

September
2014

JURUSAN TEKNIK MESIN



universitas
MALIKUSSALEH

BAHAN AJAR

ABUBAKAR, ST., MT

Mekatronika

FAKULTAS TEKNIK

Universitas Malikussaleh

Jurusan Teknik Kimia
Jurusan Teknik Industri
Jurusan Teknik Mesin
Jurusan Teknik Elektro
Jurusan Teknik Slipil
Prodi Teknik Informatika
Prodi Teknik Arsitektur

BAHAN AJAR

MEKATRONIKA

BAHAN AJAR

Diterbitkan oleh
FAKULTAS TEKNIK UNIVERSITAS MALIKUSSALEH
JURUSAN TEKNIK MESIN

Alamat
Fakultas Teknik Universitas Malikussaleh
Jl. Cot Tengku Nie, Reuleut, Muara Batu,
Aceh Utara, Provinsi Aceh

BAHAN AJAR

(JURUSAN TEKNIK MESIN)



universitas
MALIKUSSALEH

MEKATRONIKA

Disusun Oleh:

ABUBAKAR, ST, MT

**FAKULTAS TEKNIK
UNIVERSITAS MALIKUSSALEH
2014**



universitas
MALIKUSSALEH

BAHAN AJAR
JURUSAN TEKNIK MESIN
TIM PENGELOLA BAHAN AJAR
FAKULTAS TEKNIK UNIVERSITAS MALIKUSSALEH

PENASEHAT:

Ir. T. Hafli., MT
Dekan Fakultas Teknik Universitas Malikussaleh

PENANGGUNG JAWAB:

Herman Fithra, ST., MT
Pembantu Dekan I Bidang Akademik

Bustami, S.Si., M.Si
Pembantu Dekan II Bidang Keuangan

Edzwarsyah, ST., MT
Pembantu Dekan III Bidang Kemahasiswaan

Salwin, ST., MT
Pembantu Dekan IV Bidang Kerjasama dan Informasi

KETUA PENYUNTING:

Zulfikar, ST., MT
Ketua Jurusan Teknik Mesin

Reza Putra, ST., M.Eng
Sekretaris Jurusan Teknik Mesin

TATA USAHA DAN BENDAHARA:

Elizar, S. Sos
Kepala Tata Usaha

Ismail, ST
Bendahara

SAMBUTAN
KETUA JURUSAN TEKNIK MESIN
FAKULTAS TEKNIK UNIVERSITAS MALIKUSSALEH

Dalam upaya peningkatan kualitas mutu pembelajaran sesuai dengan Tri Dharma Perguruan Tinggi, salah satunya adalah penyediaan bahan ajar Mekatronika. Bahan ajar Mekatronika ini dibuat sebagai pegangan untuk dosen pengampu dan mahasiswa dalam melaksanakan proses pembelajaran.

Bahan ajar Mekatronika ini sangat penting sebagai salah satu referensi untuk kemudahan dalam proses belajar mengajar untuk mata kuliah pada Jurusan Teknik Mesin di Fakultas Teknik Unimal, Bahan ajar ini semoga dapat memberikan manfaat bagi para pembaca.

Kepada sdr Abubakar, ST,. MT penyusun yang telah membuat bahan ajar ini dan juga kepada semua pihak yang telah membantunya, kami ucapkan terimakasih.

Reuleut, 24 Agustus 2014
Ketua Jurusan Teknik Mesin
Fakultas Teknik Unimal,

Zulfikar, ST,. MT
Nip:197809182003121001

KATA PENGANTAR

Puji syukur kehadiran Allah SWT dengan rahmatnya penulis diberi kesehatan dan waktu sehingga dapat menyelesaikan penulisan bahan ajar tentang Mekatronika yang di fokuskan tentang teknologi yang mengkombinasikan mekanika dengan elektronik serta teknologi informasi untuk membentuk interaksi fungsi dan keterpaduan ruang dalam komponen modul yang kiranya menjadi panganan terutama untuk mata kuliah Mekatronika di Jurusan Teknik Mesin Fakultas Teknik Universitas Malikussaleh..

Dengan buku ini diharapkan akan menambah bahan bacaan dan rujukan mengenai aspek teoritis dan praktis dalam menuntukan bahan sesuai dengan fungsi dan kegunaannya. Atas terselesainya buku ini penulis mengucapkan terima kasih kepada Bapak Ir. T. Hafli, MT selaku Dekan Fakultas Teknik atas dukungan dan arahnya untuk meningkatkan mutu penulisan bahan ajar ini.

Lhokseumawe, 23 September 2014
Penulis.

Abubakar, ST,. MT
NIP. 196808202002121001

LEMBAR PENGESAHAN

This page is intentionally left Blank

DAFTAR ISI

Sambutan Ketua Jurusan Teknik Mesin.....	v
Kata Pengantar	vi
Lembar Pengesahan	vii
Daftar Isi	ix
Silabus Mata Kuliah.....	xii
Satuan Acara Pengajaran (SAP)	xv
BAB 1. PENGANTAR MEKATRONIKA.....	1
1.1. Deskripsi Singkat	1
1.2. Tujuan Intruksional Khusus.....	1
1.3. Penyajian.....	1
1.3.1. Pengantar	1
1.3.2. Gambaran Umum Tentang Mekatronika	3
1.3.3. Aspek Pengendalian Elektrik	5
1.3.4. Aspek Mekanika	7
1.4 Penutup.....	9
1.4.1. Bahan Diskusi Dan Tugas	9
1.5 Daftar Pustaka.....	10
1.6. Senarai.....	10
BAB 2. S E N S O R.....	11
2.1. Deskripsi Singkat	11
2.2. Tujuan Instruksional Khusus	11
2.3.2. Sensor Cahaya.....	12
2.3.3. Sensor Suhu	12
2.3.4. Sensor Proximity	13
2.3.5. Pressure Sensors.....	16
2.3.6. Displacement Transduser.....	17
2.3.7. Sensor Putaran/Velocity (Rpm Sensor)	18
2.4 Penutup.....	18
2.4.1 Bahan Diskusi Dan Tugas	18
2.5 Daftar Pustaka.....	19
2.6. Senarai.....	19
BAB 3. AKTUATOR ELEKTRIK	21
3.1. Deskripsi Singkat	21
3.2. Tujuan Intruksional Khusus.....	21
3.3. Penyajian	21
3.3.1. Pengantar	21
3.3.2. Relay	22
3.3.3. Solenoida.....	23
3.3.4. Motor Stepper	23

