan *fuzzy*, operasi himpunan *fuzzy*, variabel *linguistic*, fuzzifikasi, aturan *fuzzy*, defuzzifikasi.

BAB III Komponen Pendukung

Membahas teori-teori yang mendukung dalam perancangan dan pembuatan alat yang meliputi : Crane magnet, motor DC, sensor berat, *Optocoupler*, *Limit switch* dan LCD.

BAB IV Metodologi Penulisan

Menjelaskan tahap-tahap yang dilakukan dalam perancangan dan pembuatan alat.

BAB V Perancangan dan Pembuatan Alat

Membahas perancangan dan pembuatan sistem pengendalian kecepatan pada prototipe crane magnet dengan metode logika *fuzzy* menggunakan mikrokontroler ATmega8.

BAB VI Pengujian Alat

Membahas hasil pengujian sistem terhadap alat yang telah direalisasikan.

BAB VII Kesimpulan dan Saran

Membahas kesimpulan perancangan ini dan saran-saran yang diperlukan untuk melakukan pengembangan aplikasi selanjutnya.

BAB II

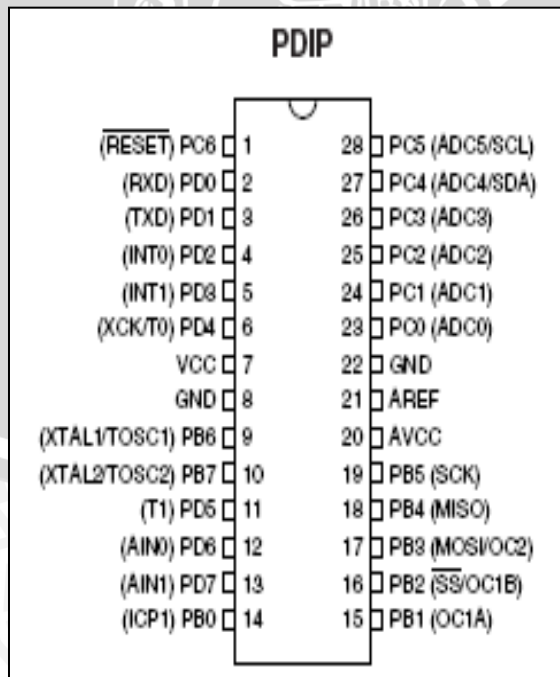
MIKROKONTROLER ATMega8 DAN KONTROL LOGIKA FUZZY

Untuk dapat memahami cara kerja rangkaian maupun dasar-dasar perancangan alat ini, maka perlu adanya penjelasan teori sistem pengontrolan yang digunakan dalam penulisan tugas akhir ini. Adapun teori-teori sistem pengontrolan yang akan dijelaskan dalam bab ini adalah:

1. Mikrokontroler ATMega8
2. Kontrol Logika Fuzzy

2.1. Mikrokontroler ATMega8

Mikrokontroler ATMega8 merupakan mikrokontroler produksi Atmel yang menggunakan arsitektur RISC (*Reduced Instruction Set Computing*) 8 bit keluarga Atmel AVR. AVR pertama kali diperkenalkan pada tahun 1996. AVR (*Alf and Vegard's Risc processor*) mengkombinasikan arsitektur RISC, memori *flash* internal dan jumlah *register* yang besar (32 buah) untuk memperoleh ukuran code program, kinerja, dan konsumsi daya yang optimal. Sebagian besar instruksi AVR dieksekusi dalam satu siklus *clock*. Kelebihan lainnya, arsitektur AVR dirancang untuk bekerja secara efisien menggunakan bahasa tingkat tinggi C.



Gambar 2.1. Susunan Pin ATMega8

Sumber: Data Sheet

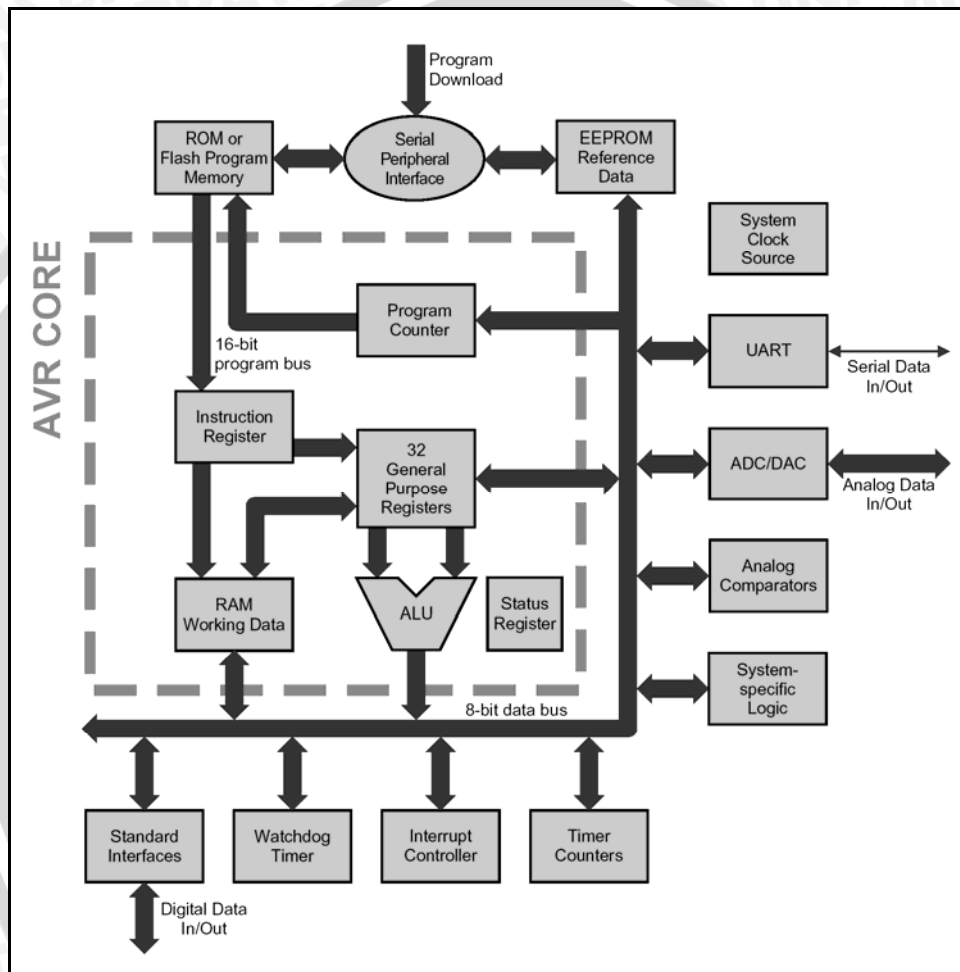
2.1.1 Fungsi Beberapa Pin ATmega8

Beberapa fungsi dari pin penting mikrokontroler ATmega8 adalah sebagai berikut :

- Vcc
Sebagai power supply DC (+5 volt).
- GND
Dihubungkan ke V_{SS} atau *ground*.
- AREF
AREF adalah pin referensi analog untuk ADC.
- AVCC
AVCC adalah pin penyuplai tegangan untuk port A dan ADC. Pin ini harus dihubungkan ke VCC, walaupun ADC tidak digunakan, dan harus dihubungkan melalui *low pass filter*.
- Port B (PB7..PB0)
Port B adalah port I/O 8 bit *bidirectional* dengan resistor pull-up internal. ICP1, OC1A, OC1B, MOSI, MISO, SCK, XTAL1(OSC1), XTAL2 (OSC2),
- Port C (PC6..PC0)
Port C adalah port I/O 7 bit *bidirectional* dengan resistor pull-up internal. ADC0, ADC1, ADC2, ADC3, ADC4/SDA, ADC5/SCL, RESET.
- Port D (PD7..PD0)
Port D adalah port I/O 8 bit *bidirectional* dengan resistor pull-up internal. Fungsi pengganti meliputi RXD (*Receive Data*), TXD (*Transmit Data*), INT0 (*Interrupt 0*), INT1 (*Interrupt 1*), XCK/T0 (*Timer 0*), T1 (*Timer 1*), AIN0, AIN1.
- RESET
Input reset. Logika rendah pada pin ini pada waktu yang lebih lama dari pada jarak pulsa minimum akan menghasilkan reset, walaupun clock sedang berjalan.
- XTAL1
merupakan pin masukan ke *oscillator* penguat pembalik dan masukan ke osilator internal clock.
- XTAL2
Keluaran dari *oscillator* penguat pembalik.

2.1.2 Arsitektur AVR

AVR menggunakan konsep arsitektur Harvard dengan memori dan bus terpisah untuk data dan program. Lebar bus program pada AVR adalah 16 bit sedangkan lebar bus data 8 bit. Memori program dieksekusi dengan *pipeline* satu tingkat. Saat instruksi sedang dieksekusi, instruksi yang berikutnya dibaca dari memori program. Konsep ini memungkinkan instruksi untuk dieksekusi dalam tiap-tiap siklus *clock*.



Gambar 2.2. Arsitektur AVR

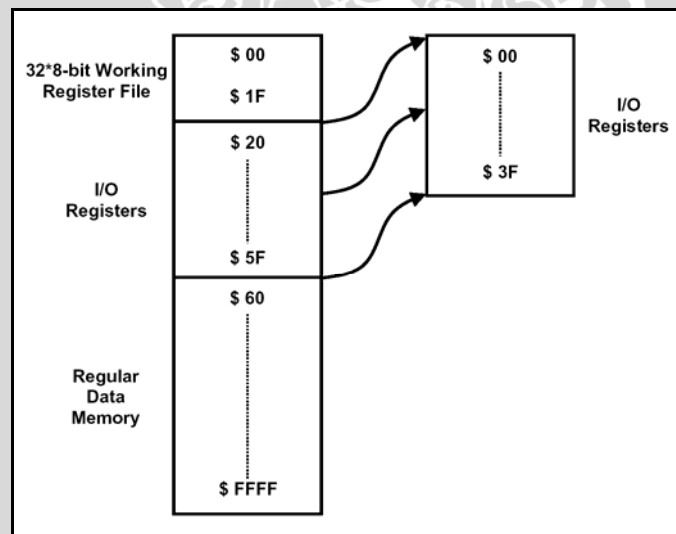
Sumber : Atmel, 1999 : 5

AVR memiliki jumlah *register* yang relatif besar untuk ukuran mikrokontroler 8 bit, yaitu 32 buah *general purpose registers*. AVR tidak memiliki *accumulator* seperti yang dimiliki sebagian besar mikroprosesor/mikrokontroler. Seluruh *register* terhubung ke ALU (*Arithmetic Logic Unit*) sehingga operasi ALU dapat dilaksanakan dengan menggunakan *general purpose registers* sebagai *operand*. Dua *operand* diambil dari *register*, operasi ALU dijalankan, dan hasil operasi disimpan ke dalam *register*, semuanya dilakukan dalam satu siklus *clock*. Dalam skala operasi yang relatif besar,

operasi ALU dapat dijalankan dengan lebih cepat. Semua *general purpose registers* juga dapat digunakan untuk mengakses data dari dan ke memori dengan instruksi *load* dan *store*.

Enam dari 32 *register* dapat digunakan sebagai pasangan *register* 16 bit. Pasangan *register* 16 bit dapat digunakan sebagai *register* pointer untuk pengalamatan area data sehingga memungkinkan perhitungan alamat menjadi lebih efisien. Ketiga *register* tersebut adalah *register* X (R26 dan R27), Y (R28 dan R29), dan Z (R30 dan R31).

Peta memori data AVR dapat dilihat dalam Gambar 2.3 Memori data dibagi menjadi 4 bagian. 32 alamat paling bawah (0000 – 001F) ditempati oleh *general purpose registers*. 64 alamat berikutnya (0020 – 005F) ditempati oleh *register* I/O yang mengatur piranti CPU seperti *register* kontrol, *timer/counter*, ADC, dan fungsi I/O lainnya. Alamat berikutnya digunakan oleh SRAM internal dan eksternal. Perlu dicatat, ukuran SRAM internal tidak sama untuk masing-masing tipe AVR, dan tidak semua tipe memiliki SRAM internal atau SRAM eksternal.



Gambar 2.3. Peta memori AVR

Sumber : Atmel, 1999 : 6

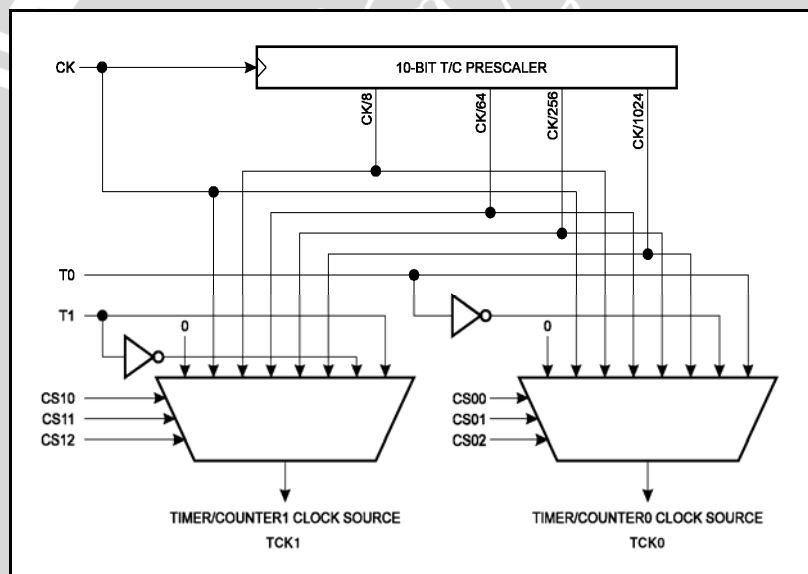
Selama interupsi dan pemanggilan *subroutine*, alamat pada *Program Counter* (PC) disimpan pada *stack*. Pada tipe AVR yang memiliki SRAM internal atau eksternal, *stack* dialokasikan di SRAM sehingga ukuran *stack* hanya terbatas oleh total ukuran dan pemakaian SRAM. Semua program harus menginisialisasi *Stack Pointer* (SP) di dalam rutin *reset*, sebelum *subroutines* atau interupsi dieksekusi. Sedangkan tipe AVR yang

tidak memiliki SRAM, alamat pada PC disimpan di *stack hardware* yang memiliki ukuran terbatas.

2.1.3 Timer/Counter

AVR memiliki beberapa jenis periperhal internal, diantaranya komparator analog, *timer/counter*, UART, *watchdog timer*, dan ADC/DAC. Periperhal yang ada dalam chip AVR tergantung tipe AVR. Periperhal yang sering digunakan adalah *timer/counter*.

Timer/counter pada AVR dibagi menjadi dua, yaitu 8 bit (Timer/Counter0) dan 16 bit (Timer/Counter1). Beberapa tipe AVR juga memiliki Timer/Counter2 yang hampir sama dengan Timer/Counter1 tetapi hanya berukuran 8 bit. *Clock timer/counter* dapat berupa *clock* internal CPU atau *clock* eksternal dari pin T0/T1. *Timer/counter* pada AVR memiliki pembagi frekuensi *clock* dari CPU (CK). Dengan menggunakan *clock* internal CPU, dapat dipilih frekuensi CK, CK/8, CK/64, CK/256, dan CK/1024.



Gambar 2.4. Blok diagram *clock timer/counter*

Sumber : Atmel, 2001 : 32

Saat *timer/counter* menggunakan *clock* eksternal, sinyal eksternal disinkronkan dengan frekwensi osilator CPU. Untuk mendapatkan sampling *clock* eksternal yang *sesuai*, waktu minimum antara dua transisi *clock* eksternal harus sedikitnya satu periode *clock* internal CPU. Sinyal *clock* eksternal disampling pada tepi naik *clock* internal CPU.

Timer/Counter0 merupakan *up counter* 8 bit (TCNT0) dengan akses baca dan tulis. Saat Timer/Counter0 overflow, akan dibangkitkan sinyal TOV0 yang dapat digunakan untuk membangkitkan interupsi dengan men-set bit TOIE dalam *Timer*

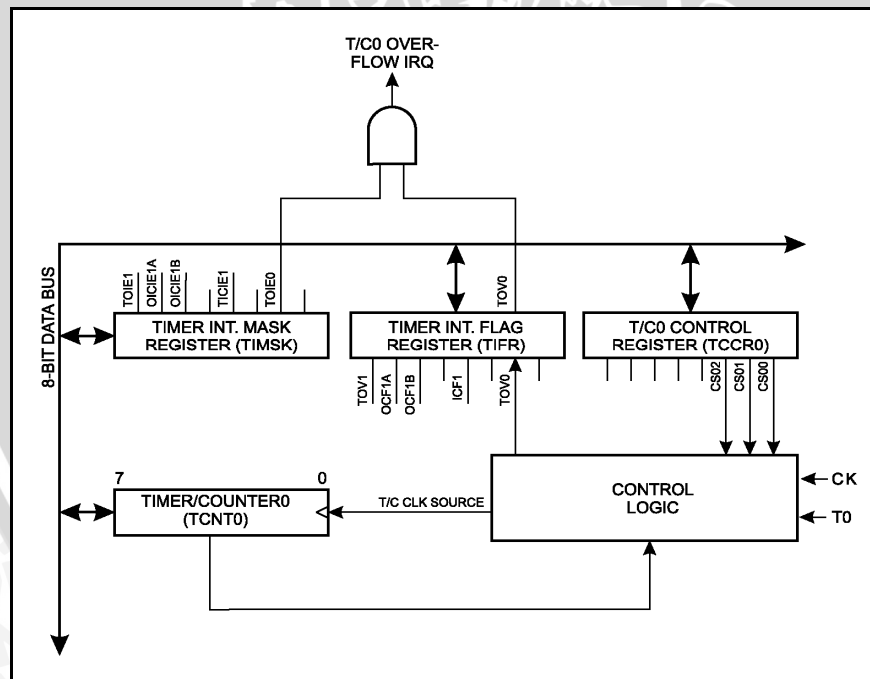
Interrupt Mask Register (TIMSK). Blok diagram Timer/Counter0 dapat dilihat dalam Gambar 2.5.

Tabel 2.1. Pemilihan *clock timer/counter* pada AVR

CSX2	CSX1	CSX0	KETERANGAN
0	0	0	Stop
0	0	1	CK
0	1	0	CK/8
0	1	1	CK/64
1	0	0	CK/256
1	0	1	CK/1024
1	1	0	Pin TX, sisi turun
1	1	1	Pin TX, sisi naik

Keterangan : X = 0 atau 1

Sumber : Atmel, 2001 : 33

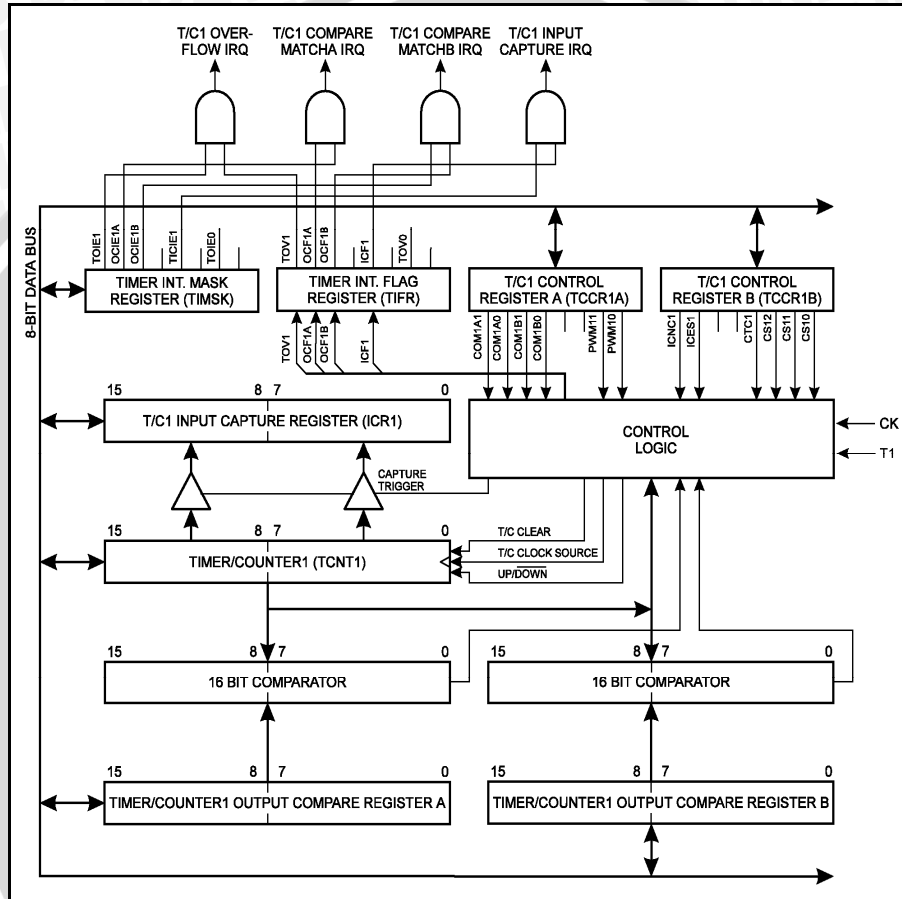


Gambar 2.5. Blok diagram Timer/Counter0

Sumber : Atmel, 2001 : 33

Timer/Counter1 merupakan *up/down counter* 16 bit (*down counter* hanya bisa digunakan saat mode PWM). Timer/Counter1 dapat dibaca atau ditulisi dengan mengakses Timer/Counter1 High (TCNT1H) dan Timer/Counter1 Low (TCNT1L). Saat CPU menulis ke TCNT1H, data disimpan ke *register* TEMP. Saat CPU menulis ke

TCNT1L, data dari CPU dikombinasikan dengan data di *register* TEMP ditulis ke Timer/Counter1, sehingga penulisan ke Timer/Counter1 harus diawali dengan penulisan ke TCNT1H. Sedangkan saat pembacaan TCNT1L, data TCNT1L dikirimkan ke CPU dan data TCNT1H di simpan ke *register* TEMP. Saat CPU membaca data dari TCNT1H, CPU menerima data dari *register* TEMP, sehingga pembacaan ke Timer/Counter1 harus diawali dengan pembacaan ke TCNT1L.



