

SKRIPSI

PERANCANGAN DAN PEMBUATAN SISTEM KENDALI JARAK PADA MOBIL ROBOT MENGUNAKAN METODE LOGIKA *FUZZY*



Disusun Oleh :

WILI HARDIANSYAH

NIM: 02.17.050

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**

MARET 2009

LEMBAR PERSETUJUAN

**PERANCANGAN DAN PEMBUATAN SISTEM KENDALI
JARAK PADA MOBIL ROBOT MENGGUNAKAN METODE
LOGIKA FUZZY**

SKRIPSI

*Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana
Teknik Elektro Strata Satu (S-1) Konsentrasi Elektronika*

Disusun Oleh :

WILI HARDIANSYAH

NIM : 02.17.050

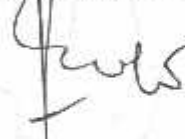
Diperiksa dan Disetujui

Dosen Pembimbing I



(I. Komang Somawirata, ST, MT)
NIP/P. 1030100361

Dosen Pembimbing II



(Ir. Eko Nurcahyo)
NIP. Y. 1028700172

Mengetahui,

Ketua Jurusan Teknik Elektro S-1



(Ir. F. Yudi Limpraptono, MT)

NIP. Y. 103 950 0274

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2009**



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa : Wili Hardiansyah
NIM : 02.17.050
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : PERANCANGAN DAN PEMBUATAN SISTEM
KENDALI JARAK PADA MOBIL ROBOT
MENGUNAKAN METODE LOGIKA FUZZY

Dipertahankan dihadapan Tim Penguji Skripsi Jenjang Strata Satu (S-1) pada:

Hari : Rabu
Tanggal : 18 Maret 2009
Dengan Nilai : B+ (79,75) *Yud*

Panitia Ujian Skripsi



Ketua Majelis Penguji

(Ir. H. Sidik Noertjahiono, MT.)
NIP.Y. 1028700163

Sekretaris Majelis Penguji

(Ir.F.Yudi Limpraptono, MT.)
NIP.Y. 1039500274

Anggota Penguji

Penguji Pertama

(M. Ashar ST, MT.)
NIP. P : 1030500408

Penguji Kedua

(Sotyhadi, ST.)
NIP. Y : 1039700309



JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG

FORMULIR PERBAIKAN SKRIPSI

NAMA : WILI HARDIANSYAH
N.I.M : 02.17.050
JURUSAN : TEKNIK ELEKTRO S-1
JUDUL SKRIPSI : PERANCANGAN DAN PEMBUATAN SISTEM
KENDALI JARAK PADA MOBIL ROBOT
MENGUNAKAN METODE LOGIKA FUZZY

No.	TANGGAL	MATERI PERBAIKAN	PARAF
1.	18 MARET 2009	Kesimpulan	
2.	18 MARET 2009	Pengujian Jarak (Depan), Stop	
3.	18 MARET 2009	Pengujian Fuzzy Logic belum ada ?	

Diperiksa Dan Disetujui :

Penguji Pertama

(M. Ashar ST, MT)
NIP. P : 1030500408

Penguji Kedua

(Sotvohadi, ST)
NIP. Y : 1039700309

Mengetahui,

Dosen Pembimbing I

(L. Komang Somawirata, ST, MT)
NIP.P : 1030100361

Dosen Pembimbing II

(Ir. Eko Nurcahyo)
NIP.P : 1028700172

Perancangan Dan Pembuatan Sistem Kendali Jarak Pada Mobil Robot Menggunakan Metode Logika *Fuzzy*

Wili Hardiansyah

Institut Teknologi Nasional Malang

Jurusan Teknik Elektro S-1, Konsentrasi Teknik Elektronika

Pembimbing : I. Komang Somawirata, ST, MT

Ir. Eko Nurcahyo

ABSTRAKSI

Kata Kunci : Logika Fuzi,Ultrasonik, Motor Servo, Mikrokontroler ATMega16.

Penggunaan kontrol logika *fuzzy* telah banyak digunakan dalam berbagai bidang, misalnya pada dunia rekayasa, teknologi komputer, sistem manufaktur, mekatronik, dan robotika. Salah satu contoh aplikasi logika *fuzzy* pada bidang robotika adalah mobil robot. Pada proposal skripsi ini akan menyajikan implementasi kontrol logika *fuzzy* untuk kendali jarak mobil robot, dimana saat mobil robot aktif semua sistem berjalan dengan otomatis, dengan kecepatan tertentu sesuai dengan permintaan pengemudi, dan jika terdapat penghalang di depannya dan kemungkinan mempunyai celah untuk belok maka mobil robot akan belok, akan tetapi jika tidak ada celah maka mobil robot akan berhenti.

Untuk menjaga agar tetap berada pada jalur, mobil robot mengambil data jarak dari tiga sensor ultrasonik dan penggunaan sistem kontrol logika *fuzzy* akan memberikan sinyal pada kedua motor servo untuk mengurangi atau menambah kecepatannya. Untuk penggunaan perangkat lunak digunakan bahasa C dari Mikrokontroler. Mobil robot ini diharapkan sebagai solusi alternatif peningkatan kenyamanan berkendara yang berfungsi sebagai asisten pengemudi, misalnya saat terjadi kemacetan, berkendara dengan kecepatan tertentu saat berada pada jalan bebas hambatan.

KATA PENGANTAR

Puji syukur penulis panjatkan atas Berkah Rahmat Allah Yang Maha Kuasa, berupa nikmat kesehatan dan kesempatan. Shalawat beserta Salam kepada Nabi Besar Muhammad SAW sebagai pembawa kebenaran dan penerang di muka bumi ini, sehingga penulis dapat menyelesaikan laporan Skripsi dengan judul:

“ PERANCANGAN DAN PEMBUATAN SISTEM KENDALI JARAK PADA MOBIL ROBOT MENGGUNAKAN METODE LOGIKA *FUZZY*”

Pembuatan Skripsi ini disusun guna memenuhi syarat akhir kelulusan pendidikan jenjang Strata-1 di Institut Teknologi Nasional Malang. Laporan Skripsi ini merupakan tanggung jawab tertulis atas ilmu pengetahuan yang didapat selama penyusunan mengikuti kuliah.

Atas terselesaikannya Skripsi ini, penulis mengucapkan terima kasih kepada :

- ❖ Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektronika S1.
- ❖ Bapak I. Komang Somawirata, ST, MT selaku Dosen Pembimbing I yang telah memberikan masukan, bimbingan, pengarahan, serta ilmu-ilmu yang sangat bermanfaat sehingga Skripsi ini dapat terselesaikan.
- ❖ Bapak Ir. Eko Nurcahyo selaku Dosen Pembimbing II yang telah memberikan masukan selama bimbingan Skripsi, baik pengarahan, serta ilmu-ilmu yang sangat bermanfaat sehingga Skripsi ini dapat terselesaikan.
- ❖ Bapak dan Ibu dosen Institut Teknologi Nasional Malang atas ilmu yang telah diberikan selama perkuliahan.
- ❖ Kedua Orang tua tercinta terima kasih atas Do'a, semua cinta, kasih sayang, yang telah banyak membantu dan selalu memberikan dorongan, semangat dan pengorbanan besar yang tulus sehingga penulis bisa mempersembahkan setitik keberhasilan ini. Semoga saya selalu menjadi anak yang shaleh dan berbakti.
- ❖ Widya Astutik istriku yang tercinta yang tanpa henti-hentinya selalu mendampingi baik dalam keadaan suka maupun duka, dan memberikan semangat serta dukungan dan do'a sehingga penulis tetap optimis untuk

tegak melangkah dalam menggapai cita-cita. Dan untuk yang spesial Lidya Komala Lindsey anakku yang tercinta, semoga kelak menjadi anak yang shalehah. Amiiin ya robbal alamin.

- ❖ Teman-teman seperjuangan T.Elektronika 2002 yang telah membantu dalam penyelesaian Skripsi ini.
- ❖ Terima kasih untuk orang-orang yang telah mendukungku dan merasa direpotkan, yang tak dapat kusebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan.

Akhir kata, penulis mohon maaf kepada semua pihak bila mana selama penyusunan Skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga Skripsi ini dapat bermanfaat bagi kita semua.

Malang, Maret 2009

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	i
BERITA ACARA	ii
FORMULIR PERBAIKAN SKRIPSI.....	iii
ABSTRAKSI	iv
KATA PENGANTAR	v
DAFTAR ISI	vii
DAFTAR GAMBAR	xii
DAFTAR TABEL	xv

BAB I PENDAHULUAN

1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan	2
1.4. Batasan Masalah	2
1.5. Metodologi	3
1.6. Sistematika Penulisan.....	4

BAB II TINJAUAN PUSTAKA

2.1. Logika <i>Fuzzy</i>	5
2.1.1. Pendahuluan	5

2.1.2.	Himpunan Logika <i>Fuzzy</i>	6
2.1.3.	Unit <i>Fuzzifikasi</i>	9
2.1.4.	Basis Pengetahuan (<i>Knowledge Base</i>)	12
2.1.4.1.	Basis Aturan.....	12
2.1.4.2.	Basis Data	13
2.1.5.	Unit Pengambilan Keputusan (<i>Inference</i>)	13
2.1.6.	<i>Defuzzifikasi</i>	14
2.2.	Mikrokontroler ATmega16	14
2.2.1.	Teori Dasar Mikrokontroler ATmega16	14
2.2.2.	Konfigurasi Pin	17
2.2.3.	Port sebagai input/output digital	19
2.2.4.	Timer	21
2.2.5.	Serial Pada ATmega16.....	30
2.3.	Sensor Ultrasonik	35
2.3.1.	Teori Dasar Sensor Ultrasonik.....	35
2.3.2.	Teori Operasi Sensor.....	37
2.4.	Motor Servo.....	38
2.4.1.	Teori Dasar Motor Servo.....	38
2.4.2.	Jenis Motor Servo.....	39

BAB III PERANCANGAN DAN PEMBUATAN ALAT

3.1.	Perancangan Perangkat Keras (<i>Hardware</i>)	41
3.1.1.	Perancangan Kontroller Logika <i>Fuzzy</i>	42

3.1.1.1. <i>Fuzzification</i>	42
3.1.1.2. <i>Rule Evaluation</i>	45
3.1.1.3. <i>Defuzzyfication</i>	46
3.1.2. Sistem Mikrokontroller ATmega16	47
3.1.3. Perancangan Dan Pembuatan Sensor	48
3.1.3.1. Sensor Ultrasonik	49
3.1.3.2. Perancangan Rangkaian Komparator	50
3.1.4. Motor Servo	50
3.2. Perancangan Perangkat Lunak (<i>Software</i>)	51
3.2.1. Desain Diagram Alir (<i>Flowchart</i>)	51
3.2.1.1. Diagram Alir (<i>Flowchart</i>) Utama	52

BAB IV PENGUJIAN DAN ANALISA DATA

4.1. Pengujian Minimum Sistem ATmega16	53
4.1.1. Tujuan	53
4.1.2. Peralatan Yang Digunakan	53
4.1.3. Tahapan Pengujian Mikrokontroller ATmega16	54
4.1.4. Persiapan	54
4.1.5. Hasil Dan Analisa.....	54
4.2. Pengujian Rangkaian Sensor Ultrasonik	54
4.2.1. Sinyal Sensor Ultrasonik	54
4.2.1.1. Tujuan.....	54
4.2.1.2. Peralatan Yang Dibutuhkan.....	55

4.2.1.3. Prosedur Pengujian.....	55
4.2.1.4. Hasil Pengujian.....	55
4.2.2. Pengujian Pemancar Ultrasonik	55
4.2.2.1. Tujuan.....	55
4.2.2.2. Peralatan Yang Dibutuhkan.....	56
4.2.2.3. Prosedur Pengujian.....	56
4.2.2.4. Hasil Pengujian.....	56
4.2.3. Pengujian Penerima Ultrasonik.....	56
4.2.3.1. Tujuan.....	56
4.2.3.2. Peralatan Yang Dibutuhkan.....	56
4.2.3.3. Prosedur Pengujian.....	57
4.2.3.4. Hasil Pengujian.....	57
4.3. Pengujian Rangkaian komparator.....	57
4.3.1. Tujuan	57
4.3.2. Peralatan Yang Dibutuhkan.....	57
4.3.3. Prosedur Pengujian	58
4.3.4. Hasil Pengujian.....	58
4.4. Pengujian Motor Servo.....	59
4.4.1. Variasi Pertama	59
4.4.2. Variasi Kedua	60
4.4.3. Variasi Ketiga.....	60
4.4.4. Variasi Keempat.....	61

BAB V PENUTUP

5.1. Kesimpulan	63
5.2. Saran	63

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar 2.1	Himpunan <i>Fuzzy</i> pada variabel temperatur	7
Gambar 2.2	Struktur kontroler logika <i>fuzzy</i> dasar	8
Gambar 2.3	<i>S-Function</i>	10
Gambar 2.4	π - <i>Function</i>	11
Gambar 2.5	<i>T-Function</i>	11
Gambar 2.6	Pin-pin ATmega16 kemasan 40-pin	17
Gambar 2.7	Blok Diagram <i>timer/counter</i>	22
Gambar 2.8	<i>Timing diagram timer/counter</i> , tanpa <i>prescaling</i>	22
Gambar 2.9	<i>Timing diagram timer/counter</i> , dengan <i>prescaling</i>	23
Gambar 2.10	<i>Timing diagram timer/counter</i> , menyeting OCFO, dengan pescaler (<i>flck_I/O/8</i>).....	23
Gambar 2.11	<i>Timing diagram timer/counter</i> , menyeting OCFO, pengosongan data timer sesuai dengan data pembanding,dengan pescaler (<i>flck_I/O/8</i>).....	24
Gambar 2.12	Register Timer/Counter 8 Bit	24
Gambar 2.13	Register Timer TCNT0.....	27
Gambar 2.14	Register Timer OCR0	28
Gambar 2.15	Register timer TIFR	29
Gambar 2.16	Block diagram <i>clock generasi logic</i>	31
Gambar 2.17	Persamaan untuk menyeting perhitungan <i>register Baud Rate</i>	34
Gambar 2.18	Sensor Ultrasonik.....	36

Gambar 2.19	Cara Kerja Sensor	37
Gambar 2.20	Diagram Waktu Sensor.....	38
Gambar 2.21	Parallax Standar Servo.....	38
Gambar 2.22	Diagram Pengkabelannya	39
Gambar 3.1	Diagram Blok Keseluruhan Sistem	41
Gambar 3.2	Bentuk <i>Membership Function Input</i> Depan	42
Gambar 3.3	Bentuk <i>Membership Function Input</i> Kiri	42
Gambar 3.4	Bentuk <i>Membership Function Input</i> Kanan	42
Gambar 3.5	<i>Membership Function Output</i> Kiri dan Kanan	46
Gambar 3.6	Perancangan Mikrokontroler ATmega16	47
Gambar 3.7	Perancangan Rangkaian Reset.....	47
Gambar 3.8	Perancangan Sensor Ultrasonik	49
Gambar 3.9	Perancangan Rangkaian Komparator.....	50
Gambar 3.10	Perancangan Motor Servo.....	51
Gambar 4.1	Diagram Blok Pengujian Rangkaian Mikrokontroler.....	54
Gambar 4.2	Sinyal Keluaran Dari Rangkaian Sensor Ultrasonik Dengan Frekuensi 40 KHz	55
Gambar 4.3	Sinyal Keluaran Dari Pemancar Dengan Frekuensi 40 KHz.....	56
Gambar 4.4	Sinyal Keluaran Dari Penerima Ultrasonik	57
Gambar 4.5	Sinyal Keluaran Dari Komparator	58
Gambar 4.6	Kondisi Variasi Robot Pertama	59
Gambar 4.7	Foto PWM Variasi Pertama.....	59
Gambar 4.8	Kondisi Variasi Robot Kedua	60

Gambar 4.9	Foto PWM Variasi Kedua.....	60
Gambar 4.10	Kondisi Variasi Robot Ketiga.....	61
Gambar 4.11	Foto PWM Variasi Ketiga	61
Gambar 4.12	Kondisi Variasi Robot Keempat.....	61
Gambar 4.13	Foto PWM Variassi Keempat.....	62

DAFTAR TABEL

Tabel 2.1	Konfigurasi pin port	20
Tabel 2.2	Deskripsi Bit Mode Pembangkit Bentuk Gelombang	25
Tabel 2.3	Mode Output Pembanding, tanpa PWM	26
Tabel 2.4	Mode Output Pembanding, Mode fast PWM.....	26
Tabel 2.5	Mode Output Pembanding, Mode phase correct PWM.....	26
Tabel 2.6	Deskripsi bit clock select.....	27
Tabel 2.7	Persamaan untuk menyetting perhitungan <i>register Baud Rate</i>	32
Tabel 3.1	<i>Rule Evaluation</i>	45
Tabel 4.1	Perbandingan Antara Perhitungan Dengan Pengukuran Pada Sensor Penerima Ultrasonik.....	58



BAB I

PENDAHULUAN

1.1. Latar Belakang

Penggunaan kontrol logika fuzzy telah banyak digunakan dalam berbagai bidang, misalnya pada dunia rekayasa, teknologi komputer, sistem manufaktur, mekatronik, dan robotika. Salah satu contoh aplikasi logika fuzzy pada bidang robotika adalah mobil robot.

Pada skripsi ini akan menyajikan implementasi kontrol logika fuzzy untuk kendali jarak mobil robot, dimana saat mobil robot aktif semua sistem berjalan dengan otomatis, dengan kecepatan tertentu sesuai dengan permintaan pengemudi, dan jika terdapat penghalang di depannya dan kemungkinan mempunyai celah untuk belok maka mobil robot akan belok, akan tetapi jika tidak ada celah maka mobil robot akan berhenti.

Untuk menjaga agar tetap berada pada jalur, mobil robot mengambil data jarak dari tiga sensor ultrasonik dan penggunaan sistem kontrol logika fuzzy akan memberikan sinyal pada kedua motor servo untuk mengurangi atau menambah kecepatannya. Untuk penggunaan perangkat lunak digunakan bahasa C dari Mikrokontroler. Mobil robot ini diharapkan sebagai solusi alternatif peningkatan kenyamanan berkendara yang berfungsi sebagai asisten pengemudi, misalnya saat terjadi kemacetan, berkendara dengan kecepatan tertentu saat berada pada jalan bebas hambatan.

1.2. Rumusan Masalah

Berdasarkan pada latar belakang yang ada, maka rumusan masalah ini adalah :

1. Bagaimana merancang dan membuat Sistem kendali jarak pada mobil robot menggunakan metode logika *fuzzy*.
2. Bagaimana membuat perangkat lunak pada alat tersebut sehingga robot dapat bergerak otomatis.
3. Bagaimana prinsip kerja dari sensor ultrasonik yang digunakan untuk mengukur jarak antara prototipe mobil robot dengan penghalang-penghalang disekitarnya.
4. Bagaimana mengaplikasikan kontrol logika *fuzzy* dan merancang sistem kemudi otomatis pada mikrokontroller untuk lintasan yang telah ditentukan.

1.3. Tujuan

Tujuan dari penulisan skripsi ini adalah merancang dan membuat sebuah sistem kendali jarak pada mobil robot menggunakan metode logika *fuzzy*.

1.4. Batasan Masalah

Dalam perancangan dan pembuatan alat ini, adapun batasan-batasannya adalah :

1. Tidak membahas karakteristik dari setiap komponen yang digunakan pada alat ini secara mendetail.
2. Tidak membahas catu daya yang digunakan.

3. Sistem kendali jarak mobil robot yang dideteksi yaitu penghalang depan, samping kanan dan samping kiri
4. Pengaturan kecepatan motor servo dilakukan hanya untuk menentukan gerakan mobil robot untuk berjalan lurus, berbelok ke kiri, atau berbelok ke kanan.
5. *Membership Function* untuk jarak robot mobil ke halangan dibatasi hanya dua label yaitu Dekat dan Jauh.
6. Gangguan yang disebabkan *mechanical error* (misalnya : getaran, dan guncangan) diabaikan.

1.5. Metodologi

Metodologi yang dipakai dalam pembuatan skripsi ini adalah :

1. Studi Literatur

Dengan mencari referensi-referensi yang berhubungan dengan perencanaan dan pembuatan alat yang akan dibuat.

2. *Field Research*

Dengan melakukan penelitian secara langsung mengenai objek-objek yang berhubungan langsung dengan perencanaan alat yang akan dibuat.

3. *Design* dan Pembuatan Alat

Yaitu meliputi pembuatan PCB, perakitan komponen serta pensolderan dan pembuatan perangkat lunak.

4. Pengujian Alat

Dengan melakukan pengujian perblok rangkaian dan kerja seluruh sistem pada alat tersebut.

5. Penyusunan Laporan Skripsi

Membuat laporan yang terdiri dari: Pendahuluan, Landasan Teori, Perencanaan dan Pembuatan Alat, Pengujian Alat dan Penutup.

1.6. Sistematika Penulisan

Sistematika penyusunan skripsi dibagi menjadi beberapa bagian utama, yaitu:

BAB I Pendahuluan

Memuat latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi pembahasan, dan sistematika pembahasan.

BAB II Teori Penunjang

Membahas teori-teori yang mendukung dalam perencanaan dan pembuatan alat.

BAB III Perencanaan dan Pembuatan Alat

Perancangan dan perealisasiian alat yang meliputi spesifikasi, perencanaan blok diagram, prinsip kerja dan realisasi alat.

BAB IV Pengujian Alat

Memuat hasil pengujian terhadap alat yang telah direalisasikan.

BAB V Kesimpulan dan Saran

Memuat kesimpulan dan saran-saran



BAB II

TINJAUAN PUSTAKA

Landasan teori sangat membantu untuk dapat memahami suatu sistem. Selain dari pada itu dapat juga dijadikan sebagai bahan acuan didalam merencanakan suatu sistem. Dengan pertimbangan hal-hal tersebut, maka landasan teori merupakan bagian yang harus dipahami untuk pembahasan selanjutnya.

2.1 Logika *Fuzzy*

2.1.1 Pendahuluan

Pada tahun 1965, logika *fuzzy* diperkenalkan oleh Prof. Lotfi. A Zadeh yang menjadi guru besar di University Of California. Beliau mengemukakan suatu himpunan yang disebut himpunan *fuzzy* (*fuzzy set*) untuk menerangkan suatu logika bertingkat. Logika ini kemudian dikenal dengan logika *fuzzy*. Zadeh memilih kata " Fuzz " untuk mempresentasikan suatu nilai logika kontinyu antara 0 (pasti salah) dan 1 (pasti benar)

Logika *fuzzy* merupakan suatu logika yang lebih dekat dengan cara berpikir manusia. Kalau pada himpunan crips, nilai keanggotaan hanya ada 2 kemungkinan yaitu 0 dan 1. Pada nilai *fuzzy* nilai keanggotaan terletak pada rentang 0 sampai 1.

Logika *fuzzy* bekerja dengan mengubah variabel control (panas dan dingin, cepat atau lambat) ke dalam tingkatan yang lebih halus (hangat atau sejuk, cukup cepat atau agak lambat) dengan derajat keanggotaan yang bervariasi.

Berbeda dengan teori logika digital, dalam logika *fuzzy* suatu kejadian tidak harus mutlak benar atau salah.

Sebuah sistem logika *fuzzy* dapat mengurangi ketidakakuratan pada sistem klasik yang memiliki persyaratan keanggotaan yang membatasi nilai anggota-anggota himpunannya hanya pada satu sampai 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 data menjadi anggota himpunan dengan tingkat keanggotaan nol sampai satu. Ada transisi yang bertahap dari unsur yang menjadi anggota atau bukan anggota himpunan dalam sebuah himpunan *fuzzy*.

2.1.2 Himpunan Logika *fuzzy*

Himpunan *Fuzzy* memiliki 2 atribut, yaitu:

- a. *Linguistik*, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: MUDA, PAROBAYA, TUA.
- b. *Numeris*, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel seperti: 10, 20, 30, 40, 50 dan sebagainya.

Ada beberapa hal yang perlu diketahui dalam memahami sistem *fuzzy*, yaitu:

- a. Variabel *fuzzy*

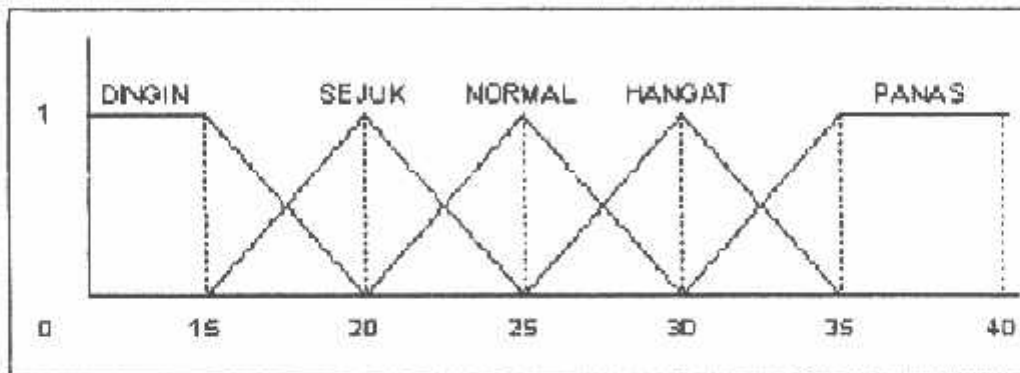
Variabel *fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *fuzzy*. Contoh: temperatur, kelembaban, umur, dan sebagainya.

- b. Himpunan *fuzzy*

Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*.

Contoh:

variabel temperatur, terbagi menjadi 5 himpunan *fuzzy*, yaitu: SANGAT DINGIN, DINGIN, NORMAL, HANGAT, dan PANAS. (Gambar 2.1).



Gambar 2.1 Himpunan Fuzzy pada variabel temperatur^[1]

c. Semesta pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk dioperasikan dalam suatu variabel *fuzzy*. Semesta pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara *monoton* dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan positif maupun negatif. Adakalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya.

Contoh:

Semesta pembicaraan untuk variabel temperatur: [0 10 20 30 40 50]

d. Domain

Domain himpunan *fuzzy* adalah keseluruhan nilai yang diijinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Seperti

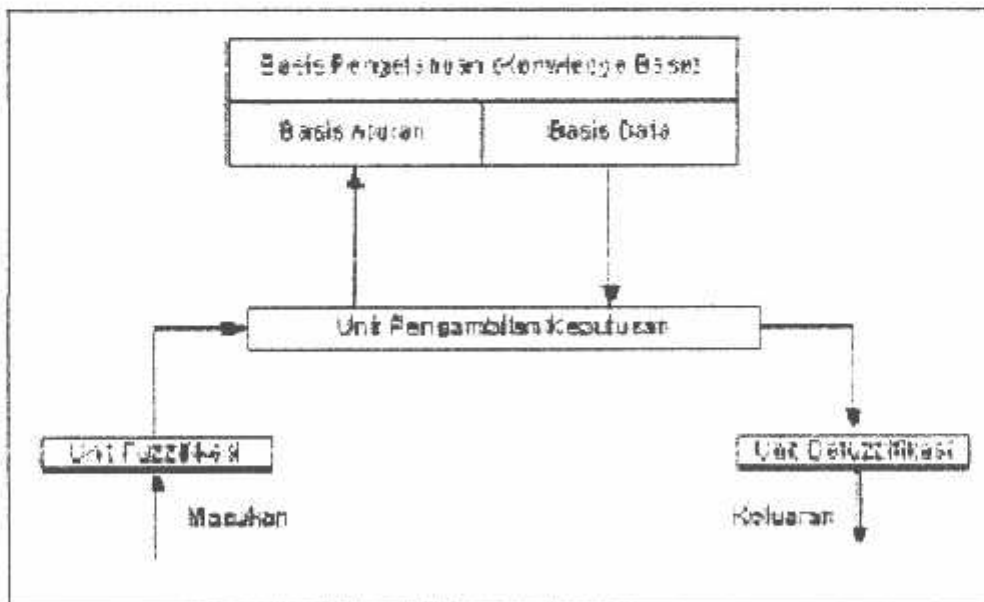
halnya semesta pembicaraan, domain merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara *monoton* dari kiri ke kanan. Nilai domain dapat berupa bilangan positif maupun negatif.

Contoh domain himpunan *fuzzy*:

- Sangat Dingin = [0 20]
- Dingin = [10 30]
- Normal = [20 40]
- Hangat = [30 50]
- Sangat Panas = [40 90]

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 $[0,1]$.

Gambar struktur kontroler logika *fuzzy* dasar digambarkan sebagai berikut:



Gambar 2.2 Struktur kontroler logika *fuzzy* dasar^[2]

2.1.3 Unit Fuzzifikasi

Seperti yang terlihat dalam Gambar 2.2. di atas, salah satu komponen kunci logika fuzzy adalah unit fuzzifikasi. Fuzzifikasi merupakan proses pengubahan dari variabel nonfuzzy (*crisp*, contohnya: 25, 40, 50), menjadi variabel fuzzy (contohnya: muda, parobaya, tua).