3.3.5. Servo Motor	25
3.3.6. Motor Dc.....	26
3.3.7. Motor Ac.....	27
3.4. Penutup.....	30
3.4.1 Bahan Diskusi Dan Tugas	30
3.5 Daftar Pustaka.....	30
3.6. Senarai.....	30
BAB 4. SISTEM MEKANIK.....	31
4.1. Deskripsi Singkat	31
4.2. Tujuan Intruksional Khusus.....	31
4.3. Penyajian	31
4.3.1. Pengantar	31
4.3.2. Transmisi Mekanik	31
4.3.3. Kinematika	34
4.3.4. Pneumatik.....	38
4.3.5. Sistem Hidrolik	42
4.4. Penutup.....	44
4.4.1. Bahan Diskusi Dan Tugas	44
4.5. Daftar Pustaka.....	44
4.6. Senarai.....	44
BAB 5. PENGENDALIAN BERBASIS MIKROPROSESSOR.....	45
5.1. Deskripsi Singkat	45
5.2. Tujuan Intruksional Khusus.....	45
5.3. Penyajian	45
5.3.1. Pengantar	45
5.3.2. Arsitektur Mikroprosesor.....	45
5.3.3. Transfer Data Paralel (Ppi 8255)	50
5.4. Sistem Interface Input/Output Antara Sistem Digital Dan Sistem Analog.....	52
5.4.1. Konversi Analog Dan Digital	53
5.5. Penutup.....	55
5.5.1. Bahan Diskusi Dan Tugas	55
5.6. Daftar Pustaka.....	56
5.7. Senarai.....	56
BAB 6. MIKRO KONTROLER	57
6.1. Deskripsi Singkat	57
6.2. Tujuan Intruksional Khusus.....	57
6.3. Penyajian	57
6.3.1. Pengantar	57
6.3.2. Konstruksi Dasar At89cx051	58
6.3.3. Memori Pada At89cx051	58
6.3.4. Dasar Pemrograman Mcs51	63
6.3.5. Pengetahuan Dasar Program Assembly	69
6.4. Penutup.....	75

6.4.1. Bahan Diskusi Dan Tugas	75
6.5. Daftar Pustaka.....	75
6.6. Senarai.....	75
BAB 7. TEKNIK PEMROGRAMAN PLC	77
7.1. Deskripsi Singkat	77
7.2. Tujuan Intruksional Khusus.....	77
7.3. Penyajian.....	77
7.3.1. Pengantar	77
7.3.2. Bahasa Pemrograman	78
7.3.3. Struktur Daerah Memori.....	79
7.3.4. Instruksi Pemrograman	79
7.3.5. Eksekusi Pemrograman	81
7.3.6. Langkah-Langkah Pembuatan Program.....	82
7.4. Penutup.....	83
7.4.1 Bahan Diskusi Dan Tugas	83
7.5 Daftar Pustaka.....	83
7.6 Senarai.....	83

SILABUS MATA KULIAH

1. Identitas Perguruan Tinggi :

- a. Perguruan Tinggi : Universitas Malikussaleh
- b. Fakultas : Teknik
- c. Jurusan : Teknik Mesin
- d. Program Studi : Teknik Mesin

2. Identitas Mata Kuliah :

- a. Nama Mata Kuliah : Mekatronika
- b. Kode Mata Kuliah : TMS 463
- c. Status Mata Kuliah : Wajib
- d. Sifat Mata Kuliah : Teori
- e. Dosen Pengampu : Abubakar, ST., MT
- f. Semester : VI
- g. Bobot SKS : 3
- h. Jumlah Pertemuan : 14 tatap muka + UTS dan UAS

3. Mata Kuliah Prasyarat

- 1). Bahasa Pemrograman
- 2). Teknik Tenaga Listrik

4. Komponen Penilaian /Evaluasi

- a. Kehadiran perkuliahan, min 75 %
- b. Tugas-tugas parsial
- c. Tugas utama (Presentasi)
- d. UTS dan UAS

5. Kompetensi dan lingkup materi Perkuliahan

a. Kompetensi

Setelah mengikuti mata kuliah mekatronika sebagaimana yang disyaratkan dalam ketentuan pelaksanaan kegiatan akademik di Jurusan Teknik Mesin Fakultas Teknik Universitas Malikussaleh, maka mahasiswa diharapkan:

- 1) Mahasiswa memahami dasar-dasar mekatronika
- 2) Mahasiswa dapat mengetahui tentang sensor dan mampu menjelaskan jenis-jenis sensor.
- 3) Mahasiswa dapat mengetahui tentang aktuator dan mampu menjelaskan bagian-bagian aktuator serta penggunaannya.
- 4) Mahasiswa dapat mengetahui tentang sistem mekanik dan bagian-bagiannya.
- 5) Mahasiswa dapat menjelaskan arsitektur mikroprosesor, mengidentifikasi komponen dasar mikroprosesor dan dapat menjelaskan mekanisme pengendalian peralatan luar menggunakan mikroprosesor.
- 6) Mahasiswa dapat memahami arsitektur dan bagian-bagian mikrokontroler serta mampu membuat program mikrokontroler.
- 7) Mahasiswa dapat merancang program kendali PLC sederhana, memasukkan

program ke dalam PLC dan dapat Mengecek kebenaran program.

b. Pokok-pokok materi Pembelajaran

- 1) Tinjauan historis mekatronika dan perkembangannya
- 2) Sensor
- 3) Aktuator Elektrik
- 4) Sistem Mekanik
- 5) Pengendalian Berbasis Mikroprosesor
- 6) Mikrokontroler
- 7) Teknik Pemrograman PLC

6. Deskripsi Mata Kuliah

Mata kuliah mekatronika difokuskan tentang teknologi yang mengkombinasikan mekanika dengan elektronik serta teknologi informasi untuk membentuk interaksi fungsi dan keterpaduan ruang dalam komponen modul dan sistem produksi.

7. Pendekatan Pembelajaran (individual/kelompok/kasikal)

Pelaksanaan pembelajaran dilakukan secara teoritis dan mahasiswa diharapkan sebelum memulai perkuliahan sudah mempelajari pokok bahasan sebelumnya. Pengajaran teoritis dilakukan secara klasikal/berkelompok, dengan menggunakan pendekatan metoda ceramah, diskusi, dan presentasi kelompok.

8. Media Pembelajaran

Media pembelajaran digunakan menurut sifat dan karakteristik materi perkuliahan yang diajarkan, meliputi penggunaan LCD proyektor, white board, dan gambar-gambar.

9. Penilaian

A. Tugas Parsial, agar mahasiswa termotivasi untuk melakukan kajian dan pendalaman materi perkuliahan dari berbagai sumber, yaitu berupa :

- 1) Kajian pustaka tentang Pengertian, sejarah, peran, dan perkembangan mekatronika, sensor dan aktuator elektrik .
- 2) Kajian pustaka tentang system mekanik dan pengendalian berbasis mikroprosesor
- 3) Kajian pustaka tentang mikrokontroler dan teknik pemrograman PLC.