Gambar 2.6. Blok diagram Timer/Counter1
 Sumber : Atmel, 2001 : 34

Tabel 2.2. Mode *compare* pada Timer/Counter1

COM1X1	COM1X0	Description
0	0	Timer/Counter1 tidak terhubung dengan OC1X
0	1	Toggle keluaran OC1X
1	0	Clear keluaran OC1X
1	1	Set keluaran OC1X

Keterangan : X = A atau B
 Sumber : Atmel, 2001 : 36

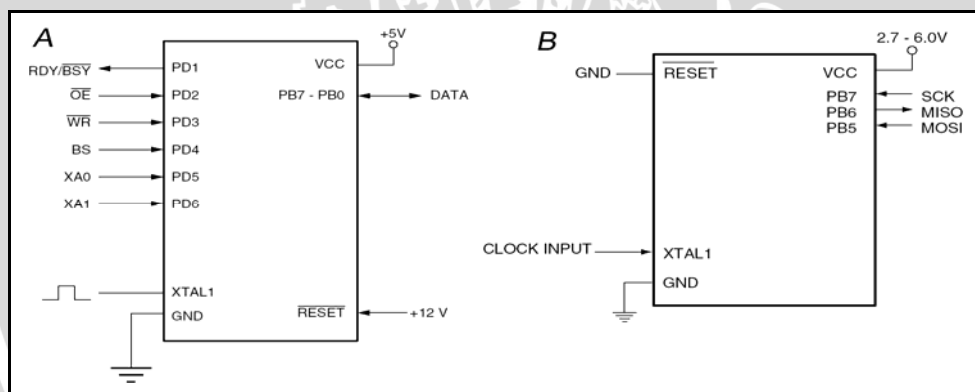
Timer/Counter1 mendukung fungsi perbandingan (*compare*) keluaran *timer/counter* dengan *Timer/Counter1 Output Compare Register* (OCR1). Pemilihan mode perbandingan dapat dilihat dalam Tabel 2.2. Dengan men-*set* bit *Clear Timer/Counter1 on Compare Match* (CTC1) pada *Timer/Counter1 Control Register B* (TCCR1B), Timer/Counter1 akan di-*clear* saat nilai *counter* sama dengan OCR1.

Fungsi Timer/Counter1 yang lain adalah *pulse width modulation* (PWM) dan *input capture*. Lebih lanjut tentang dua fungsi ini dapat dilihat dalam *datasheet*.

2.1.4 Pemrograman AVR

AVR memiliki 118 buah instruksi. Sebagian besar instruksi dieksekusi dalam satu siklus *clock*. Secara umum instruksi-instruksi dalam AVR dapat dibagi menjadi 4 kelompok yaitu : aritmatika dan logika, percabangan, transfer data, serta bit dan bit-test.

Instruksi dalam AVR disimpan dalam memori *flash* internal dalam pasangan 16 bit. Besarnya memori flash bergantung dari tipe AVR, berkisar antara 1k sampai 128k byte. Memori *flash* AVR memiliki *endurance* 1,000 siklus *write/erase*.



Gambar 2.7. Pemrograman paralel (A) dan serial (B) pada AVR
 Sumber : Atmel, 2001 : 79

AVR mempunyai dua mode pemrograman yaitu mode pemrograman paralel (mode paralel) dan mode pemrograman serial (mode ISP). Pada mode pemrograman paralel, alat yang akan diprogram dipasang pada soket programmer dan tegangan pemrograman sebesar +12 volt dikenakan pada pin *RESET*. Komunikasi antar programmer dan AVR dilakukan secara paralel, sehingga kecepatan program lebih cepat dari mode serial.

Pada mode serial, komunikasi antara AVR dan *programmer* dilakukan secara serial. Mode ini hanya memerlukan tiga jalur sinyal tanpa tegangan +12V, sehingga AVR dapat program saat masih berada dalam sistem. Mode serial juga biasa disebut

sebagai mode ISP (*In-System Programming*). Pada banyak kasus mode serial mempermudah pengembangan program karena program dapat diganti sewaktu-waktu tanpa melepas mikrokontroler dari sistem. Namun, mode pemrograman serial tidak bisa digunakan untuk memprogram *fuse bit* pada beberapa tipe AVR dan beberapa tipe tidak mempunyai fasilitas ISP. Pada kasus seperti itu pemrograman harus dilakukan dalam mode paralel. Sebagai pengecualian, tipe-tipe AVR yang hanya memiliki 8 pin menggunakan mode pemrograman serial yang menggunakan tegangan pemrograman +12V. Detil pemrograman masing-masing tipe AVR dapat dilihat pada *datasheet*.

Pada tipe-tipe AVR tertentu, flash memory dikelompokkan menjadi dua bagian, bagian aplikasi dan bagian *boot loader*. Fasilitas *boot loader* memungkinkan mikrokontroler memperbaharui kode programnya sendiri tanpa menggunakan *programmer*. Ukuran boot loader dapat dirubah dengan mengeset *fuse* BOOTSZ. Penulisan ke memori program dilakukan dengan instruksi SPM.

2.2. 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).

Ciri utama dari logika *fuzzy* adalah adanya peluang untuk memanfaatkan toleransi ketidakpastian (*inexactness*) dan ketidak tepatan (*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*.

2.2.1 Himpunan Fuzzy

Himpunan didefinisikan sebagai kelompok objek-objek tertentu yang disebut elemen. 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:

$$A = \{(x, \mu_A(x)) | x \in X\} \quad (2.1)$$

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

$$\begin{aligned} A &= \mu_1 / x_1 + \mu_2 / x_2 + \dots + \mu_n / x_n \\ A &= \sum \mu_A(x_i) / x_i \end{aligned} \quad (2.2)$$

Bila X bersifat kontinyu dan tak terbatas:

$$A = \int_x \mu_A(x) / x \quad (2.3)$$

2.2.2 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*.

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.

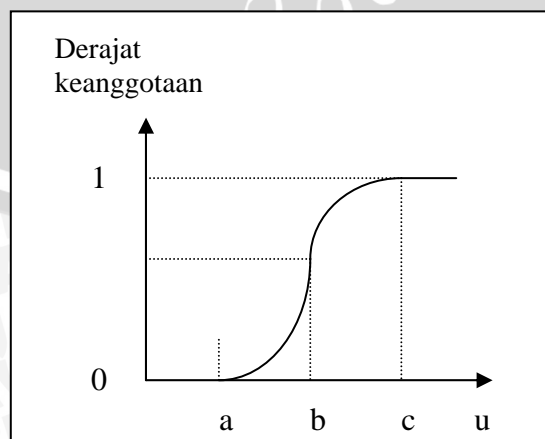
Pada himpunan *fuzzy* semua informasi terkandung dalam fungsi keanggotaannya. Oleh karena itu penting untuk mengetahui bagian-bagian pada sebuah fungsi keanggotaan *fuzzy*. Ada beberapa fungsi keanggotaan dalam logika *fuzzy*, tetapi yang sering digunakan adalah bentuk fungsi-S, fungsi- π , dan fungsi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

a. S-function

Definisi S-function adalah sebagai berikut:

$$S(u,a,b,c) = \begin{cases} 0 & u < a \\ 2\left(\frac{u-a}{c-a}\right) & a \leq u \leq b \\ 1 - 2\left(\frac{u-a}{c-a}\right) & b \leq u \leq c \\ 1 & u > c \end{cases} \quad (2.4)$$

Bentuk diagramatik S-function ditunjukkan dalam Gambar 2.2.



Gambar 2.8. S-Function

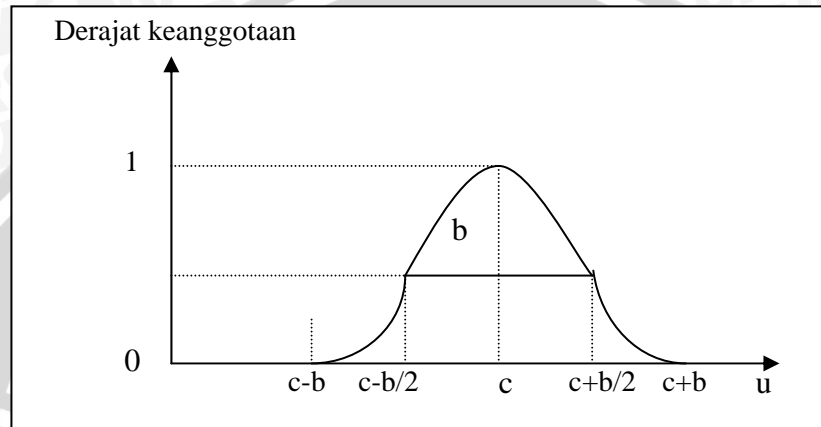
Sumber: Jun Yan, 1994: 29

b. π -function

Definisi π -Function adalah sebagai berikut:

$$\pi(u; b, c) = \begin{cases} S(u; c-b, c-\frac{b}{2}, c) & u \leq c \\ 1 - S(u; c, c+\frac{b}{2}, c+b) & u \geq c \end{cases} \quad (2.5)$$

Bentuk diagramatik π -Function ditunjukkan dalam Gambar 2.3.



Gambar 2.9. π -Function

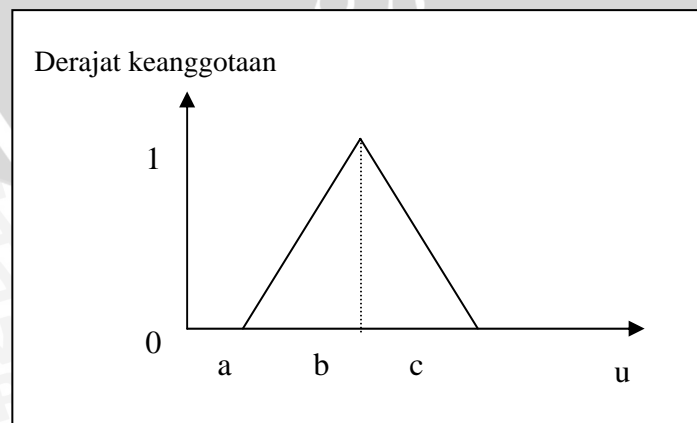
Sumber: Jun Yan, 1994: 29

c. *Triangular-function*

Triangular-function didefinisikan sebagai berikut:

$$T(u; a, b, c) = \begin{cases} 0 & u < a \\ \frac{u-a}{b-a} & a \leq u \leq b \\ \frac{c-u}{c-b} & b \leq u \leq c \\ 0 & u > c \end{cases} \quad (2.6)$$

Bentuk diagramatik *Triangular-function* ditunjukkan dalam gambar 2.4.



Gambar 2.10. T-Function

Sumber: Jun Yan, 1994: 30

2.2.3 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 komplement. Operasi dasar yang berlaku didefinisikan sebagai berikut:

a. Gabungan (*union*)

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

$$\mu_{A \cup B} = \mu_A(x) \vee \mu_B(x) = \text{maks}\{\mu_A(x), \mu_B(x)\} \quad (2.7)$$

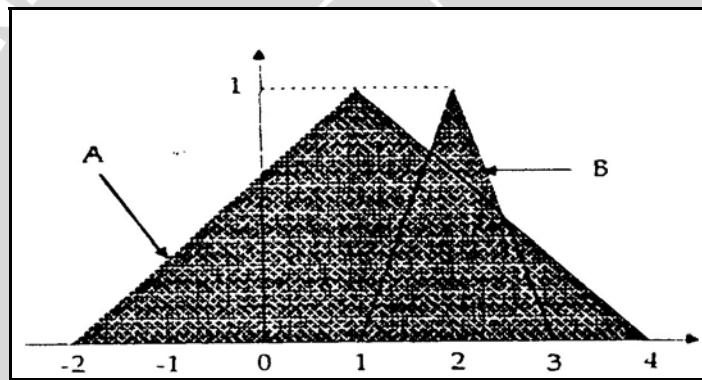
Operasi gabungan diperlihatkan dalam Gambar 2.5.

b. Irisan (*intersection*)

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

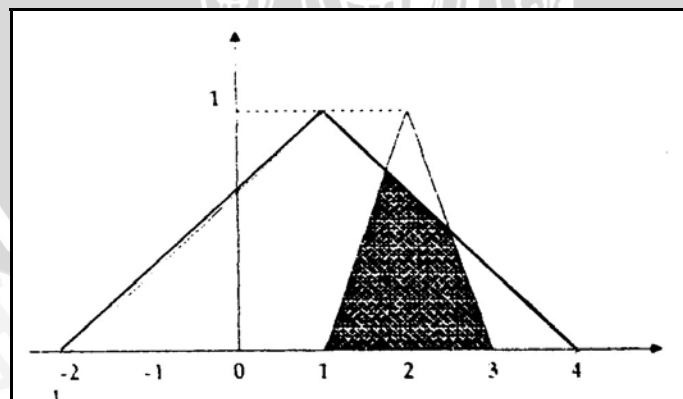
$$\mu_{A \cap B} = \mu_A(x) \wedge \mu_B(x) = \min\{\mu_A(x), \mu_B(x)\} \quad (2.8)$$

Operasi irisan diperlihatkan dalam Gambar 2.6.



Gambar 2.11. Contoh Operasi Gabungan Himpunan Fuzzy A dan B

Sumber: Jun Yan, 1994: 38



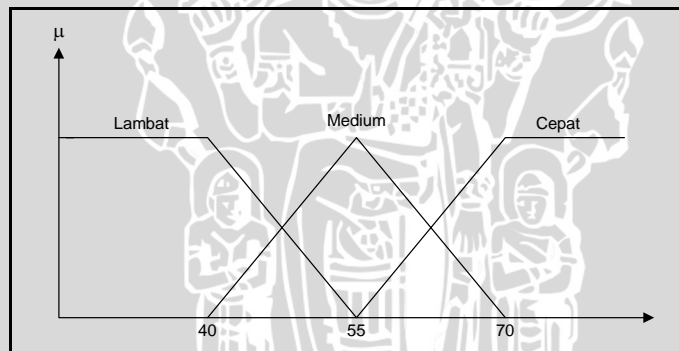
Gambar 2.12. Contoh Operasi Irisan Himpunan Fuzzy A dan B

Sumber: Jun Yan, 1994: 38

2.2.4 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 2.7.



Gambar 2.13. Diagram Representasi *Fuzzy* Kecepatan
Sumber: Jun Yan, 1994: 40

2.2.5 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. Bentuk umum dari aturan logika *fuzzy* ini sebagai berikut,

$$\text{If (kondisi), Then (aksi)} \quad (2.9)$$

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:

If A_{k1} AND ... AND A_{ki} AND ... AND A_{kn} *Then* B_k (2.10)

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:

$$\begin{aligned}\alpha_1 &= \mu_{A_1}(x_0) \wedge \mu_{B_1}(y_0) \\ \alpha_2 &= \mu_{A_2}(x_0) \wedge \mu_{B_2}(y_0)\end{aligned}\quad (2.11)$$

dengan x_0 dan y_0 menyatakan masukan *crisp*.

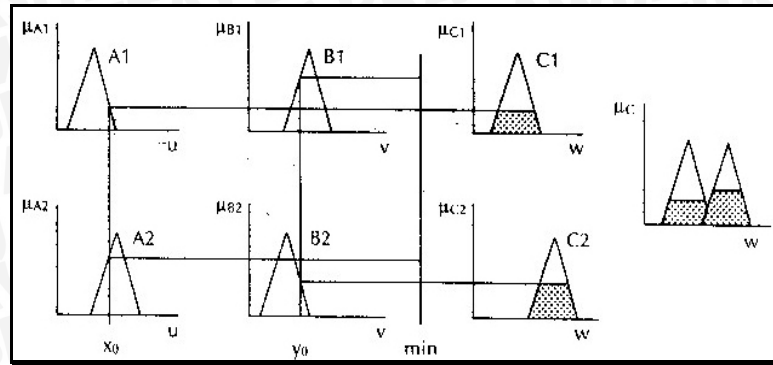
Metode-metode penalaran untuk memperoleh keputusan *fuzzy* ada dua, yaitu:

a. *Metode penalaran MAX-MIN*

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

$$\mu_c(w) = (\alpha_1 \wedge \mu_{c_1}(w)) \vee (\alpha_2 \wedge \mu_{c_2}(w)) \quad (2.12)$$

metode penalaran MAX-MIN diperlihatkan dalam Gambar 2.8.



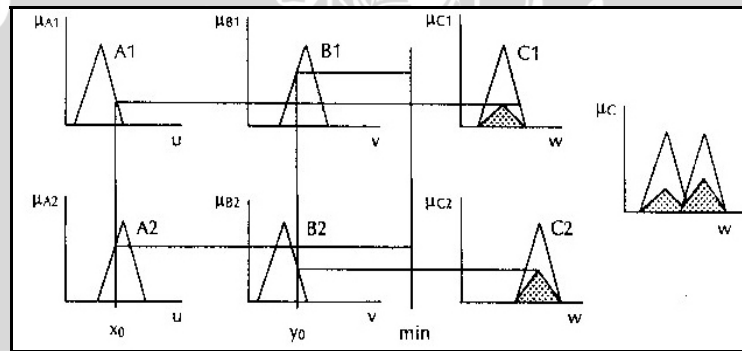
Gambar 2.14. Metode Penalaran *Fuzzy* MAX-MIN
Sumber: Jun Yan, dkk., 1994: 55

b. Metode penalaran MAX-DOT

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

$$\mu_c(w) = (\alpha_1 \cdot \mu_{c1}(w)) \vee (\alpha_2 \cdot \mu_{c2}(w)) \quad (2.13)$$

metode penalaran MAX-DOT diperlihatkan dalam Gambar 2.9.



Gambar 2.15. Metode Penalaran *Fuzzy* MAX-DOT
Sumber: Jun Yan, dkk., 1994: 55

2.2.6 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 lingustik 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.

2.2.6.1 Fuzzifikasi

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

$$x = \text{fuzzifier}(x_0) \quad (2.14)$$

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.

2.2.6.2 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:

$$y_0 = \text{defuzzifier}(y) \quad (2.15)$$

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 Yan, dkk., 1994: 56].

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

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

$$W = \frac{\sum_{i=1}^n \alpha_i H_i W_i}{\sum_{i=1}^n \alpha_i H_i} \quad (2.16)$$

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/COG

Metode COA menghasilkan aksi kontrol pada pusat berat distribusi.

$$W = \frac{\sum_{i=1}^n \alpha_i M_i}{\sum_{i=1}^n \alpha_i A_i} \quad (2.17)$$

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

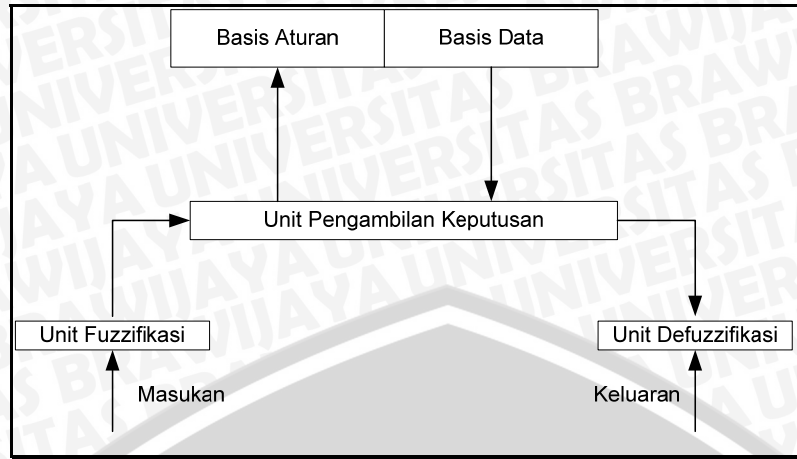
2.2.7 Struktur Kontroler Logika *Fuzzy* Dasar

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 aturan 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 2.10.

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 2.16. 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 2.3.

Tabel 2.3. 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

BAB III

KOMPONEN PENDUKUNG

Untuk memudahkan dalam memahami cara kerja rangkaian maupun dasar-dasar perancangan alat ini, maka perlu adanya penjelasan dan uraian teori komponen pendukung yang digunakan dalam penulisan tugas akhir ini. Adapun untuk teori-teori komponen pendukung yang akan dijelaskan dalam bab ini adalah:

1. Crane Magnet
2. Motor DC
3. Magnet
4. *Optocoupler*
5. Sensor Berat
6. *Limit Switch*
7. LCD

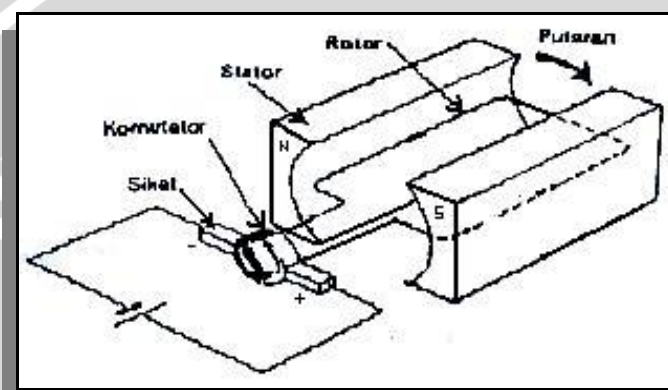
3.1 Crane Magnet

Crane magnet merupakan suatu alat yang digunakan untuk memindahkan barang jenis logam dari satu tempat ke tempat lain. Crane magnet dirancang untuk dapat mempercepat proses operasi pemindahan barang jenis logam. Jenis-jenis barang logam yang dapat dipindahkan meliputi berbagai bentuk diantaranya yaitu lembaran baja, pipa, rel, scrap, kotak yang berisi benda-benda yang terbuat dari baja dan sebagainya. Keunggulan utama crane magnet ialah tidak diperlukannya pengikatan dan pelepasan pada muatan secara manual sehingga mengurangi waktu yang diperlukan untuk proses pemindahan secara drastis. Dimungkinkan magnet pengangkat akan dapat mengatasi muatan yang jumlahnya cukup besar dengan waktu yang minimal dan peningkatan efisiensi pengangkat yang cukup besar.