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 :

$$x = \text{fuzzifier}(x_0) \quad (2.1)$$

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 5 atau 7.
3. Mengatur himpunan *fuzzy* agar saling menumpuk.
4. Menggunakan fungsi keanggotaan bentuk segitiga atau *trapezoid*.

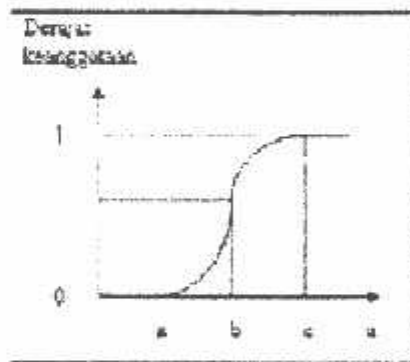
Ada beberapa fungsi keanggotaan dalam logika *fuzzy*. Adapun yang sering digunakan adalah bentuk fungsi-S, fungsi- π , fungsi T (triangular). Contoh berikut ini adalah tiga macam keanggotaan yang dinyatakan dengan fungsi keanggotaan S, π dan T (*triangular*).

a. S-function

Definisi S-function adalah sebagai berikut :

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

Bentuk diagramatik S-function ditunjukkan dalam Gambar 2.3.



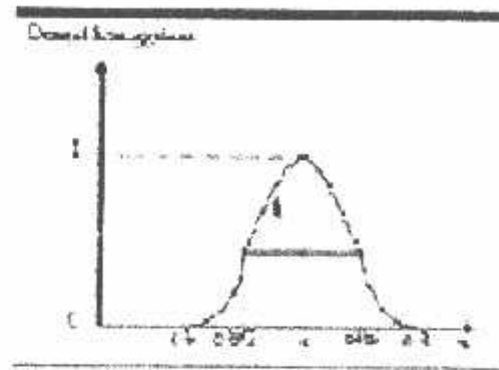
Gambar 2.3 S-Function^[2]

b. π -Function

Definisi π -function adalah sebagai berikut :

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

Bentuk diagramatik π -Function ditunjukkan dalam Gambar 2.4



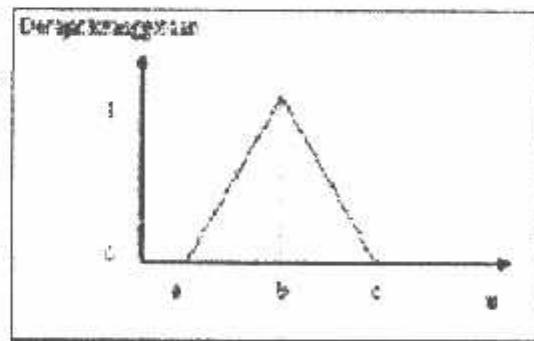
Gambar 2.4 π -Function^[2]

c. *Triangular-function.*

Triangular-function didefinisikan sebagai berikut :

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

Bentuk diagramatik *Triangular-function* ditunjukkan dalam gambar 2.5



Gambar 2.5 *T-Function*^[2]

2.1.4 Basis Pengetahuan (*Knowledge Base*)

Bagian terpenting logika *fuzzy* lainnya adalah basis pengetahuan yang mengandung dua jenis informasi utama, yaitu: basis aturan dan basis data.

Pengetahuan (*knowledge*) adalah suatu pengertian tentang beberapa wilayah subyek yang diperoleh melalui pendidikan dan pengalaman. Meskipun *fuzzy* tidak mungkin mendapat pengalaman atau belajar dan meneliti seperti manusia, tapi ia dapat memperoleh pengetahuan yang dibutuhkannya itu melalui upaya yang diberikan oleh seorang pakar manusia.

Hampir semua pangkalan pengetahuan (*knowledge base*) sangat terbatas dalam arti terfokuskan kepada suatu masalah khusus. Pada saat pangkalan pengetahuan itu sudah terbentuk, maka *fuzzy* bisa digunakan untuk memberi kemampuan baru kepada komputer atau alat agar bisa berfikir, menalar dan membuat inferensi (mengambil keputusan berdasarkan pengalaman) dan membuat pertimbangan-pertimbangan yang didasarkan kepada fakta dan hubungan-hubungannya yang terkandung dalam pangkalan pengetahuan itu.

2.1.4.1 Basis Aturan

Basis aturan terdiri atas aturan-aturan kontrol *fuzzy* yang menyatakan hubungan dalam pengontrolan. Aturan – aturan kontrol *fuzzy* disusun berdasarkan pada tujuan dan ketentuan pengontrolan yang diberikan oleh orang yang ahli pada bidangnya.

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

If (kondisi), *Then* (aksi)

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 dikarakteristikkan dalam semesta himpunan variabel masukan. Konsekuen mendefinisikan aksi kontrol yang diinginkan.

2.1.4.2 Basis Data

Basis data mendefinisikan fungsi keanggotaan himpunan-himpunan *fuzzy* variabel masukan dan keluaran pengendali logika *fuzzy*. Basis data juga menyediakan definisi yang perlu dari parameter-parameter *fuzzy* sebagai himpunan *fuzzy* dengan fungsi keanggotaan yang didefinisikan pada himpunan semesta untuk masing-masing variabel. Penyusunan basis data melibatkan pendefinisian himpunan semesta untuk masing-masing variabel, penentuan jumlah *fuzzy* serta perancangan fungsi keanggotaannya.

2.1.5 Unit Pengambilan Keputusan (Inference)

Proses untuk menghasilkan aksi keluaran (aksi control *fuzzy*) dari suatu kondisi masukan (terfuzzifikasi) dengan mengikuti aturan-aturan (rule) yang telah ditetapkan merupakan definisi dari inference.

Keputusan yang dihasilkan dari proses ini masih dalam bentuk *fuzzy*, yaitu berupa derajat keanggotaan keluaran. Salah satu unit pengambilan keputusan yang sering dipergunakan pada kontroler logika *fuzzy* adalah metode MAX-MIN.

Penyimpulan *fuzzy* adalah merupakan inti dari suatu Kontrol Logika *Fuzzy* yang mempunyai kemampuan untuk mensimulasikan keputusan manusia

yang didasarkan pada konsep fuzzy dan menyimpulkan aksi kontrol fuzzy menggunakan implikasi fuzzy dan aturan-aturan penyimpulan dalam logika fuzzy.

2.1.6 Defuzzifikasi

Defuzzifikasi merupakan proses pemetaan himpunan fuzzy ke himpunan *crisp*. Output dalam bentuk *crisp* ini digunakan untuk aksi kendali. Proses ini merupakan kebalikan dari proses fuzzifikasi berfungsi untuk mengubah keluaran linguistik ke dalam keluaran *crisp*.

Strategi defuzzifikasi diekspresikan sebagai berikut:

$$y_0 = \text{defuzzifier}(y) \quad (2.6)$$

dengan : y = aksi kontrol *fuzzy*

$$y_0 = \text{aksi kontrol } \textit{crisp}$$

defuzzifier = operator defuzzifikasi

Metode defuzzifikasi ada dua, yaitu : Metode rata-rata maksimum dan metode pusat luasan.

2.2 Mikrokontroller ATmega16

2.2.1 Teori Dasar Mikrokontroller ATmega16

AVR merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur RISC (*Reduced Instruction Set Computer*). Hampir semua instruksi dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 *register general-purpose*, *timer/counter* fleksibel dengan *mode compare*, *interrupt internal* dan *eksternal*, serial UART, *programmable Watchdog Timer*, dan *mode power saving*. Mempunyai ADC dan PWM *internal*. AVR juga mempunyai *In-System Programmable Flash on-chip* yang mengijinkan memori program untuk

diprogram ulang dalam sistem menggunakan hubungan serial SPI. Atmega16 adalah mikrokontroler CMOS 8-bit daya-rendah berbasis arsitektur RISC yang ditingkatkan.

Kebanyakan instruksi dikerjakan pada satu siklus clock, Atmega16 mempunyai *throughput* mendekati 1 MIPS per MHz membuat disainer sistem untuk mengoptimasi konsumsi daya versus kecepatan proses.

Beberapa keistimewaan dari AVR ATmega16 antara lain:

1. *Advanced RISC Architecture*

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

2. *Nonvolatile Program and Data Memories*

- *16K Bytes of In-System Self-Programmable Flash*
- *Optional Boot Code Section with Independent Lock Bits*
- *512 Bytes EEPROM*
- *1K Bytes Internal SRAM*
- *Programming Lock for Software Security*

3. *Peripheral Features*

- *Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes*

- *One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode*
- *Real Time Counter with Separate Oscillator*
- *Four PWM Channels*
- *8-channel, 10-bit ADC*
- *Byte-oriented Two-wire Serial Interface*
- *Programmable Serial USART*
- *Master/Slave SPI Serial Interface*
- *Programmable Watchdog Timer with Separate On-chip Oscillator*
- *On-chip Analog Comparator*

4. Special Microcontroller Features

- *Power-on Reset and Programmable Brown-out Detection*
- *Internal Calibrated RC Oscillator*
- *External and Internal Interrupt Sources*
- *Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby*

5. I/O and Package

- *32 Programmable I/O Lines*
- *40-pin PDIP, 44-lead TQFP, 44-lead PLCC, and 44-pad MLF*

7. Operating Voltages

- *2.7 - 5.5V for Atmega16L*
- *4.5 - 5.5V for Atmega16*

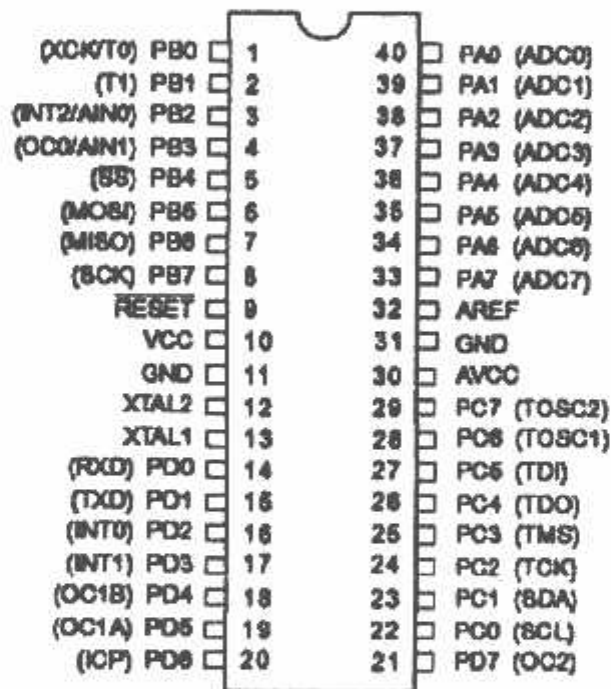
8. Speed Grades

• 0 - 8 MHz for ATmega16L

• 0 - 16 MHz for ATmega16

2.2.2 Konfigurasi Pin

Pin-pin pada ATmega16 dengan kemasan 40-pin DIP (dual inline package) ditunjukkan oleh gambar 2.2. Kemasan pin tersebut terdiri dari 4 Port yaitu Port A, Port B, Port C, Port D yang masing masing Port terdiri dari 8 buah pin. Selain itu juga terdapat RESET, VCC, GND 2 buah, VCC, AVCC, XTAL1, XTAL2 dan AREF.



Gambar 2.6 Pin-pin ATmega16 kemasan 40-pin^[3]

Diskripsi dari pin-pin ATmega 16 adalah sebagai berikut :

1. **VCC** : *Supply* tegangan digital.
2. **GND** : *Ground*

3. **PORT A:** Menjalankan analog input ke A/D converter. Port A juga berfungsi sebagai 8 bit directional I/O port jika A/D converter tidak digunakan. Ketika pin PA0-PA7 digunakan input dan secara eksternal pull-low, mereka seperti sumber arus jika internal pull-up resistor diaktifkan. Pin port A adalah tri-states ketika kondisi sebuah reset menjadi aktif, sekalipun clocknya tidak jalan.
4. **PORT B:** Port B adalah 8 bit bi-directional I/O port dengan Internal pull-up resistor. Buffer output port B ini mempunyai karakteristik symmetrical drive dengan kapabilitas source dan sink yang tinggi. Sebagai input, pin port B adalah eksternal pull-low seperti sumber arus jika pull-up resistor aktif. Pin port B adalah tri-states ketika kondisi sebuah reset menjadi aktif, sekalipun clocknya tidak jalan.
5. **PORT C:** Port C adalah 8 bit bi-directional I/O port dengan Internal pull-up resistor. Buffer output port B ini mempunyai karakteristik symmetrical drive dengan kapabilitas source dan sink yang tinggi. Sebagai input, pin port C adalah eksternal pull-low seperti sumber arus jika pull-up resistor aktif. Pin port C adalah tri-states ketika kondisi sebuah reset menjadi aktif, sekalipun clocknya tidak jalan. Jika interface JTAG enable, pull up resistor di pin PC5(TDI), PC3(TMS), dan PC2(TCK) akan aktif sekalipun reset terjadi.
6. **PORT D:** Port D adalah 8 bit bi-directional I/O port dengan Internal pull-up resistor. Buffer output port D ini mempunyai karakteristik

symmetrical drive dengan kapabilitas source dan sink yang tinggi. Sebagai input, pin port D adalah eksternal pull-low seperti sumber arus jika pull-up resistor aktif. Pin port D adalah tri-states ketika kondisi sebuah reset menjadi aktif, sekalipun locknya tidak jalan.

7. **RESET** : Sebuah low level pada pin akan lebih lama dari pada lebar pulsa minimum akan menghasilkan reset meskipun clock tidak berjalan.
8. **XTAL1** : Input inverting penguat Oscilator dan input intenal clock operasi rangkaian.
9. **XTAL2** : *Output* dari inverting penguat Oscilator.
10. **AVCC** : *Pin supply* tegangan untuk Port A dan A/D converter . Sebaiknya *eksternalnya* dihubungkan ke VCC meskipun ADC tidak digunakan. Jika ADC digunakan seharusnya dihubungkan ke VCC melalui *low pas filter*.
11. **AREF** : Pin referensi analog untuk A/D konverter.

2.2.3 Port sebagai input/output digital

ATmega16 mempunyai empat buah port yang bernama PortA, PortB, PortC, dan PortD. Keempat port tersebut merupakan jalur bi-directional dengan pilihan internal pull-up. Tiap port mempunyai tiga buah register bit, yaitu DDxn, PORTxn, dan PINxn. Huruf 'x' mewakili nama huruf dari port sedangkan huruf 'n' mewakili nomor bit. Bit DDxn terdapat pada I/O address DDRx, bit PORTxn terdapat pada I/O address PORTx, dan bit PINxn terdapat pada I/O address PINx. Bit DDxn dalam regiter DDRx (Data Direction Register) menentukan arah pin.

Bila DDxn diset 1 maka Px berfungsi sebagai pin output. Bila DDxn diset 0 maka Px berfungsi sebagai pin input. Bila PORTxn diset 1 pada saat pin terkonfigurasi sebagai pin input, maka resistor pull-up akan diaktifkan. Untuk mematikan resistor pull-up, PORTxn harus diset 0 atau pin dikonfigurasi sebagai pin output. Pin port adalah tri-state setelah kondisi reset. Bila PORTxn diset 1 pada saat pin terkonfigurasi sebagai pin output maka pin port akan berlogika 1. Dan bila PORTxn diset 0 pada saat pin terkonfigurasi sebagai pin output maka pin port akan berlogika 0. Saat mengubah kondisi port dari kondisi *tri-state* (DDxn=0, PORTxn=0) ke kondisi *output high* (DDxn=1, PORTxn=1) maka harus ada kondisi peralihan apakah itu kondisi *pull-up enabled* (DDxn=0, PORTxn=1) atau kondisi *output low* (DDxn=1, PORTxn=0). Biasanya, kondisi pull-up enabled dapat diterima sepenuhnya, selama lingkungan impedansi tinggi tidak memperhatikan perbedaan antara sebuah *strong high driver* dengan sebuah pull-up. Jika ini bukan suatu masalah, maka bit PUD pada register SFIOR dapat diset 1 untuk mematikan semua pull-up dalam semua port. Peralihan dari kondisi *input dengan pull-up* ke kondisi *output low* juga menimbulkan masalah yang sama. Kita harus menggunakan kondisi tri-state (DDxn=0, PORTxn=0) atau kondisi output high (DDxn=1, PORTxn=0) sebagai kondisi transisi. Lebih detil mengenai port ini dapat dilihat pada manual datasheet dari IC ATmega16.

Tabel 2.1 Konfigurasi pin port^[3]

DDxn	PORTxn	PUD (in SFIOR)	I/O	Pull-up	Comment
0	0	X	Input	No	Tri-state (Hi-Z)
0	1	0	Input	Yes	Pxn will source current if ext. pulled low.
0	1	1	Input	No	Tri-state (Hi-Z)
1	0	X	Output	No	Output Low (Sink)
1	1	X	Output	No	Output High (Source)

Bit 2 – PUD : Pull-up Disable

Bila bit diset bernilai 1 maka pull-up pada port I/O akan dimatikan walaupun register DDxn dan PORTxn dikonfigurasi untuk menyalakan pull-up (DDxn=0, PORTxn=1).

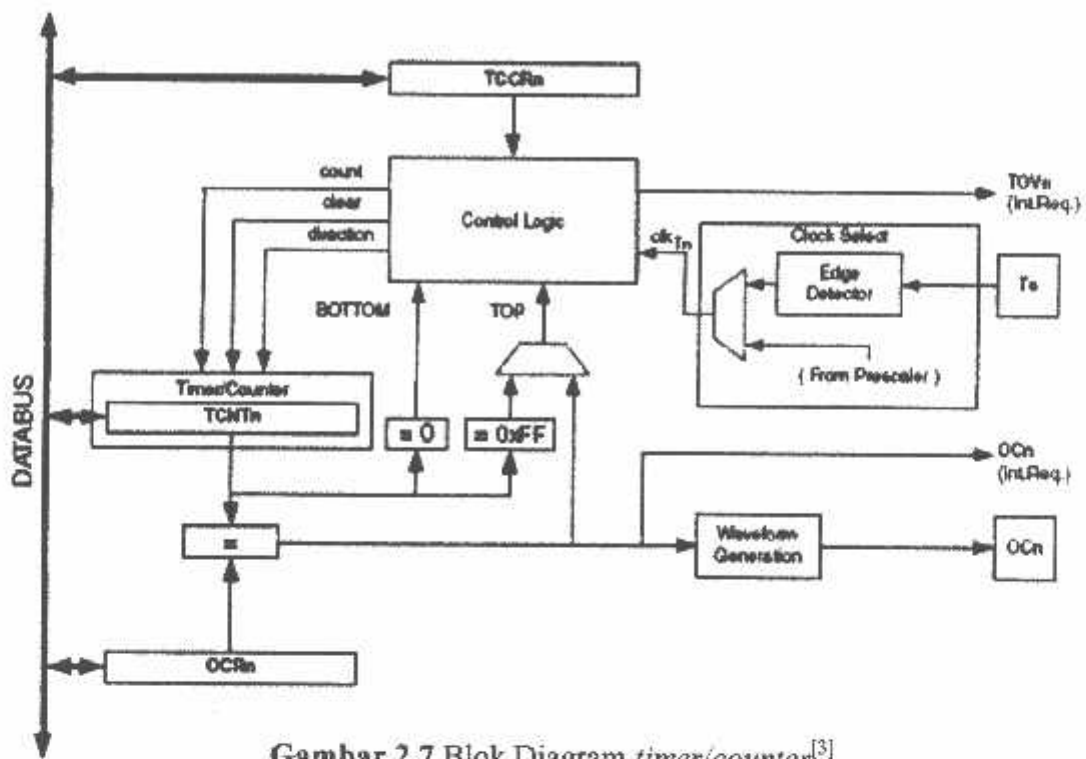
2.2.4 Timer

Timer/counter adalah tujuan umum *single channel, module 8 bit timer/counter*.

- counter channel tunggal
- pengosongan data timer sesuai dengan data pembanding
- glitch-bebas, tahap yang tepat Pulse Width Modulator (PWM)
- pembangkit frekuensi
- event counter external

➤ Gambaran Umum

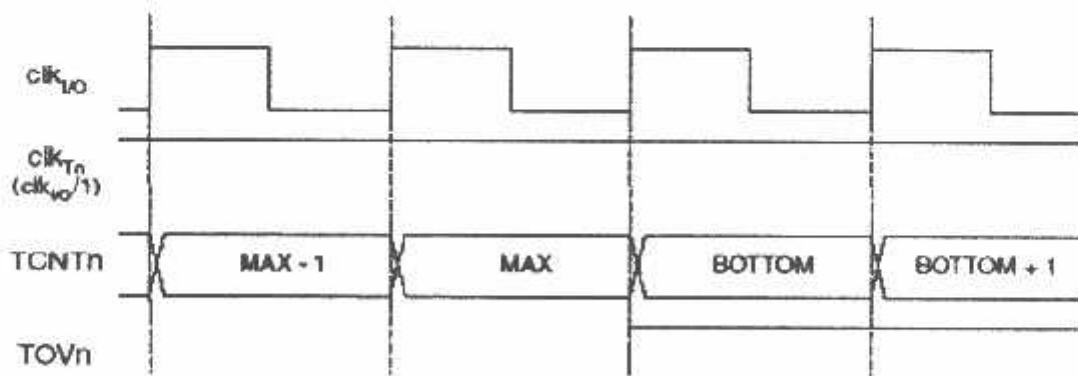
Gambar diagram *block timer/counter 8 bit* ditunjukkan pada gambar di bawah ini. Untuk penempatan pin I/O telah di jelaskan pada bagian I/O di atas. CPU dapat diakses *register I/O*, termasuk dalam pin-pin I/O dan bit I/O. *Device* khusus register I/O dan lokasi bit terdaftar pada deskripsi *timer/counter 8 bit* pada gambar 2.2 di bawah ini.



Gambar 2.7 Blok Diagram timer/counter^[3]

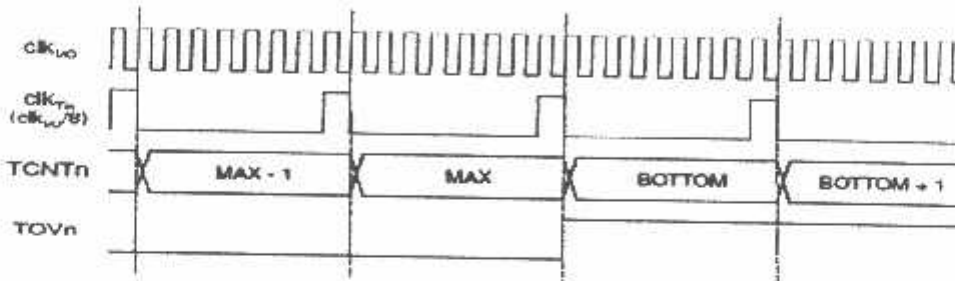
➤ **Timing Diagram Timer/Counter**

Timer/counter disain sinkron clock timer (clk_{T0}) oleh karena itu ditunjukkan sebagai sinyal *enable clock* pada gambar berikut. Gambar ini termasuk informasi ketika flag interrupt dalam kondisi set. Data timing digunakan sebagai dasar dari operasi timer/counter.



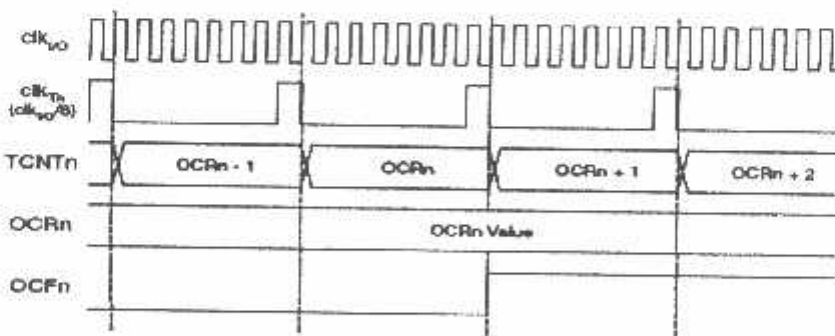
Gambar 2.8 Timing diagram timer/counter, tanpa prescaling^[3]

Sesuai dengan gambar dibawah, timing dengan timer/counter dengan prescaling maksudnya adalah counter akan menambahkan dan counter (TCNTn) ketika terjadi pulsa clock telah mencapai 8 kali pulsa dan sinyal clock pembagi aktif clock telah mencapai nilai maksimal maka nilai TCNTn akan kembali ke nol. Dan kondisi flag timer akan aktif ketika TCNTn maksimal.



Gambar 2.9 Timing diagram timer/counter, dengan prescaling^[3]

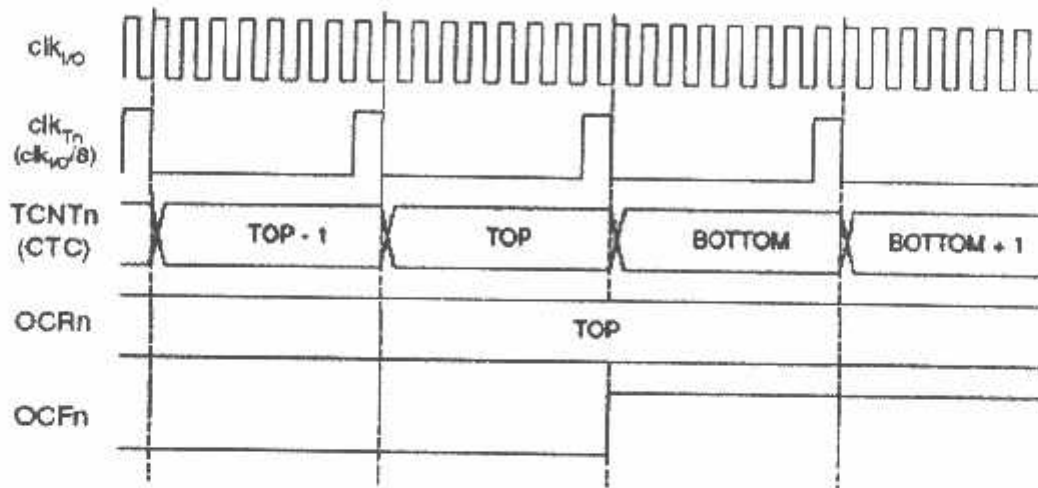
Sama halnya timing timer diatas, timing timer/counter dengan seting OCFO timer mode ini memasukkan data OCRn sebagai data input timer. Ketika nilai OCRn sama dengan nilai TCNTn maka pulsa flag timer akan aktif. TCNTn akan bertambah nilainya ketika pulsa clock telah mencapai 8 pulsa. Dan kondisi flag akan berbalik (komplemen) ketika nilai TCNTn kembali ke nilai 0 (overflow).



Gambar 2.10 Timing diagram timer/counter, menyeting OCFO, dengan pescaler

$(fclk_I/O/8)^{[3]}$

Ketika nilai ORCn sama dengan nilai TCNTn maka pulsa flag timer akan aktif. TCNTn akan bertambah nilainya ketika pulsa clock telah mencapai 8 pulsa. Dan kondisi flag akan berbalik (komplemen) ketika nilai TCNTn kembali ke nilai 0 (overflow).



Gambar 2.11 Timing diagram timer/counter, menyeting OCFO, pengosongan data timer sesuai dengan data pembanding, dengan pescaler ($clk_{I/O}/8$)^[3]

➤ Deskripsi Register Timer/Counter 8 Bit

Bit	7	6	5	4	3	2	1	0	
	FOCO	WGM00	COM01	COM00	WGM01	CS02	CS01	CS00	TCCR0
Read/Write	W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	
Initial Value	0	0	0	0	0	0	0	0	

Gambar 2.12 Register Timer/Counter 8 Bit^[3]

• Bit 7 – FOCO : perbandingan kemampuan output

FOCO hanya akan aktif ketika spesifik-spesifik bit WGM00 tanpa PWM mode. Adapun untuk meyakinkan terhadap kesesuaian dengan device-device yang akan digunakan, bit ini harus diset nol ketika TCCR0 ditulisi saat mengoperasikan mode PWM. Ketika menulisi logika satu ke bit

FOCO, dengan segera di paksakan untuk disesuaikan pada unit pembangkit bentuk gelombang. Output OCO diubah disesuaikan pada COM01: bit 0 menentukan pengaruh daya pembanding.

• **Bit 6,3 – WGM01:0: Waveform Generation Mode**

Bit ini mengontrol penghitungan yang teratur pada counter, sumber untuk harga counter maksimal (TOP), dan tipe apa dari pembangkit bentuk gelombang yang digunakan. Mode-mode operasi didukung oleh unit timer/counter sebagai berikut : mode normal, pembersih timer pada mode penyesuaian dengan pembanding (CTC), dan dua tipe mode Pulse Width Modulation (PWM).