B. Tugas Utama

Tugas ini bertujuan agar mahasiswa mampu menerapkan keilmuwan ini dalam penerapan sehari-hari

10. Sumber Pustaka/Referensi

1. J. Webb, *Industrial Control Electronics*, Macmillan Publish,. New York, 1992.
2. J. Webb, *Programmable Logic Control*, Macmillan Publish,. New York, 1992.
3. Frank D Petruzella, *Elektronik Industri* (terjemahan), Andi, Yogyakarta, 2001.

4. William Bolton, *Programmable Logic Controller (PLC)*, Sebuah Pengantar, Edisi ketiga, Penerbit Erlangga, 2004.
5. M. Budiyanto dan A. Wijaya, *Pengenalan Dasar-dasar PLC (Programmable Logic Controller)*, Penerbit Gaya Media, Yogyakarta, 2003.
6. Endra Ditawarna, *Robotika, Desain, Kontrol dan Kecerdasan Buatan*, Penerbit Andi, Yogyakarta, 2006.
7. David G. Alciator dan Michael B Histan, *Introduction to Mechatronics and Measurement Systems*, Mc Graw Hill, New York, 2003.

SATUAN ACARA PENGAJARAN (SAP)

Mata Kuliah : Mekatronika
Kode Mata Kuliah : TMS 463
SKS : 3 SKS
Waktu Pertemuan : 3 x 50 menit
Pertemuan : I

A. TUJUAN PEMBELAJARAN

1. Tujuan Pembelajaran Umum (TPU) :
Mahasiswa diharapkan dapat memahami substansi tentang mekatronika.
2. Tujuan Pembelajaran Khusus (TPK) :
Mahasiswa dapat mengetahui tentang mekatronika dan mampu menjelaskan aspek elektrik dan Mekanika

B. POKOK BAHASAN : Pengertian, sejarah, peran, dan perkembangan mekatronika

C. SUB POKOK BAHASAN :

1. Gambaran Umum Tentang Mekatronika
2. Aspek pengendalian Elektrik
3. Aspek Mekanika

D. KEGIATAN BELAJAR MENGAJAR :

TAHAPAN KEGIATAN	KEGIATAN DOSEN	KEGIATAN MAHASISWA	MEDIA & ALAT PENGAJARAN	WAKTU (MENIT)	REFER ENSI
1	2	3	4	5	6
Pendahuluan	<ol style="list-style-type: none"> 1. Menjelaskan tata tertib perkuliahan 2. Menjelaskan cakupan materi kuliah dan referensi 3. Menjelaskan TPU dan TPK 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector	3 X 50	1,2,3,4,5 ,6,7
Penyajian	Menjelaskan Tentang: <ol style="list-style-type: none"> 1. Pengertian mekatronika 2. Sejarah mekatronika 3. Pengendalian elektrik 4. Aspek mekanika 	Memperhatikan/ menyimak/ bertanya	<input checked="" type="checkbox"/> Papan Tulis		
Penutup	<ol style="list-style-type: none"> 1. Memberikan kesempatan kepada mahasiswa untuk bertanya 2. Memberikan kesempatan kepada mahasiswa untuk mencoba menjawab pertanyaan (aktivitas mahasiswa) 3. Menjawab pertanyaan 4. Menjelaskan mengenai pertemuan akan datang 	Memperhatikan/ menyimak/ bertanya	<input checked="" type="checkbox"/> Papan Tulis		

SATUAN ACARA PERKULIAHAN (SAP)

Mata Kuliah : Mekanika
Kode Mata Kuliah : TMS 463
SKS : 3 SKS
Waktu Pertemuan : 6 x 50 menit
Pertemuan : II dan III

A. TUJUAN PEMBELAJARAN

1. Tujuan Pembelajaran Umum (TPU) :
Mahasiswa diharapkan dapat memahami tentang sensor.
2. Tujuan Pembelajaran Khusus (TPK) :
Mahasiswa mampu memahami sensor dan dapat menjelaskan jenis-jenis Sensor.

B. POKOK BAHASAN : **Sensor**

C. SUB POKOK BAHASAN :

1. Sensor cahaya
2. Sensor suhu
3. Sensor proximity
4. Pressure Sensors
5. Displacement Transducer
6. Sensor Putaran/Velocity Aspek pengendalian Elektrik

D. KEGIATAN BELAJAR MENGAJAR :

TAHAPAN KEGIATAN	KEGIATAN DOSEN	KEGIATAN MAHASISWA	MEDIA & ALAT PENGAJARAN	WAKTU (MENIT)	REFERENSI
1	2	3	4	5	6
Pendahuluan	<ol style="list-style-type: none"> 1. Menjelaskan tata tertib perkuliahan 2. Menjelaskan cakupan materi kuliah dan referensi 3. Menjelaskan TPU dan TPK 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector	3 X 50	1,2,3,4,5,6,7
Penyajian	Menjelaskan Tentang: <ol style="list-style-type: none"> 1. Sensor cahaya 2. Sensor suhu 3. Sensor Proximity 4. Sensor tekanan 5. Displacement transducer 6. Sensor putaran 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		
Penutup	<ol style="list-style-type: none"> 1. Memberikan kesempatan kepada mahasiswa untuk bertanya 2. Memberikan kesempatan kepada mahasiswa untuk mencoba menjawab pertanyaan (aktivitas mahasiswa) 3. Menjawab pertanyaan 4. Menjelaskan mengenai pertemuan akan datang 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		

SATUAN ACARA PERKULIAHAN (SAP)

Mata Kuliah : Mekatronika
Kode Mata Kuliah : TMS 463
SKS : 3 SKS
Waktu Pertemuan : 6 x 50 menit
Pertemuan : IV dan V

A. TUJUAN PEMBELAJARAN

1. Tujuan Pembelajaran Umum (TPU) :
Mahasiswa diharapkan dapat memahami tentang aktuator elektrik.
2. Tujuan Pembelajaran Khusus (TPK) :
Mahasiswa mampu memahami aktuator elektrik dan dapat menjelaskan jenis-jenis aktuator elektrik.