Berdasarkan struktur mekaniknya crane magnet yang dibuat pada prototipe alat ini adalah jenis *gantry crane*. *Gantry crane* mempunyai karakteristik sebuah trolley yang bergerak diatas girder. Pergerakan trolley merupakan gerak lurus ke arah suatu arah sepanjang girder. Girder dapat dipasang diatas dua buah penopang besi seperti jembatan yang bisa bergerak secara horizontal yang disebut sebagai *bridge crane*. Type *gantry crane* ini banyak digunakan dalam banyak berbagai industri diantaranya adalah industri metalurgi, industri perakitan dan perusahaan pertambangan.

3.2 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.



Gambar 3.1. Motor DC
Sumber: Kismet, 1994: 98

3.3 Magnet

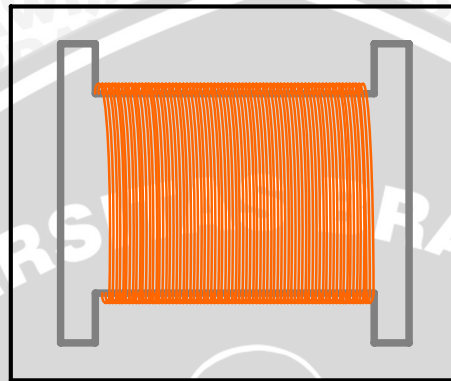
Sebuah magnet akan menarik bahan paramagnetik dan ferromagnetik, dan akan menolak bahan diamagnetik. Bahan paramagnetik adalah bahan yang ditarik oleh magnet dengan gaya yang lemah. Bahan ferromagnetik adalah bahan yang ditarik oleh magnet dengan gaya yang kuat. Bahan diamagnetik adalah bahan yang ditolak oleh magnet dengan gaya yang lemah. Bahan magnetik adalah bahan yang dapat ditarik oleh magnet. Diantara bahan-bahan magnetik tersebut adalah besi, baja, nikel, cobalt.

Tiap magnet mempunyai kutub utara dan kutub selatan. Kutub yang sejenis akan tolak-menolak dan yang tak sejenis akan tarik-menarik.

Medan magnet dapat dilukiskan dengan menggunakan garis-garis gaya magnet. Sifat-sifat dari pada garis gaya magnet adalah:

1. Garis gaya magnet membentuk kurva tertutup.
2. Garis gaya magnet berasal dari kutub utara menuju kutub selatan.
3. Garis-garis gaya magnet akan lebih banyak berkumpul pada kutub-kutub.
4. Garis-garis gaya magnet tak pernah saling berpotongan

Daerah atau ruang sekitar magnet disebut dengan medan magnet. Menurut Oersted, disekitar kawat yang dialiri arus listrik akan timbul medan magnet. Pada kumparan (lilitan kawat) yang dialiri arus listrik, maka arus-arusnya yang ada disebelah dan menyebelah sama arahnya, oleh karena itu maka medan magnet diluar kawat-kawat itu saling menguatkan. Demikian pula medan magnet yang ada diruang dalam kumparan menjadi kuat.



Gambar 3.2. Perancangan Magnet Secara Mekanik
Sumber: Perancangan

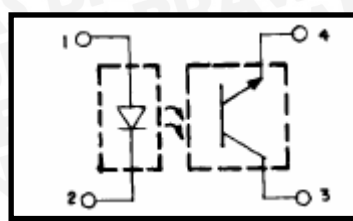
3.4 *Optocoupler*

Optocoupler atau optotransistor merupakan salah satu jenis komponen yang memanfaatkan sinar sebagai pemicu on-off. *Opto* berarti optic dan *couple* berarti pemicu. Sehingga bisa diartikan bahwa *optocoupler* merupakan sesuatu komponen yang bekerja berdasarkan pemicu cahaya *optic*. *Optocoupler* juga dikenal dengan optoisolator yaitu suatu alat yang memancarkan foton yang fluknya dikopling melalui bahan isolator yang transparan ke jenis *detector*. Alat pemancar foton ini dapat berupa lampu pijar neon ataupun led. Insulator transparan dapat berupa kaca udara atau serat optik.

Keuntungan dari *optocoupler* antara lain :

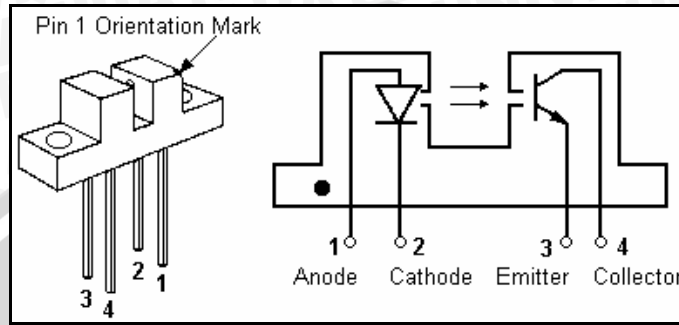
- Kecepatan operasi tinggi.
- Ukuran kecil.
- Tahan terhadap getaran dan benturan.
- Tidak mempunyai bagian yang bergerak yang dapat saling melekat.

Optocoupler dapat digunakan sebagai pemisah catu daya beserta groundingnya. Hal ini digunakan sebagai isolasi dari rangkaian luar yang dimungkinkan akan mengganggu rangkaian elektronik. Dasar rangkaian ditunjukkan pada gambar 3.3a dan 3.3b.



Gambar 3.3. Skematik *Optocoupler*

Sumber: Data sheet H22A



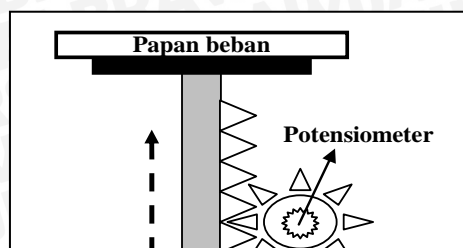
Gambar 3.4. *Optocoupler* Jenis U dan Diagram Rangkaianannya

Sumber: Data sheet H22A

3.5 Sensor Berat

Pada tugas akhir ini digunakan suatu perancangan sensor berat dengan menggunakan kombinasi potensiometer dan pegas. Potensiometer adalah sebuah instrumen yang direncanakan untuk mengukur tegangan yang tidak diketahui dengan cara membandingkannya terhadap tegangan yang diketahui. Pengukuran dengan cara perbandingan mampu menghasilkan tingkat ketelitian yang tinggi. Dengan dasar itulah dilakukan perancangan sensor berat ini [Cooper, 1999: 126].

Pada perancangan ini digunakan potensimeter *wire-wound*, yang ujung tuas penalnya dipasang sebuah *gear* yang dihubungkan pada *gear* batang (papan beban) yang diletakkan pada suatu poros. Perubahan massa beban akan menekan posisi *gear* batang (papan beban) naik turun sesuai massa beban. Perubahan posisi naik turun dari papan beban akan memutar posisi penala potensiometer. Dengan adanya perubahan posisi penala ini, nilai resistansi akan berubah, sehingga tegangan keluaran akan berubah pula. Dengan melakukan suatu kalibrasi dengan alat pengukur massa, maka sensor berat ini dapat menunjukkan perubahan massa yang terjadi setelah sensor berat yang terdapat di pasaran. Perancangan sensor berat ini secara mekanik dapat dilihat dalam Gambar 3.4.

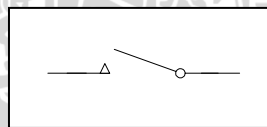


Pegas

Gambar 3.5. Perancangan Sensor Berat Secara Mekanik
Sumber: Perancangan

3.6 *Limit Switch*

Limit switch merupakan sebuah saklar yang dioperasikan secara mekanis, yang dikontrol oleh faktor-faktor secara otomatis, seperti posisi, tekanan dan suhu. *Limit switch* adalah salah satu pengendali industri yang sangat umum. *Limit switch* dirancang hanya untuk beroperasi apabila batas yang sudah ditentukan sebelumnya sudah dicapai dan saklar-saklar (*switch*) tersebut biasanya diaktifkan kontak dengan obyek tertentu. Alat tersebut menggantikan operator manusia. Saklar-saklar tersebut sering digunakan pada rangkaian pengendali dari mesin yang memproses untuk pengaturan *starting*, *stopping* atau pembalikan motor.



Gambar 3.6. Simbol *Limit Switch*
Sumber: Frank D. Petruzella, 1996:152

Pada perancangan ini *limit switch* digunakan sebagai sensor posisi, yaitu sensor atas, sensor bawah, sensor start, sensor finish, sensor barang dan sensor tujuan.

3.7 *LCD (Liquid Crystal Display)*

Liquid Crystal Display atau LCD adalah salah satu jenis penampil yang digunakan untuk menampilkan angka, karakter, atau bahkan angka dan karakter. LCD terdiri atas tumpukan tipis atau sel dari dua lembar kaca dengan pinggiran tertutup rapat. Antara dua lembar kaca tersebut diberi bahan kristal cair (*liquid crystal*), yang tembus cahaya. Permukaan luar dari masing-masing keping kaca mempunyai lapisan penghantar tembus cahaya seperti oksida timah (*tin oxide*) atau oksida indium (*indium*

oxide). Sel mempunyai ketebalan sekitar 1×10^{-5} meter dan diisi dengan kristal cair (*liquid crystal*). Blok diagram LCD ditunjukkan dalam Gambar 3.7.

LCD merupakan modul tampilan yang mempunyai konsumsi daya yang relatif rendah. LCD memiliki keistimewaan dibandingkan tampilan yang lain seperti *sevent segment* yaitu kemampuan untuk menampilkan karakter dan berbagai macam simbol. Salah satu jenis LCD diantaranya adalah LCD tipe *dot matriks 5x7*, tersusun sebanyak dua baris dan masing-masing baris terdiri atas 20 karakter. Setiap karakter dibentuk oleh 5x7 buah titik, sehingga jenis huruf yang mampu ditampilkan akan lebih banyak dan lebih baik dibandingkan dengan penampil *7 segment* atau *16 segment*.

Dalam gambar terlihat bahwa dengan adanya controller pada LCD maka LCD dapat dianggap sebagai suatu lokasi memori dari suatu unit prosesor, sehingga instruksi penampilan karakternya dapat digabungkan menjadi satu dengan unit prosesor.

Untuk dapat menggunakan LCD maka hal-hal yang perlu diperhatikan adalah sinyal kontrol dari LCD yaitu : RS, R/W dan EN. Instruksi operasi meliputi operasi dasar *register*, *busy flag*, *address counter*, *display data RAM*, *character generator ROM*.

- *Register*

Kontroler LCD mempunyai dua buah *register 8 bit*, yaitu *register* interupsi (IR) dan *register* data (DR). Kedua *register* ini dipilih melalui *register select* (RS)

- *Busy Flag*

Busy flag menunjukkan bahwa modul LCD siap untuk menerima instruksi selanjutnya. Sebagaimana terlihat pada tabel *register* seleksi, sinyal akan melalui DB₇ jika RS = 0 dan R/W = 1.

- *Address Counter (AC)*

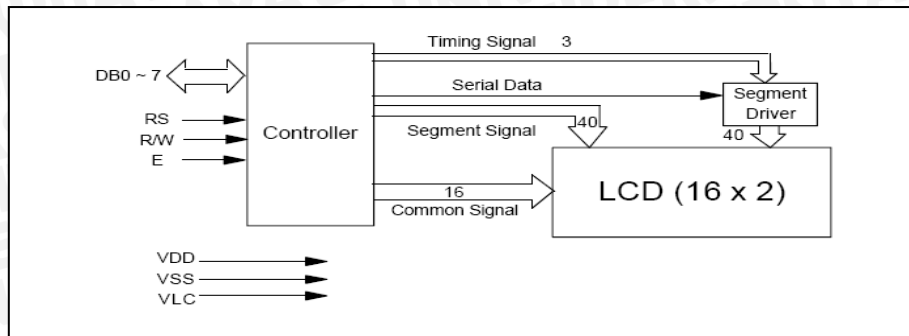
Address counter menunjukkan suatu lokasi memori dalam LCD dimana suatu data diletakkan dalam modul LCD. Pemilihan lokasi alamat diberikan melalui *register* instruksi (IR). Ketika data berada pada AC, maka secara otomatis AC akan menaikkan atau menurunkan alamat dari *entry mode set*.

- *Display Data RAM (DD RAM)*

Pada LCD masing-masing line mempunyai *range* alamat tersendiri. Alamat-alamat itu diekspresikan melalui bilangan hexadesimal. Untuk line 1 menggunakan kisaran alamat 00_H - 0F_H, sedangkan pada line 2 menggunakan kisaran alamat 40_H - 4F_H.

- *Character Generator ROM (CG ROM)*

CG ROM membangkitkan 192 buah tipe 5 X 7 karakter dot matriks. Pada LCD telah tersedia ROM sebagai pembangkit karakter dalam kode ASCII. CG RAM digunakan apabila diinginkan untuk pembuatan karakter tersendiri melalui program, dengan maksimal 8 buah karakter.



Gambar 3.7. Diagram Blok LCD Seiko Instrument M1632

Sumber: *Seiko Manual*, 1998:13

3.7.1 Konfigurasi Pin-pin LCD

Konfigurasi dan fungsi-fungsi umum pin-pin LCD ditunjukkan dalam Tabel 1

Tabel 3.1. Konfigurasi pin-pin LCD

No	SIMBOL	LEVEL	FUNGSI	
1	Vss	-	Power Supply	0 Volt (<i>Ground</i>)
2	Vcc	-		5 Volt +/- 10 %
3	Vee	-		For LCD Drive
4	RS	H/L	H = <i>Input Data</i> L = <i>Input Instruksi</i>	
5	R/W	H/L	H = Read L = <i>Write</i>	
6	E	H	Data Bus	
7	DB0	H/L		
8	DB1	H/L		
9	DB2	H/L		
10	DB3	H/L		
11	DB4	H/L		
12	DB5	H/L		
13	DB6	H/L		
14	DB7	H/L	Back Light Supply	
15	V+ BL	-		
16	V- BL	-	0 Volt (<i>Ground</i>)	

Sumber: EL-TECH Electronics, *LCD Module User Manual*, 1987:2

3.7.2 Fungsi-fungsi Terminal

Fungsi-fungsi terminal ditunjukkan dalam Tabel

Tabel 3.2. Fungsi-fungsi Terminal

NAMA SINYAL	NO. TERM.	I/O	TUJUAN	FUNGSI
DB ₀ – DB ₃	4	I/O	MPU	Lalu lintas data dan instruksi, <i>lower byte</i>
DB ₄ – DB ₇	4	I/O	MPU	Lalu lintas data dan instruksi, <i>upper byte</i>
E	1	INPUT	MPU	Sinyal Start (Read/Write)
R/W	1	INPUT	MPU	Sinyal seleksi instruksi
RS	1	INPUT	MPU	Sinyal seleksi <i>register</i>
V _{LC}	1	-	Power Supply	Driver LCD
V _{DD}	1	-	Power Supply	5 Volt
V _{SS}	1	-	Power Supply	0 Volt (<i>Ground</i>)

Sumber: EL-TECH Electronics, *LCD Module User Manual*, 1987:7

LCD tipe M1632 ini memiliki spesifikasi sebagai berikut:

- LCD ini terdiri dari 32 karakter dengan 2 baris masing-masing 16 karakter dengan *display dot matrik 5x7*
- Karakter generator ROM dengan 192 tipe karakter
- Karakter generator RAM dengan 8 bit karakter
- 80x8 bit display data RAM
- Dapat diinterfacekan ke MCU 8 atau 4
- Dilengkapi fungsi tambahan antara lain *display clear, cursor home, display on/off, cursor on/off, display character blink, cursor shift, display shift*
- *Internal data*
- *Internal* otomatis, reset pada saat *power on*
- Tegangan +5 volt PSU tunggal

Liquid Crystal Display (LCD) ini mempunyai konsumsi daya yang relatif rendah dan terdapat sebuah kontroler CMOS di dalamnya. Kontroler tersebut sebagai pembangkit dari karakter ROM/RAM dan *display data RAM*. Semua fungsi tampilan dikontrol oleh suatu instruksi dan modul LCD dapat dengan mudah untuk diinterfacekan dengan mikrokontroler. Masukan yang diperlukan untuk mengendalikan modul ini berupa bus data yang masih termultiflex dengan bus alamat serta 3 bit sinyal

kontrol. Sementara pengendalian dot matrik LCD dilakukan secara internal oleh kontroler yang sudah ada pada modul LCD.

Dasar-dasar pengoperasian LCD ini terdiri atas pengoperasian dasar pada register, *busy flag*, *address counter*, *display data RAM*

- Register

Kontroler dari LCD mempunyai 2 buah register 8 bit yaitu register instruksi (IR) dan register data (DR). IR menyimpan instruksi seperti *display clear*, *cursor shift* dan *display data* (DD RAM) serta *character generator* (CG RAM). DR menyimpan data untuk ditulis di DD RAM atau CG RAM ataupun membaca data dari DD RAM atau CG RAM. Ketika data ditulis ke DD RAM atau CG RAM maka DR secara otomatis menulis data ke DD RAM atau CG RAM. Ketika data pada DD RAM atau CG RAM akan dibaca maka alamat data ditulis pada IR sedangkan data akan dimasukkan melalui DR dan mikrokontroler membaca data dari DR.

- *Busy Flag*

Busy flag menunjukkan bahwa *module* siap untuk menerima instruksi selanjutnya. Sebagaimana yang terlihat dalam Tabel 3.3, register seleksi sinyal akan melalui DB₇ jika RS=0 dan R/W=1. Jika bernilai 1 maka modul LCD sedang melakukan kerja internal dan instruksi tidak akan diterima. Oleh karena itu status dari *flag* harus diperiksa sebelum melaksanakan instruksi selanjutnya.

Tabel 3.3. Register Seleksi

RS	R/W	OPERASI
0	0	Seleksi IR, IR Write Display Clear
0	1	Busy Flag (DB ₇) @ counter (DB ₀ -DB ₇) read
1	0	Seleksi DR, DR Write
1	1	Seleksi DR, DR Write

Sumber: *EL-TECH Electronic*, 1987:48

BAB IV

METODOLOGI PENELITIAN

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

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

- Penentuan spesifikasi alat.
- Perancangan Kontroler.
- Perancangan alat, meliputi : perangkat keras dan perangkat lunak.
- Perealisasiian alat, meliputi : perangkat keras dan perangkat lunak.
- Pengujian alat, meliputi : pengujian perangkat keras, perangkat lunak dan keseluruhan sistem alat serta analisa data.
- Pengambilan kesimpulan dan saran.

4.1 Penentuan Spesifikasi Alat

Sebelum melakukan perancangan dan perealisasiian alat, maka terlebih dahulu menentukan spesifikasi alat yang akan dibuat. Adapun spesifikasi alat yang akan direalisasiikan adalah sebagai berikut:

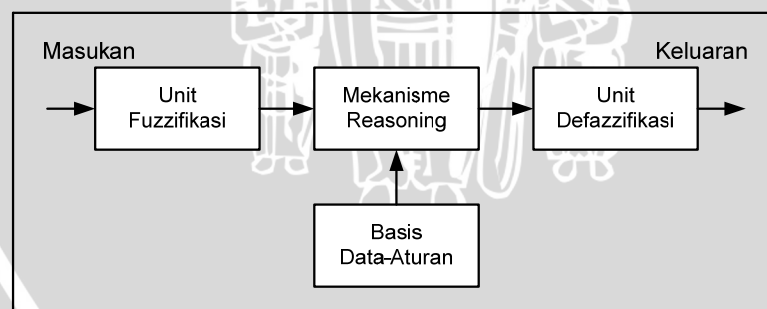
1. Menggunakan miniatur crane magnet dengan ukuran panjang 120 cm, lebar 26 cm, dan tinggi 63 cm.
2. Mikrokontroler yang digunakan adalah mikrokontroler buatan ATMEL dengan seri AVR ATmega8.
3. Kontroler yang digunakan adalah kontrol logika *fuzzy*.
4. Untuk menggerakkan crane dan magnet menggunakan motor DC *gearbox*.
5. Sensor kecepatan menggunakan rangkaian *optocoupler*.
6. Sensor berat menggunakan kombinasi potensiometer dengan pegas.
7. Sensor posisi menggunakan *limit switch*.

8. Jenis barang yang dipindahkan adalah terbuat dari jenis logam dan berbentuk kotak dengan berat maksimal 1 kg.
9. *Display* menggunakan *LCD M1632 2x16* karakter yang terpasang pada box panel.

4.2 Perancangan Kontroler

Dalam perancangan kontroler ini menggunakan Kontrol Logika *Fuzzy* (KLF). KLF ini merupakan suatu model logika yang mempresentasikan cara berpikir seorang ahli menjadi strategi kontrol otomatis (Sudjarwadi, 2003). KLF dapat mengatasi keterbatasan model matematika pengendalian karena dapat diselesaikan jika diketahui hubungan yang pasti antara masukan dan keluaran sistem (Hudallah, 2002). Dengan implementasi sistem *fuzzy* berbasis mikrokontroler ini maka pembuatan alat ini dapat segera direalisasikan.