Tabel 2.2 Deskripsi Bit Mode Pembangkit Bentuk Gelombang^[3]

Mode	WGM01 (CTC0)	WGM00 (PWM0)	Timer/Counter Mode of Operation	TOP	Update of OCR0	TOV0 Flag Set-on
0	0	0	Normal	0xFF	Immediate	MAX
1	0	1	PWM, Phase Correct	0xFF	TOP	BOTTOM
2	1	0	CTC	OCR0	Immediate	MAX
3	1	1	Fast PWM	0xFF	TOP	MAX

Catatan : definisi nama-nama bit CTC0 dan PWM0 sekarang tidak digunakan lagi

Gunakan WGM 01: 0 definisi. Bagaimanapun lokasi dan fungsional dan lokasi dari masing-masing bit sesuai dengan versi timer sebelumnya.

• **Bit 5:4 – COM01:0 Penyesuaian Pembanding Mode Output**

Bit ini mengontrol pin output compare (OCO), jika satu atau kedua bit COM01:0 diset,output OCO melebihi fungsional port normal I/O dan keduanya terhubung juga. Bagaimanapun, catatan bahwa bit Direksi Data Register (DDR) mencocokkan ke pin OCO yang mana harus diset dengan

tujuan mengaktifkan. Ketika OC0 dihubungkan ke pin, fungsi dari bit COM01:0 tergantung dari pengesetan bit WGM01:0. Tabel 2.3 di bawah menunjukkan COM functionality ketika bit-bit WGM01:0 diset ke normal atau mode CTC (non PWM).

Tabel 2.3 Mode Output Pemanding, tanpa PWM^[3]

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Toggle OC0 on compare match
1	0	Clear OC0 on compare match
1	1	Set OC0 on compare match

Tabel di bawah menunjukkan bit COM01:0 functionality ketika bit WGM01:0 diset ke mode fast PWM.

Tabel 2.4 Mode Output Pemanding, Mode fast PWM^[3]

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match, set OC0 at TOP
1	1	Set OC0 on compare match, clear OC0 at TOP

Tabel di bawah menunjukkan bit COM01:0 functionality ketika bit WGM01:0 diset ke mode phase correct PWM.

Tabel 2.5 Mode Output Pemanding, Mode phase correct PWM^[3]

COM01	COM00	Description
0	0	Normal port operation, OC0 disconnected.
0	1	Reserved
1	0	Clear OC0 on compare match when up-counting. Set OC0 on compare match when downcounting.
1	1	Set OC0 on compare match when up-counting. Clear OC0 on compare match when downcounting.

• Bit 2:0 – CS02:0 : Clock Select

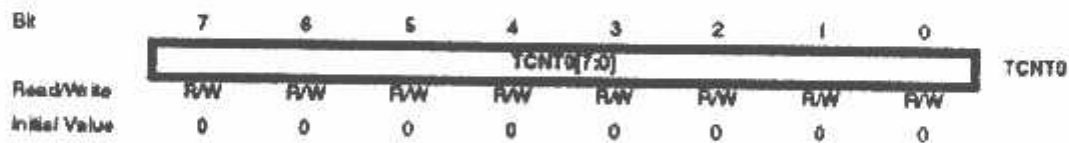
Tiga bit clock select sumber clock digunakan dengan timer/counter.

Tabel 2.6 Deskripsi bit clock select^[3]

CS02	CS01	CS00	Description
0	0	0	No clock source (Timer/Counter stopped).
0	0	1	clk _{IO} (No prescaling)
0	1	0	clk _{IO} /8 (From prescaler)
0	1	1	clk _{IO} /32 (From prescaler)
1	0	0	clk _{IO} /256 (From prescaler)
1	0	1	clk _{IO} /1024 (From prescaler)
1	1	0	External clock source on T0 pin. Clock on falling edge.
1	1	1	External clock source on T0 pin. Clock on rising edge.

Jika mode pin eksternal digunakan untuk timer counter0, perpindahan data pin T0 akan memberi clock counter dengan tetap jika pin digunakan sebagai output. Dalam hal ini software diijinkan untuk mengontrol perhitungan.

➤ Register Timer/Counter TCNT0

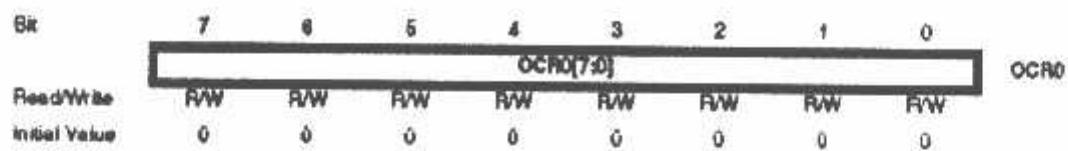


Gambar 2.13 Register Timer TCNT0^[3]

Register timer/counter memberikan akses secara langsung, keduanya digunakan untuk membaca dan menulis operasi, untuk penghitung unit 8-bit timer/counter. Menulis ke blok-blok register TCNT0 (removes) disesuaikan dengan clock timer berikutnya. Memodifikasi counter (TCNT0) ketika

perhitungan berjalan, memperkenalkan resiko kehilangan perbandingan antara TCNC0 dengan register OCR0.

➤ **Register Timer/Counter OCR0**



Gambar 2.14 Register Timer OCR0⁽³⁾

Register output pembanding berisi sebuah haraga 8 bit yang mana secara terus-menerus dibandingkan dengan harga counter (TCNT0). Sebuah penyesuaian dapat digunakan untuk membangkitkan output intrrupt pembanding, atau untuk membangkitakan sebuah output bentuk gelombang pada pin OC0.

➤ **Register Timer/Counter Interrupt Mask**

- **Bit 1-OCIE0** : Output Timer counter menyesuaikan dengan kesesuaian interrupt yang aktif.

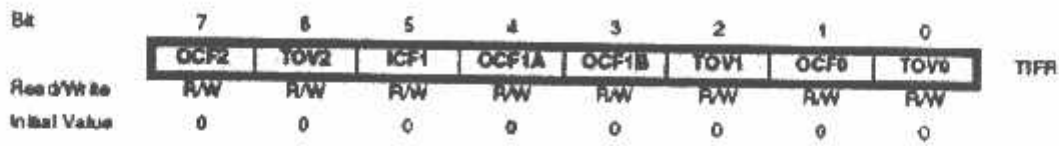
Ketika bit OCIE0 ditulis satu, dan I-bit pada register status dalam kondisi set (satu), membandingkan timer/counter pada interrupt yang sesuai diaktifkan. Mencocokkan interrupt yang dijalankan kesesuaian pembanding pada timer/counter0 terjadi, ketika bit OCF0 diset pada register penanda timer/counter-TIFR.

- **Bit 0 – TOIE0: Timer/Counter 0 Overflow Interrupt Enable**

Ketika bit TOIE0 ditulis satu, dan I-bit pada register status dalam kondisi set (satu), timer/counter melebihi interrupt diaktifkan. Mencocokkan

interrupt dijalankan jika kelebihan pada timer/counter0 terjadi, ketika bit TOV0 diset pada register penanda timer/counter-TIFR.

➤ **Register Timer/Counter Register - TIFR**



Gambar 2.15 Register timer TIFR^[3]

• **Bit 1 - OCF0: Output Compare Flag0**

OCF0 dalam kondisi set (satu) kesesuaian pembanding terjadi antara timer/counter dan data pada OCRO – Register 0 keluaran pembanding. OCF0 diclear oleh hardware ketika eksekusi pencocokan penanganan vector interrupt. Dengan alternatif mengclearkan OCF0 dengan menuliskan logika satu pada flag. Ketika I-bit pada SREG, OCIE0 (Timer/Counter0 penyesuaian pembanding interrupt enable), dan OCF0 diset (satu), timer/counter pembanding kesesuaian interrupt dijalankan.

• **Bit 0 – TOV0: Timer/Counter Overflow Flag**

Bit TOV0 di ser (satu) ketika kelebihan terjadi pada timer/counter0. TOV0 diclearkan dengan hardware ketika penjalanan pencocokan penanganan vector interrupt. Dengan alternatif, TOV0 diclearkan dengan jalan memberikan logika satu pada flag. Ketika I-bit pada SREG, TOIE0 (Timer/Counter0 *overflow* interrupt enable), dan TOV0 diset (satu), timer/counter *overflow* interrupt dijalankan. Pada tahap mode PWM yang tepat, bit ini di set ketika timer/counter merubah bagian perhitungan pada \$00.

2.2.5 Serial Pada ATmega16

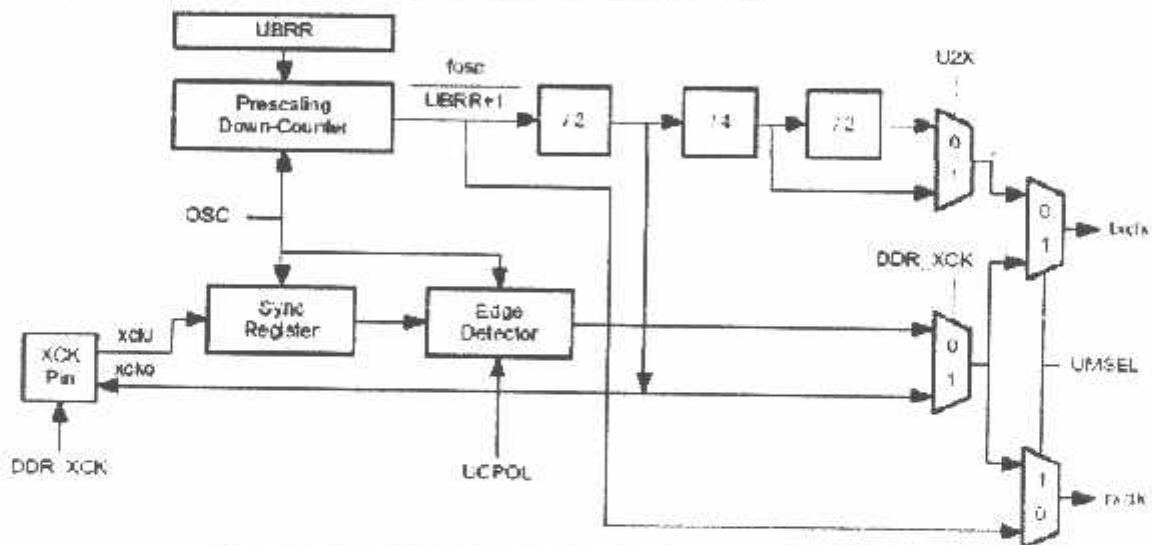
Universal synchronous dan asynchronous pemancar dan penerima serial adalah suatu alat komunikasi serial sangat fleksibel. Jenis yang utama adalah :

- Operasi *full duplex* (register penerima dan pengirim serial dapat berdiri sendiri)
- Operasi *Asynchronous* atau *synchronous*
- *Master* atau *slave* mendapat *clock* dengan operasi *synchronous*
- Pembangkit *boud rate* dengan resolusi tinggi
- Dukung frames serial dengan 5, 6, 7, 8 atau 9 Data bit dan 1 atau 2 Stop bit
- Tahap *odd* atau *even parity* dan *parity check* didukung oleh *hardware*
- Pendeteksian data *overrun*
- Pendeteksi *framing error*
- Pemfilteran gangguan (*noise*) meliputi pendeteksian *bit false start* dan pendeteksian *low pass filter digital*
- Tiga *interrupt* terdiri dari *TX complete*, *TX data register empty* dan *RX complete*.
- Mode komunikasi *multi-processor*
- Mode komunikasi *double speed asynchronous*

➤ **Generasi Clock**

Logic generasi *clock* menghasilkan dasar *clock* untuk pengirim dan penerima. USART mendukung empat *mode operasi clock* : *Normal Asynchronous*, *Double Speed Asynchronous mode Master Synchronous* dan *Slave Synchronous*.

Bit UMSEL pada USART control dan status register C (UCSRC) memilih antara operasi *Asynchronous* dan *Synchronous*. *Double speed* (hanya pada mode *Asynchronous*) dikontrol oleh U2X yang mana terdapat pada register UCSRA. Ketika menggunakan mode operasi *synchronous* (UMSEL = 1) dan data direction register untuk pin XCK (DDR_XCK) mengendalikan apakah sumber clock tersebut adalah *internal* (*master mode*) atau *eksternal* (*slave mode*) pinpin XCK hanya akan aktif ketika menggunakan mode *Synchronous*.



Gambar 2.16 Block diagram clock generasi logic^[3]

Keterangan sinyal :

Txclk : Clock pengirim (*internal clock*)

Rxclk : Clock dasar penerima (*internal clock*)

Xcki : Input dari pin XCK (sinyal *internal*).

Digunakan untuk operasi *slave synchronous*.

Xcko : clock output ke pin XCK (sinyal *internal*).

Digunakan untuk operasi *master synchronous*

Fosc : Frekuensi pin XTAL (*system clock*)

➤ **Generasi Internal Clock – Pembangkit Baud Rate**

Generasi *internal clock* digunakan untuk mode-mode operasi master *asynchronous* dan *synchronous*. Register USART *boud rate* (UBRR) dan *down-counter* dikoneksikan kepada fungsinya sebagai *programmable prescaler* atau pembangkit *boud rate*. *Down-counter*, dijalankan pada *system clock* (f_{osc}), dibebani dengan nilai UBRR setiap *counter* telah dihitung mundur ke nol atau ketika register UBRR ditulisi. *Clock* dibangkitkan setiap *counter* mencapai nol. *Clock* ini adalah pembangkit *boud rate clock output* ($f_{osc}/(UBRR+1)$). Pemancar membagi boud rete generator clock output dengan 2, 8, atau 16 cara tergantung pada mode. Pembangkit *output boud rate* digunakan secara langsung oleh penerima *clock* dan unit-unit pelindung data. Unit-unit *recovery* menggunakan suatu mesin status yang menggunakan 2, 8, atau 16 negara yang tergantung pada cara menyimpan status dari UMSEL, bit-bit U2X dan DDR_XCK. Table di bawah menunjukkan penyamaan perhitungan *boud rate* dan nilai UBRR tiap mode operasi menggunakan sumber pembangkit *clock internal*.

Tabel 2.7 Persamaan untuk menyeting perhitungan register Baud Rate^[2]

Operating Mode	Equation for Calculating Baud Rate ⁽¹⁾	Equation for Calculating UBRR Value
Asynchronous Normal Mode (U2X = 0)	$BAUD = \frac{f_{osc}}{16(UBRR + 1)}$	$UBRR = \frac{f_{osc}}{16BAUD} - 1$
Asynchronous Double Speed Mode (U2X = *)	$BAUD = \frac{f_{osc}}{8(UBRR + 1)}$	$UBRR = \frac{f_{osc}}{8BAUD} - 1$
Synchronous Master Mode	$BAUD = \frac{f_{osc}}{2(UBRR + 1)}$	$UBRR = \frac{f_{osc}}{2BAUD} - 1$

➤ **Eksternal Clock**

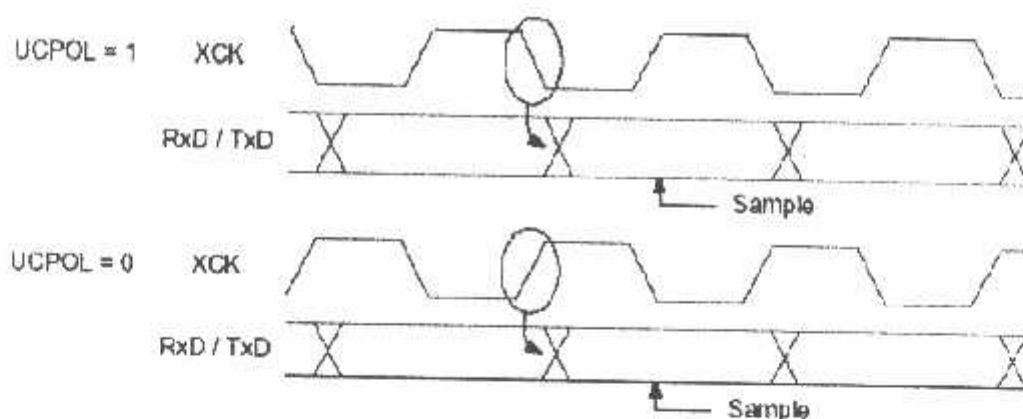
Eksternal clock digunakan untuk operasi mode *slave synchronous*. *Eksternal clock* masuk dari pin XCK dicontohkan oleh suatu daftar sinkronisasi *register* untuk memperkecil kesempatan meta-stabilitas. Keluaran dari sinkronisasi *register* kemudian harus menerobos *detector* tepi sebelum digunakan oleh pengirim dan penerima. Proses ini mengenakan dua *period delay clock* CPU dan oleh karena itu maksimal frekuensi *clock* XCK *eksternal* dibatasi oleh persamaan sebagai berikut :

$$f_{XCK} < \frac{f_{OSC}}{4}$$

Keterangan : bahwa f_{osc} tergantung pada stabilitas system sumber *clock*. Oleh karena derekomendasikan untuk menambahkan beberapa garis tepi untuk menghindari hilangnya mungkin data dalam kaitan dengan variasi frekwensi.

➤ **Operasi Synchronous Clock**

Ketika mode sinkron digunakan (UMSEL=1), pin XCK akan digunakan sama seperti *clock input (slave)* atau *clock output (master)*. Dengan ketergantungan antara tepi *clock* dan data sampling atau perubahan data menjadi sama. Prinsip dasarnya adalah data *input (on RxD)* dicontohkan pada *clock* XCK berlawanan tepi dari tepi data *output (TxD)* sehingga mengalami perubahan.



Gambar 2.17 Operasi *Synchronous Clock*^[3]

UCPOL bit UCRSC memilih yang mana tepi clock XCK digunakan untuk data sampling dan yang mana digunakan untuk perubahan data. Seperti yang ditunjukkan pada gambar di atas, ketika UCPOL nol data akan diubah pada tepi kenaikan XCK dan dicontohkan pada tepi XCK saat jatuh. Jika UCPOL dalam kondisi set, data akan mengalami perubahan pada saat tepi XCK jatuh dan data akan dicontohkan pada saat tepi XCK naik.

➤ **Inisialisasi USART**

USART harus diinisialisasi sebelum komunikasi manapun dapat berlansung. Proses inisialisasi normalnya terdiri dari pengesetan *boud rate*, penyetingan frame format dan pengaktifan pengirim atau penerima tergantung pada pemakaian. Untuk interrupt menjalankan operasi USART, *global interrupt flag* (penanda) sebaiknya dibersihkan (dan *interrupt global disable*) ketika inisialisasi dilakukan.

Sebelum melakukan inisialisasi ulang dengan mengubah *boud rate* atau frame format, untuk meyakinkan bahwa tidak ada transmisi berkelanjutan sepanjang periode *register* yang diubah. *Flag TXC* dapat digunakan untuk

mengecek bahwa pemancar telah melengkapi semua pengiriman, dan *flag* RXC dapat digunakan untuk mengecek bahwa tidak ada data yang tidak terbaca pada *buffer* penerima. Tercatat bahwa *flag* TXC harus dibersihkan sebelum tiap transmisi (sebelum UDR ditulisi) jika itu semua digunakan untuk tujuan tersebut.

USART sederhana inisialisasi kode contoh berikut menunjukkan fungsi satu assembly dan satu C itu mempunyai kesamaan dalam kemampuan. Pada contoh tersebut mengasumsikan bahwa operasi asinkron menggunakan metode poling (tidak ada *interrupt enable*) frame format yang tetap. *Boud rate* diberikan sebagai fungsi parameter. Untuk kode assembly, parameter *boud rate* diasumsikan untuk di simpan pada register r16, r17. Ketika menulis fungsi pada register UCSRC, bit URSEL (MSB) harus diset dalam kaitan dengan pembagian penempatan I/O oleh UBRRH dan UCSRC.

Lebih mengedepankan inisialisasi rutin dapat dibuat seperti itu meliputi frame format sebagai parameter, *disable interrupt* dan lain-lain. Bagai manapun juga banyak aplikasi menggunakan seting tetap *boud* dan *register control*, dan untuk aplikasi jenis ini dapat ditempatkan secara langsung pada keseluruhan *routine*, atau dikombinasikan dengan inisialisasi kode untuk modul I/O yang lain.

2.3. Sensor Ultrasonik

2.3.1. Teori Dasar Sensor Ultrasonik

Sensor ultrasonik ini memiliki rangkaian yang sudah jadi yaitu berasal dari modul parallax. Pada robot ini menggunakan tiga buah sensor ultrasonik yang dipasang pada bagian depan robot yang digunakan untuk mendeteksi halangan depan, sensor kiri untuk mendeteksi halangan samping kiri dan sensor kanan

untuk mendeteksi halangan samping kiri. Sensor ultrasonik ini mengirim data ke mikrokontroller secara terus-menerus sehingga semua sensor ultrasonik akan aktif dalam pengiriman data. Untuk itu proses scanning sensor ultrasonik terhadap objek dilakukan secara bergantian agar data yang didapat lebih mudah dan valid. *Output* dari modul ini berupa data pwm sehingga data jarak sama dengan *duty cycle* dari sinyal *output*. Semakin jauh objek maka semakin besar *duty cycle*.



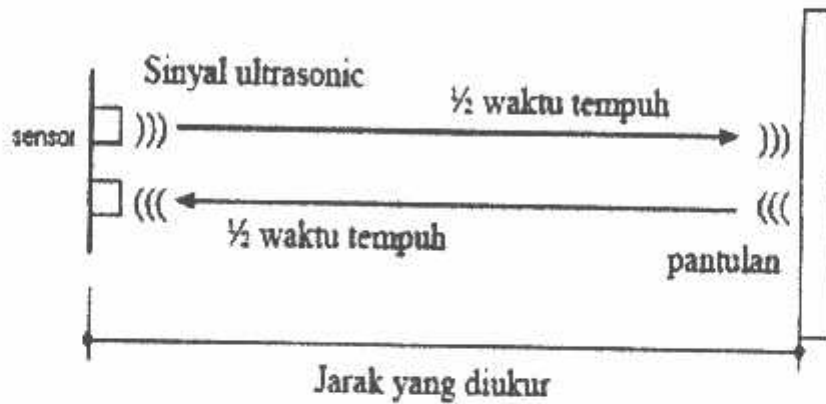
Gambar 2.18 Sensor ultrasonik^[4]

Dibandingkan dengan infrared, ultrasonic mempunyai kemampuan mendeteksi obyek lebih jauh terutama untuk benda-benda yang keras. Pada benda-benda yang mempunyai permukaan keras gelombang ini akan dipantulkan lebih kuat daripada benda-benda yang mempunyai permukaan lunak.

Sesuai dengan namanya, ultrasonic adalah sebuah gelombang yang mempunyai frekwensi di atas pendengaran manusia yaitu di atas 20 KHz. Pada umumnya rangkaian ultrasonic menggunakan frekwensi 40 KHz yang dihasilkan oleh rangkaian osilator. Pengenalan obyek atau jarak antara sensor dengan obyek dapat dikenali dengan menghitung perbedaan waktu dari saat sinyal ultrasonic pertama kali dipancarkan hingga diterima kembali oleh sensor.

Sensor ultrasonik adalah sensor yang bekerja berdasarkan prinsip pantulan gelombang suara, dimana sensor ini menghasilkan gelombang suara yang

kemudian menangkapnya kembali dengan perbedaan waktu sebagai dasar pengindraannya. Perbedaan waktu antara gelombang suara yang dipancarkan dengan gelombang suara ditangkap kembali tersebut adalah berbanding lurus dengan jarak atau tinggi objek yang memantulkannya. Jenis objek yang dapat diindra diantaranya adalah: objek padat, cair, butiran maupun tekstil.

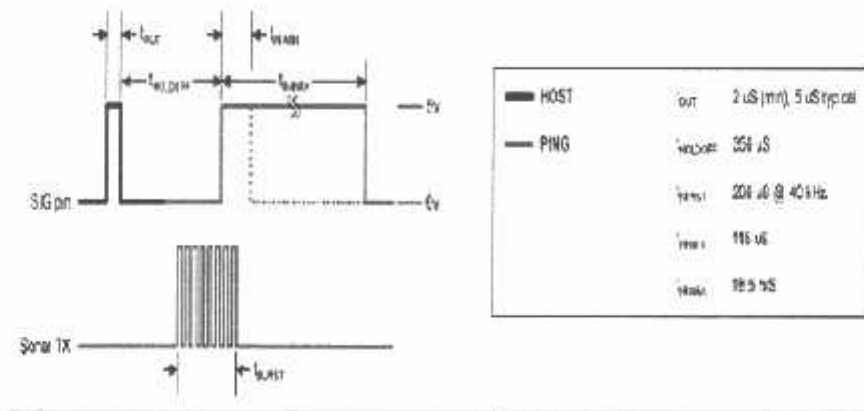


Gambar 2.19 Cara kerja sensor^[4]

2.3.2 Teori operasi sensor :

Sensor mendeteksi jarak obyek dengan cara memancarkan gelombang ultrasonik (40 kHz) selama $t_{BURST}(200 \mu s)$ kemudian mendeteksi pantulannya. Sensor Ping memancarkan gelombang ultrasonik sesuai dengan kontrol dari mikrokontroler pengendali (pulsa *trigger* dengan t_{OUT} min. $2 \mu s$). Gelombang ultrasonik ini melalui udara dengan kecepatan 344 meter per detik, mengenai halangan dan memantul kembali ke sensor. Ping mengeluarkan pulsa *output high* pada pin SIG setelah memancarkan gelombang ultrasonik dan setelah gelombang pantulan terdeteksi Ping akan membuat *output low* pada pin SIG. Lebar pulsa *High* (t_{IN}) akan sesuai dengan lama waktu tempuh gelombang ultrasonik untuk

2x jarak ukur dengan halangan. Maka jarak yang diukur adalah $[(t_{IN} \times 344 \text{ m/s}) \div 2]$ meter.



Gambar 2.20 Diagram waktu sensor^[4]

2.4. Motor Servo

2.4.1. Teori Dasar Motor Servo

Motor servo adalah motor yang mampu bekerja dua arah (CW dan CCW) dimana arah dan sudut pergerakan rotornya dapat dikendalikan hanya dengan memberikan pengaturan duty cycle sinyal PWM pada bagian pin kontrolnya.

Motor Servo merupakan sebuah motor DC yang memiliki rangkaian *control elektronik* dan *internal gear* untuk mengendalikan pergerakan dan sudut sudutnya.

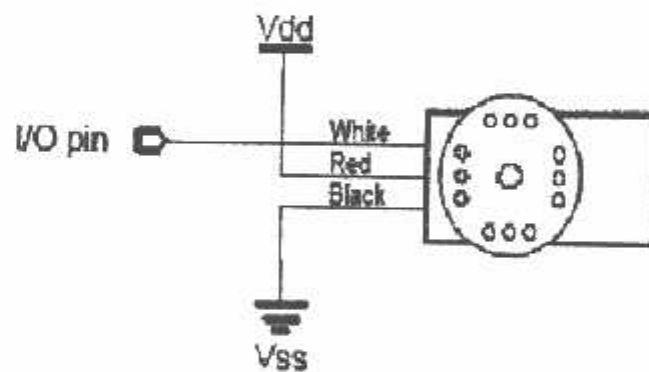


Gambar 2.21 Parallax standar servo^[5]

Motor servo adalah motor yang berputar lambat, dimana biasanya ditunjukkan oleh *rate* putarannya yang lambat, namun demikian memiliki torsi yang kuat karena *internal gear*nya.

Lebih dalam dapat digambarkan bahwa sebuah motor servo memiliki :

- jalur kabel : *power*, *ground*, dan *control*
- Sinyal *control* mengendalikan posisi
- Operasional dari servo motor dikendalikan oleh sebuah pulsa selebar ± 20 ms, dimana lebar pulsa antara 0.5 ms dan 2 ms menyatakan akhir dari range sudut maksimum.
- Konstruksi didalamnya meliputi *internal gear*, *potensiometer*, dan *feedback control*.



Gambar 2.22 Diagram pengkabelannya⁽⁵⁾

2.4.2. Jenis Motor Servo

Adapun jenis-jenis motor servo adalah :

- ⌚ Motor Servo Standar 180°

Motor servo jenis ini hanya mampu bergerak dua arah (CW dan CCW) dengan defleksi masing-masing sudut mencapai 90° sehingga total defleksi sudut dari kanan – tengah – kiri adalah 180°.

⌚ *Motor Servo Continuous*

Motor servo jenis ini mampu bergerak dua arah (CW dan CCW) tanpa batasan defleksi sudut putar (dapat berputar secara kontinyu).