B. POKOK BAHASAN : Aktuator Elektrik

C. SUB POKOK BAHASAN :

1. Relay
2. Solenoida
3. Motor stepper
4. Servo Motor
5. Motor DC
6. Motor AC

D. KEGIATAN BELAJAR MENGAJAR :

TAHAPAN KEGIATAN	KEGIATAN DOSEN	KEGIATAN MAHASISWA	MEDIA & ALAT PENGAJARAN	WAKTU (MENIT)	REFERENSI
1	2	3	4	5	6
Pendahuluan	<ol style="list-style-type: none"> 1. Menjelaskan tata tertib perkuliahan 2. Menjelaskan cakupan materi kuliah dan referensi 3. Menjelaskan TPU dan TPK 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector	6 X 50	1,2,3,4,5,6, 7
Penyajian	Menjelaskan Tentang: <ol style="list-style-type: none"> 1. Relay 2. Solenoida 3. Motor Stepper 4. Servo Motor 5. Motor DC 6. Motor AC 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		
Penutup	<ol style="list-style-type: none"> 1. Memberikan kesempatan kepada mahasiswa untuk bertanya 2. Memberikan kesempatan kepada mahasiswa untuk mencoba menjawab pertanyaan (aktivitas mahasiswa) 3. Menjawab pertanyaan 4. Menjelaskan mengenai pertemuan akan datang 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		

SATUAN ACARA PERKULIAHAN (SAP)

Mata Kuliah : Mekatronika
Kode Mata Kuliah : TMS 463
SKS : 3 SKS
Waktu Pertemuan : 6 x 50 menit
Pertemuan : VI dan VII

A. TUJUAN PEMBELAJARAN

1. Tujuan Pembelajaran Umum (TPU) :
Mahasiswa diharapkan dapat memahami tentang system mekanik.
2. Tujuan Pembelajaran Khusus (TPK) :
Mahasiswa mampu memahami system mekanik dan dapat menjelaskan jenis-jenis sistem mekanik.

B. POKOK BAHASAN : **Sistem Mekanik**

C. SUB POKOK BAHASAN :

1. Transmisi Mekanik
2. Kinematika
3. Pneumatik
4. Sistem Hidrolik

D. KEGIATAN BELAJAR MENGAJAR :

TAHAPAN KEGIATAN	KEGIATAN DOSEN	KEGIATAN MAHASISWA	MEDIA & ALAT PENGAJARAN	WAKTU (MENIT)	REFERENSI
1	2	3	4	5	6
Pendahuluan	<ol style="list-style-type: none"> 1. Menjelaskan tata tertib perkuliahan 2. Menjelaskan cakupan materi kuliah dan referensi 3. Menjelaskan TPU dan TPK 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector	6 X 50	1,2,3,4,5,6,7
Penyajian	Menjelaskan Tentang: <ol style="list-style-type: none"> 1. Transmisi mekanik 2. Kinematika 3. Pneumatik 4. Sistem hidrolik 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		
Penutup	<ol style="list-style-type: none"> 1. Memberikan kesempatan kepada mahasiswa untuk bertanya 2. Memberikan kesempatan kepada mahasiswa untuk mencoba menjawab pertanyaan (aktivitas mahasiswa) 3. Menjawab pertanyaan 4. Menjelaskan mengenai pertemuan akan datang 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		

SATUAN ACARA PERKULIAHAN (SAP)

Mata Kuliah : Mekatronika
Kode Mata Kuliah : TMS 463
SKS : 3 SKS
Waktu Pertemuan : 6 x 50 menit
Pertemuan : IX dan X

A. TUJUAN PEMBELAJARAN

1. Tujuan Pembelajaran Umum (TPU) :

Mahasiswa diharapkan dapat memahami tentang Pengendalian berbasis mikroprosesor.

2. Tujuan Pembelajaran Khusus (TPK) :

Mahasiswa mampu memahami dan dapat menjelaskan pengendalian berbasis mikroprosesor.

B. POKOK BAHASAN : Pengendalian Berbasis Mikroprosesor

C. SUB POKOK BAHASAN :

1. Arsitektur Mikroprosesor
2. Transfer Data Paralel
3. Sistem Interface
4. Konversi Analog dan Digital

D. KEGIATAN BELAJAR MENGAJAR :

TAHAPAN KEGIATAN	KEGIATAN DOSEN	KEGIATAN MAHASISWA	MEDIA & ALAT PENGAJARAN	WAKTU (MENIT)	REFERENSI
1	2	3	4	5	6
Pendahuluan	<ol style="list-style-type: none"> 1. Menjelaskan tata tertib perkuliahan 2. Menjelaskan cakupan materi kuliah dan referensi 3. Menjelaskan TPU dan TPK 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector	6 X 50	1,2,3,4,5,6,7
Penyajian	Menjelaskan Tentang: <ol style="list-style-type: none"> 1. Arsitektur mikroprosesor 2. Transfer data paralel 3. Sistem interface 4. Konversi analog dan digital 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		
Penutup	<ol style="list-style-type: none"> 1. Memberikan kesempatan kepada mahasiswa untuk bertanya 2. Memberikan kesempatan kepada mahasiswa untuk mencoba menjawab pertanyaan (aktivitas mahasiswa) 3. Menjawab pertanyaan 4. Menjelaskan mengenai pertemuan akan datang 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		

SATUAN ACARA PERKULIAHAN (SAP)

Mata Kuliah : Mekatronika
Kode Mata Kuliah : TMS 463
SKS : 3 SKS
Waktu Pertemuan : 6 x 50 menit
Pertemuan : XI dan XII

A. TUJUAN PEMBELAJARAN

1. Tujuan Pembelajaran Umum (TPU) :
Mahasiswa diharapkan dapat memahami tentang mikro kontroler.
2. Tujuan Pembelajaran Khusus (TPK) :
Mahasiswa mampu memahami dan dapat menjelaskan tentang mikro kontroler.

B. POKOK BAHASAN : **Mikro Kontroler**

C. SUB POKOK BAHASAN :

1. Konstruksi dasar AT89Cx051
2. Memori Pada AT89Cx051
3. Dasar Pemrograman MCS51
4. Pengetahuan Dasar Program Assembly

D. KEGIATAN BELAJAR MENGAJAR :

TAHAPAN KEGIATAN	KEGIATAN DOSEN	KEGIATAN MAHASISWA	MEDIA & ALAT PENGAJARAN	WAKTU (MENIT)	REFERENSI
1	2	3	4	5	6
Pendahuluan	<ol style="list-style-type: none"> 1. Menjelaskan tata tertib perkuliahan 2. Menjelaskan cakupan materi kuliah dan referensi 3. Menjelaskan TPU dan TPK 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector	6 X 50	1,2,3,4,5,6,7
Penyajian	Menjelaskan Tentang : <ol style="list-style-type: none"> 1. Konstruksi dasar AT89Cx051 2. Memori Pada AT89Cx051 3. Dasar Pemrograman MCS51 4. Pengetahuan Dasar Program Assembly 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		
Penutup	<ol style="list-style-type: none"> 1. Memberikan kesempatan kepada mahasiswa untuk bertanya 2. Memberikan kesempatan kepada mahasiswa untuk mencoba menjawab pertanyaan (aktivitas mahasiswa) 3. Menjawab pertanyaan 4. Menjelaskan mengenai pertemuan akan datang 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		

SATUAN ACARA PERKULIAHAN (SAP)

Mata Kuliah : Mekatronika
Kode Mata Kuliah : TMS 463
SKS : 3 SKS
Waktu Pertemuan : 9 x 50 menit
Pertemuan : XIII, XIV dan XV

A. TUJUAN PEMBELAJARAN

1. Tujuan Pembelajaran Umum (TPU) :
Mahasiswa diharapkan dapat memahami tentang Teknik Pemrograman PLC.
2. Tujuan Pembelajaran Khusus (TPK) :
Mahasiswa mampu memahami dan dapat membuat program PLC.