Sedangkan struktur sederhana dari pengendali kontrol logika *fuzzy* secara umum meliputi langkah *fuzzifikasi*, yaitu mengubah sinyal masukan yang berasal dari sensor kecepatan, posisi dan berat menjadi *crisp input* sehingga dapat diproses oleh kontrol logika *fuzzy*, *knowledge base* (basis data dan aturan), yaitu membuat fungsi keanggotaan *fuzzy* serta *rule-base fuzzy system* (*fuzzy inference system*), mekanisme reasoning dan *defuzzifikasi*, yaitu mengubah *crisp output* yang telah diproses oleh kontrol logika *fuzzy* menjadi sinyal keluaran.



Gambar 4.1. Pengendali Kontrol Logika Fuzzy
Sumber: Perancangan

4.3 Perancangan Alat

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

4.3.1 Perancangan Perangkat Keras

Adapun untuk perancangan perangkat keras adalah meliputi:

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

4.3.2 Perancangan Perangkat Lunak

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 program utama sistem beserta subrutin-subrutinnya, kemudian merancang kontroler yang akan digunakan yakni kontrol logika *fuzzy*.

4.4 Perealisasian Alat

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

4.4.1 Perealisasian Perangkat Keras (*Hardware*)

Adapun tahap perealisasian 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.

4.4.2 Perealisasian Perangkat Lunak (*Software*)

Perealisasian perangkat lunak dilakukan dengan mengimplementasikan diagram alir dari program yang telah dirancang, penulisan program dalam bahasa C untuk kemudian dikonversi ke dalam bahasa mesin.

4.5 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.

4.5.1 Pengujian Perangkat Keras

Pengujian perangkat keras dilakukan dengan tujuan untuk memastikan bahwa sistem ini berjalan dengan baik maka dilakukan pengujian per blok maupun keseluruhan sistem. Pengujian tersebut adalah :

1. Pengujian sensor, yaitu sensor kecepatan, sensor berat dan *limit switch*.
2. Pengujian Mikrokontroler ATmega8.
3. Pengujian LCD.
4. Pengujian *driver* motor DC.

4.5.2 Pengujian Perangkat Lunak

Pengujian perangkat lunak dilakukan dengan melakukan pengujian pada subrutin-subrutin program dengan memberi masukan dan mengamati keluarannya.

4.5.3 Pengujian Keseluruhan Sistem dan Analisa Data

Tujuan pengujian sistem secara keseluruhan adalah untuk mengetahui unjuk kerja alat setelah perangkat keras dan perangkat lunak diintegrasikan bersama.

4.6 Pengambilan Kesimpulan

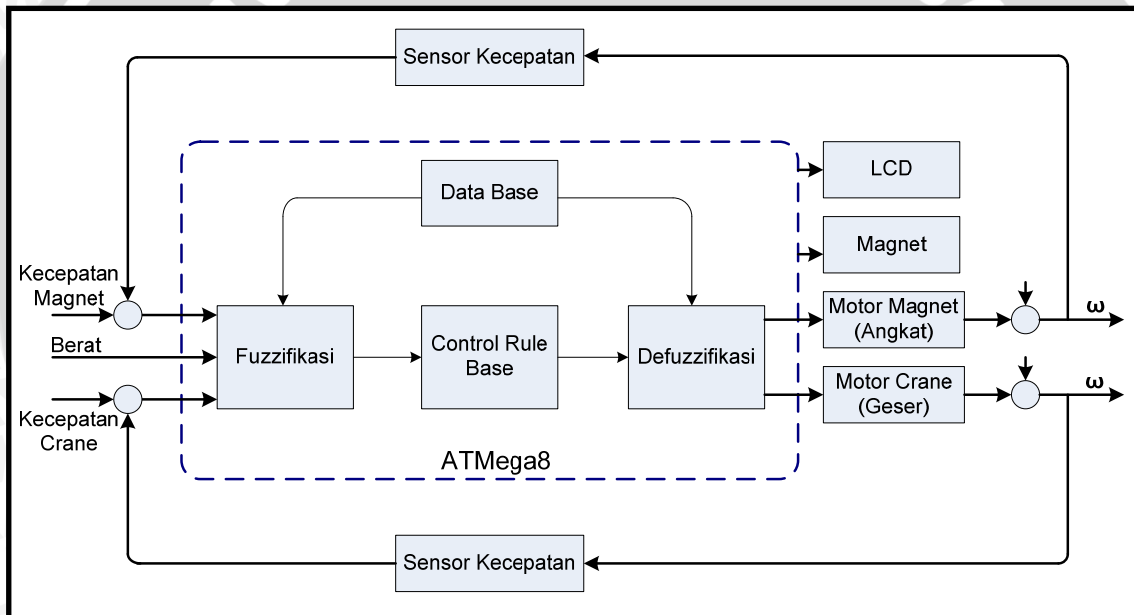
Kesimpulan diperoleh berdasarkan hasil perealisasiian dan pengujian pada sistem secara menyeluruh, akankah sesuai dengan tujuan dan rumusan masalah yang telah ditentukan. Sedangkan saran diberikan setelah melihat adanya kekurangan dalam sistem tersebut, dengan harapan sistem tersebut dapat dikembangkan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

BAB V PERANCANGAN DAN PEMBUATAN ALAT

Perancangan alat ini dilakukan secara bertahap yakni blok demi blok, sehingga akan memudahkan dalam penganalisaan disetiap bloknya maupun secara keseluruhan. Perancangan ini terdiri atas dua bagian yaitu perancangan perangkat keras (*hardware*) dan perancangan perangkat lunak (*software*). Beberapa aspek lain yang akan dijelaskan dalam bab ini adalah diagram blok sistem beserta prinsip kerjanya.

5.1 Diagram Blok Sistem

Diagram blok sistem pengendalian kecepatan pada prototipe crane magnet ini ditunjukkan dalam Gambar 5.1.



Gambar 5.1. Diagram Blok Sistem
Sumber: Perancangan

Keterangan blok diagram:

- Sensor berat digunakan untuk mendeteksi berat barang yang akan dipindahkan.
- Sensor kecepatan digunakan untuk mendeteksi kecepatan gerak crane magnet.
- Mikrokontroler berfungsi untuk mengendalikan keseluruhan proses kerja sistem.
- *LCD* digunakan untuk menampilkan kecepatan dan beban yang di pindahkan.
- Magnet berfungsi sebagai alat penanganan barang yang dipindahkan.
- Motor magnet berfungsi untuk menggerakkan naik turun magnet.

- Motor crane berfungsi untuk menggerakkan crane.

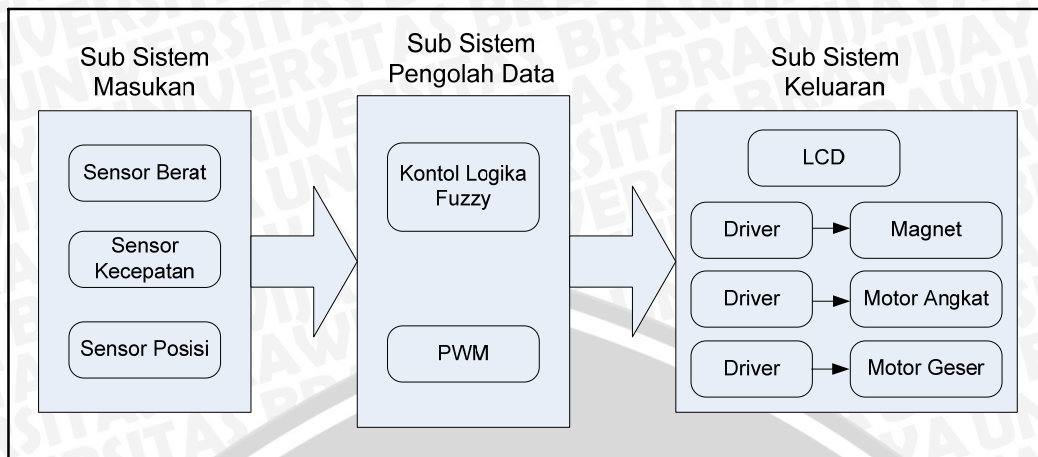
5.2 Prinsip Kerja Alat

Secara singkat prinsip kerja keseluruhan dari sistem ini adalah sebagai berikut:

- Jika tidak ada barang pada tempat barang, crane magnet akan tetap diam diposisinya, yaitu tepat diatas posisi barang yang akan dipindahkan.
- Jika ada barang yang terdeteksi oleh *limit switch* barang, magnet akan turun (motor magnet) sampai magnet menempel pada barang yang akan dipindahkan.
- Pada saat magnet menempel pada barang, *limit switch* magnet yang ada pada magnet akan aktif yang artinya bahwa magnet sudah menempel pada barang sehingga motor akan berhenti.
- Setelah barang menempel pada magnet, barang akan diangkat keatas dan motor berhenti ketika magnet menyentuh *limit switch* crane (magnet sudah sampai posisi paling atas).
- Kemudian barang akan digeser/dipindahkan oleh crane ke tempat tujuan, motor akan berhenti ketika crane menyentuh *limit switch* finish (crane sudah sampai tujuan).
- Setelah crane sampai diposisi tujuan, magnet (motor magnet) bergerak turun sampai barang sampai ditempat tujuan, yaitu pada saat barang menyentuh *limit switch* tujuan barang sehingga motor magnet akan berhenti.
- Setelah barang sampai ditempat tujuan, magnet akan melepas barang tersebut dengan memutuskan aliran arus dan kemudian magnet (motor magnet) akan naik keatas sampai magnet menempatkan posisinya pada crane dengan menyentuh *limit switch* crane.
- Kemudian crane akan bergerak atau kembali ke posisi semula hingga motor crane berhenti ketika menyentuh *limit switch* start, yaitu pada posisi start tepat diatas barang yang akan dipindahkan.
- Dan kemudian akan mendeteksi kembali barang yang akan dipindahkan, selama tidak ada barang yang terdeteksi maka crane akan tetap diam dan begitu seterusnya.

5.3 Perancangan Perangkat Keras (*Hardware*)

Pada perancangan perangkat keras (*hardware*) ini meliputi perancangan di setiap bloknya, sedangkan diagram blok perancangan perangkat keras dapat dilihat dalam Gambar 5.2.



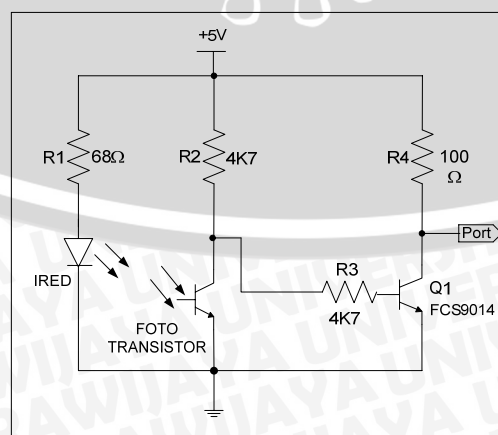
Gambar 5.2. Diagram Blok Perancangan Perangkat Keras
Sumber: Perancangan

5.3.1. Sub Sistem Masukan

Sub sistem ini terdiri tiga masukan yaitu kecepatan, berat dan posisi. Untuk masukan berupa kecepatan, diperoleh dari sensor kecepatan. Untuk masukan berupa berat, diperoleh dari sensor berat. Dan untuk masukan posisi adalah berupa rangkaian *limit switch*.

5.3.1.1. Perancangan Rangkaian Sensor Kecepatan

Optocoupler jenis U digunakan untuk mendeteksi kecepatan putaran motor DC, poros motor DC tersebut terhubung dengan sebuah disk berlubang. Perancangan rangkaian sensor kecepatan ini dapat dilihat dalam Gambar 5.3. Jika cahaya yang dipancarkan LED terhalang oleh bagian disk maka keluaran *optocoupler* akan menghasilkan tegangan sekitar 5 volt (logika tinggi). Sebaliknya, pada saat cahaya yang dipancarkan LED diterima oleh *phototransistor*, maka keluaran *optocoupler* akan menghasilkan tegangan sekitar 0 volt (logika rendah).



Gambar 5.3. Rangkaian Sensor Kecepatan
Sumber: Perancangan

Rangkaian sensor yang ditunjukkan dalam gambar di atas memiliki LED yang berfungsi sebagai pembangkit cahaya, sedangkan *phototransistor* sebagai sensor cahaya. Agar LED dapat menghasilkan cahaya yang dapat dideteksi oleh *phototransistor*, maka perlu dilakukan pengaturan arus. Berdasarkan *datasheet*, diperoleh data bahwa tegangan jatuh pada IRED (V_{IRED}) sebesar 1 Volt dan arus maksimal yang mengalir pada IRED (I_1) adalah 60mA. Maka besarnya nilai resistor (R_1) dapat ditentukan, yaitu:

$$R_{1(\min)} = \frac{V_{CC} - V_{IRED}}{I_1}$$

$$R_{1(\min)} = \frac{(5-1)\text{volt}}{60\text{mA}}$$

$$R_{1(\min)} = 66.7\Omega$$

$$R_1 \approx 68\Omega$$

Jadi nilai resistor (R_1) yang digunakan dalam perancangan ini adalah 68 Ω .

Kemudian jika *phototransistor* terkena cahaya dari LED, maka besar $V_{CE(\text{saturasi})}$ maksimum saat I_c sebesar 1,2 mA adalah 0 V. Sehingga besarnya nilai resistor (R_2) dapat ditentukan melalui persamaan berikut:

$$R_2 = \frac{V_{CC} - V_{ce(\text{on})}}{I_C}$$

$$R_2 = \frac{(5-0)V}{1,2\text{mA}}$$

$$R_2 = 4.17\text{k}\Omega$$

$$R_2 \approx 4\text{K}7$$

Jadi nilai resistor (R_2) yang digunakan dalam perancangan ini adalah 4K7.

Berdasarkan *datasheet*, transistor FCS9014 memiliki β_{\min} sebesar 100 dan arus kolektor (I_C) maksimal 100 mA. Maka nilai arus basis (I_B) maksimum adalah:

$$I_{B(\max)} = \frac{I_{C(\max)}}{\beta_{(\min)}}$$

$$I_{B(\max)} = \frac{100\text{mA}}{100}$$

$$I_{B(\max)} = 1\text{mA}$$

Untuk membatasi arus basis (I_B) agar tidak melebihi dari $I_{B(\max)}$ maka besar resistansi pada basis transistor (R_B) dapat diketahui dengan perhitungan berdasarkan

persamaan dibawah. Diinginkan arus yang mengalir pada basis agar transistor tetap saturasi pada level arus $I_C = 50 \text{ mA}$ adalah $0,5 \text{ mA}$, maka besar R_B :

$$V_{CC} = I_B(R_2 + R_3) + V_{BE}$$

$$R_B = R_3$$

$$5V = 0,5mA(4,7K + R_B) + 0,7V$$

$$4,3V = 0,5mA(4,7K + R_B)$$

$$R_B = \frac{4,3V}{0,5mA} - 4,7K$$

$$R_B = 8,6K - 4,7K = 3,9K$$

$$R_B \approx 4K7$$

Jadi nilai resistor (R_3/R_B) yang digunakan dalam perancangan ini adalah $4K7$.

Sedangkan untuk mencari nilai resistor pada kolektor (R_C) dapat ditentukan dengan mengetahui $I_C = 50 \text{ mA}$ dan $V_{CE} = 0$ pada saat transistor saturasi adalah:

$$R_C = \frac{V_{CC} - V_{CE(CO)}}{I_C}$$

$$R_C = R_4$$

$$R_C = \frac{V_{CC} - 0}{I_C}$$

$$R_4 = \frac{5V}{50mA}$$

$$R_4 = 100\Omega$$

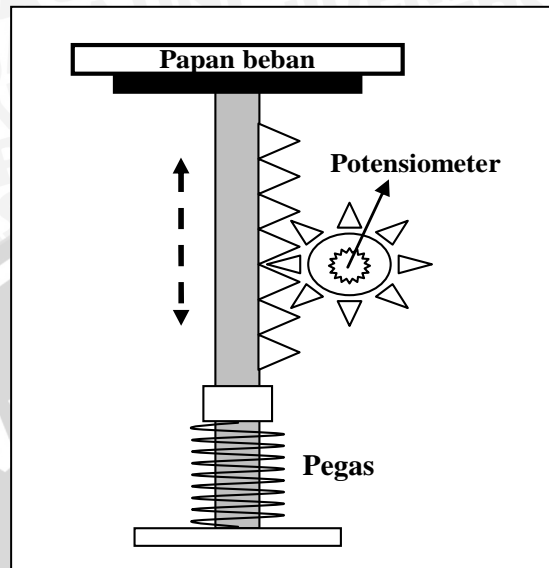
Jadi nilai resistor (R_4/R_C) yang digunakan dalam perancangan ini adalah 100Ω .

5.3.1.2. Perancangan Rangkaian Sensor Berat

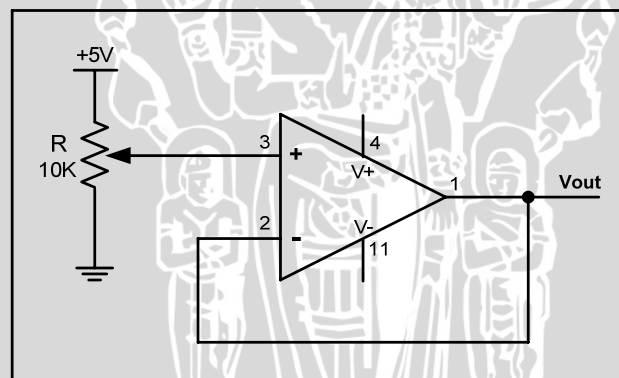
Sensor berat yang digunakan dalam perancangan ini adalah potensiometer *wire-wound*, yang ujung tuas penalnya dihubungkan pada papan beban yang diletakkan pada suatu poros. Perubahan massa beban akan menggeser posisi papan beban naik turun sesuai massa beban. Perubahan posisi naik turun dari papan beban akan menggeser posisi penala potensiometer. Dengan adanya perubahan posisi penala ini, nilai resistansi akan berubah, sehingga bila dihubungkan dengan sumber tegangan, maka tegangan keluaran akan berubah pula. Nilai tegangan yang dihasilkan sama dengan tegangan pada potensiometer yang difungsikan sebagai hambatan putar.

$$V_{out} = V_R$$

Perancangan sensor berat ini secara mekanik dapat dilihat dalam Gambar 5.4. Sedangkan rangkaian tegangan keluarannya ditunjukkan dalam Gambar 5.5.



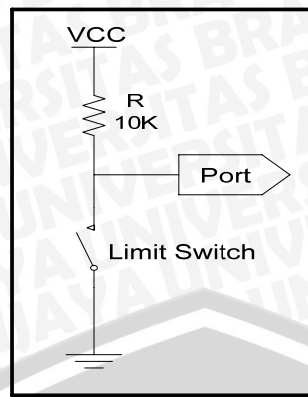
Gambar 5.4. Perancangan Sensor Berat Secara Mekanik
Sumber: Perancangan



Gambar 5.5. Rangkaian Tegangan Keluaran
Sumber: Perancangan

5.3.1.3. Perancangan Rangkaian *Limit Switch*

Dalam perancangan ini terdapat 6 buah *limit switch* sebagai sensor posisi yaitu *limit switch* barang, *limit switch* magnet, *limit switch* crane, *limit switch* start, *limit switch* finish dan *limit switch* tujuan. Untuk perancangan rangkaian sensor posisi *limit switch* dapat dilihat pada gambar 5.6.



Gambar 5.6. Rangkaian *Limit Switch*

Sumber: Perancangan

Pada saat *limit switch* dalam keadaan tertutup maka tegangan di keluaran (yang akan masuk ke IC ATmega8) sebesar 0 V, tetapi pada saat *limit switch* dalam keadaan terbuka, diharapkan tegangan keluaran mendekati 5 Volt. Kondisi ini dapat dicapai dengan pertimbangan, tegangan suply (V_{cc}) yang digunakan adalah 5 Volt, arus yang dibutuhkan masukan IC ATmega8 adalah 0.01mA dan tegangan keluaran yang diharapkan adalah mendekati 5 Volt sehingga analisisnya adalah sebagai berikut:

$$V_{cc} = I \cdot R + V_{out}$$

$$5 = (0,01 \cdot 10^{-3} \cdot R) + 4,9$$

$$R = \frac{0,1}{0,01 \cdot 10^{-3}}$$

$$R = 10000\Omega = 10k\Omega$$

Jadi nilai resistor (R) yang digunakan dalam perancangan ini adalah 10K.

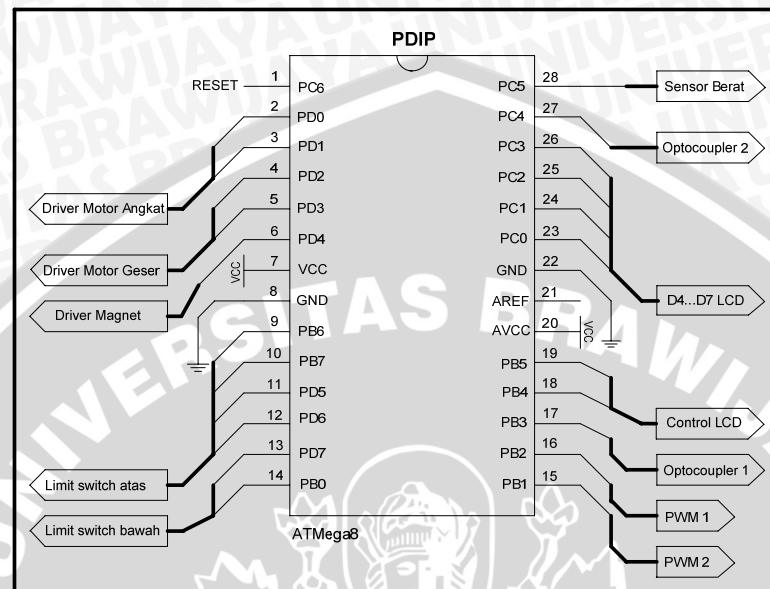
5.3.2. Sub Sistem Pengolah Data

Sesuai dengan Gambar 5.2 bahwa pada sub sistem ini merupakan tempat pemrosesan data yang berasal dari sub sistem masukan untuk kemudian diumpankan ke sub sistem keluaran.