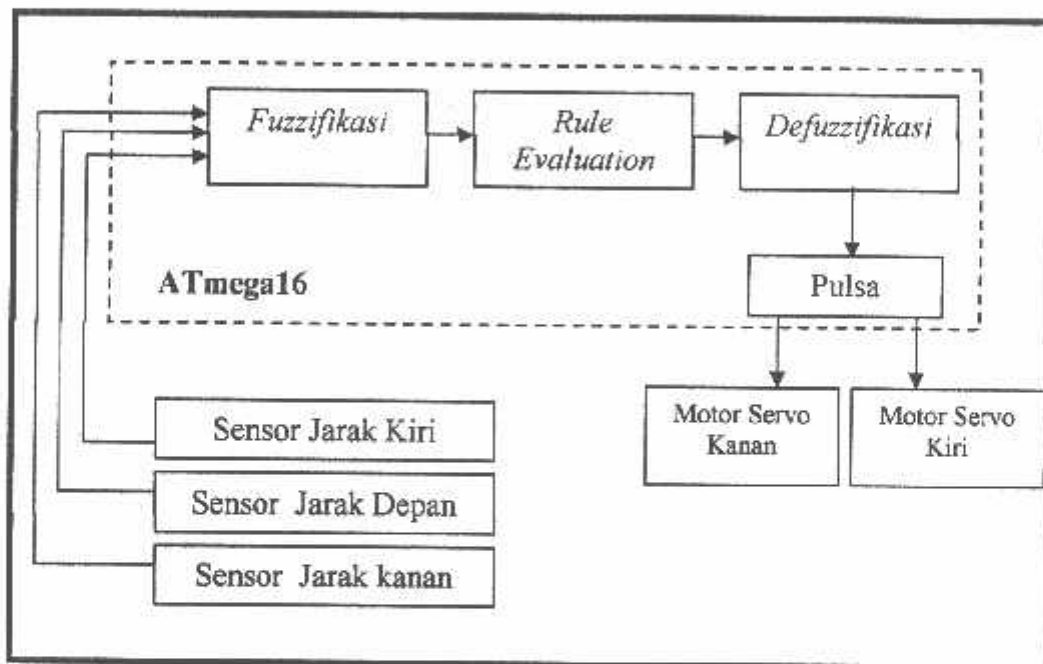


BAB III

PERANCANGAN DAN PEMBUATAN ALAT

Secara umum sistem kontrol yang terdapat pada robot penghindar halangan terdapat dua bagian dasar, yaitu bagian perangkat keras (*hardware*) dan perangkat lunak (*software*).

3.1. Perancangan Perangkat Keras (*Hardware*)



Gambar 3.1 Diagram Blok Keseluruhan Sistem

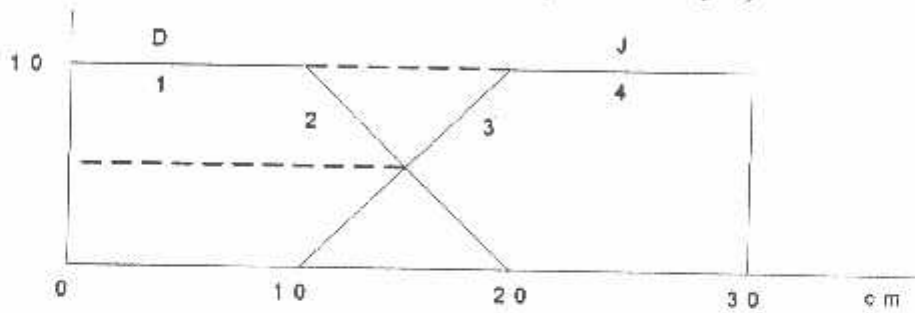
Fungsi dari tiap-tiap blok diagram dijelaskan sebagai berikut :

1. Unit system MCU ATmega16 merupakan unit pengolah data.
2. Unit Sensor berfungsi sebagai *input* untuk mendeteksi halangan.
3. Unit servo berfungsi sebagai *output* untuk menggerakkan roda mobil robot.

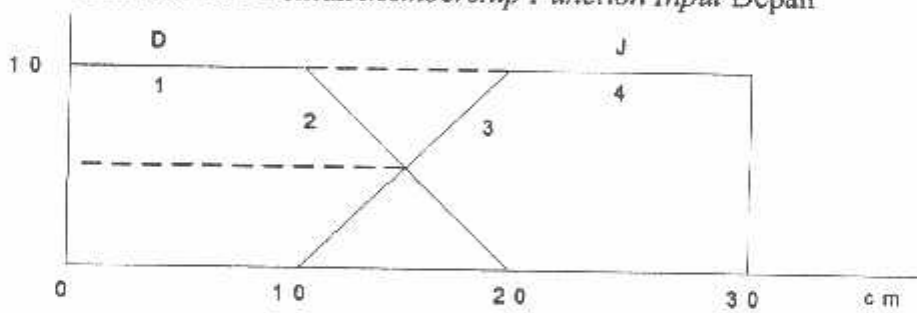
3.1.1. Perancangan Kontroller Logika Fuzzy

3.1.1.1. Fuzzification

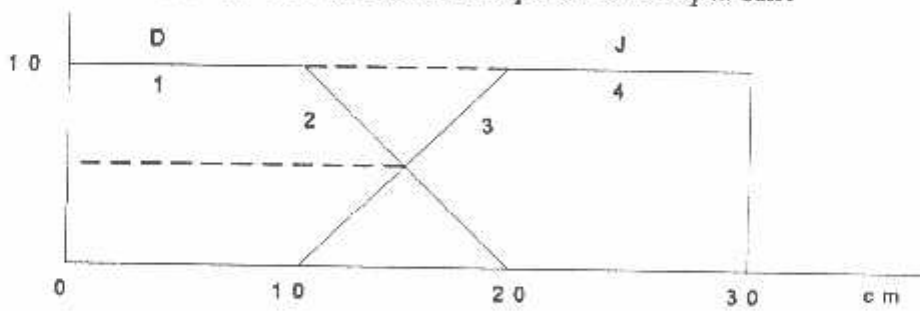
Dalam Proses *Fuzzification* ini terdapat 3 *input* yaitu sensor ultrasonik sisi kiri, depan dan kanan. Masing-masing *input* ini diwakili oleh *variable input* 8bit. Dimana *input* 8bit ini menyatakan jarak robot mobil terhadap dinding atau penghalang. Bentuk *membership function* yang dipilih adalah *Trapezoid* dan *triangles*. Untuk setiap *input* diberi label jauh (J) dan dekat (D).



Gambar 3.2 Bentuk *Membership Function* Input Depan



Gambar 3.3 Bentuk *Membership Function* Input Kiri



Gambar 3.4 Bentuk *Membership Function* Input Kanan

Dari gambar di atas tersebut terdapat sepuluh persamaan garis. Setiap persamaan garis ini harus didapatkan persamaan garisnya supaya dapat diperoleh *Fuzzy Input*-nya. Dimana untuk mendapatkan persamaan tersebut mengikuti rumus persamaan garis yang sudah ada :

$$Y - Y_1 = m (X - X_1)$$

$$\text{Dimana, } m = \frac{Y_2 - Y_1}{X_2 - X_1}$$

Untuk persamaan garis 1 :

$$Y_{(x)} = 10 \quad , \text{ untuk } X \leq 10$$

Persamaan garis 2 :

$$M = -1$$

$$Y - 10 = -1 (X - 10)$$

$$Y = 20 - X$$

$$Y_{(x)} = 20 - X \quad , \text{ untuk } 10 < X < 20$$

Persamaan garis 3 :

$$M = 1$$

$$Y - 0 = 1 (X - 20)$$

$$Y_{(x)} = X - 20 \quad , \text{ untuk } 10 < X < 20$$

Persamaan garis 4 :

$$M = 10 \quad , \text{ untuk } X < 20$$

Dari perhitungan persamaan garis di atas maka dapat dikelompokkan *membership function input* sebagai berikut :

- Untuk *input* sensor depan adalah :

$$D \begin{cases} 10 & , \text{ untuk } X \leq 10 \\ 20 - X & , \text{ untuk } 10 < X < 20 \\ 0 & , \text{ untuk } X \geq 20 \end{cases}$$

$$J \begin{cases} 0 & , \text{ untuk } X \leq 10 \\ X - 20 & , \text{ untuk } 10 < X < 20 \\ 10 & , \text{ untuk } X \geq 20 \end{cases}$$

- Untuk *input* sensor kiri adalah :

$$D \begin{cases} 10 & , \text{ untuk } X \leq 10 \\ 20 - X & , \text{ untuk } 10 < X < 20 \\ 1 & , \text{ untuk } X \geq 20 \end{cases}$$

$$J \begin{cases} 0 & , \text{ untuk } X \leq 10 \\ X - 20 & , \text{ untuk } 10 < X < 20 \\ 10 & , \text{ untuk } X \geq 20 \end{cases}$$

- Untuk *input* sensor kanan adalah :

$$D \begin{cases} 10 & , \text{ untuk } X \leq 10 \\ 20 - X & , \text{ untuk } 10 < X < 20 \\ 2 & , \text{ untuk } X \geq 20 \end{cases}$$

$$J \begin{cases} 0 & , \text{ untuk } X \leq 10 \\ X - 20 & , \text{ untuk } 10 < X < 20 \\ 10 & , \text{ untuk } X \geq 20 \end{cases}$$

3.1.1.2. Rule Evaluation

Rule yang digunakan dalam sistem ini, diekstrak dari logika manusia. Sebagai contoh bila sensor depan dekat dan sensor kiri dekat dan sensor kanan dekat dengan penghalang maka motor kiri akan berhenti dan motor kanan akan berhenti.. Tabel 3.1 menunjukkan sebagian dari rule yang telah dirancang.

Tabel 3.1 Rule Evaluation

Inputs			Outputs	
Sensor Depan	Sensor Kiri	Sensor Kanan	Motor Kiri	Motor Kanan
Dekat	Dekat	Jauh	Stop	Stop
Dekat	Jauh	Jauh	Cepat	Pelan
Jauh	Jauh	Dekat	Pelan	Cepat
Jauh	Jauh	Jauh	Cepat	Cepat
Jauh	Dekat	Jauh	Cepat	Pelan
Dekat	Jauh	Dekat	Cepat	Cepat

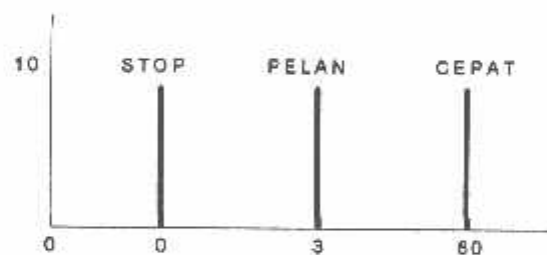
Dari pernyataan tadi maka dapat diekstrak *rule*

1. *IF* Sensor Depan *is* Dekat *AND* Sensor Kiri *is* Dekat *AND* Sensor Kanan *is* Dekat *THEN* Motor Kiri *is* Stop *AND* Motor Kanan *is* Stop
2. *IF* Sensor Depan *is* Dekat *AND* Sensor Kiri *is* Jauh *AND* Sensor Kanan *is* Jauh *THEN* Motor Kiri *is* Cepat *AND* Motor Kanan *is* Pelan
3. *IF* Sensor Jauh *is* Dekat *AND* Sensor Kiri *is* Jauh *AND* Sensor Kanan *is* Dekat *THEN* Motor Kiri *is* Pelan *AND* Motor Kanan *is* Cepat

4. *IF* Sensor Jauh *is* Dekat *AND* Sensor Kiri *is* Jauh *AND* Sensor Kanan *is* Jauh
THEN Motor Kiri *is* Cepat *AND* Motor Kanan *is* Cepat
5. *IF* Sensor Depan *is* Jauh *AND* Sensor Kiri *is* Dekat *AND* Sensor Kanan *is* Jauh
THEN Motor Kiri *is* Cepat *AND* Motor Kanan *is* Pelan
6. *IF* Sensor Depan *is* Dekat *AND* Sensor Kiri *is* Jauh *AND* Sensor Kanan *is* Dekat
THEN Motor Kiri *is* Cepat *AND* Motor Kanan *is* Cepat

3.1.1.3. Defuzzification

Proses terakhir dari *fuzzy logic* adalah *Defuzzifikasi* yaitu untuk menentukan suatu nilai *crisp output*. Suatu nilai *fuzzy output* yang berasal dari *rule evaluation* diambil kemudian dimasukkan ke dalam suatu *membership function output*.



Gambar 3.5 *Membership Function Output* Kiri dan Kanan

Besar nilai *fuzzy output* dinyatakan sebagai *degree of membership function output*. Nilai-nilai tersebut dimasukkan ke dalam suatu rumus yang dinamakan *COG (Center Of Gravity)* untuk menentukan hasil akhir yang disebut *crisp output*. Rumus yang digunakan dalam proses ini adalah :

$$\text{Crisp Output [Y]} = \frac{S_i (\text{fuzzy output}) \times \left(\begin{array}{c} \text{Singleton position} \\ \text{On x axis i} \end{array} \right)}{S_i (\text{fuzzy output})}$$

This simple demo illustrates the use of the PING))) ultrasonic range finder class with the Javelin Stamp:

```
import stamp.core.*;
import stamp.peripheral.sensor.Ping;

public class testPing {

    public static final char HOME = 0x01;

    public static void main() {

        Ping range = new Ping(CPU.pin0);
        StringBuffer msg = new StringBuffer();

        int distance;

        while (true) {
            // measure distance to target in inches
            distance = range.getIn();

            // create and display measurement message
            msg.clear();
            msg.append(HOME);
            msg.append(distance);
            msg.append(" \n");
            System.out.print(msg.toString());

            // wait 0.5 seconds between readings
            CPU.delay(5000);
        }
    }
}
```

```

/* File include */
#include <mega16.h>
#include <delay.h>

/* Pendefinisian */

#define servo2    PORTC.0
#define servol    PORTC.1

#define pingka    PORTA.0
#define pongka    PINA.1
#define pingta    PORTA.2
#define pongta    PINA.3
#define pingki    PORTA.4
#define pongki    PINA.5

/* Inisialisasi variabel global */
unsigned char tanda;
int a,b;
long int ;
unsigned int i,timer,jki,jka,jta,j;
bit tka,tki,trx,tb;
long int k1,kawal;
//flash int
konstanta_timing[25]={1,666,333,111,167,133,112,96,84,74,66,60,56,
52,48,44,42,40,38,36,34,32,30,28,28};

/*Inisialisasi input output */
void init_port()
{
    DDRC=0b00110011;
    DDRB=0b00011111;
    DDRD=0b00111100;
    DDRA=0b00010101;
    //PORTA.5=1;
    //PORTA=0b11111111;
    //scl=1;sdad=1;
}

void putar1(unsigned char pu)
{
    unsigned char ii,jj;

    servol=1;
    for(jj=0;jj<pu;jj++) delay_us(10);
    servol=0;delay_ms(20);
}

void putar2(unsigned char pu)
{
    unsigned char ii,jj;

```

```

servo2=1;
for(jj=0;jj<pu;jj++) delay_us(10);
servo2=0;delay_ms(20);
}
void maju(int kiri,int kanan)
{
int aa,bb;
aa=120-kanan;if(aa<40) aa=40;
bb=120+kiri;if(bb>200) bb=200;
putar1(aa);
putar2(bb);
}
int ultrakiri()
{
pingki=1;delay_us(5);
pingki=0;delay_us(15);
k1=0;

while(pongki==1) ;
while(pongki==0) k1++;
//kawal=k1*0/3;
return k1/10;
}
int ultrakanan()
{
pingka=1;delay_us(5);
pingka=0;delay_us(15);
k1=0;

while(pongka==1) ;
while(pongka==0) k1++;
//kawal=k1*0/3;
return k1/10;
}
int ultratengah()
{
pingta=1;delay_us(5);
pingta=0;delay_us(15);
k1=0;

while(pongta==1) ;
while(pongta==0) k1++;
//kawal=k1*0/3;
return k1/10;
}
/* Program Utama */
void main()
{

/* Inisialisasi */
init_port();
tka=0;tki=0;
tanda=0;
a=30;

```

```

        b=80;

        do
        {
j=ultrakanan();
jka=j;
    tampil(jka);
    j=ultrakiri();
jki=j;
tampil(jki);
    j=ultratengah();
jta=j;
    tampil(jta);
if(tanda==0)
{
    if(jka>20 && jki>20 && jta>20) maju(80,80);
    else if(jka<10 && jki<10 && jta<10);
    else if( jki<jka) maju(80,3);
    else if(jki>jka) maju(3,80);
    else if( jki==jka) maju(80,80);
    else maju(80,80);
}
else if(tanda==bkanan)
{
    if(jka<9) tanda=0;
    maju(2,80);
}
    else if(tanda==bkiri)
    {
    if(jki<9) tanda=0;
    maju(80,2);
}

        }while(1);
}

```

DATA SHEET
MOTOR SERVO

Standard Servo Motor

AN-08

Oleh: Tim Digiware

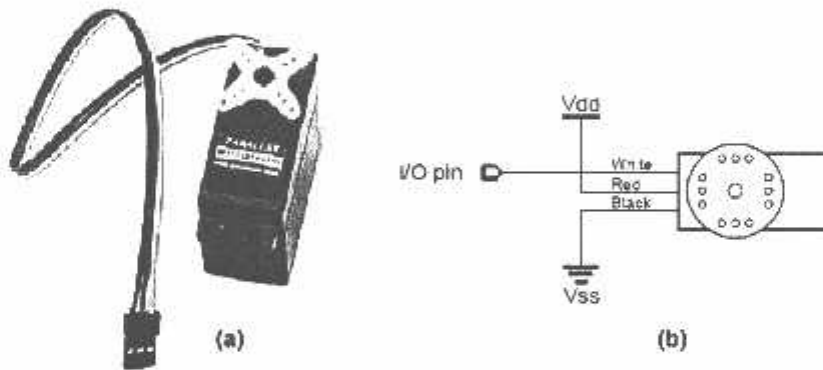
S *tandard servo motor* seringkali digunakan sebagai divais penggerak dalam aplikasi robotika, karena motor dengan *built-in* umpan balik posisi ini memiliki torsi relatif cukup kuat meskipun hanya mampu berputar 0 sampai 180 derajat. Penggunaan *standard servo motor* ini cukup sederhana, yaitu dengan memberi pulsa digital yang memiliki lebar tertentu melalui sebuah pin kontrol untuk membuat motor ini berputar ke arah jarum jam, berlawanan jarum jam, atau bergerak ke posisi tengah. Dalam aplikasi ini DT-51™ Low Cost Nano System akan mengendalikan posisi sudut dari poros *standard servo motor* sesuai perintah dari komputer. Bahasa pemrograman mikrokontrolernya menggunakan C dengan *compiler* µC/51© dan Borland Delphi 5 untuk pembuatan program aplikasi komputernya.

Aplikasi ini membutuhkan:

- 1 DT-51™ Low Cost Nano System,
- 1 *Standard servo motor* (pada AN ini menggunakan Parallax Standard Servo).

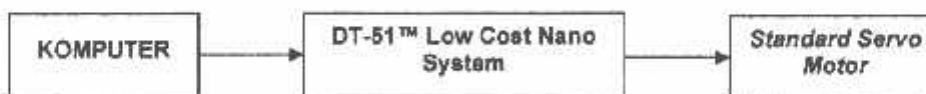
Parallax Standard Servo (Gambar 1) memiliki spesifikasi sebagai berikut:

- Catu daya : 6 VDC (maksimum),
- Waktu putar : 1,5 detik / 180 derajat (rata-rata),
- Berat fisik : 45 gram,
- Torsi putar : 3,40 kg-cm,
- Ukuran fisik : 40,5 mm (P) x 20 0 mm (L) x 38,0 mm (T)



Gambar 1
Parallax Standard Servo (a) dan Diagram Pengkabelannya (b)

A dapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



Gambar 2
Blok Diagram AN-08

Hubungan antara modul-modul tersebut adalah sebagai berikut:

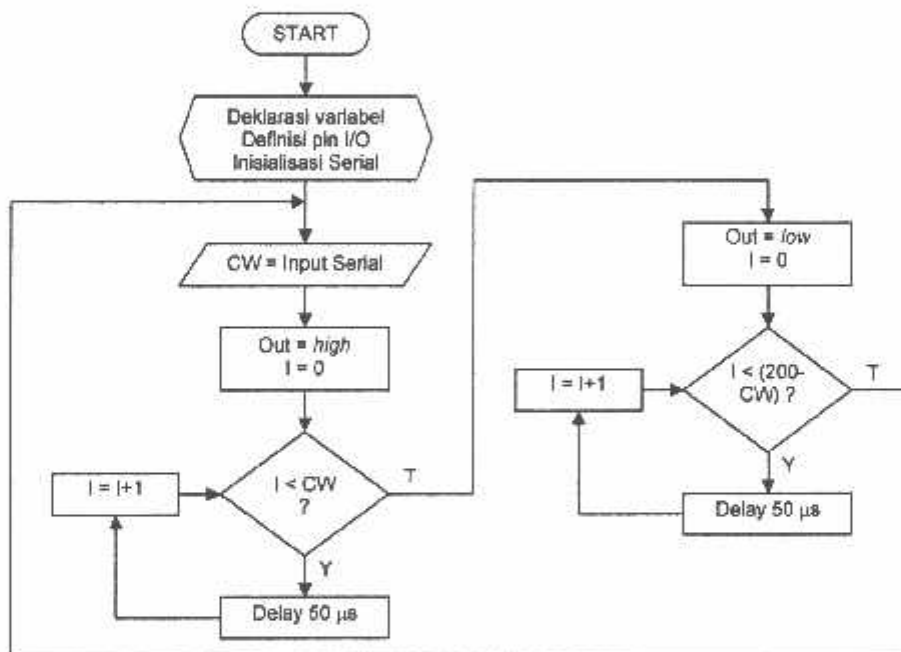
DT-51™ Low Cost Nano System	Parallax Standard Servo
VCC	VCC (<i>red</i>)
GND	GND (<i>black</i>)
P1.1*	INPUT (<i>white</i>)

* Pin ini tidak mutlak dan dapat diganti pin lain tetapi harus mengubah program

Tabel 1
Hubungan DT-51™ Low Cost Nano System dengan Parallax Standard Servo

Gunakan kabel serial DT-51™ Low Cost Nano System untuk menghubungkan modul ke COM port komputer. Lepaslah AT89C2051 dari board DT-51™ Low Cost Nano System, lalu gunakan TOP2004 Universal Programmer atau DT-HiQ Programmer atau DT-51™ MinSys v3.0 + DT-51™ ProgPAL atau divais *parallel programmer* lain untuk memprogram Servo.hex ke dalam AT89C2051. Setelah terisi dengan program Servo.hex, pasanglah AT89C2051 kembali ke board DT-51™ Low Cost Nano System. Dan setelah rangkaian dan sumber tegangan terhubung dengan benar, jalankan program Servo.exe pada komputer.

Flowchart program untuk DT-51™ Low Cost Nano System adalah sebagai berikut:



Gambar 3
Flowchart Program untuk DT-51™ Low Cost Nano System

Program utama dari Servo.c akan diproses sebagai berikut:

1. Proses yang pertama dilakukan adalah deklarasi variabel CW dan I serta definisi pin Out.

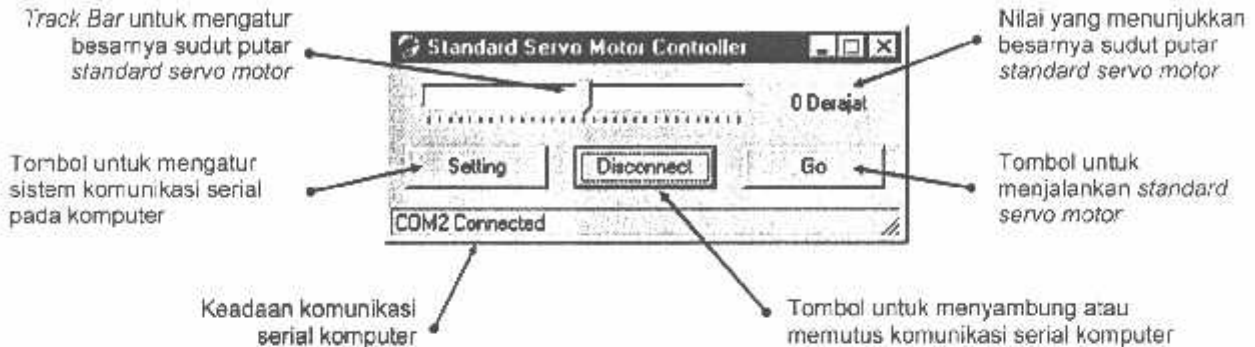
CW adalah variabel bertipe *character* yang akan berisi nilai yang menentukan besarnya lebar pulsa *high* yang akan dikirimkan oleh DT-51™ Low Cost Nano System ke *standard servo motor* (50 μ s tiap satuannya).

I adalah variabel bertipe *character* yang akan dipakai sebagai indeks pada proses pengulangan.

Out adalah definisi untuk pin P1.1 (P1_B1) yang terhubung dengan pin input *standard servo motor*.

2. Proses selanjutnya adalah inisialisasi port serial yaitu menentukan mode (1 start bit, 8 bit data, 1 stop bit, dan tanpa bit *parity*) serta *baudrate* (9600 bps) untuk jalur komunikasi serial antara komputer dengan modul DT-51™ Low Cost Nano System.

Pada AN-08.zip telah disertakan program Servo.exe yang bekerja dalam sistem operasi Windows® untuk mengirimkan data sudut putaran *standard servo motor* dari komputer.



Gambar 4
Tampilan Program Servo.exe

3. Pada tahap berikutnya program akan menunggu masukan serial dari komputer yang nantinya akan diletakkan ke dalam variabel CW.
4. Kemudian pin Out diberi logika *high* dan proses pengulangan dimulai yaitu dari $I = 0$ sampai dengan $I < CW$ dengan nilai I ditambah 1 pada tiap kali pengulangan. Serta pada tiap kali pengulangan akan dijalankan prosedur delay 50 μ s, sehingga akan terjadi penundaan selama $CW \times 50 \mu$ s.
5. Setelah itu pin Out diberi logika *low* dan proses pengulangan dimulai yaitu dari $I = 0$ sampai dengan $I < (200 - CW)$ dengan nilai I ditambah 1 pada tiap kali pengulangan. Serta pada tiap kali pengulangan akan dijalankan prosedur delay 50 μ s, sehingga akan terjadi penundaan selama $(200 - CW) \times 50 \mu$ s.
6. Kembali ke langkah 3.

Listing program terdapat pada AN-08.zip.

DATA SHEET IC LM 358

LM158/LM258/LM358/LM2904 Low Power Dual Operational Amplifiers

General Description

The LM158 series consists of two independent, high gain, internally frequency compensated operational amplifiers which were designed specifically to operate from a single power supply over a wide range of voltages. Operation from split power supplies is also possible and the low power supply current drain is independent of the magnitude of the power supply voltage.

Application areas include transducer amplifiers, dc gain blocks and all the conventional op amp circuits which now can be more easily implemented in single power supply systems. For example, the LM158 series can be directly operated off of the standard +5V power supply voltage which is used in digital systems and will easily provide the required interface electronics without requiring the additional $\pm 15V$ power supplies.

Unique Characteristics

- In the linear mode the input common-mode voltage range includes ground and the output voltage can also swing to ground, even though operated from only a single power supply voltage.
- The unity gain cross frequency is temperature compensated.
- The input bias current is also temperature compensated.