B. POKOK BAHASAN : Teknik Pemrograman PLC

C. SUB POKOK BAHASAN :

1. Bahasa Pemrograman
2. Struktur Daerah Memori
3. Instruksi Pemrograman
4. Eksekusi Pemrograman
5. Langkah-langkah pembuatan program

D. KEGIATAN BELAJAR MENGAJAR :

TAHAPAN KEGIATAN	KEGIATAN DOSEN	KEGIATAN MAHASISWA	MEDIA & ALAT PENGAJARAN	WAKTU (MENIT)	REFERENSI
1	2	3	4	5	6
Pendahuluan	<ol style="list-style-type: none"> 1. Menjelaskan tata tertib perkuliahan 2. Menjelaskan cakupan materi kuliah dan referensi 3. Menjelaskan TPU dan TPK 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector	9 X 50	1,2,3,4,5,6,7
Penyajian	Menjelaskan Tentang : <ol style="list-style-type: none"> 1. Bahasa Pemrograman 2. Struktur Daerah Memori 3. Instruksi Pemrograman 4. Eksekusi Pemograman 5. Langkah-langkah pembuatan program 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		
Penutup	<ol style="list-style-type: none"> 1. Memberikan kesempatan kepada mahasiswa untuk bertanya 2. Memberikan kesempatan kepada mahasiswa untuk mencoba menjawab pertanyaan (aktivitas mahasiswa) 3. Menjawab pertanyaan 4. Menjelaskan mengenai pertemuan akan datang 	Memperhatikan/ menyimak/ bertanya	Papan tulis (<i>white board</i>), laptop, projector		

BAB 1

PENGANTAR

MEKATRONIKA

1.1. Deskripsi Singkat

Tujuan dari mempelajari mekatronika adalah untuk dapat memadukan aspek elektronika dengan mekanika dalam teknologi.

1.2. Tujuan Instruksional Khusus

Setelah materi ini diajarkan mahasiswa dapat mengetahui tentang mekatronika dan mampu menjelaskan aspek elektronika dan mekanika.

1.3. Penyajian

1.3.1. Pengantar

Mekatronika adalah kata baru yang lahir di Jepang pada awal tahun 1970-an yang merupakan gabungan antara 2 kata yaitu *mechanics* dan *electronics*. Sekarang kita sering melihat di sekeliling kita barang-barang mekatronik seperti robot, mesin bubut NC, kamera digital, printer dan lain sebagainya. Persamaan dari barang-barang mekatronik ini adalah objek yang dikendalikan adalah gerakan mesin. Jika dibandingkan dengan gerakan mesin konvensional maka gerakan mesin tersebut lebih bersifat fleksibel dan lebih memiliki kecerdasan. Hal ini dimungkinkan karena memanfaatkan kemajuan iptek micro-electronics. Artinya dengan bantuan micro-electronics mesin dapat bergerak dengan lebih cerdas. Jika seseorang memberikan sebuah perintah, lalu semua dapat dipasrahkan ke mesin yang dapat bergerak secara otomatis. Ini sangat membantu menciptakan mesin atau alat yang praktis dan mudah digunakan. Sehingga sumber daya pada manusia seperti waktu dan otak dapat dipakai untuk pekerjaan yang lain, sehingga dapat menciptakan nilai tambah.

Pada awalnya mekatronik diarahkan pada 3 target yaitu: penghematan energi (energy saving), pengecilan dimensi dan peringanan berat, dan peningkatan kehandalahan (reliability). Sekarang, setelah 30 tahun lebih berlalu dari kelahirannya, perlu dirumuskan kembali arah mekatronik sesuai dengan perkembangan jaman. Dan khususnya untuk Indonesia sebagai negara yang masih berkembang dengan segudang permasalahannya, rasanya arah mekatronik perlu ditentukan agar dapat membantu memecahkan masalah-masalah yang ada dengan tetap memperhatikan lingkungan regional dan global.

Mekatronik adalah teknologi atau rekayasa yang menggabungkan teknologi tentang mesin elektronika, dan informatika untuk merancang, memproduksi, mengoperasikan dan memelihara sistem untuk mencapai tujuan yang diamanatkan. Seperti diketahui dari definisi, mekatronika adalah gabungan disiplin teknik mesin, teknik elektro, teknik informatika, dan teknik kendali. Pada awalnya, secara khusus tidak ada disiplin mekatronika. Untuk menggabungkan beberapa disiplin iptek tersebut, mekatronika memerlukan teori kendali dan teori sistem.

Secara sempit pengertian mekatronika mengarah pada teknologi kendali numerik yaitu teknologi mengendalikan mekanisme menggunakan aktuator untuk mencapai tujuan tertentu dengan memonitor informasi kondisi gerak mesin menggunakan sensor, dan memasukkan informasi tersebut ke dalam mikro-prosesor. Ini menyumbangkan kemajuan yang spektakuler jika dibandingkan dengan kontrol otomatis menggunakan instrument analog, karena dapat merubah skenario kontrol secara fleksibel dan dapat memiliki fungsi pengambilan keputusan tingkat tinggi.

Contoh klasik barang mekatronik adalah lengan robot dan mesin bubut control numerik. Barang-barang ini dapat melakukan pekerjaan-pekerjaan yang berbeda-beda dengan cara merubah program mereka sesuai kondisi yang diminta, karena telah ditambahkan kemampuan kendali aktif yang canggih terhadap mekanisme yang telah ada. Beberapa manfaat penerapan mekatronik adalah sebagai berikut :

1. **Meningkatkan fleksibilitas.** Manfaat terbesar yang dapat diperoleh dari penerapan mekatronik adalah meningkatkan fleksibilitas mesin dengan menambahkan fungsi-fungsi baru yang mayoritas merupakan kontribusi mikro-prosesor. Sebagai contoh, lengan robot industri dapat melakukan berbagai jenis pekerjaan dengan merubah program peranti lunak di mikro-prosesornya seperti halnya lengan manusia. Ini yang menjadi faktor utama dimungkinkannya proses produksi produk yang beraneka ragam tipenya dengan jumlah yang sedikit-sedikit.