5.3.2.1. Perancangan Mikrokontroler ATmega8

Mikrokontroler merupakan komponen utama dalam perancangan ini, karena komponen ini merupakan otak dari sistem agar dapat bekerja dengan baik dan optimal. Sistem mikrokontroler bertugas mengatur operasi perangkat keras lain yang meliputi dua sensor kecepatan, sensor berat, *limit switch*, dua buah driver motor dan driver magnet.

Mikrokontroler AVR digunakan sebagai kendali sistem alat ini. Pemilihan mikrokontroler ATmega8 mengingat AVR ini terdapat pin khusus sebagai PWM dan ADC. Selain itu AVR ini sangat mudah didapatkan dipasaran dengan harga yang relatif murah.



Gambar 5.7. Rangkaian Minimum Sistem

Sumber: Perancangan

Dalam Gambar 5.7 menunjukkan pin-pin rangkaian mikrokontroler yang digunakan dalam perancangan pengendalian kecepatan pada crane magnet ini. Adapun fungsi-fungsi pin-pin tersebut adalah sebagai berikut :

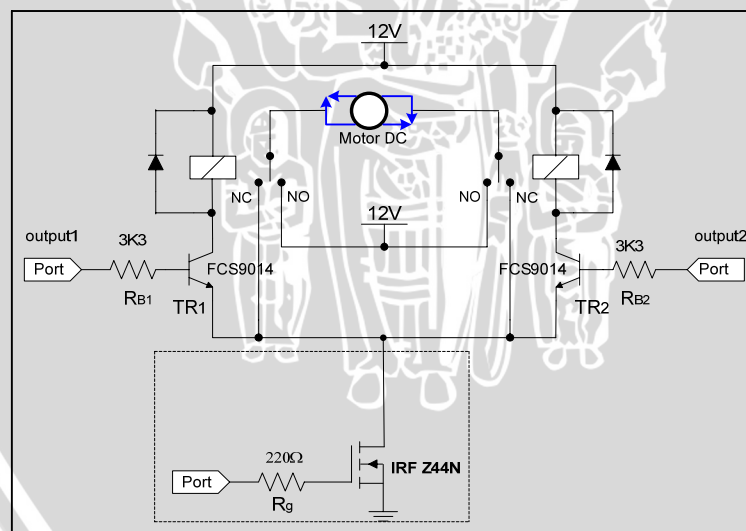
- PC.0 - PC.3 digunakan untuk penampilan data pada LCD.
- PB.4 - PB.5 digunakan untuk control LCD (pin RS & E)
- PB.2 digunakan untuk pengontrolan PWM 1 pada motor DC magnet (angkat).
- PB.1 digunakan untuk pengontrolan PWM 2 pada motor DC crane (geser).
- PC.5 dihubungkan dengan sensor berat.
- PC.4 dihubungkan dengan sensor kecepatan (*optocoupler 2*) motor DC magnet (angkat).
- PB.3 dihubungkan dengan sensor kecepatan (*optocoupler 1*) motor DC crane (geser).
- PB.6, PB.7, PD.5, PD.6 dihubungkan dengan *limit switch* Atas (*limit switch* start, *limit switch* magnet, *limit switch* crane dan *limit switch* finish).
- PB.0 & PD.7 dihubungkan dengan *Limit Switch* Bawah (*limit switch* barang dan *limit switch* tujuan)

- PD.2 dan PD.3 digunakan untuk mengendalikan relay untuk menggerakkan motor DC magnet (angkat).
- PD.0 dan PD.1 digunakan untuk mengendalikan relay untuk menggerakkan motor DC crane (geser).
- $\overline{\text{RESET}}$ difungsikan sebagai input pengendali reset.

5.3.3. Sub Sistem Keluaran

5.3.3.1 Rangkaian Driver Motor Magnet dan Crane

Rangkaian driver motor DC magnet dan motor DC crane memiliki rangkaian driver yang sama. Gambar rangkaian driver motor DC dapat dilihat pada gambar 5.8. Rangkaian *driver* digunakan untuk menggerakkan *relay* 12 volt yang dihubungkan ke motor. Jika output1 mikrokontroler berlogika *high* dan output2 mikrokontroler berlogika *low*, maka kontak *relay1* terhubung ke tegangan 12 volt dan kontak *relay2* terhubung ke GND sehingga motor akan bergerak kesuatu arah. Sedangkan apabila output1 mikrokontroler berlogika *low* dan output2 mikrokontroler berlogika *high*, maka kontak *relay1* terhubung ke GND dan kontak *relay2* akan terhubung ke tegangan 12 volt sehingga motor akan bergerak ke arah sebaliknya.



Gambar 5.8. Rangkaian Driver Motor
Sumber: Perancangan

Jika $R_{\text{relay}} = 400 \Omega$ (terukur), sedangkan *relay* membutuhkan arus sebesar:

$$\begin{aligned}
 I_{\text{relay}} &= I_C(\text{sat}) \\
 &= \frac{V_{CC}}{R_{\text{relay}}}
 \end{aligned}$$

$$I_{relay} = \frac{12}{400} = 30mA$$

Berdasarkan datasheet ATmega8, arus keluaran mikrokontroler pada saat logika high adalah 20mA. Sehingga dari data tersebut nilai $\beta_{(min)}$ yang diperlukan oleh sebuah transistor yang akan digunakan adalah:

$$\beta_{(min)} = \frac{I_{relay}}{I_{OH}}$$

$$\beta_{(min)} = \frac{30mA}{20mA}$$

$$\beta_{(min)} = 1,5$$

Sehingga transistor yang digunakan harus memiliki β lebih besar dari 1,5 dan arus I_C yang lebih besar dari 30 mA. Maka transistor FCS9014 memenuhi spesifikasi diatas, berdasarkan datasheet transistor ini memiliki β_{min} sebesar 100 dan arus kolektor (I_C) maksimal 100 mA.

Transistor FCS9014 memerlukan arus basis untuk memastikan transistor akan tetap saturasi pada level arus *relay*, sehingga arus basis transistor harus lebih besar daripada arus *relay* dibagi dengan penguatan minimum transistor.

$$I_B \gg \frac{I_{relay}}{\beta_{(min)}}$$

$$I_B \gg \frac{30mA}{100}$$

$$I_B \gg 0,3mA$$

Untuk nilai maksimum I_B adalah:

$$I_{B(max)} = \frac{I_{C(max)}}{\beta_{(min)}}$$

$$I_{B(max)} = \frac{100mA}{100}$$

$$I_{B(max)} = 1mA$$

Sehingga nilai resistor pembatas arus pada basis adalah:

$$V_{OH(min)} = I_B * R_B + V_{BE(max)}$$

$$R_B = \frac{V_{OH(\min)} - V_{BE(\max)}}{I_B}$$

$$R_B = \frac{(4,2 - 1)V}{1mA}$$

$$R_B = 3,3K$$

$$R_B = R_{B1} = R_{B2} = 3K3$$

Jadi nilai resistor (R_B) yang digunakan dalam perancangan ini adalah 3K3.

Untuk mencari nilai R_g dengan menggunakan rumus:

$$I_g(\max) = \frac{V_{OH}}{R_g(\min)}$$

$$20mA = \frac{4.2V}{R_g(\min)}$$

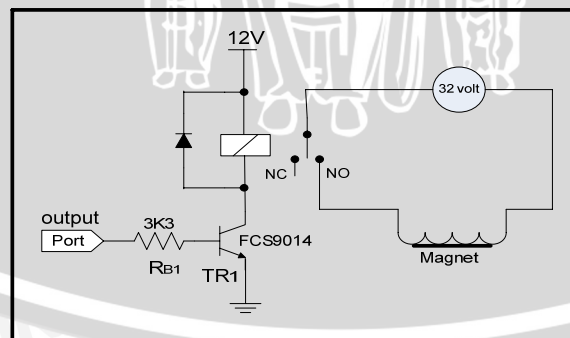
$$R_g(\min) = \frac{4.2V}{20mA}$$

$$R_g(\min) = 210\Omega \approx 220\Omega$$

Jadi nilai resistor (R_g) yang digunakan dalam perancangan ini adalah 220 Ω .

5.3.3.2 Rangkaian Driver Magnet

Rangkaian *driver* magnet digunakan untuk menggerakkan *relay* 12 volt yang dihubungkan pada magnet. Jika output mikrokontroler berlogika *high*, maka kontak *relay* terhubung ke tegangan 25 volt sehingga magnet akan aktif. Sedangkan apabila output mikrokontroler berlogika *low*, maka kontak *relay* terhubung ke NC sehingga magnet akan OFF.



Gambar 5.9. Rangkaian Driver Magnet

Sumber: Perancangan

Jika $R_{relay} = 400\ \Omega$ (terukur), sedangkan *relay* membutuhkan arus sebesar:

$$\begin{aligned}
 I_{relay} &= I_C(\text{sat}) \\
 &= \frac{V_{CC}}{R_{relay}} \\
 I_{relay} &= \frac{12}{400} = 30\text{mA}
 \end{aligned}$$

Berdasarkan datasheet ATmega8, arus keluaran mikrokontroler pada saat logika high adalah 20mA. Sehingga dari data tersebut nilai $\beta_{(\min)}$ yang diperlukan oleh sebuah transistor yang akan digunakan adalah:

$$\begin{aligned}
 \beta_{(\min)} &= \frac{I_{relay}}{I_{OH}} \\
 \beta_{(\min)} &= \frac{30\text{mA}}{20\text{mA}} \\
 \beta_{(\min)} &= 1,5
 \end{aligned}$$

Sehingga transistor yang digunakan harus memiliki β lebih besar dari 1,5 dan arus I_C yang lebih besar dari 30 mA. Maka transistor FCS9014 memenuhi spesifikasi diatas, berdasarkan datasheet transistor ini memiliki β_{\min} sebesar 100 dan arus kolektor maksimal 100 mA.

Transistor FCS9014 memerlukan arus basis untuk memastikan transistor akan tetap saturasi pada level arus *relay*, sehingga arus basis transistor harus lebih besar daripada arus *relay* dibagi dengan penguatan minimum transistor.

$$\begin{aligned}
 I_B &>> \frac{I_{relay}}{\beta_{(\min)}} \\
 I_B &>> \frac{30\text{mA}}{100} \\
 I_B &>> 0,3\text{mA}
 \end{aligned}$$

Untuk nilai maksimum I_B adalah:

$$\begin{aligned}
 I_{B(\max)} &= \frac{I_{C(\max)}}{\beta_{(\min)}} \\
 I_{B(\max)} &= \frac{100\text{mA}}{100}
 \end{aligned}$$

$$I_{B(\max)} = 1\text{mA}$$

Sehingga nilai resistor pembatas arus pada basis adalah:

$$V_{OH(\min)} = I_B * R_B + V_{BE(\max)}$$

$$R_B = \frac{V_{OH(\min)} - V_{BE(\max)}}{I_B}$$

$$R_B = \frac{(4,2 - 1)V}{1\text{mA}}$$

$$R_B = 3,3K$$

$$R_B = R_{B1} = R_{B2} = 3K3$$

Jadi nilai resistor (R_B) yang digunakan dalam perancangan ini adalah 3K3.

5.3.3.3 Liquid Crystal Display

LCD banyak digunakan pada alat-alat elektronika yang memerlukan penampilan, sehingga pemakai dapat mengerti dengan informasi yang ditampilkan oleh alat. Konfigurasi terminal I/O pada sebuah LCD biasanya akan tampak seperti dalam Tabel 5.1.

Tabel 5.1. Tabel Terminal I/O pada LCD

No	Simbo l	Level	Fungsi	
1	Vss	-	Power Supply	
2	Vcc	-		0V
3	Vee	-		5V ± 10%
4	RS	H/L	Kontras LCD	
5	R/W	H/L	H = data input L = instruction input	
6	E	H to L	H = read L = write	
7	DB0	H/L	Enable signal	
8	DB1	H/L		
9	DB2	H/L		
10	DB3	H/L		
11	DB4	H/L		
12	DB5	H/L		

Data bus

13	DB6	H/L	Data bus	
14	DB7	H/L		
15	V+BL	-	Back	4 – 4,2V
16	V-BL	-	light	0V (GND)

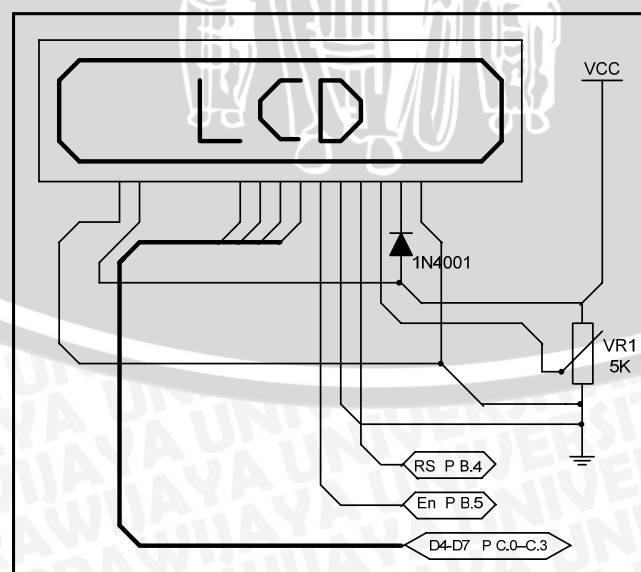
Sumber: Hitachi, 1998

Dari Tabel 5.1 terlihat bahwa LCD TM162ABC mempunyai 4 bit data (*bidirectional bus*) dan 3 buah sinyal kontrol yaitu RS, R/W dan E. Ketiga sinyal kontrol tersebut mempunyai fungsi sebagai berikut:

- 1 RS digunakan untuk memilih register yaitu register IR (*instruction register*) atau DR (*data register*)
- 2 R/W digunakan untuk memilih fungsi membaca atau menulis pada kedua *register* IR dan DR.
- 3 E berguna untuk memberikan sinyal pada bahwa data akan ditulis atau dibaca ke *register*.

Mode pengiriman data antara mikrokontroler dengan LCD dapat dilakukan dengan Mode transfer data 4 bit, semua jalur bus data (D4 sampai D7) digunakan dalam proses pengiriman.

LCD yang digunakan adalah LCD TM162ABC. LCD ini dikontrol oleh AVR ATmega8. Gambar rangkaian LCD TM162ABC dapat dilihat dalam Gambar 5.10.



Gambar 5.10. Rangkaian LCD

Sumber: Perancangan

Gambar 5.10 menunjukkan antarmuka LCD dengan mikrokontroler. Pin RS dari LCD terhubung ke P B.4 pada mikrokontroler dan Pin *Enable* dari LCD terhubung ke port P B.5 dari mikrokontroler. Seperti dilihat dalam Tabel 5.1 saat RS = 0 berfungsi untuk menulis instruksi ke LCD dan saat RS = 1 berfungsi untuk menulis data/karakter ke LCD.

5.4 Perancangan Kontroler Logika Fuzzy

Kontroler yang dirancang bertujuan untuk mengendalikan kecepatan motor ketika menggerakkan crane magnet, sehingga kecepatan motor dapat stabil sesuai dengan yang diinginkan. Kontroler yang digunakan adalah kontrol logika *fuzzy*, adapun tahapan dalam perancangan kontroler yang digunakan adalah sebagai berikut:

- a. Menentukan variabel masukan dan variabel keluaran.

Variabel masukan untuk kontrol logika *fuzzy* adalah kecepatan dan berat. Sedangkan untuk variabel keluarannya adalah sinyal kontrol berupa *duty cycle* yang akan mempengaruhi kecepatan dari motor DC.

- b. Menentukan jenis fungsi keanggotaan (*membership function*) masukan dan keluaran serta nilai dari labelnya.

Fungsi keanggotaan untuk masukan berbentuk *trapezoid* dan *triangular*, sedangkan untuk fungsi keanggotaan keluaran berbentuk *singleton*.

- c. Fuzzifikasi.

Fuzzifikasi adalah proses mengubah nilai masukan (*crisp input*) menjadi input *fuzzy*. Fungsi keanggotaan masukan untuk kontrol logika *fuzzy* adalah berat dan kecepatan.

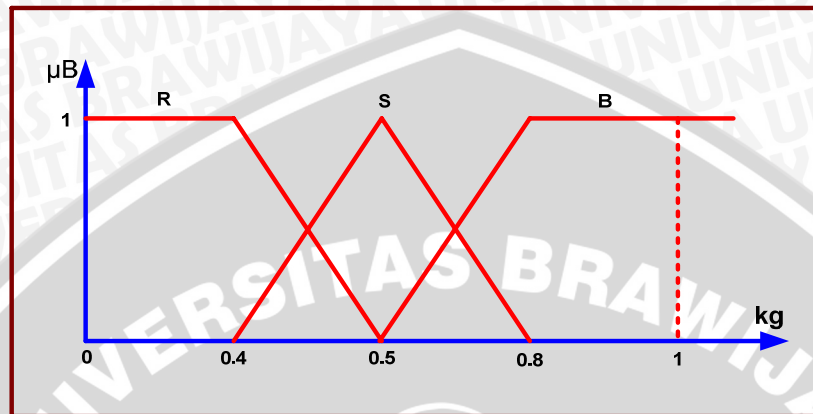
Untuk pengendalian kecepatan crane magnet ini terdapat dua pengendalian, yaitu pengendalian kecepatan motor magnet (angkat) dan pengendalian kecepatan motor crane (geser) sehingga terdapat dua fungsi keanggotaan masukan dan satu fungsi keanggotaan keluaran untuk pengendalian kecepatan motor magnet. Dan dua fungsi keanggotaan masukan dan satu fungsi keanggotaan keluaran untuk pengendalian kecepatan motor crane.

Fungsi keanggotaan untuk masukan berbentuk *trapezoid* dan *triangular*, sedangkan untuk fungsi keanggotaan keluaran berbentuk *singleton*.

1. Fungsi keanggotaan untuk pengendalian kecepatan motor magnet.

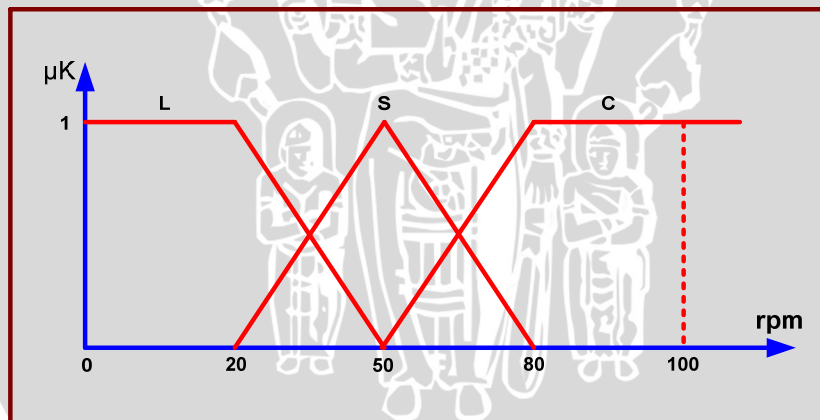
Fungsi keanggotaan masukan untuk pengendalian kecepatan motor magnet adalah berat dan kecepatan.

Fungsi keanggotaan berat mempunyai tiga label, yaitu ringan (R), sedang (S), dan Berat (B). Fungsi keanggotaan berat ditunjukkan dalam Gambar 5.11.



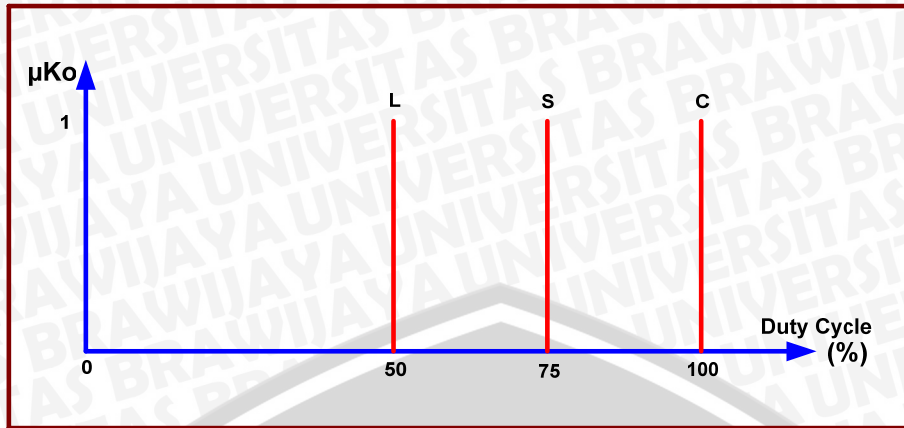
Gambar 5.11. Fungsi Keanggotaan *Crisp Input* Berat
Sumber: Perancangan

Fungsi keanggotaan kecepatan mempunyai tiga label, yaitu lambat (L), sedang (S), dan cepat (C). Fungsi keanggotaan kecepatan ditunjukkan dalam Gambar 5.12.



Gambar 5.12. Fungsi Keanggotaan *Crisp Input* Kecepatan
Sumber: Perancangan

Fungsi keanggotaan keluaran adalah kecepatan yang mempunyai tiga label, yaitu lambat (L), sedang (S), dan cepat (C). Fungsi keanggotaan keluaran ditunjukkan dalam Gambar 5.13.