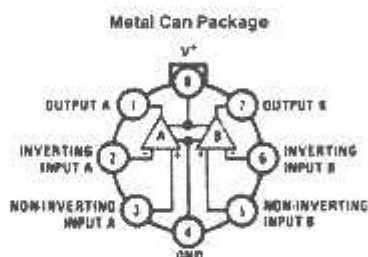
Advantages

- Two internally compensated op amps in a single package
- Eliminates need for dual supplies
- Allows directly sensing near GND and V_{OUT} also goes to GND
- Compatible with all forms of logic
- Power drain suitable for battery operation
- Pin-out same as LM1558/LM1458 dual operational amplifier

Features

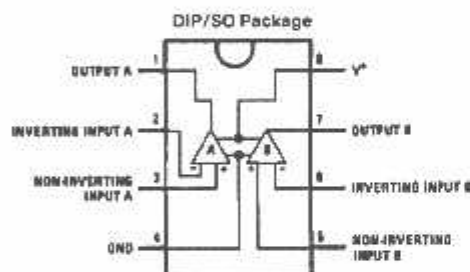
- Internally frequency compensated for unity gain
- Large dc voltage gain 100 dB
- Wide bandwidth (unity gain) 1 MHz (temperature compensated)
- Wide power supply range:
 - Single supply 3V to 32V
 - or dual supplies $\pm 1.5V$ to $\pm 16V$
- Very low supply current drain (500 μA)—essentially independent of supply voltage
- Low input offset voltage 2 mV
- Input common-mode voltage range includes ground
- Differential input voltage range equal to the power supply voltage
- Large output voltage swing 0V to $V^+ - 1.5V$

Connection Diagrams (Top Views)



TJ-117767-1

Order Number LM158AH, LM158AH/863*,
LM158H, LM158H/883*, LM258H or LM358H
See NS Package Number H06C



TJ-117767-2

Order Number LM158J, LM158J/883*,
LM158AJ or LM158AJ/883*
See NS Package Number J08A
Order Number LM358M, LM358AM or LM2904M
See NS Package Number M08A
Order Number LM358AN, LM358N or LM2904N
See NS Package Number N08E

*LM158 is available per BMD #5962-8771001
LM158A is available per BMD #5962-8771002

Absolute Maximum Ratings

If Military/Aerospace specified devices are required, please contact the National Semiconductor Sales Office/Distributors for availability and specifications. (Note 9)

	LM158/LM258/LM358 LM158A/LM258A/LM358A	LM2904	LM158/LM258/LM358 LM158A/LM258A/LM358A	LM2904
Supply Voltage, V^+	32V	26V	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C
Differential Input Voltage	32V	26V	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C
Input Voltage	-0.3V to +32V	-0.3V to +26V	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C
Power Dissipation (Note 1)	830 mW	830 mW	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C
Molded DIP	530 mW	530 mW	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C
Metal Can	530 mW	530 mW	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C
Small Outline Package (M)	530 mW	530 mW	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C
Output Short-Circuit to GND (Once Amplified) (Note 2)	Continuous	Continuous	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C
$V^+ \leq 15V$ and $T_A = 25^\circ C$	Continuous	Continuous	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C
Input Current ($V_{IN} < -0.3V$) (Note 3)	50 mA	50 mA	0°C to +70°C -25°C to +85°C -55°C to +125°C	260°C

Electrical Characteristics $V^+ = +5.0V$, unless otherwise stated

Parameter	Conditions	LM158A		LM358A		LM159/LM258		LM358		LM2904		Units
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min	
Input Offset Voltage	(Note 5), $T_A = 25^\circ C$	1	2	2	3	2	5	2	7	2	7	mV
Input Bias Current	$I_{IN(+)}$ or $I_{IN(-)}$, $T_A = 25^\circ C$, $V_{CM} = 0V$, (Note 6)	20	50	45	100	45	150	45	250	45	250	nA
Input Offset Current	$I_{IN(+)} - I_{IN(-)}$, $V_{CM} = 0V$, $T_A = 25^\circ C$	2	10	5	30	3	30	5	50	5	50	nA
Input Common-Mode Voltage Range	$V^+ - 30V$, (Note 7) (LM2904, $V^+ = 26V$), $T_A = 25^\circ C$	0	$V^+ - 1.5$	0	$V^+ - 1.5$	0	$V^+ - 1.5$	0	$V^+ - 1.5$	0	$V^+ - 1.5$	V
Supply Current	Over Full Temperature Range $R_L = \infty$ on All Op Amps $V^+ = 30V$ (LM2904 $V^+ = 26V$) $V^- = 5V$	1	2	1	2	1	2	1	2	1	2	mA
		0.5	1.2	0.5	1.2	0.5	1.2	0.5	1.2	0.5	1.2	mA

Electrical Characteristics (Continued) $V_{I-} = +5.0V$, Note 4, unless otherwise stated

Parameter	Conditions	LM158A		LM358A		LM158/LM258		LM358		LM2504		Units	
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min		Typ
Large Signal Voltage Gain	$V_{I+} = 15V$, $T_A = 25^\circ C$, $R_L \geq 2k\Omega$, (For $V_O = 1V$ to 11V)	50	100	25	100	50	100	25	100	25	100	V/mV	
Common-Mode Rejection Ratio	$T_A = 25^\circ C$, $V_{CM} = 0V$ to $V_{I+} - 1.5V$	70	85	65	85	70	85	65	85	50	70	dB	
Power Supply Rejection Ratio	$V_{I+} = 5V$ to 30V (LM2504, $V_{I+} = 5V$ to 26V), $T_A = 25^\circ C$	65	100	65	100	65	100	65	100	50	100	dB	
Amplifier-to-Amplifier Coupling	$f = 1$ kHz to 20 kHz, $T_A = 25^\circ C$ (Input Referred), (Note 8)	-	-	-120	-	-120	-	-120	-	-120	-	dB	
Output Current	Source	$V_{IN+} = 1V$, $V_{IN-} = 0V$, $V_{I+} = 15V$, $V_O = 2V$, $T_A = 25^\circ C$	20	40	20	40	20	40	20	40	20	40	mA
	Sink	$V_{IN+} = 1V$, $V_{IN-} = 0V$, $V_{I+} = 15V$, $T_A = 25^\circ C$, $V_O = 2V$	10	20	10	20	10	20	10	20	10	20	mA
Short Circuit to Ground	$V_{IN+} = 1V$, $V_{IN-} = 0V$, $T_A = 25^\circ C$, $V_O = 200$ mV, $V_{I+} = 15V$	12	50	12	50	12	50	12	50	12	50	μA	
Input Offset Voltage	$T_A = 25^\circ C$, (Note 2), $V_{I+} = 15V$	40	50	40	50	40	50	40	50	40	50	mV	
Input Offset Voltage Drift	(Note 5)	4	15	4	15	4	15	4	15	4	15	$\mu V/^\circ C$	
Input Offset Current	$R_S = 0.1\Omega$	7	20	7	20	7	20	7	20	7	20	nA	
Input Offset Current Drift	$I_{IK(+)} - I_{IK(-)}$	30	75	30	75	30	75	30	75	30	75	nA/°C	
Input Bias Current	$R_S = 0.1\Omega$	10	200	10	300	10	300	10	300	10	300	pA/°C	
Input Bias Current	$I_{IK(+)}$ or $I_{IK(-)}$	40	100	40	200	40	300	40	500	40	500	nA	

Electrical Characteristics (Continued) $V^+ = +5.0V$, Note 4, unless otherwise stated

Parameter	Conditions	LM158A		LM358A		LM158/LM258		LM358		LM2904		Units	
		Min	Typ	Max	Min	Typ	Max	Min	Typ	Max	Min		Typ
Input Common-Mode Voltage Range	$V^- = -30V$, (Note 7) (LM2904, $V^+ = -26V$)	0		$V^+ - 2$	0	$V^+ - 2$	0	$V^+ - 2$	0	$V^+ - 2$	0	$V^+ - 2$	V
Large Signal Voltage Gain	$V^+ = +15V$ ($V_O = 1V$ to $11V$) $R_L \geq 2k\Omega$	25			15		25		15		15		V/mV
Output Voltage Swing	V_{OH}				26		26		26		22		V
	V_{OL}				27		27		27		23		V
Output Current	Source				5		5		5		5		100
	Sink				10		10		10		10		20

Note 1: For operating at high temperatures, the LM258/LM358A, LM2904 must be derated based on a $1.25^\circ C$ maximum junction temperature and a thermal resistance of $120^\circ C/W$ which applies for the device soldered in a printed circuit board, operating in a still air ambient. The LM258/LM358A and LM158/LM258 can be derated based on a $+150^\circ C$ maximum junction temperature. The deration is the total of both amplifiers—use external resistors, where possible, to allow the amplifier to separate or to reduce the power which is dissipated in the integrated circuit.

Note 2: Short circuits from the output to V^- can cause excessive heating and eventual destruction. When considering short-circuits can exceed the power dissipation ratings and cause eventual destruction. Destructive dissipation can result from simultaneous shorts on all amplifiers. V^+ . At values of supply voltage in excess of $+15V$, continuous short-circuits can exceed the power dissipation ratings and cause eventual destruction.

Note 3: This input current will only exist when the voltage at any of the input leads is driven negative. It is due to the collector-base junction of the input PNP transistors becoming forward biased and thereby acting as input diode clamps. In addition to this diode action, there is also lateral NPN parasitic transistor action on the IC chip. The transistor action can cause the output voltage of the op amp to go to the V^- voltage level for a large overdrive for the time duration that an input is driven negative. This is not destructive and normal output states will re-establish when the input voltage, which was negative, again returns to a value greater than $-0.2V$ (at $25^\circ C$).

Note 4: These specifications are limited to $-50^\circ C \leq T_A \leq +125^\circ C$ for the LM158/LM158A, with the LM258/LM258A, all temperature specifications are limited to $-25^\circ C \leq T_A \leq +85^\circ C$, the LM358/LM358A temperature specifications are limited to $0^\circ C \leq T_A \leq +70^\circ C$, and the LM2904 specifications are limited to $-40^\circ C \leq T_A \leq +45^\circ C$.

Note 5: $V_O = 1.4V$, $R_S = 0\Omega$ with V^+ from $5V$ to $36V$, and over the full input common-mode range ($0V$ to $V^+ - 1.2V$) at $25^\circ C$. For LM2904, V^+ from $0V$ to $26V$.

Note 6: The direction of the input current is out of the IC due to the PNP input stage. This current is essentially constant, independent of the state of the output so no loading change exists on the input lines.

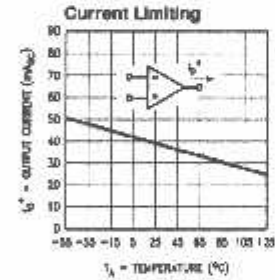
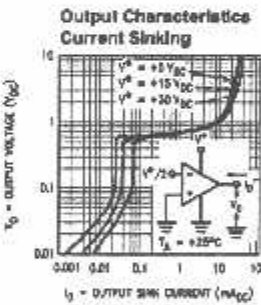
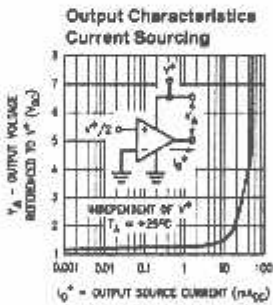
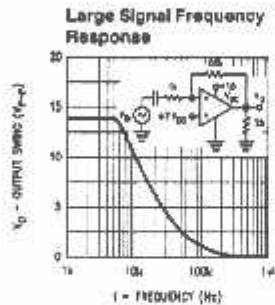
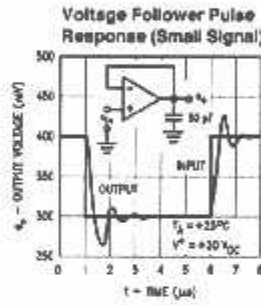
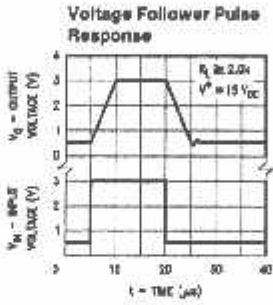
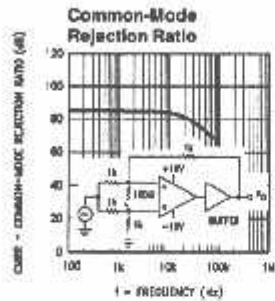
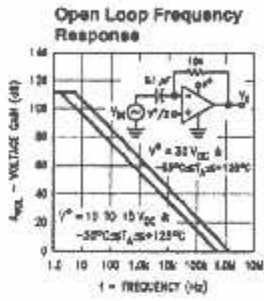
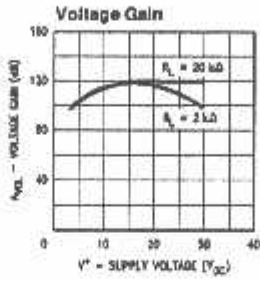
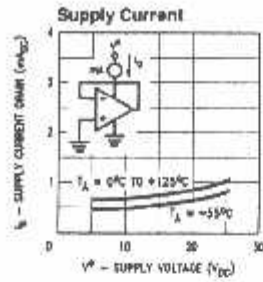
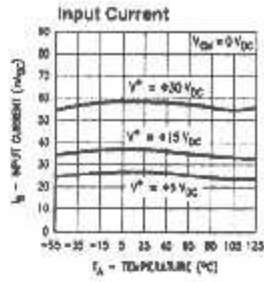
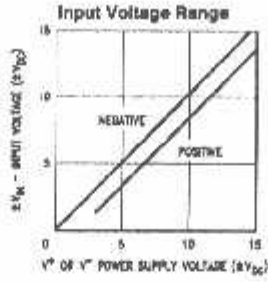
Note 7: The input common-mode voltage of either input signal voltage should not be allowed to go negative by more than $0.3V$ (at $25^\circ C$). The upper end of the common-mode voltage range is $V^+ - 1.5V$ (at $25^\circ C$), but either or both inputs can go to $+20V$ without damage ($+26V$ for LM2904), independent of the magnitude of V^- .

Note 8: Due to proximity of external components, insure that coupling is not originating via stray capacitance between these external parts. This typically can be detected as this type of capacitance increases at higher frequencies.

Note 9: Refer to RET158AX for LM158A military specifications and to RET158BAX for LM158 military specifications.

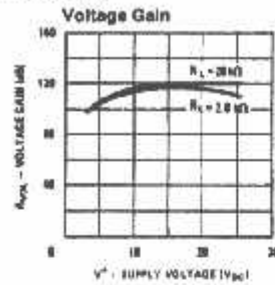
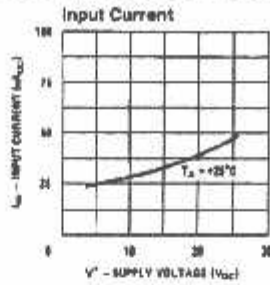
Note 10: Human body model, $1.5k\Omega$ in series with $100pF$.

Typical Performance Characteristics



TL147767-4

Typical Performance Characteristics (Continued) (LM2902 only)



TL767-5

Application Hints

The LM158 series are op amps which operate with only a single power supply voltage, have true-differential inputs, and remain in the linear mode with an input common-mode voltage of 0 V_{DC}. These amplifiers operate over a wide range of power supply voltage with little change in performance characteristics. At 25°C amplifier operation is possible down to a minimum supply voltage of 2.3 V_{DC}.

Precautions should be taken to insure that the power supply for the integrated circuit never becomes reversed in polarity or that the unit is not inadvertently installed backwards in a test socket as an unlimited current surge through the resulting forward diode within the IC could cause fusing of the internal conductors and result in a destroyed unit.

Large differential input voltages can be easily accommodated and, as input differential voltage protection diodes are not needed, no large input currents result from large differential input voltages. The differential input voltage may be larger than V⁺ without damaging the device. Protection should be provided to prevent the input voltages from going negative more than -0.3 V_{DC} (at 25°C). An input clamp diode with a resistor to the IC input terminal can be used.

To reduce the power supply current drain, the amplifiers have a class A output stage for small signal levels which converts to class B in a large signal mode. This allows the amplifiers to both source and sink large output currents. Therefore both NPN and PNP external current boost transistors can be used to extend the power capability of the basic amplifiers. The output voltage needs to raise approximately 1 diode drop above ground to bias the on-chip vertical PNP transistor for output current sinking applications.

For ac applications, where the load is capacitively coupled to the output of the amplifier, a resistor should be used, from the output of the amplifier to ground to increase the class A bias current and prevent crossover distortion. Where the load is directly coupled, as in dc applications, there is no crossover distortion.

Capacitive loads which are applied directly to the output of the amplifier reduce the loop stability margin. Values of 50 pF can be accommodated using the worst-case non-inverting unity gain connection. Large closed loop gains or resistive isolation should be used if larger load capacitance must be driven by the amplifier.

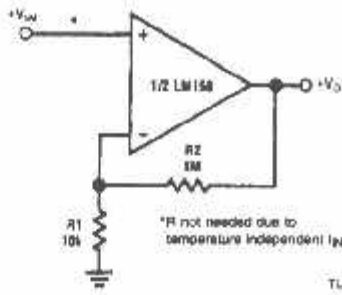
The bias network of the LM158 establishes a drain current which is independent of the magnitude of the power supply voltage over the range of 3 V_{DC} to 30 V_{DC}.

Output short circuits either to ground or to the positive power supply should be of short time duration. Units can be destroyed, not as a result of the short circuit current causing metal fusing, but rather due to the large increase in IC chip dissipation which will cause eventual failure due to excessive junction temperatures. Putting direct short-circuits on more than one amplifier at a time will increase the total IC power dissipation to destructive levels, if not properly protected with external dissipation limiting resistors in series with the output leads of the amplifiers. The larger value of output source current which is available at 25°C provides a larger output current capability at elevated temperatures (see typical performance characteristics) than a standard IC op amp.

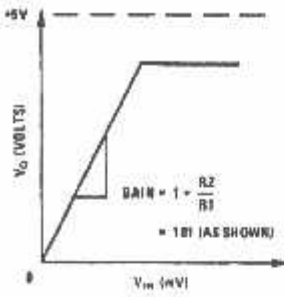
The circuits presented in the section on typical applications emphasize operation on only a single power supply voltage. If complementary power supplies are available, all of the standard op amp circuits can be used. In general, introducing a pseudo-ground (a bias voltage reference of V⁺/2) will allow operation above and below this value in single power supply systems. Many application circuits are shown which take advantage of the wide input common-mode voltage range which includes ground. In most cases, input biasing is not required and input voltages which range to ground can easily be accommodated.

Typical Single-Supply Applications ($V^+ = 5.0 V_{DC}$)

Non-Inverting DC Gain (0V Input = 0V Output)

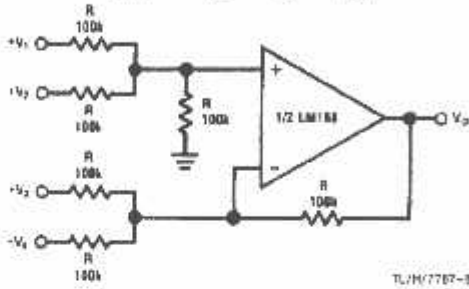


TL/H/7787-6



TL/H/7787-7

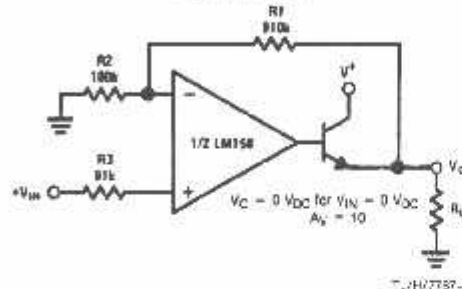
DC Summing Amplifier
($V_{INs} > 0 V_{DC}$ and $V_O > 0 V_{DC}$)



TL/H/7787-8

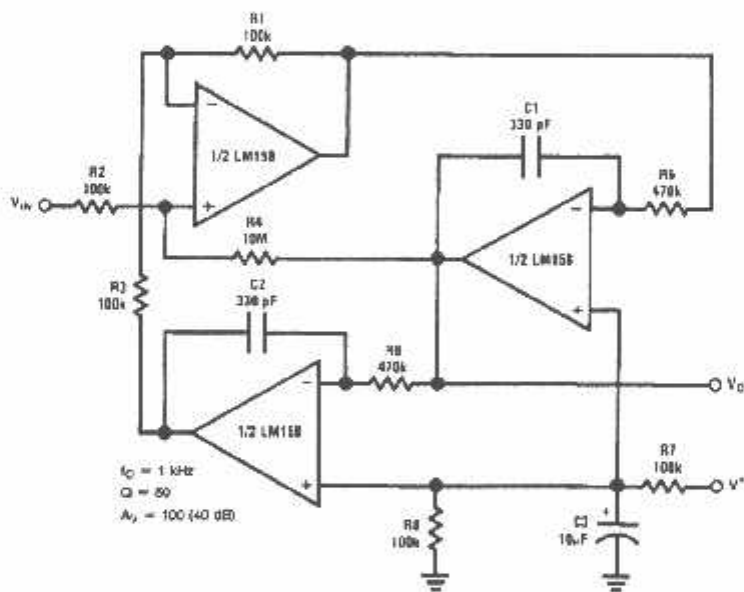
Where: $V_O = V_1 + V_2 + V_3 + V_4$
 $(V_1 - V_2) \geq (V_3 + V_4)$ to keep $V_O > 0 V_{DC}$

Power Amplifier



TL/H/7787-9

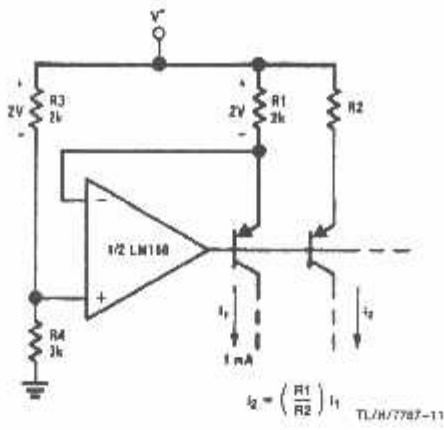
"BI-QUAD" RC Active Bandpass Filter



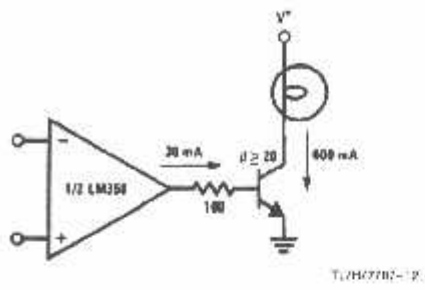
TL/H/7787-10

Typical Single-Supply Applications ($V^+ = 5.0 V_{DD}$) (Continued)

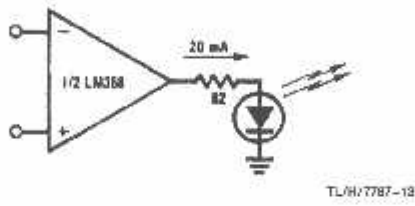
Fixed Current Sources



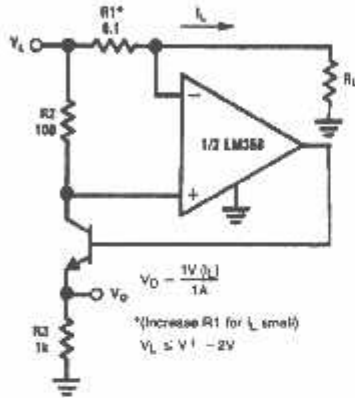
Lamp Driver



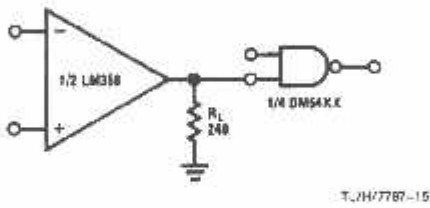
LED Driver



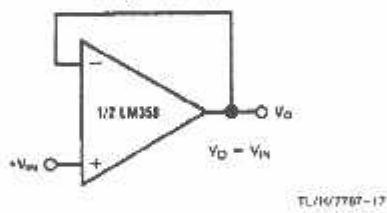
Current Monitor



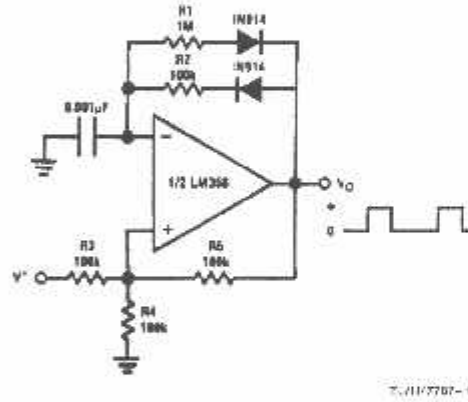
Driving TTL



Voltage Follower

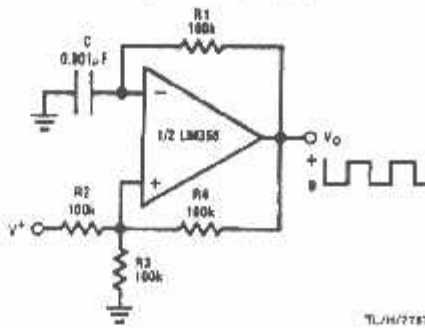


Pulse Generator



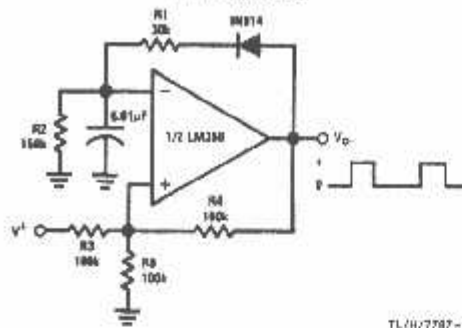
Typical Single-Supply Applications (V₊ = 5.0 V_{DC}) (Continued)

Squarewave Oscillator



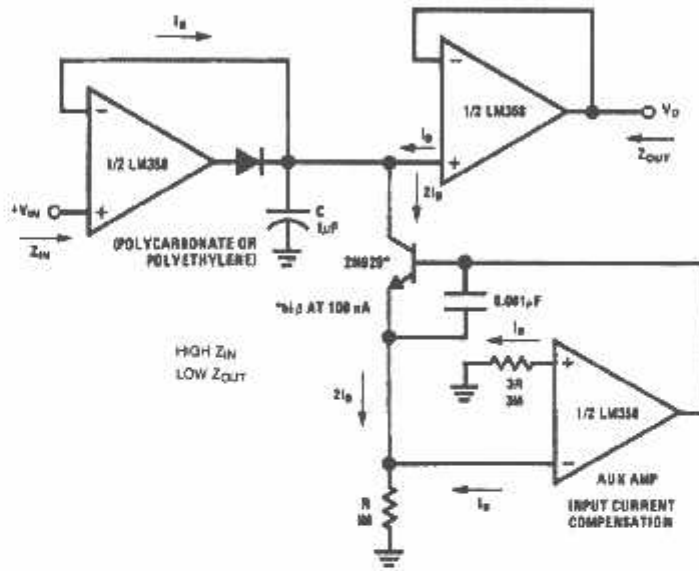
TL/H/7787-18

Pulse Generator



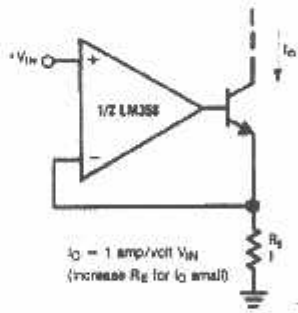
TL/H/7787-19

Low Drift Peak Detector



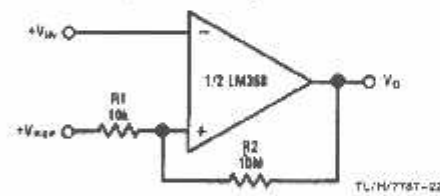
TL/H/7787-20

High Compliance Current Sink



TL/H/7787-21

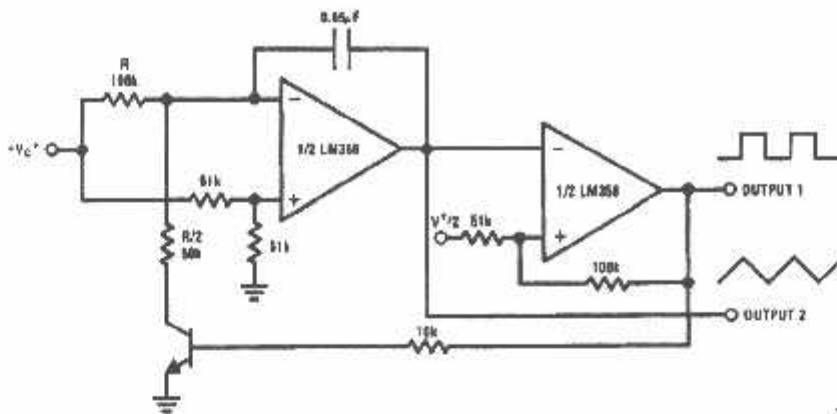
Comparator with Hysteresis



TL/H/7787-22

Typical Single-Supply Applications ($V^+ = 5.0 V_{DC}$) (Continued)

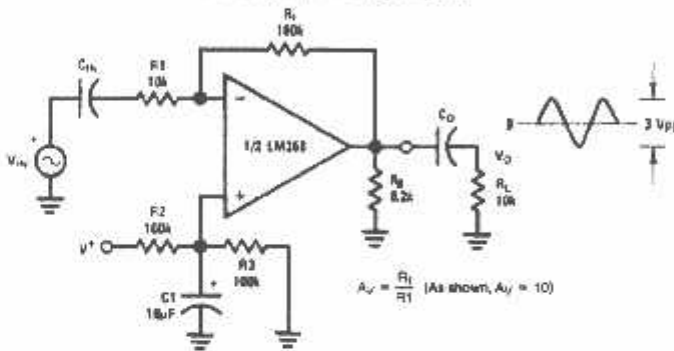
Voltage Controlled Oscillator (VCO)