2. **Meningkatkan kehandalan.** Pada mesin-mesin konvensional (manual) muncul berbagai masalah yang diakibatkan oleh berbagai jenis gesekan pada mekanisme yang digunakan seperti: keusangan, masalah sentuhan, getaran dan kebisingan. Pada penggunaan mesin-mesin tersebut diperlukan sarana dan operator yang jumlahnya banyak untuk mencegah timbulnya masalah-masalah tersebut. Dengan menerapkan switch semikonduktor misalnya, maka masalah-masalah akibat sentuhan tersebut dapat diminimalkan sehingga meningkatkan kehandalan. Selain itu, dengan menggunakan komponen-komponen elektronika untuk mengendalikan gerakan, maka komponen-komponen mesin pengendali gerak bisa dikurangi sehingga meningkatkan kehandalan.

3. **Meningkatkan presisi dan kecepatan.** Pada mesin-mesin konvensional (manual) yang sebagian besar menggunakan komponen-komponen mesin sebagai pengendali gerak, tingkat presisi dan kecepatan telah mencapai garis saturasi yang sulit untuk diangkat lagi. Dengan menerapkan kendali digital dan teknologi elektronika, maka tingkat presisi mesin dan kecepatan gerak mesin dapat diangkat lebih tinggi lagi sampai batas tertentu. Batas ini misalnya adalah rigiditas mesin yang menghalangi kecepatan lebih tinggi karena munculnya getaran. Hal ini melahirkan tantangan baru yaitu menciptakan sistem mesin yang memiliki rigiditas lebih tinggi.

Struktur mekatronik dapat dipilah menjadi 2 buah dunia yaitu dunia mekanika dan dunia elektronika. Di dunia mekanika terdapat mekanisme mesin sebagai objek yang dikendalikan. Di dunia elektronika terdapat beberapa elemen mekatronika yaitu: sensor, kontroler, rangkaian penggerak, aktuator dan sumber energi. Elemen-elemen mekatronik dapat dijelaskan sebagai berikut :

Mekanisme mesin. Ini adalah objek kendali yang bisa berupa lengan robot, mekanisme penggerak otomotif, generator pembangkit listrik dan lain sebagainya. **Sensor.** Ini adalah elemen yang bertugas memonitor keadaan objek yang dikendali. Sensor ini dilengkapi dengan rangkaian pengkondisi sinyal yang berfungsi memproses sinyal listrik menjadi sinyal yang mengandung informasi yang bisa dimanfaatkan. **Kontroler.** Ini adalah elemen yang mengambil keputusan apakah keadaan objek kendali telah sesuai dengan nilai referensi yang diinginkan, dan kemudian memproses informasi untuk menetapkan nilai komando guna merevisi keadaan objek kendali. **Rangkaian penggerak.** Ini adalah elemen yang berfungsi menerima sinyal komando dari kontroler dan mengkonversinya menjadi energi yang mampu menggerakkan aktuator untuk melaksanakan komando dari kontroler. Elemen ini selain menerima informasi dari kontroler juga menerima catu daya berenergi tinggi. **Aktuator.** Ini adalah elemen yang berfungsi mengkonversi energi dari energi listrik ke energi mekanik. Bentuk konkrit aktuator ini misalnya: motor listrik, tabung hidrolis, tabung pneumatik, dan lain sebagainya. **Sumber energi.** Ini adalah elemen yang mencatu energi listrik ke semua elemen yang membutuhkannya. Salah satu bentuk konkrit sumber energi adalah baterai untuk sistem yang berpindah tempat, atau adaptor AC-DC untuk sistem yang stasionari (tetap di tempat).

Struktur mekatronik yang digambarkan di sini dari segi teori kendali disebut sistem umpan balik (closed loop). Sistem umpan balik ini menyerupai makhluk hidup, dimana dalam melakukan kegiatan selalu merevisi tindakannya berdasarkan informasi umpan balik yang dikirim oleh indra ke otak. Dengan demikian mekatronik adalah merealisasikan sistem mekanik yang mampu melakukan pekerjaan seperti halnya seorang manusia yang memiliki kondisi yang sempurna.

1.3.2. Gambaran umum tentang mekatronika

Batas formal antara berbagai disiplin ilmu rekayasa (engineering) saat ini semakin kabur seiring dengan perkembangan teknologi IC (Integrated Circuit = rangkaian elektronika terpadu) dan komputer. Hal ini terutama terlihat jelas pada bidang mekanik dan elektronik yaitu semakin banyak produk yang merupakan integrasi dari kedua bidang tersebut, sehingga berkembang suatu bidang yang disebut mekatronika, yang merupakan perluasan cakupan dari bidang elektromekanik.

Beberapa definisi dari yang diambil dari berbagai sumber di antaranya :

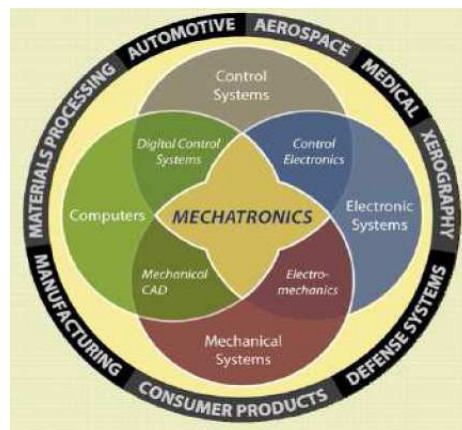
“Integration of microprocessor control system, electrical systems and mechanical system” (Bolton, Mechatronics).

“The synergistic combination of precision mechanical engineering, electronic control and systems thinking in the design of products and manufacturing processes” (Journal of Mechatronics).

“The synergistic use of precision engineering, control theory, computer science, and sensor and actuator technology to design improved products and processes“ (MEMagazine).

“The interdisciplinary field of engineering dealing with the design of products whose function relies on the synergistic integration of mechanical and electronic components coordinated by a control architecture.“ (Alciatore, and Histan)

Dari berbagai pengertian di atas maka dicoba disusun pengertian dari mekatronika yaitu integrasi dari sistem mekanik dan elektronik yang dikendalikan dengan komputer dan dimanfaatkan pada produk maupun proses produksi. Saat ini mekatronika sudah dianggap sebagai suatu bidang tersendiri, meskipun tidak terlepas hubungannya dengan bidang ilmu lainnya. Gambar 1.1 menunjukkan cakupan dari bidang mekatronika beserta bidang-bidang yang memanfaatkan perkembangan bidang mekatronika.