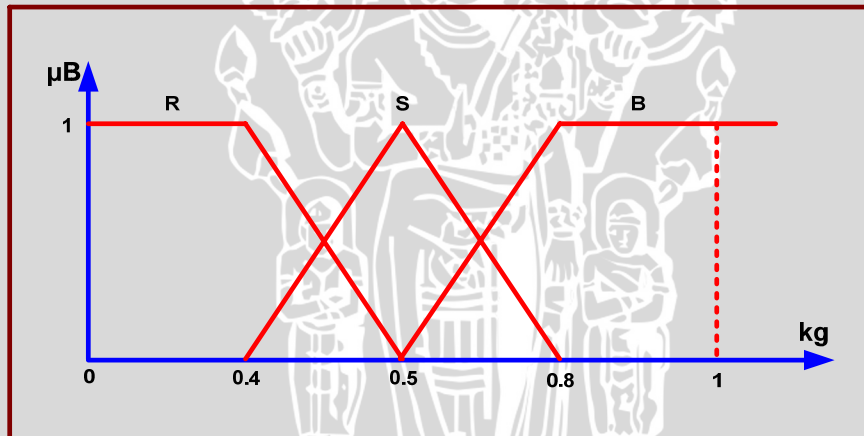


Gambar 5.13. Fungsi Keanggotaan *Crisp Output* Kecepatan Motor Magnet
 Sumber: Perancangan

2. Fungsi keanggotaan untuk pengendalian kecepatan motor crane.

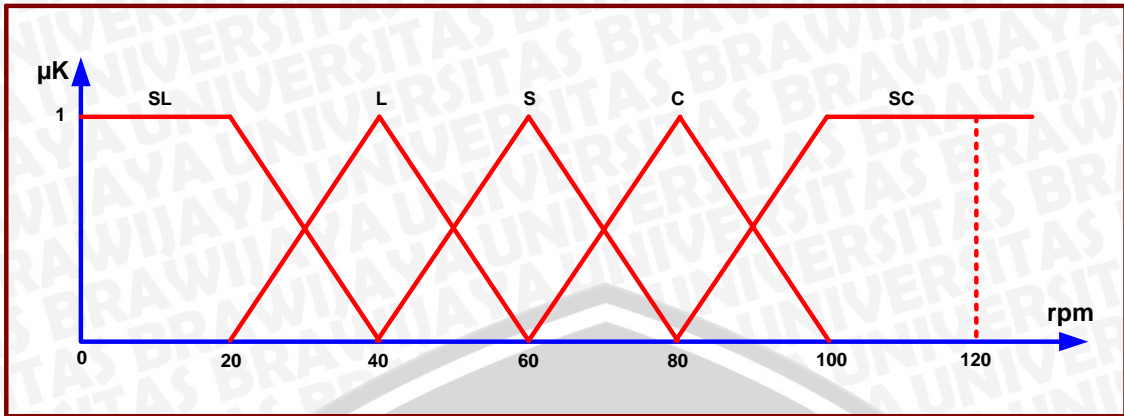
Fungsi keanggotaan masukan untuk pengendalian kecepatan motor crane adalah berat dan kecepatan.

Fungsi keanggotaan berat mempunyai tiga label, yaitu ringan (R), sedang (S), dan Berat (B). Fungsi keanggotaan berat ditunjukkan dalam Gambar 5.14.



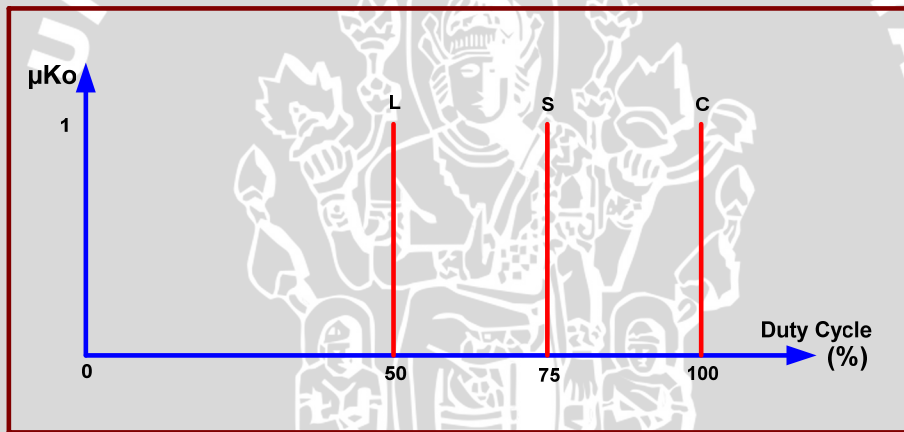
Gambar 5.14. Fungsi Keanggotaan *Crisp Input* Berat
 Sumber: Perancangan

Fungsi keanggotaan kecepatan mempunyai lima label, yaitu sangat lambat (SL), lambat (L), sedang (S), cepat (C), sangat cepat (SC). Fungsi keanggotaan kecepatan ditunjukkan dalam Gambar 5.15.



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

Fungsi keanggotaan keluaran adalah kecepatan yang mempunyai tiga label, yaitu lambat (L), sedang (S), dan cepat (C). Fungsi keanggotaan keluaran ditunjukkan dalam Gambar 5.16.



Gambar 5.16. Fungsi Keanggotaan *Crisp Output* Kecepatan Motor Crane
 Sumber: Perancangan

- d. Kaidah atur kontrol logika *fuzzy*.

Kaidah atur kontrol logika *fuzzy* didasarkan pada pengalaman dan dalam bentuk Jika-Maka. Setelah *crisp input* diubah menjadi *input fuzzy*, selanjutnya akan diolah sesuai dengan kaidah aturnya. Dalam pengendalian kali ini dibuat dua kaidah atur kontrol logika *fuzzy*.

Tabel 5.2. Tabel Kaidah Atur Pengaturan Kecepatan Motor Magnet

Kec Berat	Lambat	Sedang	Cepat
Ringan	Sedang	Lambat	Lambat
Sedang	Cepat	Sedang	Lambat
Berat	Cepat	Cepat	Sedang

Tabel 5.3. Tabel Kaidah Atur Pengaturan Kecepatan Motor Crane

Kec Berat	Sangat Lambat	Lambat	Sedang	Cepat	Sangat Cepat
Ringan	Sedang	Sedang	Sedang	Lambat	Lambat
Sedang	Cepat	Cepat	Sedang	Lambat	Lambat
Berat	Cepat	Cepat	Sedang	Sedang	Sedang

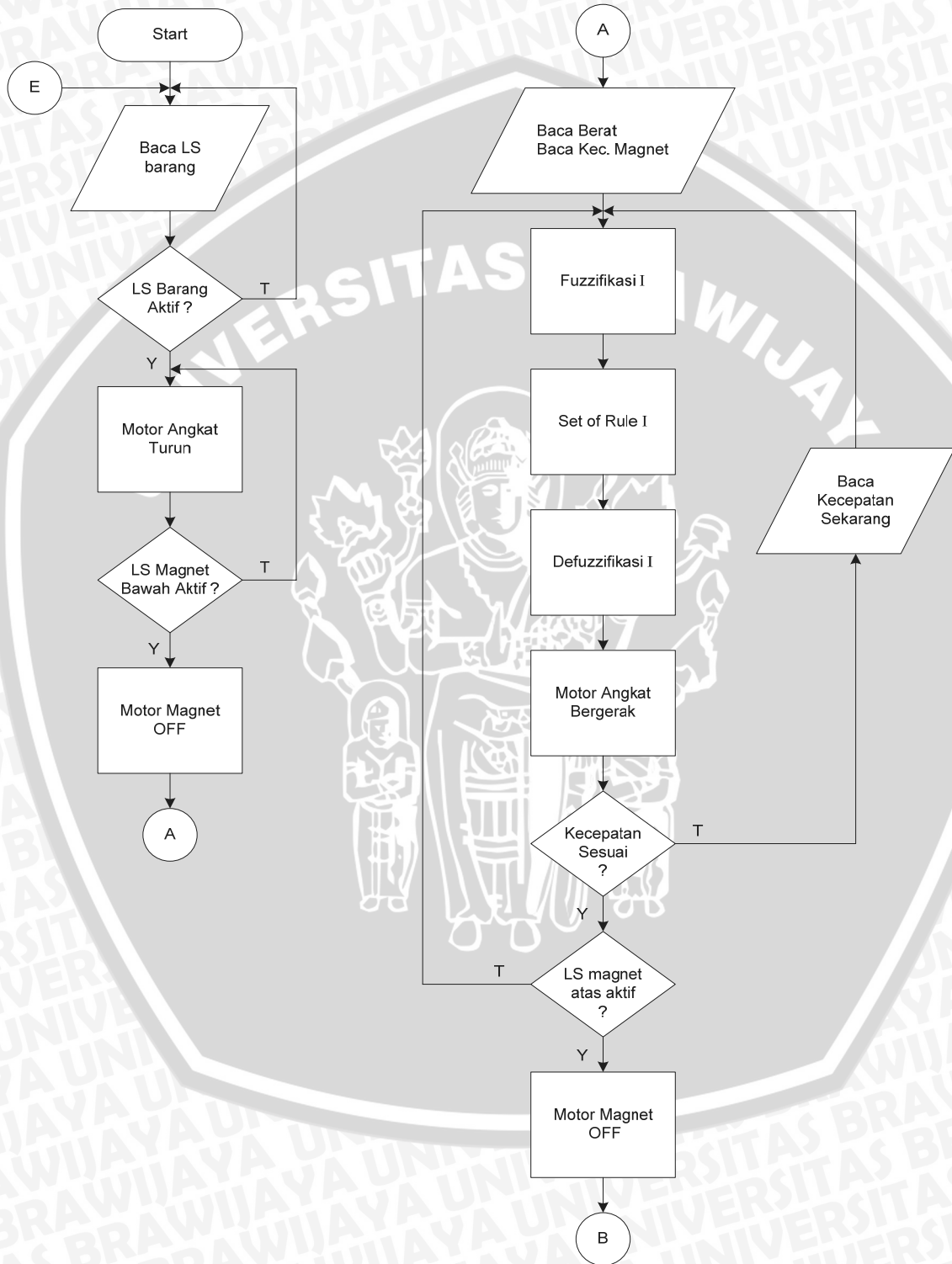
e. Defuzzifikasi

Defuzzifikasi adalah proses untuk merubah *output fuzzy* menjadi *crisp output*. Proses ini berfungsi untuk menentukan suatu nilai *crisp output*. *Crisp output* adalah suatu nilai analog yang akan kita butuhkan untuk mengolah data pada sistem yang telah dirancang. Prosesnya adalah untuk nilai fuzzy output yang berasal dari *rule evaluation* diambil kemudian dimasukkan ke dalam suatu *membership function output*. Bentuk bangun yang digunakan dalam membership function adalah *singleton* yaitu garis lurus vertikal ke atas, untuk besar nilai *fuzzy output* dinyatakan sebagai *degree of membership function output*, nilai-nilai tersebut kemudian dimasukkan ke dalam suatu rumus COG (*Center Of Gravity*) untuk mendapatkan hasil akhir yang disebut *crisp output*.

Hasil defuzzifikasi inilah yang digunakan untuk mengatur nilai *duty cycle*. Metode defuzzifikasi yang digunakan adalah COG (*Center Of Gravity*) karena kesederhanaan implementasinya pada komputasi mikrokontroler. Keluaran defuzzifikasi yang juga merupakan hasil akhir prosedur *fuzzy* merupakan nilai *duty cycle* PWM antara 0 sampai dengan 100% yang diperlukan untuk pengendalian motor. Nilai ini akan diolah lebih lanjut oleh algoritma pembangkit PWM.

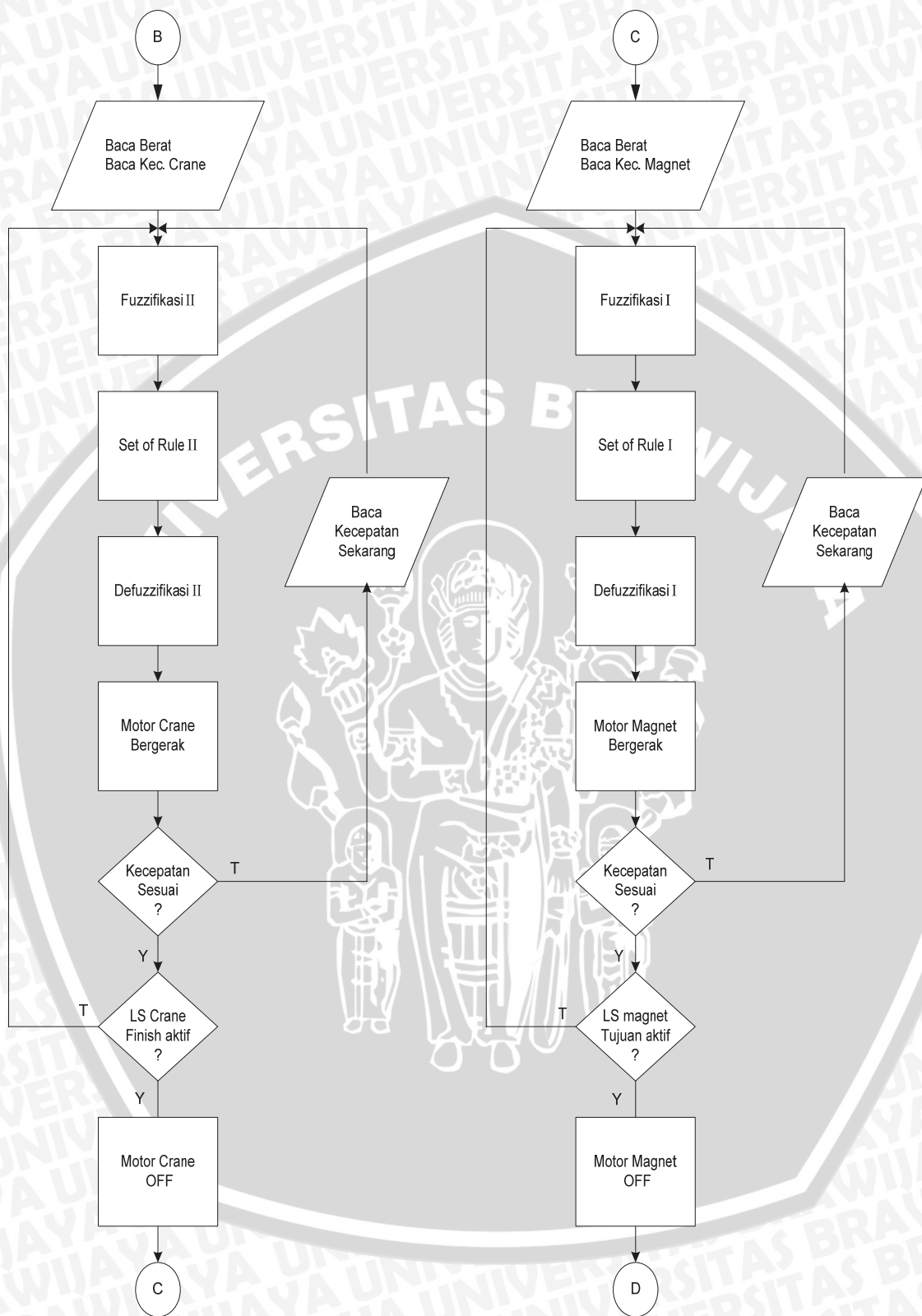
5.5 Perancangan Perangkat Lunak (*Software*)

Diagram alir perancangan perangkat lunak (*software*) dari sistem ini secara garis besar dapat dilihat pada Gambar 5.17abc.

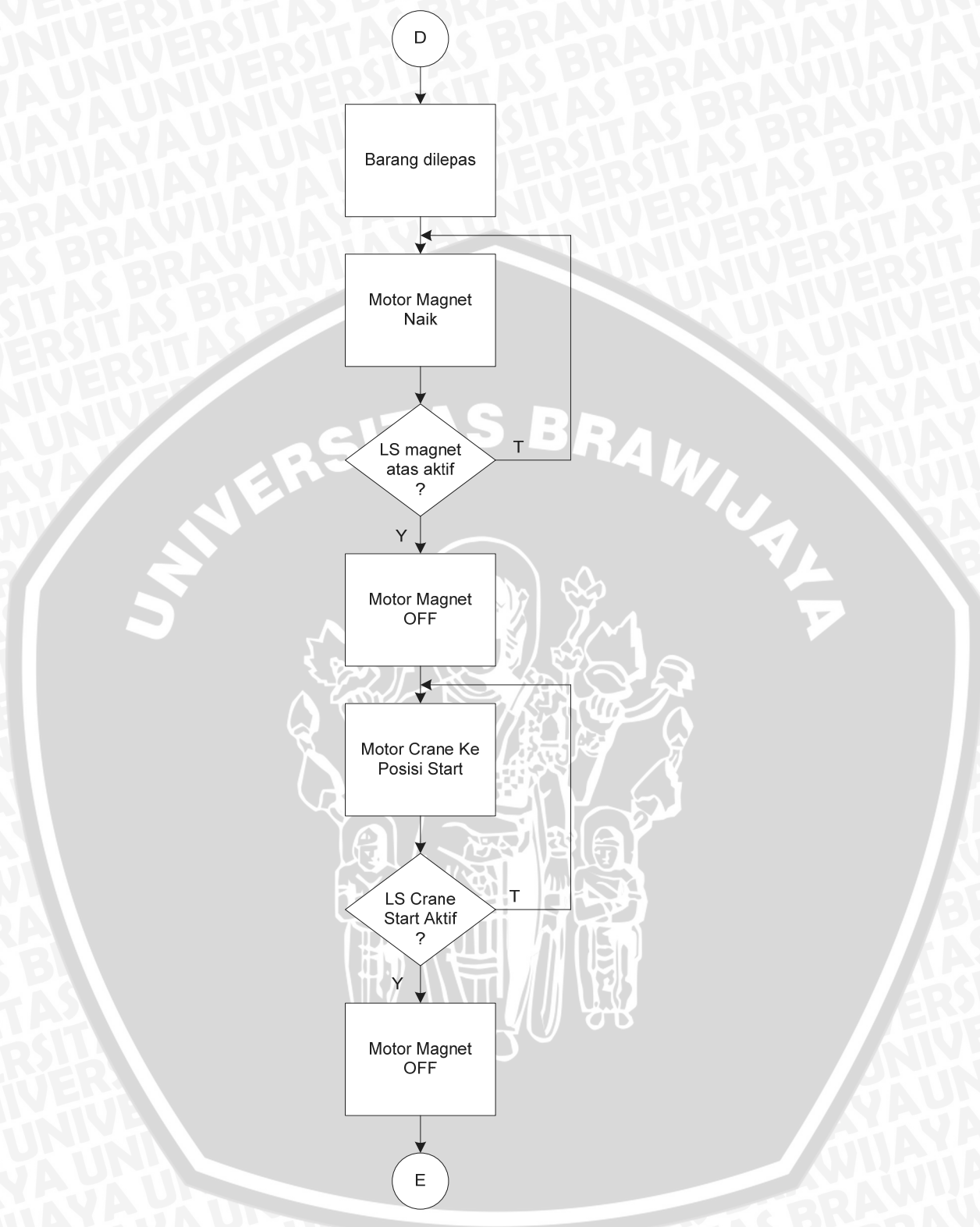


Gambar 5.17a. Diagram Alir Perangkat Lunak Sistem

Sumber: Perancangan



Gambar 5.17b. Lanjutan Diagram Alir Perangkat Lunak Sistem
Sumber: Perancangan



Gambar 5.17c. Lanjutan Diagram Alir Perangkat Lunak Sistem
 Sumber: Perancangan

BAB VI

PENGUJIAN ALAT

Tujuan pengujian alat adalah untuk menentukan apakah alat yang telah dibuat berfungsi dengan baik dan sesuai dengan perancangan. Pengujian ini meliputi pengujian setiap blok maupun pengujian secara keseluruhan. Pengujian setiap blok dilakukan untuk mempermudah analisis apabila alat tidak bekerja sesuai dengan perancangan. Pengujian yang dilakukan meliputi:

1. Pengujian sensor, yaitu: sensor kecepatan, sensor berat dan *limit switch*.
2. Pengujian Mikrokontroler.
3. Pengujian LCD.
4. Pengujian *driver* motor DC.
5. Pengujian sistem secara keseluruhan.

6.1. Pengujian Rangkaian Sensor

6.1.1. Pengujian Rangkaian Sensor Kecepatan

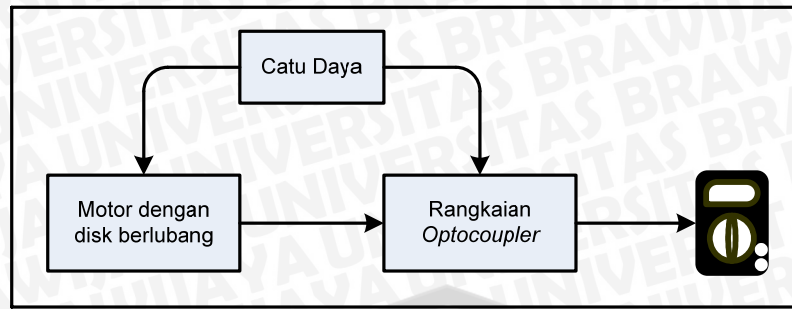
Tujuan pengujian rangkaian sensor kecepatan adalah untuk mengetahui level tegangan *output* rangkaian sensor kecepatan, ketika cahaya yang dipancarkan oleh LED terhalang atau tidak terhalang oleh bagian disk.

❖ Peralatan yang digunakan adalah:

- Rangkaian *optocoupler*.
- Motor dengan sebuah disk yang terdapat pada porosnya.
- Catu daya.
- Multimeter digital.