TLH/7787-23

*WIDE CONTROL VOLTAGE RANGE: $0 V_{DD} < V_C < 2 (V^+ - 1.5V_{DD})$

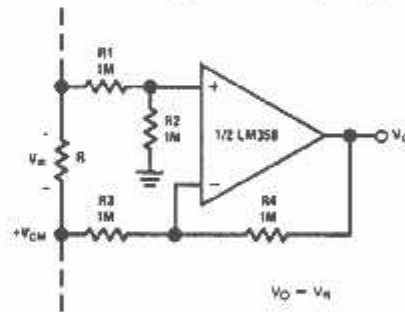
AC Coupled Inverting Amplifier



$$A_v = \frac{R_f}{R_1} \text{ (As shown, } A_v = 10 \text{)}$$

TLH/7787-24

Ground Referencing a Differential Input Signal

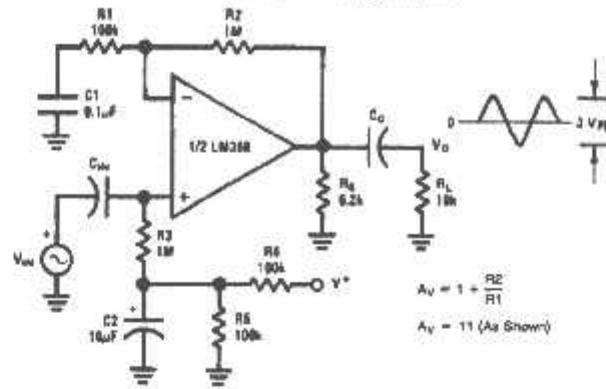


$$V_O = V_{IN}$$

TLH/7787-25

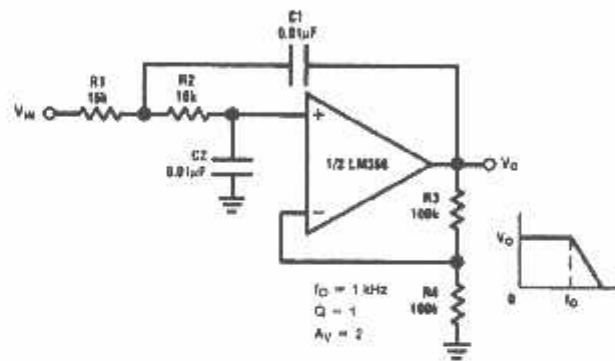
Typical Single-Supply Applications ($V^- = 5.0 V_{DC}$) (Continued)

AC Coupled Non-Inverting Amplifier



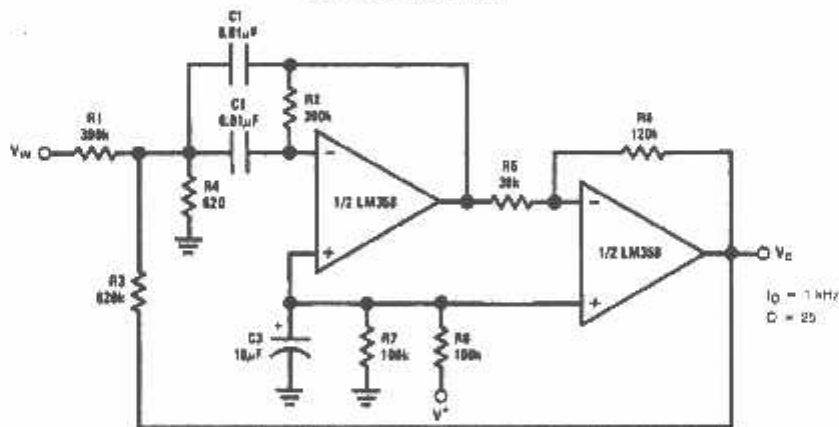
TL/H/7767-26

DC Coupled Low-Pass RC Active Filter



TL/H/7767-27

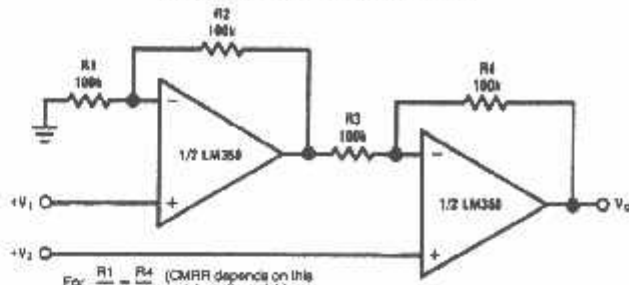
Bandpass Active Filter



TL/H/7767-28

Typical Single-Supply Applications ($V^+ = 5.0 \text{ V}_{\text{DC}}$) (Continued)

High Input Z, DC Differential Amplifier



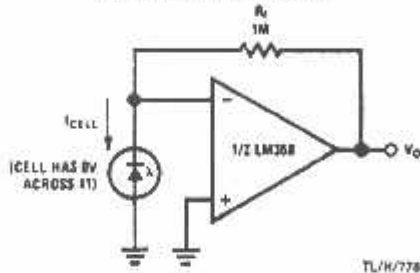
For $\frac{R1}{R2} = \frac{R4}{R3}$ (CMRR depends on this resistor ratio match)

$$V_O = 1 + \frac{R4}{R3} (V_2 - V_1)$$

As Shown: $V_O = 2 (V_2 - V_1)$

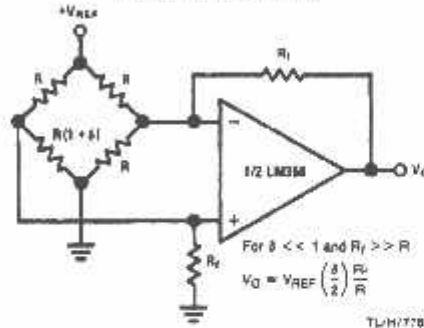
TL/H/7787-20

Photo Voltaic-Cell Amplifier



TL/H/7787-30

Bridge Current Amplifier

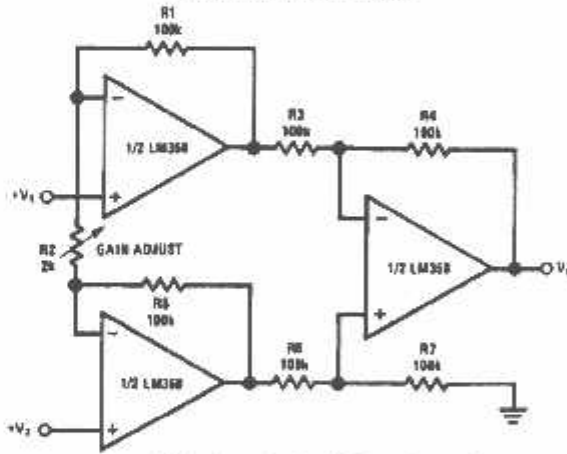


For $\delta \ll 1$ and $R_f \gg R$

$$V_O = V_{\text{REF}} \left(\frac{\delta}{2} \right) \frac{R_f}{R}$$

TL/H/7787-33

High Input Z Adjustable-Gain DC Instrumentation Amplifier



If $R1 = R5$ & $R3 = R4 = R6 = R7$ (CMRR depends on match)

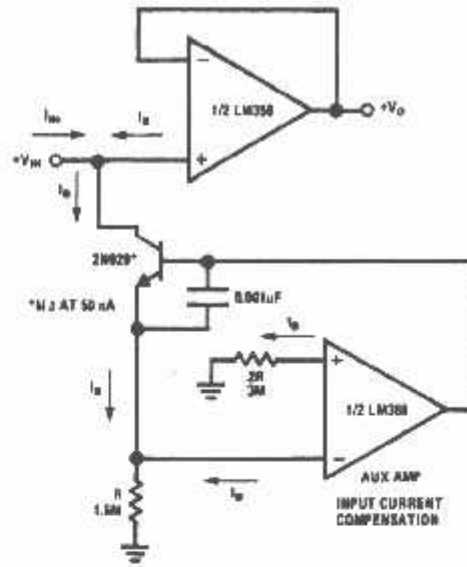
$$V_O = 1 + \frac{2R1}{R2} (V_2 - V_1)$$

As shown $V_O = 101 (V_2 - V_1)$

TL/H/7787-01

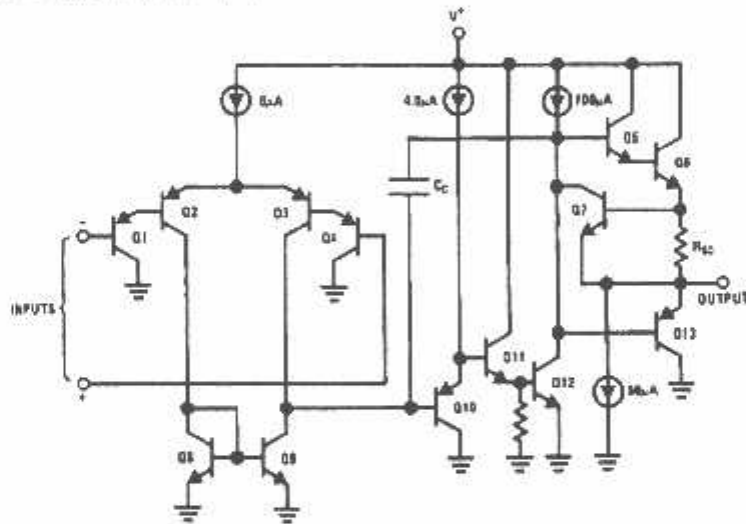
Typical Single-Supply Applications ($V^+ = 5.0 V_{DD}$) (Continued)

Using Symmetrical Amplifiers to Reduce Input Current (General Concept)



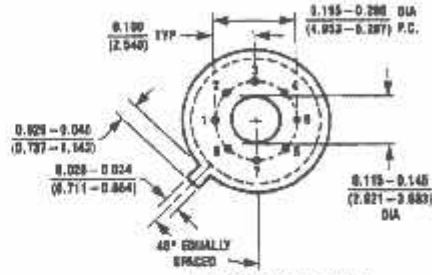
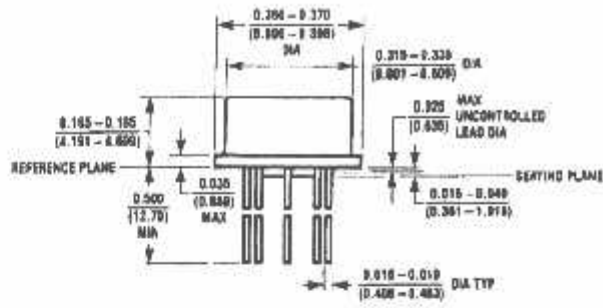
TL11/7707-3E

Schematic Diagram (Each Amplifier)



TL11/7707-3

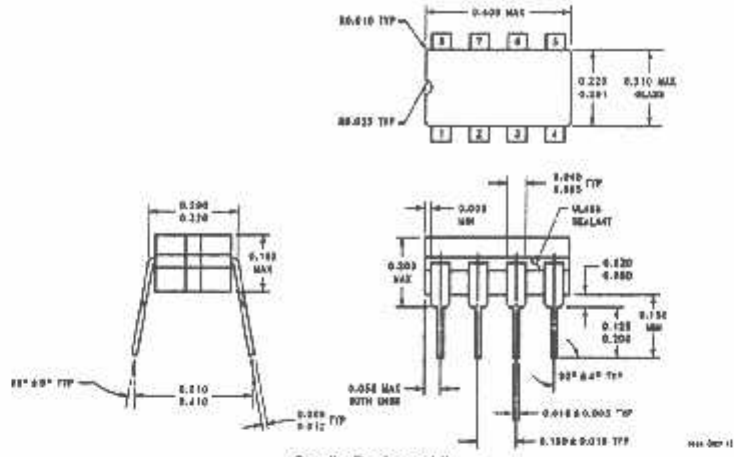
Physical Dimensions inches (millimeters)



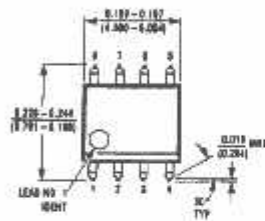
H06C REV B

Metal Can Package (H)
Order Number LM158AH, LM158AH/883, LM158H,
LM158H/883, LM258H or LM358H
NS Package Number H06C

Physical Dimensions inches (millimeters) (Continued)

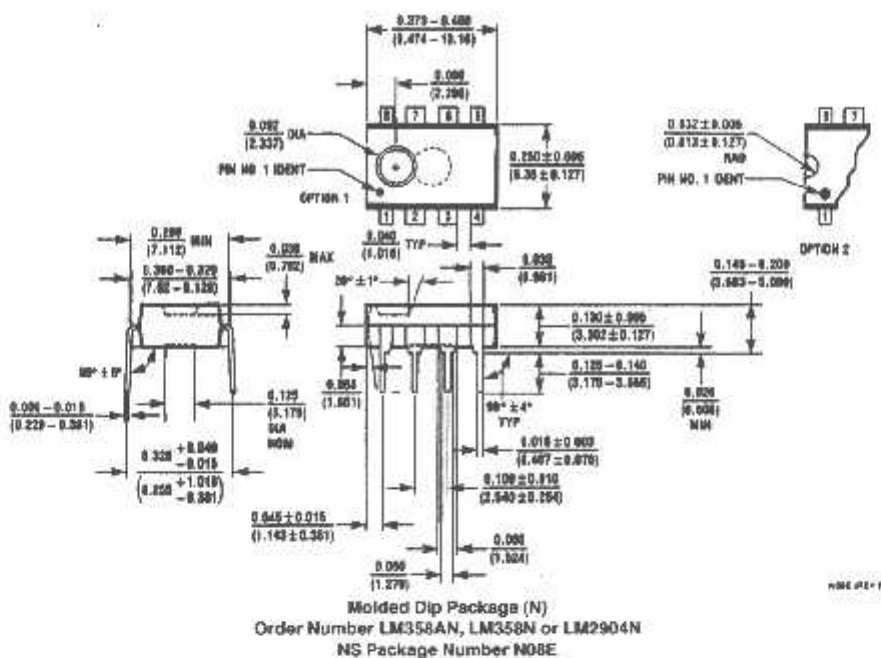


Cerdip Package (J)
 Order Number LM158J, LM158J/883, LM158AJ or LM158AJ/883
 NS Package Number J08A



S.O. Package (M)
 Order Number LM358M, LM358AM or LM2904M
 NS Package Number M08A

Physical Dimensions inches (millimeters) (Continued)



LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.
2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.

 <p>National Semiconductor Corporation 111 West Bendin Road Arlington, TX 76011 Tel: (817) 277-9679 Fax: (817) 277-7018</p>	<p>National Semiconductor Europe Fax: (+49) 0-180-630 35 68 Email: crjago@nsmv2.nsc.com Deutsch: Tel: (+49) 0-180-630 35 66 English: Tel: (+49) 0-180-632 78 32 Français: Tel: (+49) 0-180-632 35 58 Italiano: Tel: (+49) 0-180-634 18 80</p>	<p>National Semiconductor Hong Kong Ltd. 12th Floor, Straits Block, Ocean Centre, 5 Canton Rd. Tsimshatsui, Kowloon Hong Kong Tel: (852) 2737-1000 Fax: (852) 2736-8999</p>	<p>National Semiconductor Japan Ltd. Tel: 01-043-200-2508 Fax: 01-043-250-2408</p>
--	---	---	--

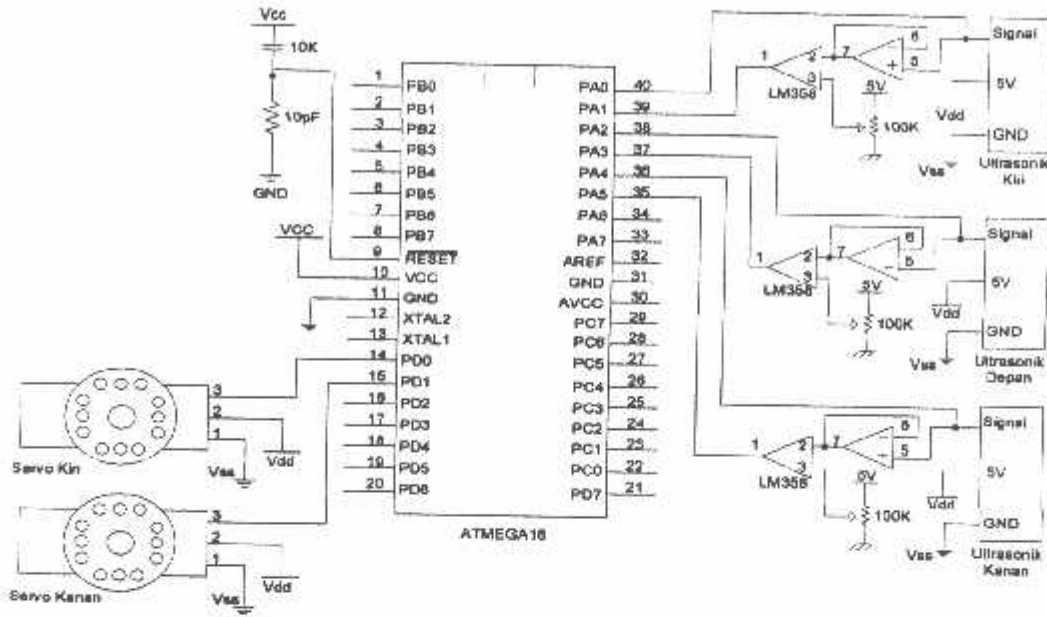
National does not assume any responsibility for use of any circuitry described, no circuit patent licenses are implied and National reserves the right at any time without notice to change product design and specifications.

This datasheet has been download from:

www.datasheetcatalog.com

Datasheets for electronics components.

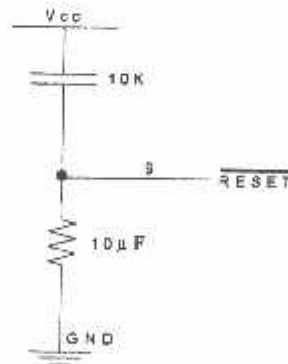
3.1.2 Sistem Mikrokontroler ATmega16



Gambar 3.6 Perancangan Mikrokontroler ATmega16

- **Reset**

Untuk mereset mikrokontroler ATmega16, maka pin RST diberi logika tinggi selama sekurangnya dua siklus mesin (24 periode osilator). Untuk membangkitkan sinyal reset kapasitor dihubungkan dengan Vcc dan sebuah resistor yang dihubungkan ke ground. Rangkaian reset ditunjukkan dalam gambar 3.7 sebagai berikut :



Gambar 3.7 Perancangan Rangkaian Reset

Karena kristal yang digunakan mempunyai frekuensi sebesar 12 MHz, maka satu periode membutuhkan waktu sebesar :

$$\tau = \frac{1}{f_{XTAL}} = \frac{1}{12 \text{ MHz}} \text{ S} = 8,333 \times 10^{-8} \text{ s}$$

Sehingga waktu minimal logika tinggi yang dibutuhkan untuk mereset mikrokontroller adalah :

$$\begin{aligned} \text{Reset (min)} &= \tau \times \text{periode yang dibutuhkan} \\ &= 8,333 \times 10^{-8} \times 24 = 1,999 \mu\text{s} \end{aligned}$$

Jadi mikrokontroller membutuhkan waktu minimal 1,999 μs untuk mereset. Waktu minimal inilah yang dijadikan pedoman untuk menentukan nilai R dan C. Dari persamaan konstanta waktu $\tau = R \times C$ dan jika nilai R ditentukan sebesar 10 k Ω , maka nilai C adalah :

$$\begin{aligned} C &= \frac{\tau}{R} \\ &= \frac{1,999 \times 10^{-6}}{10 \times 10^3} \\ &= 199,9 \times 10^{-12} \text{ F} \end{aligned}$$

Kapasitor minimal yang dibutuhkan adalah 199,9 pF. Dengan menggunakan kapasitor sebesar 10 μF , maka akan menjamin waktu reset di atas nilai minimal waktu yang dibutuhkan untuk mereset mikrokontroller.

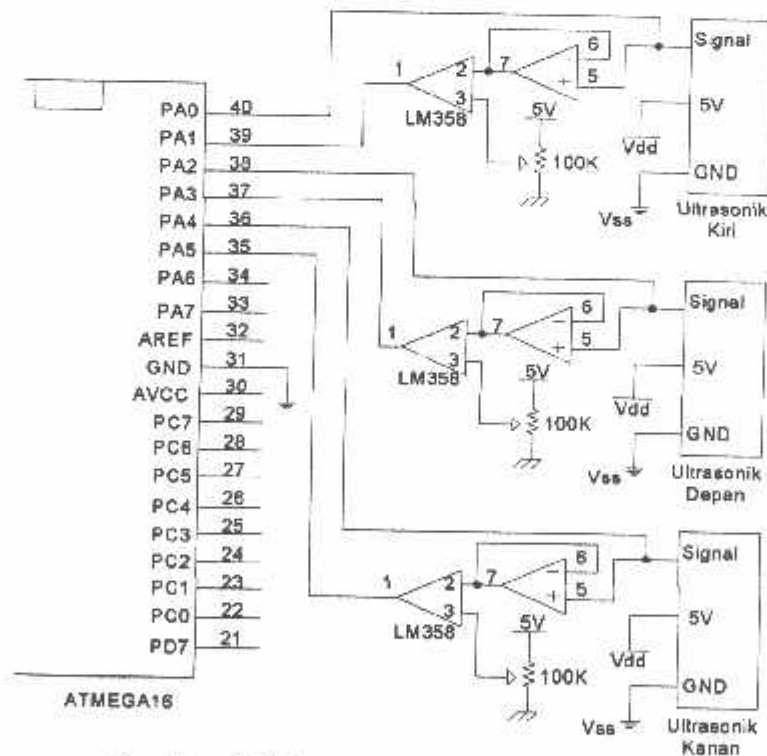
3.1.3 Perancangan dan Pembuatan Sensor

Sistem sensor digunakan sebagai masukan untuk piranti pengontrol robot, sistem sensor tersebut adalah sensor ultrasonik.

3.1.3.1 Sensor Ultrasonik

Sistem sensor ultrasonik digunakan sebagai masukan dari proses pengontrolan robot. Sensor ultrasonik ini memiliki rangkaian yang sudah jadi yaitu berasal dari modul parallax. Pada robot ini menggunakan tiga sensor ultrasonik yang dipasang pada bagian depan, samping kiri dan samping kanan robot yang digunakan untuk mengamati halangan.

Sensor ultrasonik ini mengirim data ke mikrokontroler secara terus-menerus sehingga semua sensor ultrasonik akan aktif dalam pengiriman data. Untuk itu proses *scanning* sensor ultrasonik terhadap halangan dilakukan secara bergantian agar data yang didapat lebih mudah. *Output* dari modul ini berupa data pulsa sehingga data jarak sama dengan *duty cycle* dari sinyal *output*. Semakin jauh halangan maka semakin besar *duty cycle*.

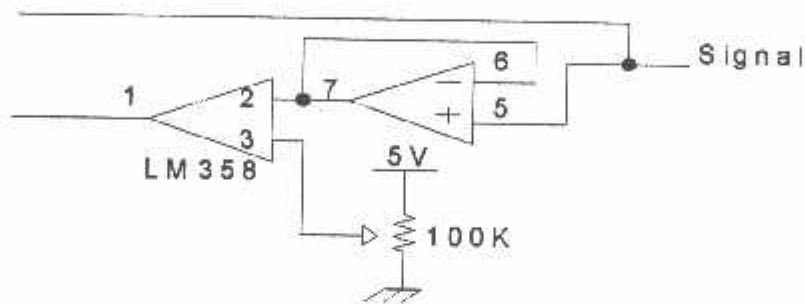


Gambar 3.8 Perancangan Sensor Ultrasonik

Untuk mengaktifkan sensor maka modul diberi triger pulsa maka sensor akan mengeluarkan sinyal pwm dan *duty cycle* tersebut sebagai jarak halangan dengan sensor.

3.1.3.2. Perancangan Rangkaian Komparator

Karena kembalian dari sensor ultrasonik kecil yaitu 0,7V sedangkan sinyal ultrasonik adalah 5V, maka pada rangkaian sensor ultrasonik diberikan rangkaian komparator sebagai berikut.



Gambar 3.9 Perancangan Rangkaian Komparator

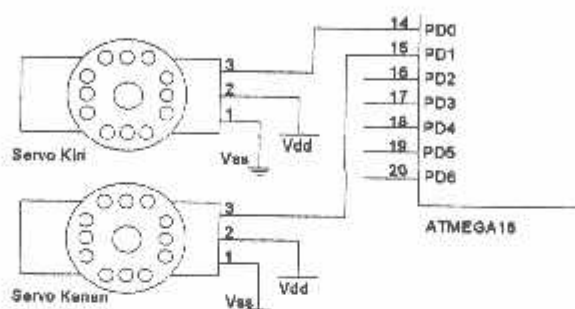
3.1.4. Motor Servo

Motor servo adalah motor yang mampu bekerja dua arah (CW dan CCW) dimana arah dan sudut pergerakan rotornya dapat dikendalikan hanya dengan memberikan pengaturan *duty cycle* sinyal pulsa pada bagian pin kontrolnya. Motor Servo merupakan sebuah motor DC yang memiliki rangkaian kontrol elektronik dan *internal gear* untuk mengendalikan pergerakan dan sudut sudutnya.

Motor servo adalah motor yang berputar lambat, dimana biasanya ditunjukkan oleh *rate* putarannya yang lambat, namun demikian memiliki torsi yang kuat karena *internal gear*nya.

Lebih dalam dapat digambarkan bahwa sebuah motor servo memiliki :

- jalur kabel : power, ground, dan control
- Sinyal kontrol mengendalikan posisi
- Operasional dari servo motor dikendalikan oleh sebuah pulsa selebar ± 20 ms, dimana lebar pulsa antara 0.5 ms dan 2 ms menyatakan akhir dari range sudut maksimum.
- Konstruksi didalamnya meliputi internal gear, potensiometer, dan *feedback* kontrol.



Gambar 3.10 Perancangan Motor Servo

3.2. Perancangan Perangkat Lunak

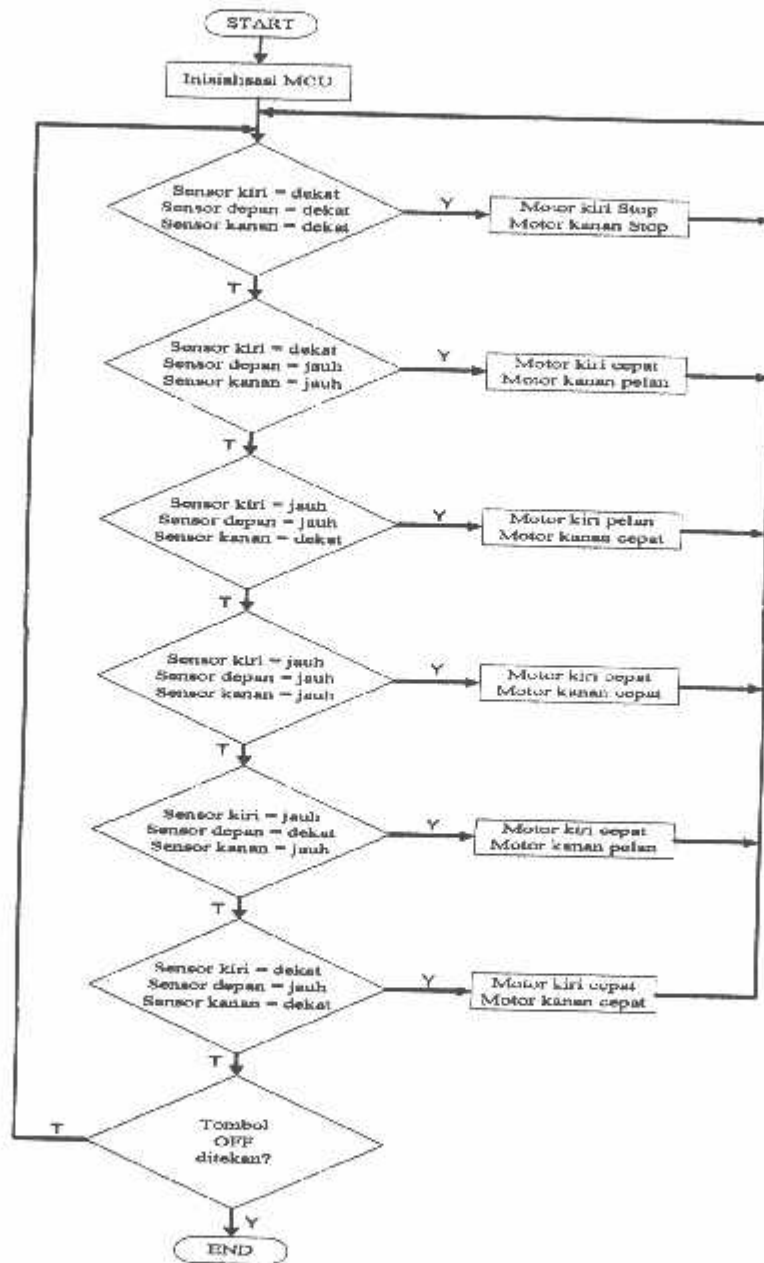
Perancangan perangkat lunak sangat diperlukan oleh *programmer* dalam mempermudah menentukan langkah-langkah atau alur dari program. Selain mempermudah langkah-langkah pemrograman, diagram alir juga difungsikan supaya program sesuai dan sinkron dengan kerja perangkat keras (*hardware*), sehingga sesuai dengan apa yang direncanakan. Mikrokontroler ATmega16 menggunakan bahasa C.