Gambar 1.1. Ilustrasi sistem mekatronika sebagai perpotongan dari system mekanik, elektronik, control dan computer (sumber : Alciatore & Hisdtand, 2003).

Sistem kendali elektronik dan sebagian besar di antaranya menggunakan computersaat ini pengendalian system mekanik hamper seluruhnya dilakukan menggunakan computer. Contohnya adalah mesin mobil, dahulu sistem pembakaran yang terjadi pada silinder dikendalikan sepenuhnya secara mekanis. Banyaknya bahan bakar dan udara diatur langsung dari pedal lewat perantaraan kabel dengan perbandingan yang telah disetel sebelumnya. Katup terbuka dan tertutup diatur secara mekanik menggunakan camshaft tergantung posisi piston.

Saat ini banyak sekali sensor yang terlibat pada sistem pembakaran mobil yaitu di antaranya sensor kecepatan dan posisi poros engkol, sensor temperatur udara dan bahan bakar, dan sensor pada pedal gas. Semua informasi dari sensor tersebut

diolah oleh sistem pengendali berupa komputer yang disebut Engine Control Unit untuk digunakan mengatur waktu dan besarnya bukaan katup serta perbandingan bahan bakar-udara yang dapat disesuaikan dengan kondisi mesin ataupun pengendara.

Pada mobil juga terdapat berbagai sistem lain yang saat ini menerapkan system mekatronika, yaitu sistem transmisi otomatis, sistem suspensi aktif, sistem anti-lock braking system (ABS), sistem pengkondisi udara, serta display kecepatan, putaran mesin dan level bahan bakar. Selain kendaraan bermotor, mekatronika juga diterapkan pada berbagai hal antara lain :

- Perancangan sensor/transduser.
- Peralatan rumah tangga dan perkantoran : mesin cuci, mesin isap debu, timbangan digital, microwave, remote control, pembuat kopi, sistem HVAC, kamera, mesin foto kopi dan masih banyak lagi.
- Berbagai peranti pada komputer : mouse, printer, disk drive, CD ROM drive, keyboard.
- Dunia penerbangan : pengendalian pesawat terbang secara Fly By Wire (FBW).
- Peralatan medis dan laboratorium
- Bidang industri : monitoring dan kendali berbagai peralatan industri
- Bidang robotika.

Komponen utama pada suatu sistem mekatronika adalah sensor, aktuator, dan kontroler. Sensor digunakan untuk mendeteksi variabel pada sistem. Aktuator berfungsi untuk memberikan aksi pada sistem yang dikendalikan.

1.3.3. Aspek pengendalian Elektrik

Sistem kendali mempunyai tiga unsur yaitu : input, proses dan output.



Gambar 1.2. Unsur-unsur sistem kendali

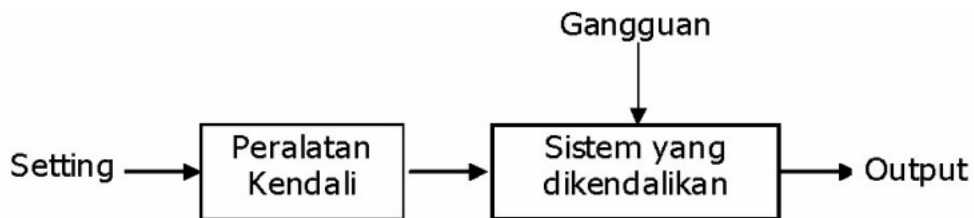
Input pada umumnya berupa sinyal dari sebuah transduser, yaitu alat yang dapat merubah besaran fisik menjadi besaran listrik, misalnya tombol tekan, saklar batas, termostat, dan lain-lain. Transduser memberikan informasi mengenai besaran yang diukur, kemudian informasi ini diproses oleh bagian proses. Bagian proses dapat berupa rangkaian kendali yang menggunakan peralatan yang dirangkai secara listrik, atau juga berupa suatu sistem kendali yang dapat diprogram misalnya sistem berbasis mikroprosesor, mikrokontroler atau PLC.

Pemrosesan informasi (sinyal input) menghasilkan sinyal output yang selanjutnya digunakan untuk mengaktifkan aktuator (peralatan output) yang dapat berupa motor listrik, kontaktor, katup selenoid, lampu, dan sebagainya. Dengan peralatan output, besaran listrik diubah kembali menjadi besaran fisik.

Sistem kendali dibedakan menjadi dua, yaitu sistem kendali loop terbuka dan sistem kendali loop tertutup.

a) Sistem Kendali Loop Terbuka

Sistem kendali loop terbuka adalah proses pengendalian di mana variable input mempengaruhi output yang dihasilkan. Gambar 1.3 menunjukkan diagram blok system kendali loop terbuka.



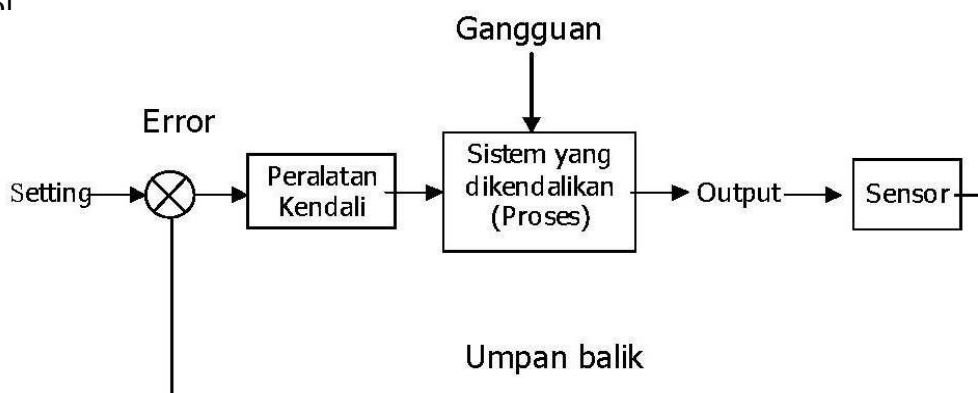
Gambar 1.3. Diagram blok sistem kendali loop terbuka.

Dari gambar 1.3 di atas, dapat dipahami bahwa tidak ada informasi yang diberikan oleh peralatan output kepada bagian proses sehingga tidak diketahui apakah hasil output sesuai dengan yang dikehendaki.

b) Sistem Kendali Loop Tertutup

Sistem kendali loop tertutup adalah suatu proses pengendalian di mana variabel yang dikendalikan (output) disensor secara kontinyu, kemudian dibandingkan dengan besaran acuan.