❖ Langkah pengujian meliputi:

- Merangkai blok pengujian rangkaian sensor kecepatan seperti Gambar 6.1.
- Mengaktifkan catu daya dan motor DC.
- Menghubungkan multimeter digital dengan *output* rangkaian sensor kecepatan.
- Mencatat hasil pengujian, yaitu berupa tegangan yang tampak pada multimeter digital.



Gambar 6.1. Diagram Blok Pengujian Rangkaian Sensor Kecepatan

❖ Hasil Pengujian dan Analisis.

Tabel 6.1. Hasil Pengujian Rangkaian Sensor Kecepatan

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

Dalam Tabel 6.1 dapat ditarik kesimpulan bahwa pada saat cahaya LED terhalang oleh bagian disk, maka sensor kecepatan akan berlogika “0”. Sedangkan pada saat cahaya LED tidak terhalang oleh bagian disk, maka sensor kecepatan akan berlogika “1”. Sehingga sensor kecepatan ini dapat berjalan dengan baik pada aplikasi yang telah dirancang.

6.1.2. Pengujian Pengukuran Kecepatan

Tujuan pengujian pengukuran kecepatan adalah untuk mengetahui tingkat kesalahan pengukuran kecepatan motor DC secara *software* dengan sensor kecepatan.

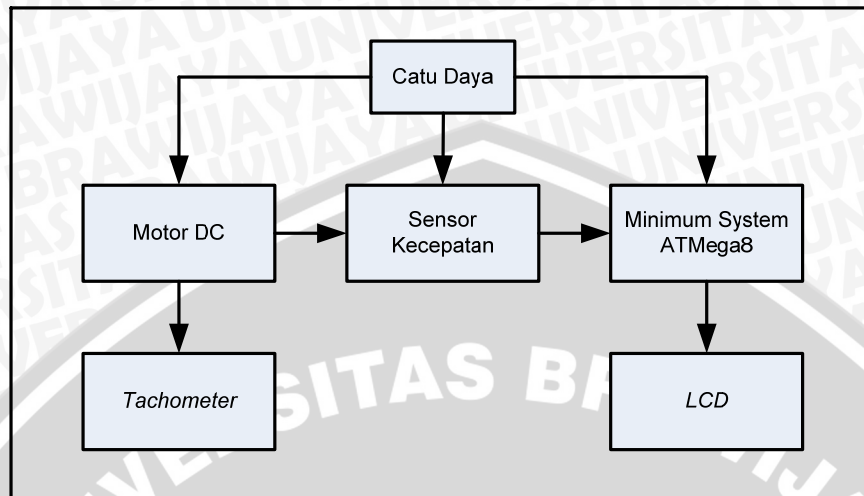
❖ Peralatan

- Rangkaian *optocoupler* sebagai sensor kecepatan.
- Catu Daya.
- Indikator LED 16 bit.
- Tachometer.

❖ Langkah Pengujian

- Merangkai blok pengujian pengukuran kecepatan seperti dalam Gambar 6.2.
- Mengaktifkan catu daya.

- Mengamati dan mencatat kondisi indikator LED 16 bit yang mewakili kecepatan motor yang terjadi serta pembacaan dari *tachometer* untuk setiap perubahan tegangan catu daya.



Gambar 6.2. Diagram Blok Pengujian Pengukuran Kecepatan

❖ Hasil Pengujian dan Analisis

Untuk menghitung kecepatan putaran motor dengan menggunakan *software* adalah sebagai berikut :

Ditetapkan: $T_{clock\ timer} = 125\ \mu s$

Periode putaran motor = Jumlah hitungan timer $\times 1\ \mu s$
 $= 8 \times register\ timer \times 125\ \mu s$

$$\begin{aligned}
 \text{Kecepatan putaran motor} &= \frac{1}{\text{Periode putaran motor}} \text{ rps} \\
 &= \frac{60}{\text{Periode putaran motor}} \text{ rpm} \\
 &= \frac{60 \times 10^6}{8 \times register\ timer \times 125} \text{ rpm} \\
 &= \frac{60 \times 10^3}{register\ timer} \text{ rpm}
 \end{aligned}$$

Tabel 6.2. Hasil Pengujian Kecepatan

Duty cycle (%)	Pembacaan Software		Pembacaan Tachometer (rpm)	Deviasi (rpm)	Kesalahan (%)
	Register Timer (desimal)	RPM			
20	0	0	0	0	0
40	4344	13.81	14	0.19	1.35
60	1245	48.19	49	0.81	1.65
80	837	71.68	73	1.32	1.80
100	612	98.03	100	1.99	1.97

Dalam Tabel 6.2 terlihat kesalahan data berkisar mulai 1.35% sampai dengan 1.97% dengan rata-rata kesalahan 1.69%. Dengan rata-rata kesalahan kurang dari 5% maka sensor kecepatan ini bisa dikatakan berjalan dengan baik pada aplikasi yang telah dirancang.

6.1.3. Pengujian Rangkaian Sensor Berat

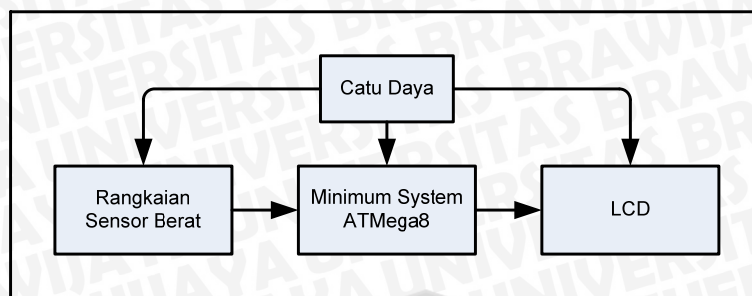
Tujuan pengujian rangkaian sensor berat adalah untuk mengetahui tegangan yang dihasilkan oleh rangkaian sensor berat pada saat diberikan beban mulai 0 kg sampai 1 kg.

❖ Peralatan yang digunakan adalah:

- Rangkaian sensor berat.
- Catu daya.
- Multimeter digital
- Box barang
- Timbangan

❖ Langkah Pengujian meliputi:

- Merangkai blok pengujian rangkaian sensor berat seperti dalam Gambar 6.3.
- Mengaktifkan catu daya.
- Menaruh timbel timbangan di atas sensor berat, mulai 0 kg sampai 1 kg dengan kenaikan beban 0,1 kg.
- Mencatat hasil pengujian, yakni berupa tegangan yang tampak pada multimeter digital untuk pengukuran tiap-tiap level beban.



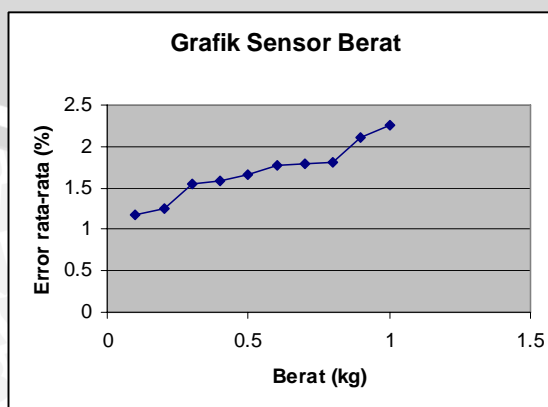
Gambar 6.3. Blok Diagram Pengujian Rangkaian Sensor Berat

❖ Hasil Pengujian dan Analisis

Tabel 6.3. Hasil Pengujian Rangkaian Sensor Berat

Berat Barang	Tampilan Pengujian LCD						Kesalahan rata-rata
	Ke-1	Ke-2	Ke-3	Ke-4	Ke-5	Ke-6	
0.1	0.101	0.102	0.101	0.102	0.101	0.1	1.167
0.2	0.204	0.201	0.203	0.204	0.201	0.202	1.250
0.3	0.305	0.306	0.302	0.304	0.305	0.306	1.556
0.4	0.409	0.405	0.407	0.409	0.401	0.407	1.583
0.5	0.512	0.508	0.507	0.509	0.509	0.505	1.667
0.6	0.61	0.611	0.61	0.611	0.612	0.61	1.778
0.7	0.713	0.711	0.713	0.712	0.712	0.714	1.786
0.8	0.816	0.815	0.814	0.812	0.813	0.817	1.813
0.9	0.919	0.92	0.919	0.917	0.919	0.92	2.111
1	1.021	1.019	1.02	1.028	1.023	1.024	2.250

Dalam Tabel 6.3. terlihat kesalahan data berkisar mulai 1.167% sampai dengan 2.250% dengan rata-rata kesalahan 1.70%. Dengan rata-rata kesalahan kurang dari 5% maka sensor berat ini bisa dikatakan berjalan dengan baik pada aplikasi yang telah dirancang.

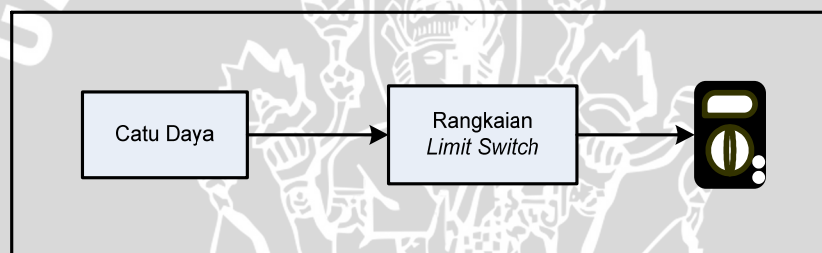


Gambar 6.4. Grafik Error Sensor Berat

6.1.4. Pengujian Rangkaian *Limit Switch*

Tujuan pengujian rangkaian *limit switch* adalah untuk mengetahui tegangan *output* rangkaian *limit switch* pada saat aktif dan tidak aktif.

- ❖ Peralatan yang digunakan adalah:
 - Rangkaian *limit switch*.
 - Catu daya.
 - Multimeter digital.
- ❖ Langkah Pengujian meliputi:
 - Merangkai blok pengujian rangkaian *limit switch* seperti dalam Gambar 6.5.
 - Mengaktifkan catu daya.
 - Memberikan perubahan kondisi pada *limit switch*.
 - Mencatat hasil pengujian, yakni berupa tegangan yang tampak pada multimeter digital.



Gambar 6.5. Diagram Blok Pengujian Rangkaian *Limit Switch*

- ❖ Hasil Pengujian dan Analisis

Tabel 6.4. Hasil Pengujian Rangkaian *Limit Switch*

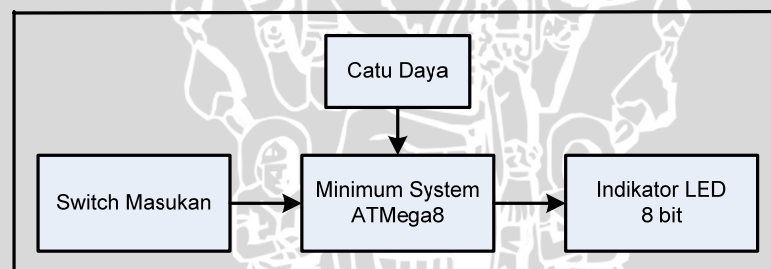
Kondisi <i>limit switch</i>	Tegangan pada multimeter digital (Volt)	Logika
Tidak Aktif	4.91	1
Aktif	0	0

Dalam Tabel 6.4. dapat ditarik kesimpulan bahwa pada saat *limit switch* aktif, maka rangkaian *limit switch* akan berlogika “1”. Sedangkan pada saat *limit switch* tidak aktif, maka rangkaian *limit switch* akan berlogika “0”. Sehingga rangkaian *limit switch* yang sebagai sensor posisi ini bekerja dengan baik pada aplikasi yang telah dirancang.

6.2. Pengujian Mikrokontroler

Tujuan pengujian mikrokontroler adalah untuk mengetahui kondisi awal dari sistem mikrokontroler agar sesuai dengan yang diharapkan.

- ❖ Peralatan yang digunakan adalah:
 - *Minimum system* ATmega8.
 - Catu daya.
 - Indikator LED 8 bit.
- ❖ Langkah Pengujian meliputi:
 - Merangkai blok pengujian mikrokontroler seperti dalam Gambar 6.6.
 - Mengisi mikrokontroler dengan program sederhana yakni meletakkan data 0Fh dan F0h pada *accumulator* secara bergantian kemudian mengeluarkan data tersebut melalui port 1 yang telah dihubungkan dengan indikator LED.
 - Mengaktifkan catu daya.
 - Mereset program untuk mengeluarkan data mikrokontroler.
 - Mencatat data *output* dalam bentuk biner yang diwakili oleh aktifnya indikator LED 8 bit.



Gambar 6.6. Diagram Blok Pengujian Mikrokontroler

- ❖ Hasil Pengujian dan Analisis

Tabel 6.5. Hasil Pengujian Mikrokontroler

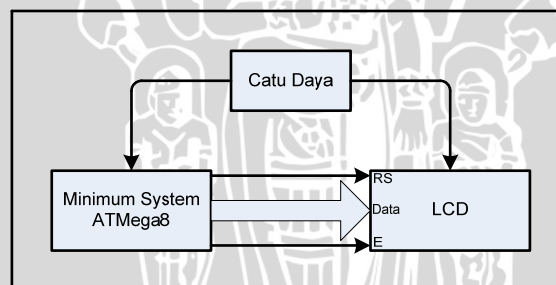
Kondisi	Keluaran pada Indikator LED							
	Bit 0	Bit 1	Bit 2	Bit 3	Bit 4	Bit 5	Bit 6	Bit 7
I	1	1	1	1	0	0	0	0
II	0	0	0	0	1	1	1	1

Jika diasumsikan indikator LED aktif berlogika “1” dan tidak aktif berlogika “0”, maka dalam Tabel 6.5. dapat ditarik kesimpulan bahwa port 1 mikrokontroler menghasilkan logika 0Fh dan F0h secara bergantian sesuai dengan program.

6.3. Pengujian Rangkaian LCD

Tujuan pengujian rangkaian LCD adalah untuk mengetahui tampilan pada *display LCD*, ketika diberi data masukan dari *minimum system ATmega8*.

- ❖ Peralatan yang digunakan adalah:
 - Display LCD.
 - Rangkaian *driver* LCD.
 - *Minimum system ATmega8*.
 - Catu daya.
 - Seperangkat komputer.
 - *Projectboard*.
- ❖ Langkah Pengujian meliputi:
 - Merangkai blok pengujian *display LCD* seperti dalam Gambar 6.7.
 - Menentukan yang ingin ditampilkan.
 - Mengaktifkan catu daya.
 - Membuat program untuk pengujian *display LCD* ini, kemudian *download* program tersebut ke dalam mikrokontroler.
 - Hasil pengujian dapat diketahui dengan cara melihat pada *display LCD* tersebut.



Gambar 6.7. Diagram Blok Pengujian LCD

- ❖ Hasil Pengujian dan Analisis



Gambar 6.8. Hasil Pengujian LCD.

Dalam Gambar 6.8. dapat disimpulkan bahwa LCD bekerja dengan baik, yaitu dengan dapat menampilkan karakter yang diinginkan sesuai dengan data yang dimasukkan.

6.4. Pengujian Rangkaian *Driver Motor DC*.

6.4.1. Pengujian Rangkaian *Driver Motor DC Magnet (angkat)*.

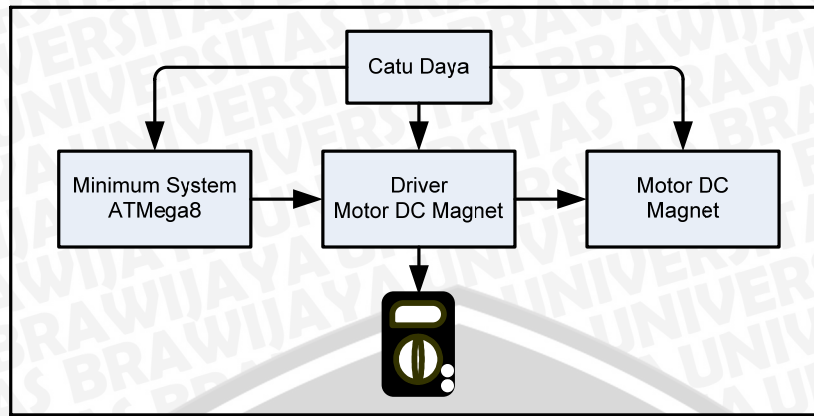
Tujuan pengujian rangkaian driver motor DC magnet adalah untuk mengetahui cara kerja *driver* motor DC sebagai pengangkat dan menurunkan magnet (barang).

- ❖ Peralatan yang digunakan adalah:
 - *Minimum system* ATmega8.
 - Catu daya 5V dan 12V.
 - *Driver Motor DC*.
 - Motor DC.
 - Multimeter digital.
- ❖ Langkah Pengujian meliputi:
 - Merangkai blok pengujian *driver* motor DC magnet seperti dalam Gambar 6.9.
 - Mengisi mikrokontroler dengan program sederhana yakni dengan memberikan data *input* pada pin PD.0 dan PD.1 serta input PWM pada pin PB.2 sesuai dengan tabel berikut:

Tabel 6.6. Data *Input* Rangkaian *Driver Motor DC Magnet*

Kondisi	Logika PD.0	Logika PD.1	Logika PB.2	Kondisi Motor Crane
1	0	0	1	Berhenti
			0	Berhenti
2	1	0	1	Putar Kiri
			0	Berhenti
3	0	1	1	Putar Kanan
			0	Berhenti
4	1	1	1	Berhenti
			0	Berhenti

- Mengaktifkan catu daya dan mereset mikrokontroler.
- Mengamati dan mencatat hasil pengujian, yakni berupa tegangan yang tampak pada multimeter digital dan gerakan motor magnet.



Gambar 6.9. Diagram Blok Pengujian Rangkaian *Driver* Motor DC Magnet

❖ Hasil Pengujian dan Analisis

Tabel 6.7. Hasil Pengujian Rangkaian *Driver* Motor DC Magnet

Tegangan PD.0 (Volt)	Tegangan PD.1 (Volt)	Output Motor (Volt)	Gerak Motor Crane
0	0	0	Diam
4,97	0	11,92	Putar Kiri
0	4,97	11,92	Putar Kanan
4,97	4,97	0	Diam

Dalam Tabel 6.7 dapat disimpulkan bahwa motor magnet bergerak ketika pin PD.0 berlogika *high* (1) sedangkan pin PD.1 berfungsi untuk membalik polaritas dari motor DC. Maka rangkaian *driver* ini bekerja sesuai dengan perancangan.

6.4.2. Pengujian Rangkaian *Driver* Motor DC Crane (geser).

Tujuan pengujian rangkaian driver motor DC crane adalah untuk mengetahui cara kerja *driver* motor DC sebagai penggeser crane.

❖ Peralatan yang digunakan adalah:

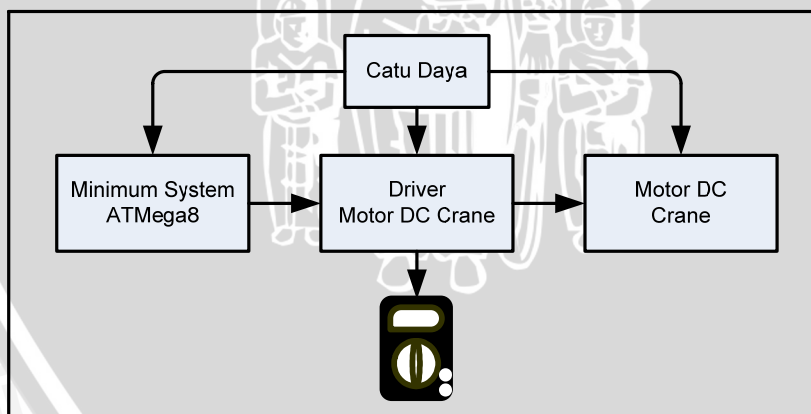
- *Minimum system* ATmega8.
- Catu daya 5V dan 24V.
- *Driver* Motor DC.
- Motor DC.
- Multimeter digital.

- ❖ Langkah Pengujian meliputi:
 - Merangkai blok pengujian *driver* motor DC seperti dalam Gambar 6.10.
 - Mengisi mikrokontroler dengan program sederhana yakni dengan memberikan data *input* pada pin PD.2 dan PD.3 serta input PWM pada pin PB.1 sesuai dengan tabel berikut:

Tabel 6.8. Data *Input* Rangkaian *Driver* Motor DC

Kondisi	Logika PD.2	Logika PD.3	Logika PB.1	Kondisi Motor Crane
1	0	0	1	Berhenti
			0	Berhenti
2	1	0	1	Putar Kiri
			0	Berhenti
3	0	1	1	Putar Kanan
			0	Berhenti
4	1	1	1	Berhenti
			0	Berhenti

- Mengaktifkan catu daya dan mereset mikrokontroler.
- Mengamati dan mencatat hasil pengujian, yakni berupa tegangan yang tampak pada multimeter digital dan gerakan motor crane.



Gambar 6.10. Diagram Blok Pengujian Rangkaian *Driver* Motor DC Crane.

❖ Hasil Pengujian dan Analisis

Tabel 6.9. Hasil Pengujian Rangkaian *Driver* Motor DC Magnet

Tegangan PD.2 (Volt)	Tegangan PD.3 (Volt)	Output Motor (Volt)	Gerak Motor Crane
0	0	0	Diam
4,97	0	23,74	Putar Kiri
0	4,97	23,74	Putar Kanan
4,97	4,97	0	Diam

Dalam Tabel 6.9. dapat disimpulkan bahwa motor crane (geser) bergerak ketika pin PD.2 berlogika *high* (1) sedangkan pin PD.3 berfungsi untuk membalik polaritas dari motor DC crane. Maka rangkaian *driver* ini bekerja sesuai dengan perancangan.

6.5. Pengujian Rangkaian *Driver* Magnet.

Tujuan pengujian rangkaian *driver* magnet adalah untuk mengetahui cara kerja *driver* magnet sebagai alat penanganan barang jenis logam.