3.2.1 Desain Diagram Alir (*Flowchart*)

Mendesain *flowchart* sebelum membuat suatu program sangatlah penting, sehingga program yang akan dirancang sedemikian rupa, dapat memaksimalkan bahasa C. Selain itu juga memudahkan dalam penganalisaan bila terjadi kesalahan

saat membuat program, dengan demikian akan menghasilkan suatu program yang waktu mengeksekusi bahasa atau menjalankan program sangat cepat. Agar perancangan perangkat lunak ini dalam pemahamannya lebih mudah, maka ditunjukkan seperti dalam gambar di bawah ini :

3.2.1.1 Diagram Alir (*Flowchart*) Utama



Gambar 3.11 Diagram Alir (*Flowchart*) Utama



BAB IV

PENGUJIAN DAN ANALISA

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

Pengujian yang dilakukan pada bab ini antara lain :

1. Pengujian minimum sistem mikrokontroler ATmega16
2. Pengujian sensor ultrasonik
3. Pengujian motor servo
4. Pengujian sistem keseluruhan

4.1 Pengujian Minimum Sistem ATmega16

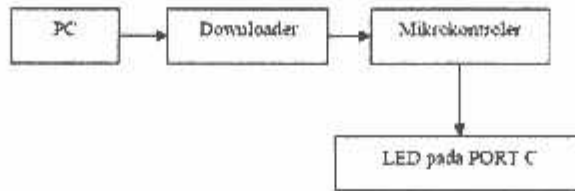
4.1.1. Tujuan :

- Untuk mengetahui apakah mikrokontroler dapat bekerja dengan baik

4.1.2. Peralatan:

- Minimum sistem mikrokontroler ATmega16
- Rangkaian led pada PORT C
- Catu daya 5 volt
- Seperangkat downloader ATmega16 dan program code vision AVR

4.1.3. Tahapan pengujian mikrokontroler ATmega 16 sebagai berikut :



Gambar 4.1 Diagram Blok Pengujian Rangkaian Mikrokontroler

4.1.4. Persiapan :

- Memasang rangkaian seperti yang ditunjukkan pada gambar
- Mengetik *Source* program untuk pengujian
- Mendownload program
- Menjalankan program

4.1.5. Hasil dan analisa :



Setelah program didownload maka akan ada tampilan led yang ada pada port C yang bergeser perbit kekanan. Dari hasil tersebut dapat dianalisa bahwa minimum sistem mikrokontroler ATmega16 dapat berfungsi dengan baik dan dapat diprogram untuk aplikasi pergerakan robot selanjutnya.

4.2 Pengujian Rangkaian Sensor Ultrasonik

4.2.1 Sinyal Sensor Ultrasonik

4.2.1.1. Tujuan :

- Untuk mengetahui sinyal keluaran dari sensor Ultrasonik

4.2.1.2. Peralatan yang dibutuhkan :

- Tranduser Ultrasonik

4.2.1.2. Peralatan yang dibutuhkan :

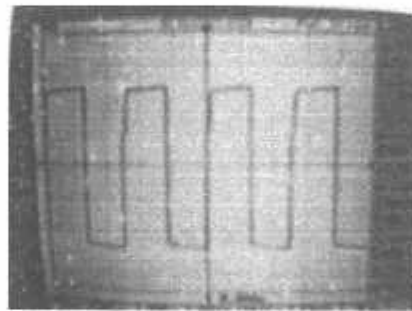
- Transduser Ultrasonik
- Oscilloscope

4.2.1.3. Prosedur pengujian :

- Rangkaian diberi catu daya 5 volt
- Letakan ujung penyidik oscilloscope pada rangkaian sensor ultrasonik

4.2.1.4. Hasil pengujian :

- Frekuensi yang dipancarkan : 40 KHz



Gambar 4.2 Sinyal Keluaran Dari Rangkaian Sensor Ultrasonik Dengan Frekuensi 40 KHz

Gambar diatas menunjukkan gambar sinyal keluaran dari rangkaian sensor ultrasonik yang memancarkan sinyal frekuensi sebesar 40 KHz.

4.2.2 Pengujian Pemancar Ultrasonik

4.2.2.1. Tujuan :

- Untuk mengetahui sinyal keluaran dari pemancar Ultrasonik

4.2.2.2. Peralatan yang dibutuhkan :

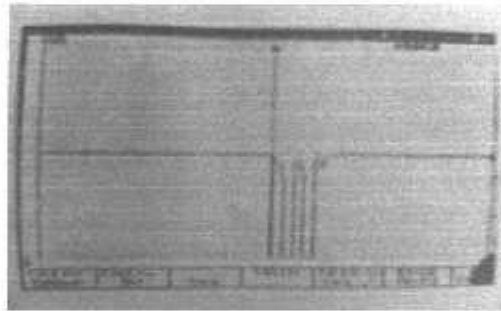
- Transduser Ultrasonik
- Oscilloscope

4.2.2.3. Prosedur pengujian :

- Rangkaian diberi catu daya 5 volt
- Letakan ujung penyidik oscilloscope pada Pemancar Ultrasonik

4.2.2.4. Hasil pengujian :

- Frekuensi yang dipancarkan : 40 KHz



Gambar 4.3 Sinyal Keluaran Dari Pemancar Dengan Frekuensi 40 KHz

Gambar diatas menunjukkan gambar sinyal pemancar yang memancarkan sinyal frekuensi sebesar 40 KHz.

4.2.3 Pengujian Penerima Ultrasonik

4.2.3.1. Tujuan :

- Untuk mengetahui sinyal keluaran dari Penerima Ultrasonik

4.2.3.2. Peralatan yang dibutuhkan :

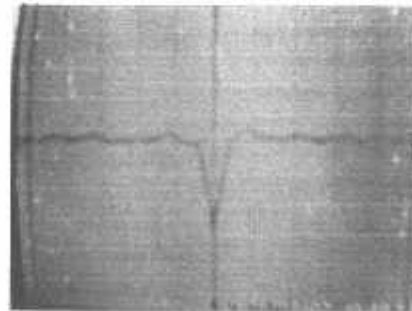
- Transduser Ultrasonik

- Oscilloscope

4.2.3.3. Prosedur pengujian :

- Rangkaian diberi catu daya 5 volt
- Letakan ujung penyidik oscilloscope pada Penerima Ultrasonik

4.2.3.4. Hasil pengujian tampak pada gambar



Gambar 4.4 Sinyal Keluaran Dari Penerima Ultrasonik.

Karena kembalian dari sensor ultrasonik kecil yaitu 0,7V sedangkan sinyal ultrasonik adalah 5V, maka pada rangkaian sensor ultrasonik diberikan rangkaian komparator.

4.3. Pengujian Rangkaian Komparator

4.3.1. Tujuan :

- Untuk mengetahui sinyal keluaran dari Komparator

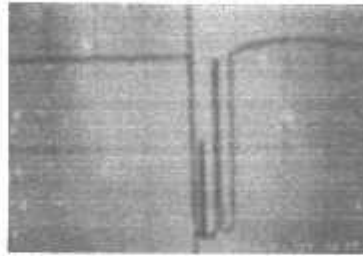
4.3.2. Peralatan yang dibutuhkan :

- Transduser Ultrasonik
- Oscilloscope

4.3.3. Prosedur pengujian :

- Rangkaian diberi catu daya 5 volt
- Letakan ujung penyidik oscilloscope pada rangkaian output dari komparator

4.3.4. Hasil pengujian tampak pada gambar



Gambar 4.5 Sinyal Keluaran Dari Komparator

Gambar diatas menunjukkan sinyal komparator dimana pada saat sinyal menunjukkan low maka mikrokontroller mendeteksi bahwa sinyal tersebut low. Ini merupakan perbandingan antara perhitungan dengan pengukuran pada sensor penerima ultrasonik.

Tabel 4.1 Perbandingan Antara Perhitungan Dengan Pengukuran Pada Sensor Penerima Ultrasonik

No	Jarak	Vin Penerima	Pengukuran	Perhitungan	Kondisi Sensor (Tembok)
			Vout Komparator	Vout Komparator	
1	10 cm	0,025 V	3,51 V	3,6 V	Ada
2	20 cm	0,010 V	3,86 V	3,9 V	Tidak
3	30 cm	0,008 V	4,24 V	4,3 V	Tidak

- Pada jarak 10 cm

$$Error = \left| \frac{3,6 - 3,51}{3,6} \right| \times 100\%$$

$$Error = 2,5 \%$$

- Pada jarak 20 cm

$$Error = \left| \frac{3,9 - 3,86}{3,9} \right| \times 100\%$$

$$Error = 1 \%$$

- Pada jarak 30 cm

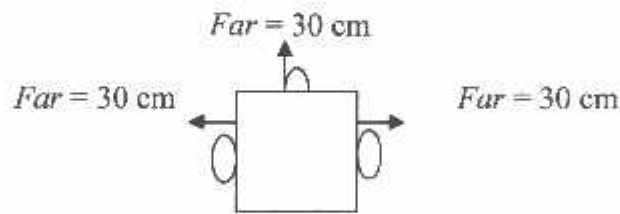
$$Error = \left| \frac{4,3 - 4,24}{4,3} \right| \times 100\%$$

$$Error = 1,4 \%$$

4.4 Pengujian Motor Servo

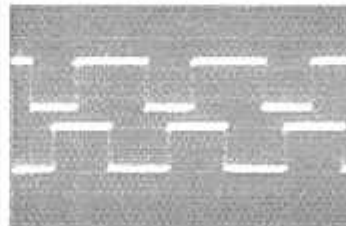
4.4.1. Variasi Pertama :

Dalam variasi ini diberikan kondisi untuk ketiga sensor adalah *far* dengan nilai jarak 30 cm.



Gambar 4.6 Kondisi Variasi Robot Pertama

Motor Kiri :



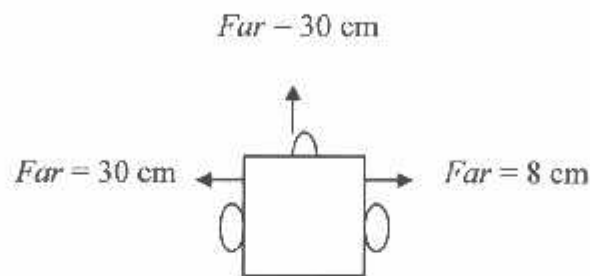
Motor Kanan :

Gambar 4.7 Foto PWM Variasi Pertama

Dengan variasi input seperti ini, maka robot akan bergerak lurus, dengan keadaan motor servo berputar untuk motor kiri dan kanan cepat.

4.4.2. Variasi Kedua :

Pada variasi kedua ini diberikan kondisi sebagai berikut sensor kiri dan sensor depan diberikan kondisi daerah *far* dengan nilai 30 cm, sedangkan sensor kanan diberikan kondisi daerah *near* dengan nilai 8 cm.



Gambar 4.8 Kondisi Variasi Robot Kedua

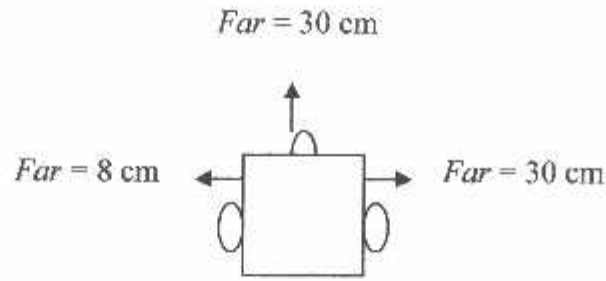


Gambar 4.9 FotoPWM Variasi Kedua

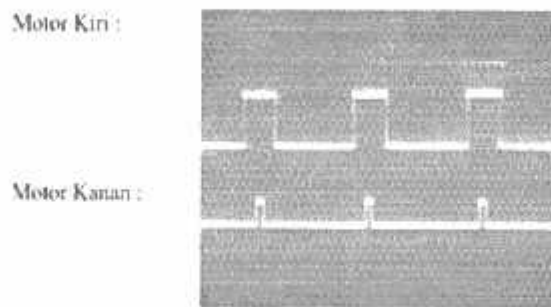
Dengan melihat analisa diatas, maka robot bergerak ke kiri, dengan keadaan motor servo berputar untuk motor kiri pelan dan kanan cepat.

4.4.3. Variasi Ketiga :

Pada variasi ketiga ini diberikan kondisi sebagai berikut sensor kanan dan sensor depan diberikan kondisi daerah *far* dengan nilai 30 cm, sedangkan sensor kiri diberikan kondisi daerah *near* dengan nilai 8 cm.



Gambar 4.10 Kondisi Variasi Robot Ketiga

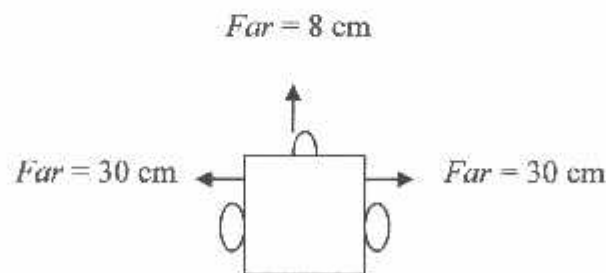


Gambar 4.11 FotoPWM Variasi Ketiga

Dengan melihat analisa diatas, maka robot bergerak ke kanan, dengan keadaan motor servo berputar untuk motor kiri cepat dan kanan pelan.

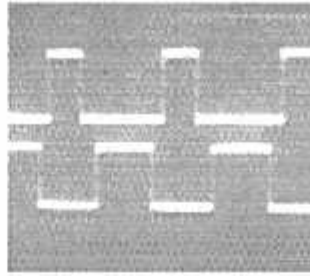
4.4.4. Variasi Keempat :

Pada variasi keempat ini diberikan kondisi sebagai berikut sensor kanan dan sensor kiri diberikan kondisi daerah *far* dengan nilai 30 cm, sedangkan sensor depan diberikan kondisi daerah *near* dengan nilai 8 cm.



Gambar 4.12 Kondisi Variasi Robot Keempat

Motor Kanan



Motor Kiri :

Gambar 4.13 FotoPWM Variasi Keempat

Dengan melihat analisa diatas, maka robot bergerak ke kiri, dengan keadaan motor servo berputar untuk motor kiri pelan dan kanan pelan.



BAB V

PENUTUP

5.1. Kesimpulan

Dari percobaan yang telah dilakukan, maka dapat diambil kesimpulan :

- Dari hasil pengujian jarak di peroleh error diantaranya :
 - Pada jarak 10 cm dengan *error* 2,5%,
 - Pada jarak 20 cm dengan *error* 1%
 - Pada jarak 30 cm dengan *error* 1,4%.
- Sistem logika *fuzzy* dapat diimplementasikan dalam sistem pengendali jarak untuk sebuah robot mobil terhadap penghalang.

5.2. Saran

Dalam perancangan dan pembuatan sistem kendali jarak pada mobil robot menggunakan metode logika *fuzzy* ini, masih didapatkan hal-hal yang bisa menjadikan fungsi alat lebih optimal dengan adanya penambahan atau pengembangan. Hal tersebut antara lain :

- Respon sistem masih dapat diperbaiki dengan men-tuning bentuk MF dan *rule* yang digunakan.
- Pengujian dapat dilanjutkan dengan menggunakan *membership function* yang jumlah labelnya lebih banyak sehingga dimungkinkan mendapatkan respon yang lebih baik.

DAFTAR PUSTAKA

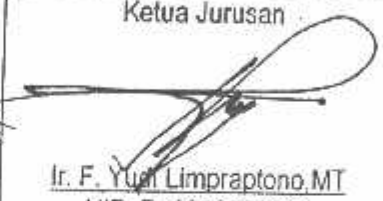
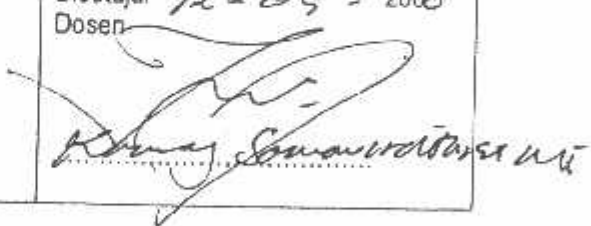
- [1] Kusumadewi, Sri, 2003, *Artificial Intelligence (Teknik dan Aplikasinya)*, Graha Ilmu, Yogyakarta
- [2] Yan, Jun, 1994, *Using Fuzzy Logic*, Prentice Hall: New Jersey
- [3] ATMEL, 2007, *Data Sheet 8 bit Microcontroler with 16 K bytes Flash AVR ATMega16*. ATMEL Corp. (www.atmel.com.)
- [4], "electronicsdownload_filesartikelAN73", (www.innovativeelectronics.com)
- [5], "digiwaredownload_filesartikelAN08", (www.digiware.com)

LAMPIRAN



LEMBAR PENGAJUAN JUDUL SKRIPSI JURUSAN TEKNIK ELEKTRO S-1

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika*)

1.	Nama Mahasiswa: <u>WILI HARDIANCYAH</u>	Nim: <u>0217050</u>
2.	Waktu Pengajuan	Tanggal: _____ Bulan: _____ Tahun: _____
3.	Spesifikasi Judul (berilah tanda silang)**)	
	a. Sistem Tenaga Elektrik b. Energi & Konversi Energi c. Tegangan Tinggi & Pengukuran d. Sistem Kendali Industri	(e.) Elektronika & Komponen f. Elektronika Digital & Komputer g. Elektronika Komunikasi h. lainnya
4.	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen*) <u>7. Kang Samudra, ST</u>	Ketua Jurusan  Ir. F. Yudi Limpraptono, MT NIP. P. 1039500274
5.	Judul yang diajukan mahasiswa:	<u>PERANCANGAN DAN PEMBUATAN SISTEM KENDALI JARAK MOBIL ROBOT DENGAN PENGHALANG DEPAN, SAMPING KANAN DAN SAMPING KIRI MENGGUNAKAN METODE LOGIKA F22</u>
6.	Perubahan judul yang disetujui Dosen sesuai materi bidang ilmu
	Catatan: <u>ditambah Rumus COA</u>	
7.	Persetujuan Judul skripsi yang dikonsultasikan kepada Dosen materi bidang ilmu	Disetujui Dosen <u>12-05-2008</u> 

Perhatian:

1. Formulir pengajuan ini harap dikembalikan kepada jurusan paling lambat satu minggu setelah disetujui kelompok dosen keahlian dengan dilampirkan proposal skripsi beserta persyaratan skripsi sesuai form S-1
2. Keterangan: *) Coret yang tidak perlu
**) dilingkari a, b, c, atau g sesuai bidang keahlian

Lampiran : 1 (satu) berkas
Pembimbing Skripsi

Kepada : Yth. Bapak I Komang Somawirata, ST, MT
Dosen Institut Teknologi Nasional
MALANG

Yang bertanda tangan di bawah ini :

Nama : Wili Hardiansyah
Nim : 0217050
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika

Dengan ini mengajukan permohonan, kiranya Bapak bersedia menjadi Dosen Pembimbing Utama / ~~Pendamping~~ *), untuk penyusunan Skripsi dengan judul (proposal terlampir) :

**Perancangan Dan Pembuatan Sistem Kendali Jarak Mobil Robot
Menggunakan Metode Logika Fuzzy**

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik.
Demikian permohonan kami dan atas kesediaan Bapak kami ucapkan terima kasih.

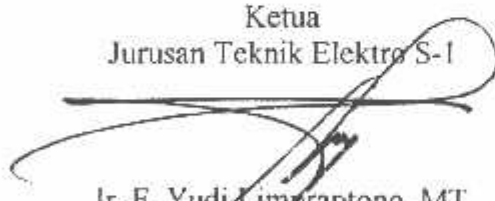
Malang,

Hormat kami,



Wili Hardiansyah

Ketua
Jurusan Teknik Elektro S-1



Ir. F. Yudi Limpraptono, MT
Nip. 1039500274

*) coret yang tidak perlu

INSTITUT TEKNOLOGI NASIONAL
Jl. Bandung Sigura-gura No. 2
MALANG

Lampiran : 1 (satu) berkas
Pembimbing Skripsi

Kepada : Yth. Bapak Eko Nurcahyo, ST, MT
Dosen Institut Teknologi Nasional
MALANG

Yang bertanda tangan di bawah ini :

Nama : Wili Hardiansyah
Nim : 0217050
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika

Dengan ini mengajukan permohonan, kiranya Bapak bersedia menjadi Dosen Pembimbing ~~Utama~~ / Pendamping *), untuk penyusunan Skripsi dengan judul (proposal terlampir) :


**Perancangan Dan Pembuatan Sistem Kendali Jarak Mobil Robot
Menggunakan Metode Logika Fuzzy**

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik.


Demikian permohonan kami dan atas kesediaan Bapak kami ucapkan terima kasih.

Malang,

Hormat kami,


Wili Hardiansyah

Ketua
Jurusan Teknik Elektro S-1


Ir. F. Yudi Limpraptono, MT
Nip. 1039500274

*) coret yang tidak perlu

10/01/2023

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : Wili Hardiansyah

Nim : 0217050

Semester : 12 (duabelas)

Jurusan : Teknik Elektro S-1

Konsentrasi : Teknik Elektronika

Dengan ini Menyatakan bersedia / ~~tidak bersedia~~ *) Membimbing Skripsi dari mahasiswa tersebut, dengan judul :

Perancangan Dan Pembuatan Sistem Kendali Jarak Mobil Robot Menggunakan Metode Logika Fuzzy

Demikian surat pernyataan ini kami buat agar dapat digunakan seperlunya.

Malang,

Kami yang membuat pernyataan,



I Komang Somawirata, ST, MT
NIP. 1300100361

Catatan :
Setelah disetujui agar formulir ini
Diserahkan mahasiswa/i yang bersangkutan
Kepada Jurusan untuk diproses lebih lanjut.
*) coret yang tidak perlu

Form S-3b

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : Wili Hardiansyah

Nim : 0217050

Semester : 12 (duabelas)

Jurusan : Teknik Elektro S-1

Konsentrasi : Teknik Elektronika

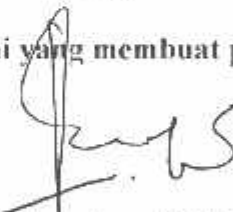
Dengan ini Menyatakan bersedia / ~~tidak bersedia~~ *) Membimbing Skripsi dari mahasiswa tersebut, dengan judul :

Perancangan Dan Pembuatan Sistem Kendali Jarak Mobil Robot Menggunakan Metode Logika Fuzzy

Demikian surat pernyataan ini kami buat agar dapat digunakan seperlunya.

Malang,

Kami yang membuat pernyataan,



Eko Nurcahyo, ST, MT
NIP. 1028700172

Catatan :

Setelah disetujui agar formulir ini
Diserahkan mahasiswa/i yang bersangkutan
Kepada Jurusan untuk diproses lebih lanjut.

*) coret yang tidak perlu



PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 10 Juni 2008

Nomor : ITN-046/7/TA /2008
Lampiran :
Perihal : Bimbingan Skripsi

Kepada : Yth. Sdr. I KOMANG SOMAWIRATA, ST, MT
Dosen Pembimbing
Jurusan Teknik Elektro S-1
di
Malang

Dengan hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi
untuk mahasiswa:

Nama : WILI HARDIANSYAH
Nim : 0217050
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya
kepada Saudara/i selama masa waktu 6 (enam) bulan, terhitung mulai
tanggal:

07 JUNI 2008 S/D 07 DESEMBER 2008

Sebagai satu syarat untuk menempuh Ujian sarjana.
Demikian atas perhatian serta kerjasama yang baik kami ucapkan
terima kasih



Ketua Jurusan
Teknik Elektro S-1

F. F. Yudi Limpraptono, MT
NIP. Y. 1039500274

Tindakan:

1. Mahasiswa yang Bersangkutan
2. Arsip

Form S-4a



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. 103411 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 10 Juni 2008

Nomor : ITN-047/7/TA /2008
Lampiran :
Perihal : Bimbingan Skripsi

Kepada : Yth. Sdr. **IR. EKO NURCAHYO**
Dosen Pembimbing
Jurusan Teknik Elektro S-1
di
Malang

Dengan hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi
untuk mahasiswa:

Nama : **WILI HARDIANSYAH**
Nim : **0217050**
Fakultas : **Teknologi Industri**
Jurusan : **Teknik Elektro S-1**
Konsentrasi : **Teknik Elektronika**

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya
kepada Saudara/i selama masa waktu 6 (enam) bulan, terhitung mulai
tanggal:

07 JUNI 2008 S/D 07 DESEMBER 2008

Sebagai satu syarat untuk menempuh Ujian sarjana.
Demikian atas perhatian serta kerjasama yang baik kami ucapkan
terima kasih



Ketua Jurusan
Teknik Elektro S-1

H. F. Yudi Limpraptono, MT
NIP. Y. 1039500274

Tindakan:

1. Mahasiswa yang Bersangkutan
2. Arsip

Form S-4a



FORMULIR BIMBINGAN SKRIPSI

Nama : Wili Hardiansyah
NIM : 02.17.050
Masa Bimbingan : 07 Januari 2009 s/d 07 Juni 2009
Judul Skripsi : Perancangan dan Pembuatan Sistem Kendali Jarak Pada Mobil Robot Menggunakan Metode Logika Fuzzy

NO	Tanggal	Uraian	Paraf Pembimbing
1	29 - 01 - 2009	REVISI BAB III DITAMBAH RULE/ATURAN- ATURAN FUZZY, DIPELAJARI SENSOR	
2	06 - 02 - 2009	SOFTWARE BELUM ALGORITMA FUZZY	
3	06 - 02 - 2009	PENGUJIAN BAB IV	
4	06 - 02 - 2009	UJI PENGARUH JARAK TERHADAP MOTOR	
5	07 - 03 - 2009	KONSULTASI BAB I, BAB II, BAB III, BAB IV, BAB V,	
6	11 - 03 - 2009	ACC BAB I, BAB II, BAB III, BAB IV, BAB V,	
7	16 - 03 - 2009	BUAT MAKALAH SEMINAR HASIL	
8	23 - 03 - 2009	ACC SKRIPSI	
9			
10			

Malang,
Dosen Pembimbing I

(I Komang Somawirata, ST, MT)
NIP. 1030100361

Form S-4B



FORMULIR BIMBINGAN SKRIPSI

Nama : Wili Hardiansyah
NIM : 02.17.050
Masa Bimbingan : 07 Januari 2009 s/d 07 Juni 2009
Judul Skripsi : Perancangan dan Pembuatan Sistem Kendali Jarak Pada Mobil Robot Menggunakan Metode Logika Fuzzy

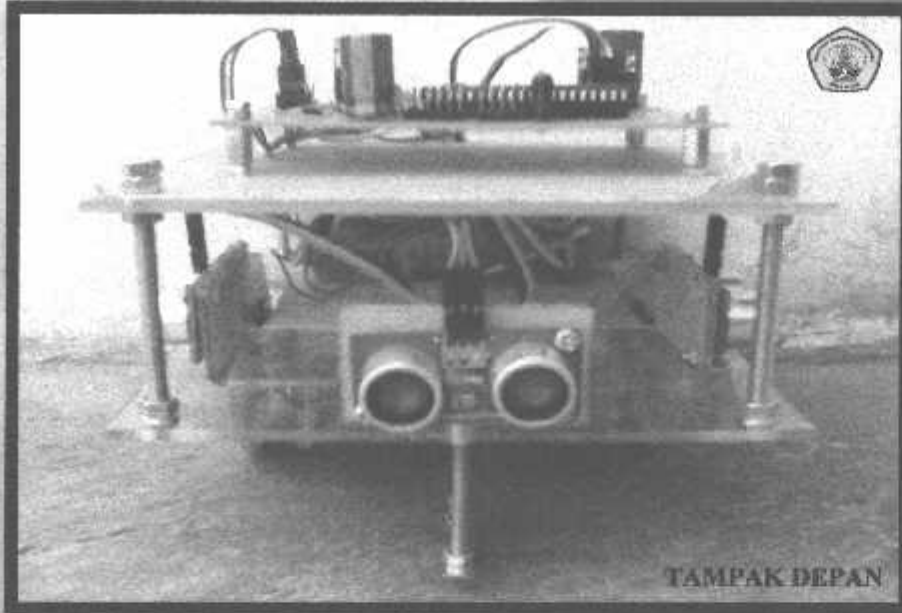
NO	Tanggal	Uraian	Paraf Pembimbing
1	21 - 01 - 2009	BAB I	
2	23 - 01 - 2009	ACC BAB I, BAB II	
3	27 - 01 - 2009	ACC BAB II, BAB III	
4	29 - 01 - 2009	ACC BAB III, BAB IV	
5	31 - 01 - 2009	ACC BAB IV, BAB V	
6	02 - 02 - 2009	ACC BAB V, PENUTUP, KESIMPULAN	
7	09 - 02 - 2009	ACC PENUTUP, KESIMPULAN, MAKALAH SEMINAR HASIL	
8	16 - 02 - 2009	ACC MAKALAH SEMINAR HASIL	
9	02 - 03 - 2009	SKRIPSI	
10	11 - 03 - 2009	ACC SKRIPSI	

Malang,
Dosen Pembimbing II

(Ir. Eko Nurcahyo)
NIP.V. 1028700172

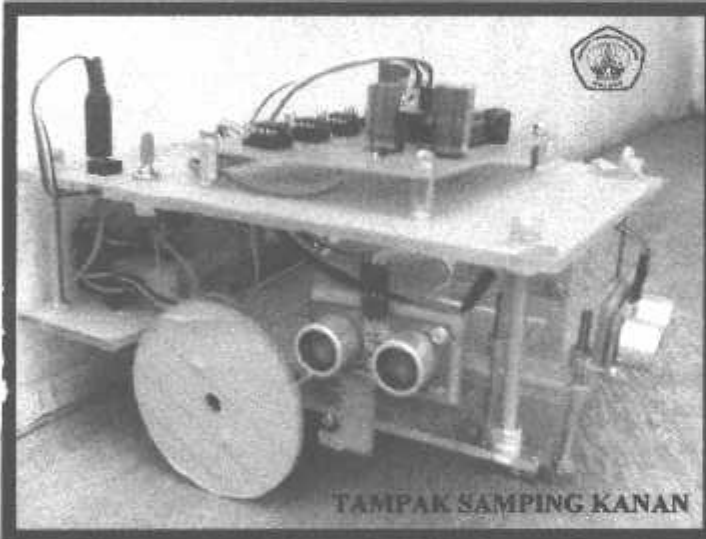
Form S-4B

FOTO ALAT

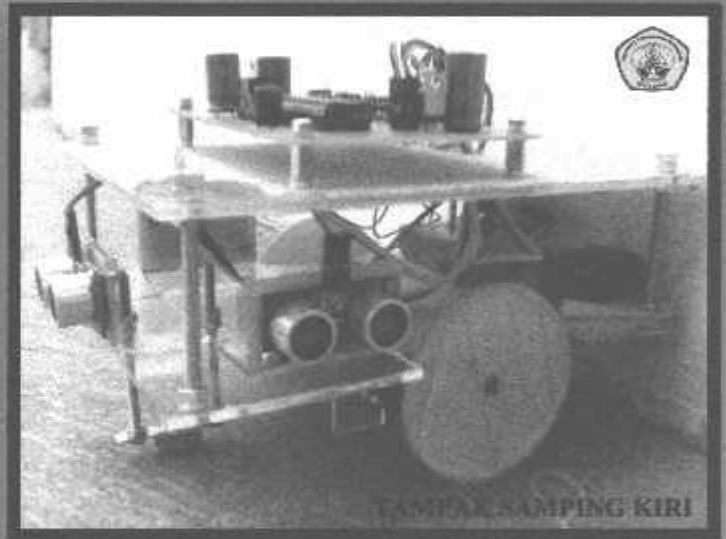


TAMPAK DEPAN

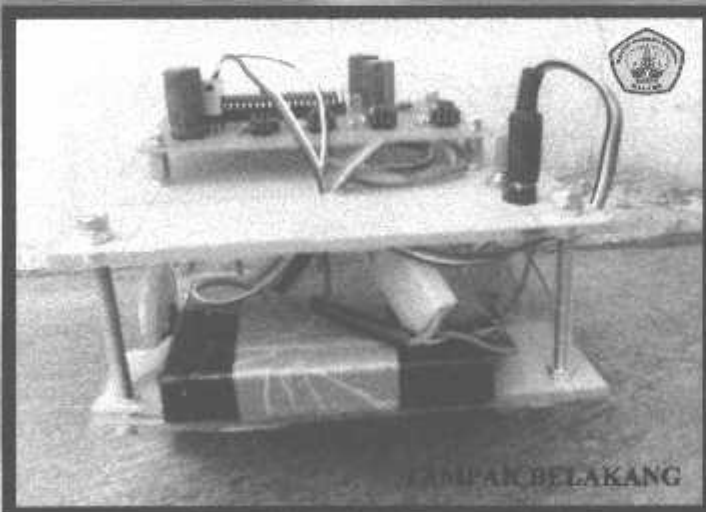
Untuk menjaga agar tetap berada pada jalur, mobil robot mengambil data jarak dari tiga sensor ultrasonik dan penggunaan sistem kontrol logika fuzzy akan memberikan sinyal pada kedua motor servo untuk mengurangi atau menambah kecepatannya. Untuk penggunaan perangkat lunak digunakan bahasa C dari Mikrokontroler. Mobil robot ini diharapkan sebagai solusi alternatif peningkatan kenyamanan berkendara yang berfungsi sebagai asisten pengemudi, misalnya saat terjadi kemacetan, berkendara dengan kecepatan tertentu saat berada pada jalan bebas hambatan.