❖ Peralatan yang digunakan adalah:

- *Minimum system* ATmega8.
- Catu daya 5V, 12V dan 24Volt.
- *Driver* Magnet.
- Multimeter digital.

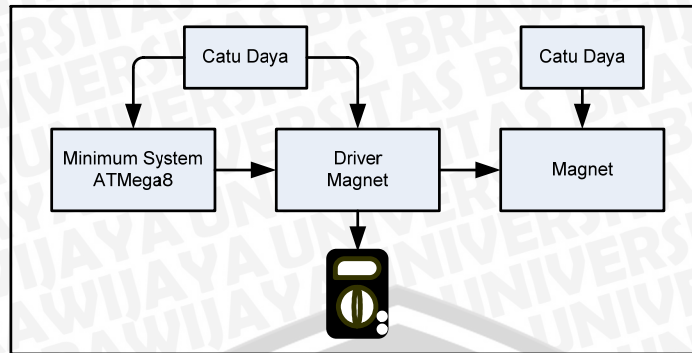
❖ Langkah Pengujian meliputi:

- Merangkai blok pengujian *driver* magnet seperti dalam Gambar 6.11.
- Mengisi mikrokontroler dengan program sederhana yakni dengan memberikan data *input* pada pin PD.4 sesuai dengan tabel berikut:

Tabel 6.10. Data *Input* Rangkaian *Driver* Magnet

Kondisi	Logika PD.4	Kondisi Magnet
1	0	Tidak Aktif
2	1	Aktif

- Mengaktifkan catu daya dan mereset mikrokontroler.
- Mengamati dan mencatat hasil pengujian, yakni berupa tegangan yang tampak pada multimeter digital dan kondisi magnet.



Gambar 6.11. Diagram Blok Pengujian Rangkaian *Driver Magnet*.

❖ Hasil Pengujian dan Analisis

Tabel 6.11. Hasil Pengujian Rangkaian *Driver Magnet*

Tegangan PD.4 (Volt)	Output Magnet (Volt)	Kondisi Magnet
0	0	Tidak Aktif
4,97	23,96	Aktif

Dalam Tabel 6.11. dapat disimpulkan bahwa magnet akan aktif ketika pin PD.4 berlogika *high* (1). Maka rangkaian *driver magnet* ini bekerja sesuai dengan perancangan.

6.6. Pengujian Sistem Secara Keseluruhan

Tujuan pengujian sistem ini adalah untuk mengetahui unjuk kerja perangkat lunak (*software*) dengan perangkat keras (*hardware*) ketika diintegrasikan bersama.

❖ Peralatan yang digunakan adalah:

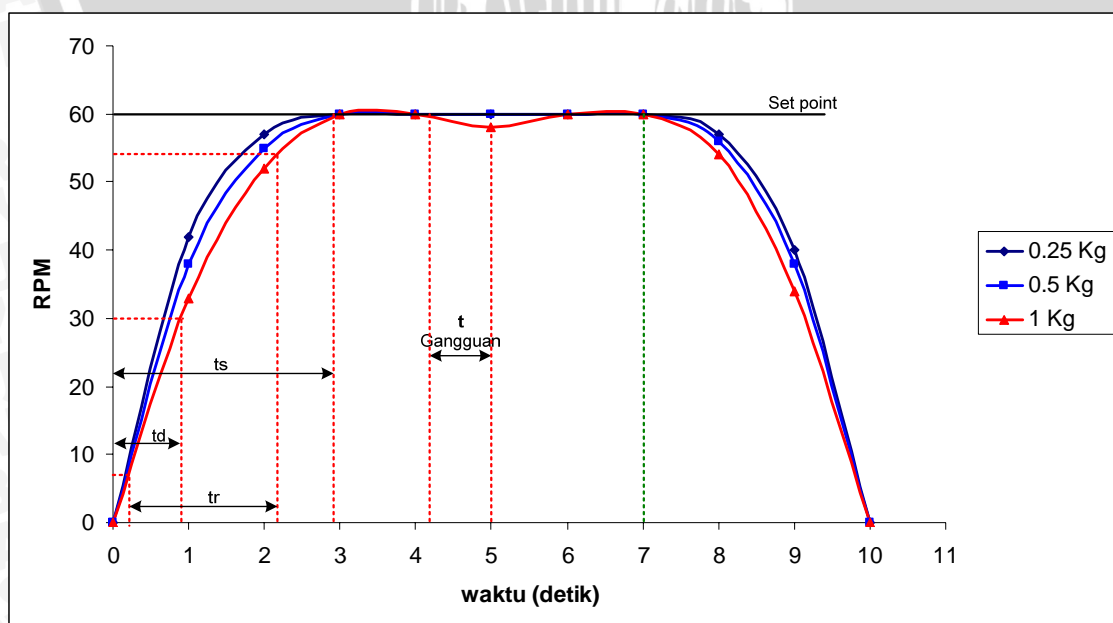
- Prototipe crane magnet.
- *Minimum system* ATmega8.
- Catu daya 5V, 12V dan 24V.
- *Stopwatch*.
- *Driver motor* DC.
- Motor DC.
- *LCD*.
- Rangkaian sensor kecepatan.
- Rangkaian sensor berat.
- Rangkaian *limit switch*.
- *Push Button*.

- ❖ Langkah Pengujian meliputi:
 - Menggabungkan seluruh perangkat keras (*hardware*) beserta programnya dengan mekanik prototipe crane magnet.
 - Mengaktifkan catu daya.
 - Memasukkan beban pada tempat yang sudah ditentukan, yaitu tepatnya pada sensor berat.
 - Mengamati dan mencatat hasil pengujian.
- ❖ Hasil Pengujian dan Analisis

Untuk kecepatan motor magnet adalah 60 RPM, sedangkan untuk motor crane adalah 80 RPM. Beban diberikan mulai 0.25 Kg, 0.5 Kg dan 1 Kg.

Tabel 6.12. Hasil Pengujian Motor Magnet Naik dengan Set Point Kecepatan 60 RPM

Waktu (detik)	RPM setelah diberikan beban		
	0.25 Kg	0.5 Kg	1 Kg
0	0	0	0
1	42	38	33
2	57	55	52
3	60	60	60
4	60	60	60
5	60	60	58
6	60	60	60
7	60	60	60
8	57	56	54
9	40	38	34
10	0	0	0



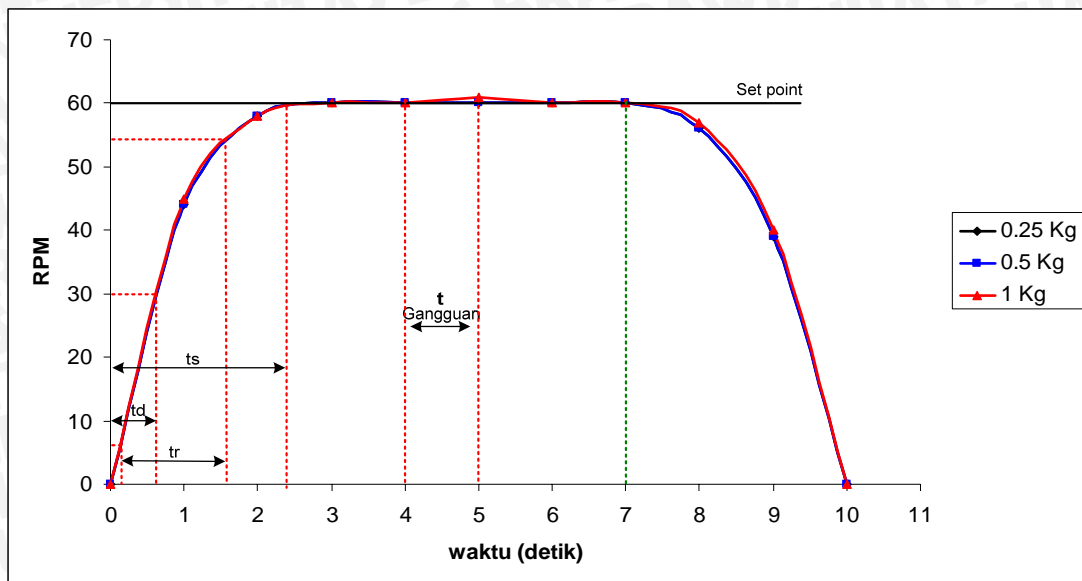
Gambar 6.12. Grafik Keluaran Motor Magnet Naik 60 RPM..

Dengan melihat grafik diatas maka didapatkan data (untuk beban 1 kg) sebagai berikut:

1. Waktu tunda, t_d : waktu yang diperlukan respon untuk mencapai setengah nilai akhir untuk waktu yang pertama.
 - t_d yang didapatkan dari pengujian ini adalah ± 0.95 detik.
2. Waktu naik, t_r : waktu yang diperlukan respon untuk naik dari 10 - 90 % dari harga akhirnya.
 - t_r yang didapatkan dari pengujian ini adalah ± 2.1 detik
3. Waktu turun, t_s : waktu yang diperlukan untuk mencapai *steady state*.
 - t_s yang diperoleh adalah ± 2.95 detik.
4. Mekanisme pemberian gangguan pada kecepatan berupa pemberian beban tambahan pada magnet sebesar 0.25 Kg. Dan didapatkan penyimpangan kecepatan dari set point sebesar 2 RPM sehingga terjadi *error steady state* sebesar ± 3.33 %.

Tabel 6.13. Hasil Pengujian Motor Magnet Turun dengan Set Point Kecepatan 60 RPM

Waktu (detik)	RPM setelah diberikan beban		
	0.25 Kg	0.5 Kg	1 Kg
0	0	0	0
1	44	44	45
2	58	58	58
3	60	60	60
4	60	60	60
5	60	60	61
6	60	60	60
7	60	60	60
8	56	56	57
9	39	39	40
10	0	0	0



Gambar 6.13. Grafik Keluaran Motor Magnet Turun 60 RPM.

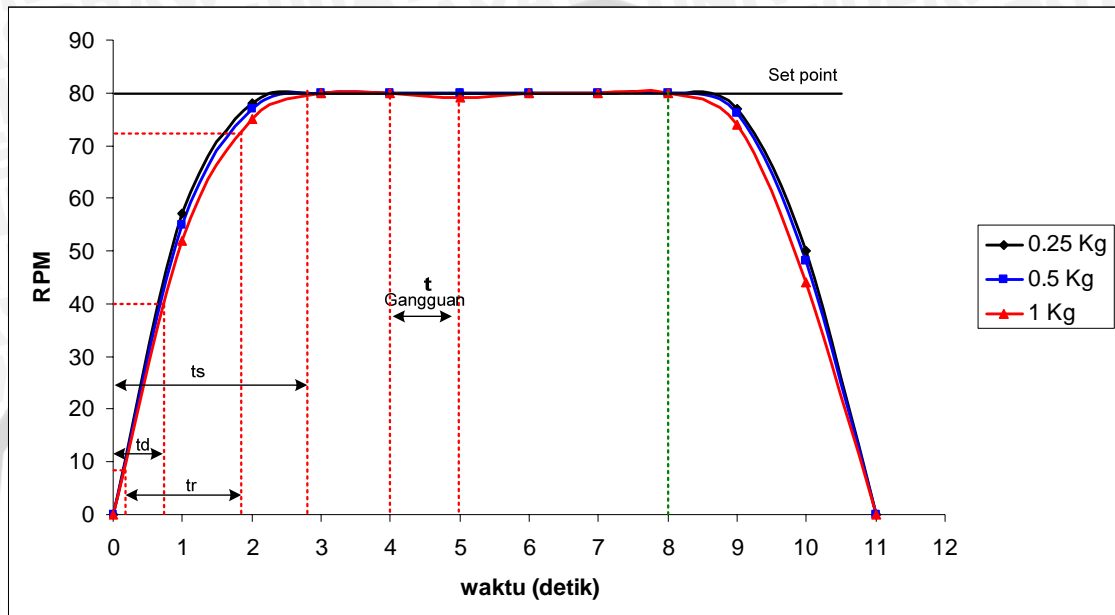
Dengan melihat grafik diatas maka didapatkan data (untuk beban 1 kg) sebagai berikut:

1. Waktu tunda, t_d : waktu yang diperlukan respon untuk mencapai setengah nilai akhir untuk waktu yang pertama.
 - t_d yang didapatkan dari pengujian ini adalah ± 0.7 detik.
2. Waktu naik, t_r : waktu yang diperlukan respon untuk naik dari 10 - 90 % dari harga akhirnya.
 - t_r yang didapatkan dari pengujian ini adalah ± 1.7 detik
3. Waktu turun, t_s : waktu yang diperlukan untuk mencapai *steady state*.
 - t_s yang diperoleh adalah ± 2.3 detik.
4. Mekanisme pemberian gangguan pada kecepatan berupa pemberian beban tambahan pada magnet sebesar 0.25 Kg. Dan didapatkan penyimpangan kecepatan dari set point sebesar 1 RPM sehingga terjadi *error steady state* sebesar ± 1.67 %

Tabel 6.14. Hasil Pengujian Motor Crane (geser) dengan Set Point Kecepatan 80 RPM

Waktu (detik)	RPM setelah diberikan beban		
	0.25 Kg	0.5 Kg	1 Kg
0	0	0	0
1	57	55	52
2	78	77	75
3	80	80	80
4	80	80	80
5	80	80	79

6	80	80	80
7	80	80	80
8	80	80	80
9	77	76	74
10	50	48	44
11	0	0	0



Gambar 6.14. Grafik Keluaran Motor Crane (geser) 80 RPM.

Dengan melihat grafik diatas maka didapatkan data (untuk beban 1 kg) sebagai berikut:

1. Waktu tunda, t_d : waktu yang diperlukan respon untuk mencapai setengah nilai akhir untuk waktu yang pertama.
 - t_d yang didapatkan dari pengujian ini adalah ± 0.8 detik.
2. Waktu naik, t_r : waktu yang diperlukan respon untuk naik dari 10 - 90 % dari nilai akhirnya.
 - t_r yang didapatkan dari pengujian ini adalah ± 1.95 detik
3. Waktu turun, t_s : waktu yang diperlukan untuk mencapai *steady state*.
 - t_s yang diperoleh adalah ± 2.9 detik.
4. Mekanisme pemberian gangguan pada kecepatan berupa pemberian beban tambahan pada crane sebesar 0.5 Kg. Dan didapatkan penyimpangan kecepatan dari set point sebesar 1 RPM sehingga terjadi *error steady state* sebesar ± 1.25 %

BAB VII

KESIMPULAN DAN SARAN

7.1. Kesimpulan

Setelah melakukan perancangan, pembuatan, dan dilanjutkan dengan pengujian beserta analisis sistem, maka dapat diambil kesimpulan sebagai berikut:

- a. Sensor kecepatan yang digunakan mampu mendeteksi kecepatan motor DC sebagai penggerak crane dengan prosentase kesalahan pendeteksian sebesar 1.69%. Sensor berat yang digunakan mampu mengubah beban menjadi tegangan sesuai dengan beban yang diberikan dengan rata-rata kesalahan pengukuran sebesar 1.70%.
- b. Kontroler yang digunakan bekerja sesuai dengan perancangan, sehingga pergerakan crane magnet sesuai dengan set point kecepatan yang sudah ditentukan dengan adanya perubahan beban barang yang dipindahkan.
- c. Semakin berat beban yang dipindahkan oleh crane magnet, maka waktu untuk mencapai *steady state* untuk bergerak naik agak lebih lama. Hal ini dikarenakan ketika bergerak ke atas, magnet melawan gaya gravitasi sehingga memperlambat pergerakannya. Namun ketika crane magnet bergerak turun, waktu untuk mencapai *steady state* agak lebih cepat.

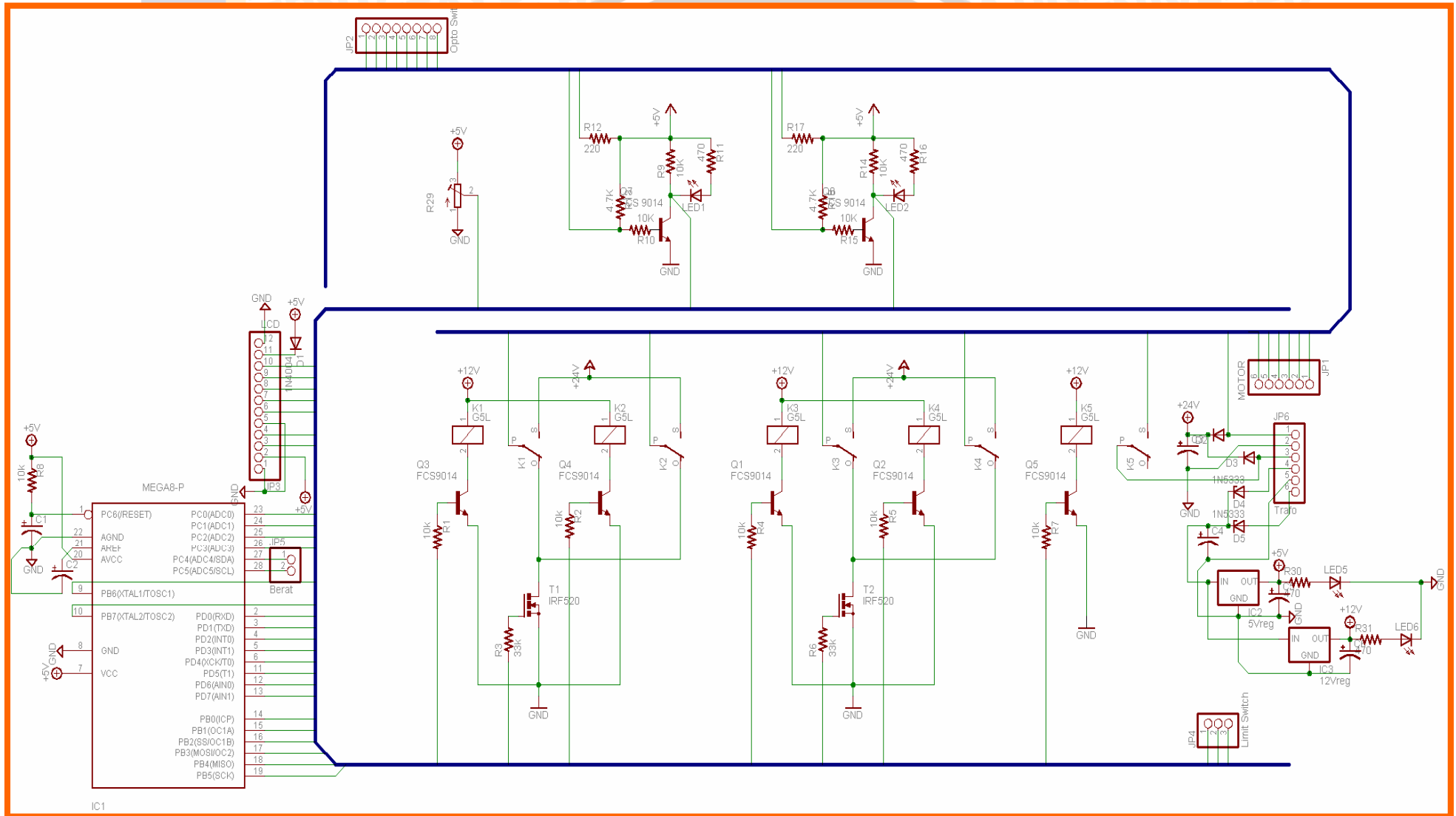
7.2. Saran

Dalam perancangan dan pembuatan alat ini masih terdapat kelemahan dan kekurangan. Agar tercapainya penyempurnaan kinerja alat dan pengembangan lebih lanjut disarankan:

- a. Perlu adanya penyempurnaan dalam perancangan sensor berat, karena pegas yang digunakan dalam pemakaian yang cukup tinggi konstanta pegasnya akan berubah, sehingga perlu dilakukan kalibrasi secara berkala.
- b. Untuk pemakaian yang sebenarnya, perlu adanya perubahan tentang sensor posisi yang digunakan, untuk menjaga keamanan dan kontinuitas pemakaiannya.

DAFTAR PUSTAKA

- Anonymous*. 2000. *H22A1/2/3 Slotted Optical Switch*. Fairchild Semiconductor.
Akses dari: www.alldatasheet.com/datasheet-pdf/view/52741/FAIRCHILD/H22A/datasheet.pdf.
- Anonymous*. 2002. *SS9014 NPN Epitaxial Silicon Transistor*. Fairchild Semiconductor. Akses dari: www.alldatasheet.com/datasheet-pdf/view/52741/FAIRCHILD/SS9014/datasheet.pdf.
- Anonymous*. 2005. *8-Bit Microcontroller with 8 kBytes Flash*. Atmel corporation.
Akses dari: <http://www.atmel.com>.
- Anonymous*. 1998. *Liquid Cristal Display, Standart Character Modules Application Notes*. Seiko Instruments, Germany. Akses dari: [http://www.seiko.com/seiko manual.pdf](http://www.seiko.com/seiko/manual.pdf)
- JunYan, Michael Ryan, and James Power. 1994. *Using Fuzzy Logic*. Prentice Hall International, New Jersey.
- Kuswadi, Son A. 2000. *Kendali Cerdas (Intelligent Control)*. Surabaya : EEPIS Press.
- Kismet. F. 1997. *Ilmu Listrik*. Bandung: Angkasa
- Ogata, Katsuhiko. 1987. *Teknik Kontrol Automatik*. Jilid 1. Bandung : Erlangga
- Petruszella, Frank D. 1996 *Elektronika Industri*. Yogyakarta : Penerbit Andi
- Ross, Timothy J. 1995. *Fuzzy Logic With Engineering Application*. Mc Graw-Hill Inc. USA
- Rudenko, N. 1992. *Mesin Pemindah Bahan*. Terjemahan Foed, Nazar. 1994 Jakarta : Penerbitan Erlangga.



Gambar Rangkaian Keseluruhan