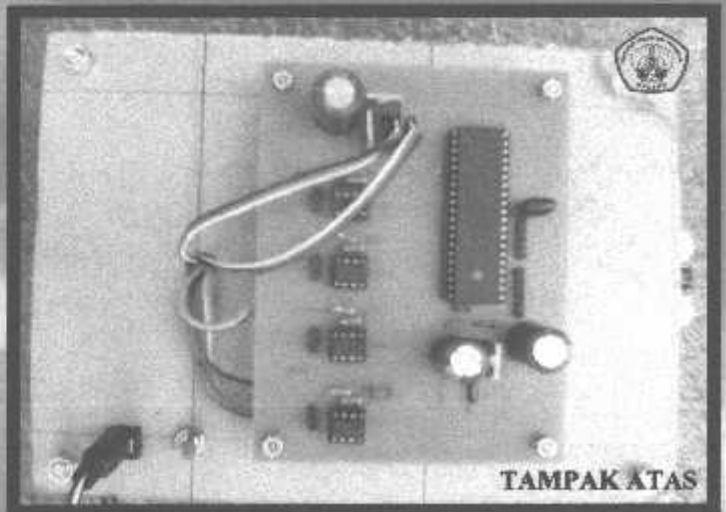
TAMPAK SAMPING KANAN



TAMPAK SAMPING KIRI

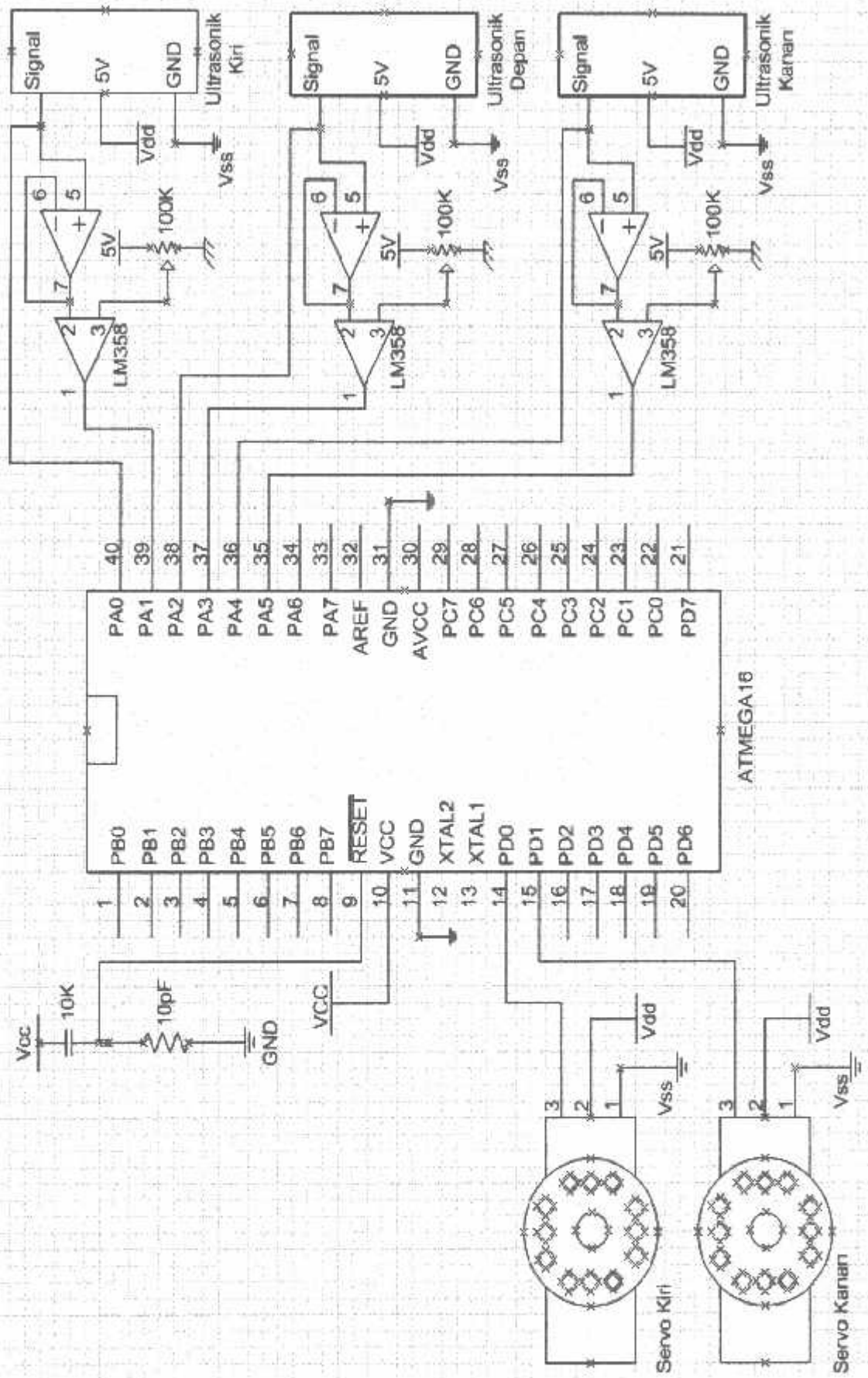


TAMPAK BELAKANG



TAMPAK ATAS

**GAMBAR RANGKAIAN
KESELURUHAN SISTEM**



SOFTWARE ALAT

```

/* File include */
#include <mega16.h>
#include <delay.h>

/* Pendefinisian */

#define servo2    PORTC.0
#define servo1    PORTC.1

#define pingka    PORTA.0
#define pongka    PINA.1
#define pingta    PORTA.2
#define pongta    PINA.3
#define pingki    PORTA.4
#define pongki    PINA.5

/* Inisialisasi variabel global */
unsigned char tanda;
int a,b;
long int ;
unsigned int i,timer,jki,jka,jta,j;
bit tka,tki,trx,tb;
long int kl,kawal;
//flash int
konstanta_timing[25]={1,666,333,111,167,133,112,96,84,74,66,60,56,52,48,44,42,40,
38,36,34,32,30,28,28};

/*Inisialisasi input output */
void init_port()
{
    DDRC=0b00110011;
    DDRB=0b00011111;
    DDRD=0b00111100;
    DDRA=0b00010101;
    //PORTA.5=1;
    //PORTA=0b11111111;
    //scl=1;sdad=1;
}

void putar1(unsigned char pu)
{
    unsigned char ii,jj;

    servo1=1;
    for(jj=0;jj<pu;jj++) delay_us(10);
    servo1=0;delay_ms(20);
}

void putar2(unsigned char pu)
{
    unsigned char ii,jj;

    servo2=1;
    for(jj=0;jj<pu;jj++) delay_us(10);
    servo2=0;delay_ms(20);
}

void maju(int kiri,int kanan)
{
    int aa,bb;
    aa=120-kanan;if(aa<40) aa=40;
    bb=120+kiri;if(bb>200) bb=200;
    putar1(aa);
}

```

```

putar2(bb);
}

int ultrakiri()
{
    pingki=1;delay_us(5);
    pingki=0;delay_us(15);
    k1=0;

    while(pongki==1) ;
    while(pongki==0) k1++;
    //kawal=k1*0/3;
    return k1/10;
}

int ultrakanan()
{
    pingka=1;delay_us(5);
    pingka=0;delay_us(15);
    k1=0;

    while(pongka==1) ;
    while(pongka==0) k1++;
    //kawal=k1*0/3;
    return k1/10;
}

int ultratengah()
{
    pingta=1;delay_us(5);
    pingta=0;delay_us(15);
    k1=0;

    while(pongta==1) ;
    while(pongta==0) k1++;
    //kawal=k1*0/3;
    return k1/10;
}

/* Program Utama */
void main()
{
    /* Inisialisasi */
    init_port();
    tka=0;tki=0;
    tanda=0;
    a=80;
    b=80;

    do
    {
        j=ultrakanan();
        jka=j;
        tampil(jka);
        j=ultrakiri();
        jki=j;
        tampil(jki);
        j=ultratengah();
        jta=j;
        tampil(jta);
    }
    if(tanda==0)
    {
        if(jka>20 && jki>20 && jta>20) maju(80,80);
        else if(jka<10 && jki<10 && jta<10);
        else if( jki<jka) maju(80,3);
        else if(jki>jka) maju(3,80);
        else if( jki==jka) maju(80,80);
        else maju(80,80);
    }
}

```

```
else if(tanda==bkanan)
{
if(jka<9)tanda=0;
maju(2,80);
}
else if(tanda==bkiri)
{
if(jki<9)tanda=0;
maju(80,2);
}
}
}while(1);
```


DATA SHEET *FUZZY*
LOGIC

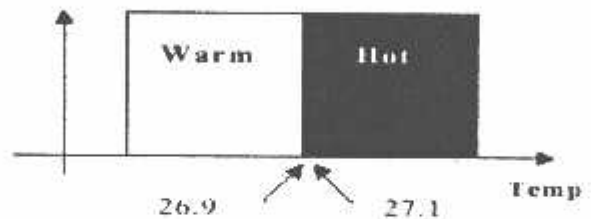
Fuzzy Logic

Pengenalan Fuzzy Logic

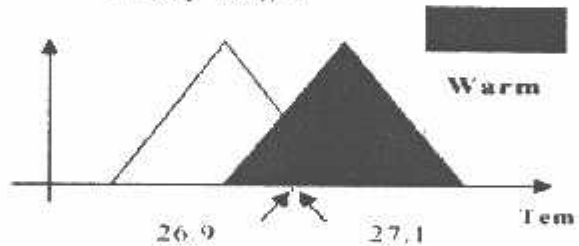
- Fuzzy logic digagas oleh Prof Lotfi Zadeh pada tahun 1965
- Pada dasarnya merupakan logika berpikir yang mengizinkan nilai antara yang didefinisikan di antara nilai logika boolean konvensional seperti benar/salah, putih/hitam, ya/tidak dsb
- Fuzzy logic mempunyai cara berpikir sepertimanusia, yang mengenali misalnya hangat panas, sangat panas, dsb

	Boolean Logic			Fuzzy Logic	
IF	TRUE	FALSE	IF	TRUE	FALSE
AND/OR	TRUE	FALSE	AND/OR	TRUE	FALSE
THEN	TRUE	FALSE	THEN	TRUE	FALSE

Boolean Logic



Fuzzy Logic



How tall is Kevin?



Kevin is 6' 2"

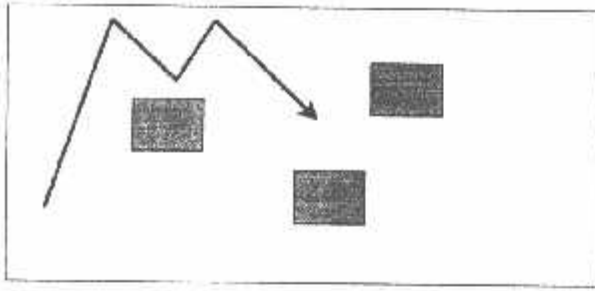
- Very Tall - 16%
- Tall - 84%
- Average - 0%
- Short - 0%
- Very Short - 0%

How tall is Kevin?

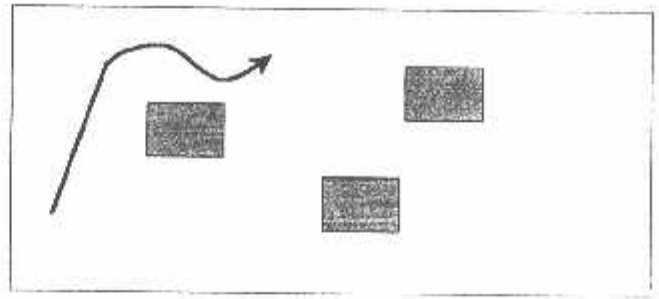


- Very Tall (7 feet)?
- Tall (6 feet)?
- Average (5 feet)?
- Short (4 feet)?
- Very Short (3 feet)?

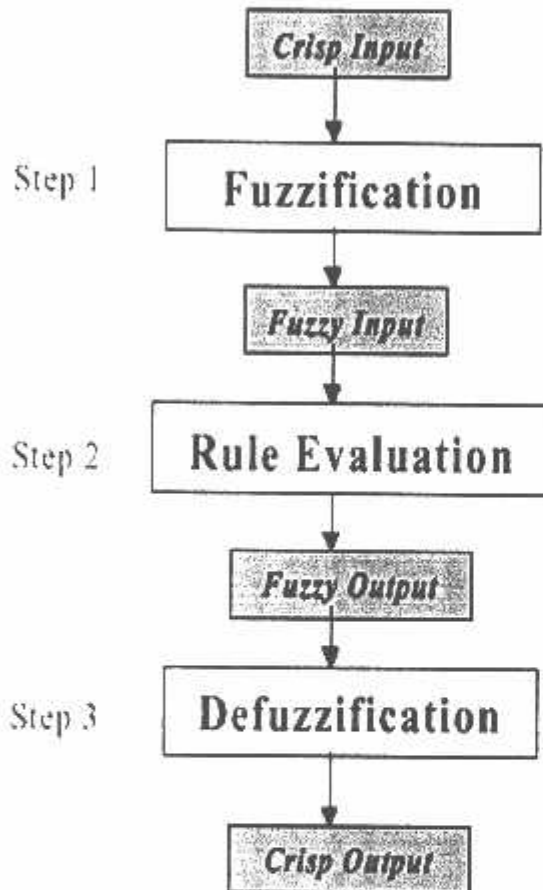
Without Fuzzy Logic



With Fuzzy Logic



Operasi Sistem Fuzzy



Fuzzy Operators: AND

$F_{AND}(A,B)$ - Fuzzy AND = $\min(A,B)$

$F_{AND}(100, 30) = 30$

$F_{AND}(20, 250) = 20$

$F_{AND}(1, 0) = 0$ -- Just like boolean logic

$F_{AND}(1, 1) = 1$ -- Geeeeee. This too!

Fuzzy Operators: OR

$F_{OR}(A,B)$ - Fuzzy OR = $\max(A,B)$

$F_{OR}(100, 30) = 100$

$F_{OR}(20, 250) = 250$

$F_{OR}(1, 0) = 1$ -- Just like boolean logic

$F_{OR}(0, 0) = 0$ -- Geeeeee. This too!

Fuzzy Operators: NOT

$F_{NOT}(A)$ - Fuzzy NOT = $100\% - A$
(100% defined as 255)

$F_{NOT}(100) = 155$

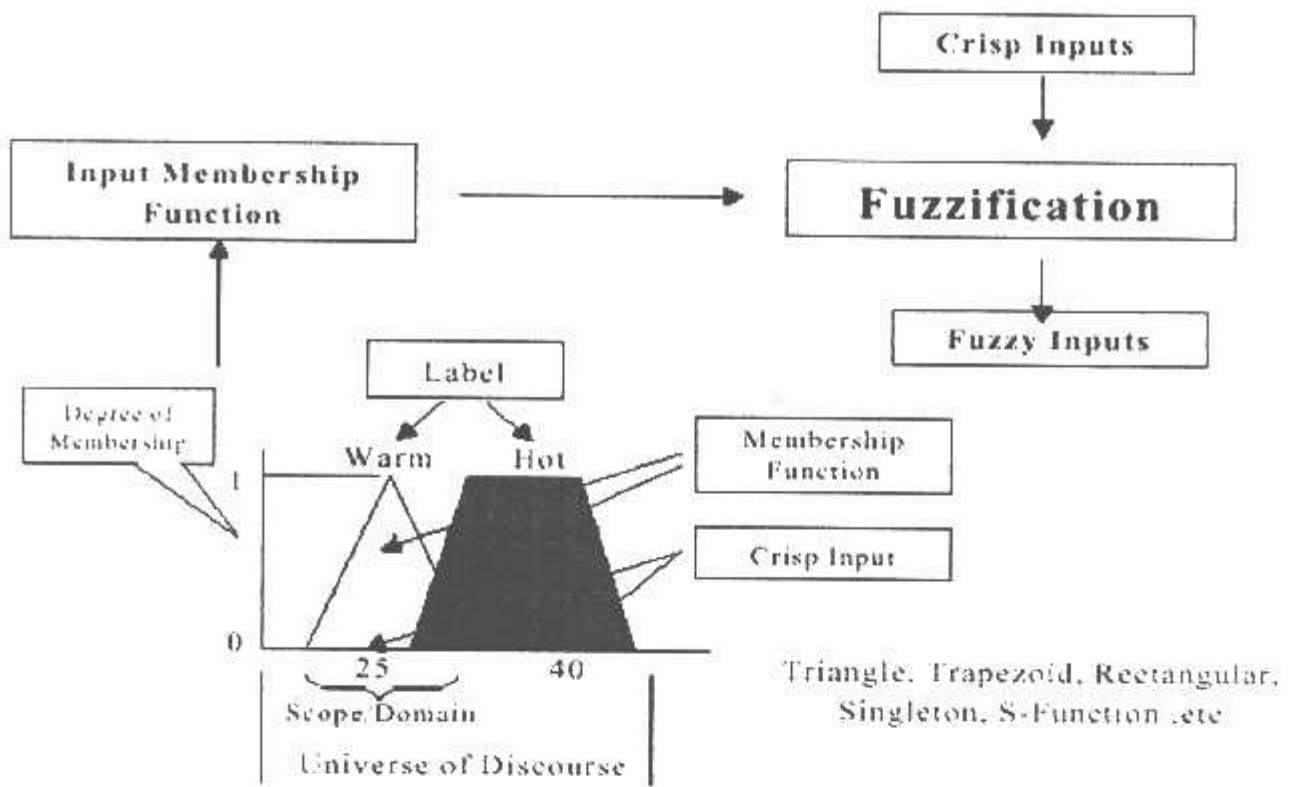
$F_{NOT}(250) = 5$

$F_{NOT}(255) = 0$

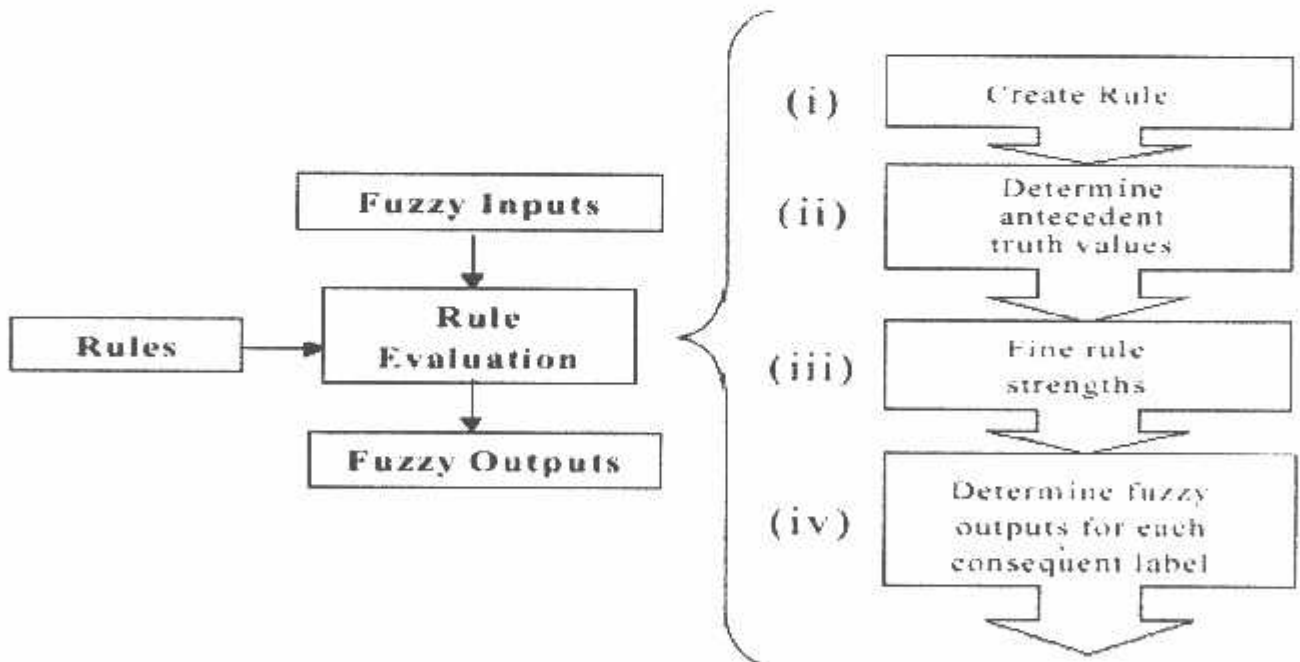
$F_{NOT}(0) = 255$

- See the similarity to Boolean logic?????

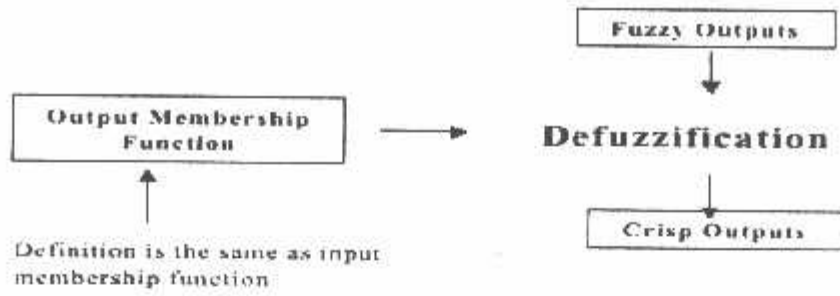
Fuzzifikasi & Membership Function



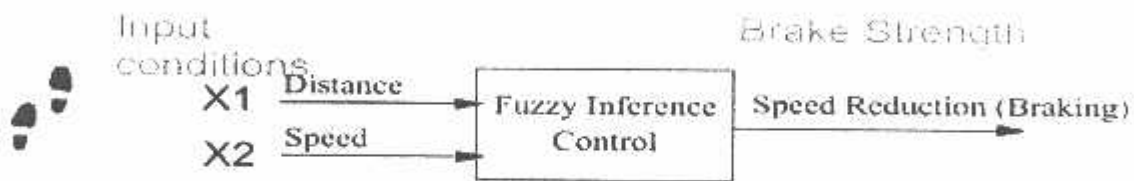
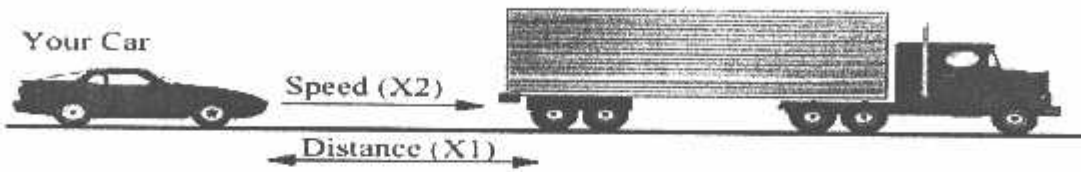
Evaluasi Rule



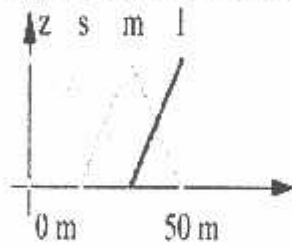
Defuzzifikasi



Brake control

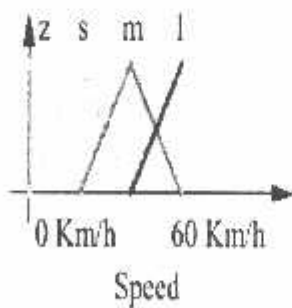


First Variable: Distance between two cars (X1)



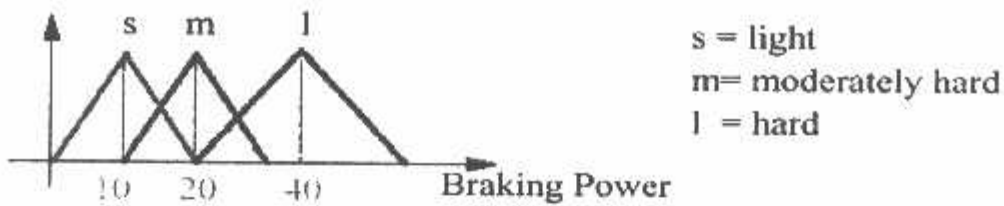
- z = zero
- s = short
- m = moderately long
- l = long

Second Variable: Speed (X2)



- z = zero
- s = low
- m = moderately high
- l = high

A Step to Step Example: Output Fuzzy Sets and Rules Definition



**Rule 1: IF the (10-20) IS Short AND the Speed IS High
THEN the Braking Power IS Hard**

**Rule 2: IF the (10-30) IS Moderately Long AND the Speed IS High
THEN the Braking Power IS Moderately Hard**

**Rule 3: IF the (20-50) IS Long OR the Speed IS Low
THEN the Braking Power IS Light**

(i) Create Rule Base

Antecedent 1		Temperature				
Antecedent 2		Cold	Cool	Normal	Warm	Hot
Moisture	Wet	short	short	short	short	short
	Moist	short	med.	med.	med.	med.
	Dry	long	long	long	long	long

Sample rules extract from the table in above table

- If temperature is *hot* **AND** soil is *dry*, then water duration is *long*.
- If temperature is *cold* **AND** soil is *dry*, then water duration is *long*.