Variabel yang dikendalikan dapat berupa hasil pengukuran temperatur, kelembaban, posisi mekanik, kecepatan putaran, dan sebagainya. Hasil pengukuran tersebut diumpan-balikkan ke pembanding (komparator) yang dapat berupa peralatan mekanik, listrik, elektronik, atau pneumatik. Pembanding membandingkan sinyal sensor yang berasal dari variabel yang dikendalikan dengan besaran acuan, dan hasilnya berupa sinyal kesalahan. Selanjutnya, sinyal kesalahan diumpankan kepada peralatan kendali dan diproses untuk memperbaiki kesalahan sehingga menghasilkan output sesuai dengan yang dikehendaki. Dengan kata lain, kesalahan sama dengan nol



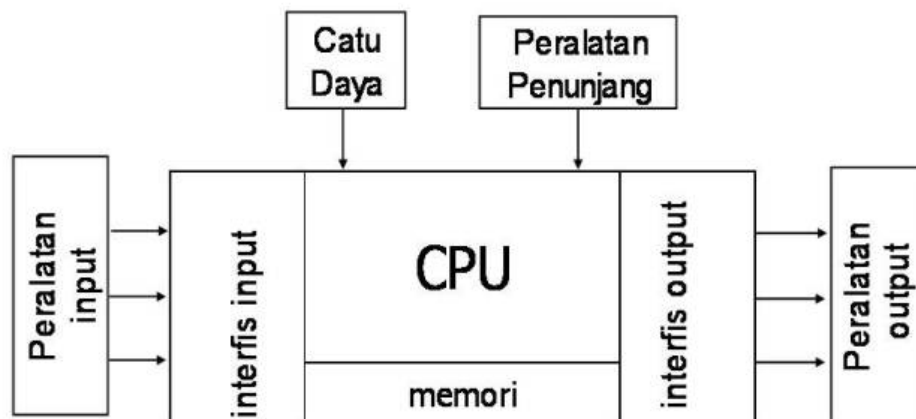
Gambar 1.4. Sistem kendali loop tertutup

Peran Teknologi Digital dalam Pengendalian

Perkembangan Teknologi rangkaian terintegrasi khususnya system mikroprosesor memberikan kontribusi yang signifikan pada bidang kendali digital. Teknologi mikroprosesor telah banyak diterapkan untuk pengendalian berbagai peralatan hingga rumah tangga. Kehadiran mikrokontroler menjadi penerapan pengendali digital relatif mudah diterapkan pada berbagai aplikasi yang portabel.

Hingga akhir tahun 1970, sistem otomasi mesin dikendalikan oleh relai elektromagnet. Dengan semakin meningkatnya perkembangan teknologi, tugas-tugas pengendalian dibuat dalam bentuk pengendalian terprogram yang dapat dilakukan antara lain menggunakan PLC (Programmable Logic Controller). Dengan PLC, sinyal dari berbagai peralatan luar diinterfisi sehingga fleksibel dalam mewujudkan sistem kendali. Disamping itu, kemampuannya dalam komunikasi jaringan memungkinkan penerapan yang luas dalam berbagai operasi pengendalian sistem. Dalam sistem otomasi, PLC merupakan 'Jantung' sistem kendali. Dengan program yang disimpan dalam memori PLC, dalam eksekusinya, PLC dapat memonitor keadaan sistem melalui sinyal dari peralatan input, kemudian didasarkan atas logika program menentukan rangkaian aksi pengendalian peralatan output luar.

PLC dapat digunakan untuk mengendalikan tugas-tugas sederhana yang berulang-ulang, atau di-interkoneksi dengan yang lain menggunakan komputer melalui sejenis jaringan komunikasi untuk mengintegrasikan pengendalian proses yang kompleks. Cara kerja sistem kendali PLC dapat dipahami dengan diagram blok seperti ditunjukkan pada Gambar 1.5.



Gambar 1.5. Diagram blok PLC

1.3.4. Aspek Mekanika

Otomotif adalah ilmu yang mempelajari tentang alat-alat transportasi darat yang menggunakan mesin, terutama mobil dan sepeda motor. Otomotif mulai berkembang sebagai cabang ilmu seiring dengan diciptakannya mesin mobil. Dalam perkembangannya, mobil menjadi alat transportasi kompleks yang terdiri dari ribuan komponen yang tergolong dalam puluhan sistem dan subsistem.

Pompa bahan bakar adalah komponen penting dalam sebuah mobil atau mesin kombusi dalam lainnya. Bahan bakar harus dipompa dari tangki bensin ke mesin dan diantar dalam tekanan rendah ke karburator atau dalam tekanan tinggi ke sistem injeksi bahan bakar. Beberapa mesin injeksi bahan bakar memiliki 2 macam

pompa untuk tujuan ini: satu pompa tekanan rendah/volume besar di tangki dan satu tekanan tinggi/volume rendah di atau dekat mesin.

Suspensi adalah kumpulan komponen tertentu yang berfungsi meredam kejutan, getaran yang terjadi pada kendaraan akibat permukaan jalan yang tidak rata yang dapat meningkatkan kenyamanan berkendara dan pengendalian kendaraan. Ada dua jenis utama suspensi yaitu sistem suspensi dependen dan sistem suspensi independen.

Peredam kejut, shock absorber, shock breaker, atau damper adalah sebuah alat mekanik yang didesain untuk meredam hentakan yang disebabkan oleh energi kinetik. Peredam kejut adalah bagian penting dalam suspensi kendaraan bermotor, roda pendaratan pesawat terbang, dan mendukung banyak mesin industri. Peredam kejut berukuran besar juga digunakan dalam arsitektur dan teknik sipil untuk mengurangi kelemahan struktur akibat gempa bumi dan resonansi. Dalam kendaraan, alat ini berfungsi untuk mengurangi efek dari kasarnya permukaan jalan. Tanpa peredam kejut, kendaraan dapat terlempar, seperti energi yang disimpan dalam per/pegas, kemudian dilepaskan pada kendaraan, barangkali melebihi gerakan suspensi. Kontrol gerakan berlebih pada suspensi tanpa peredam kejut diredam secara paksa oleh per yang kaku, yang dapat menyebabkan ketidaknyamanan dalam berkendara.

Peredam kejut diperkenankan menggunakan per yang lembut yang mengontrol gerakan suspensi dalam merespon gundukan atau lubang. Dan juga, berhubungan dengan pelambatan efek fisik dalam ban itu sendiri, mengurangi gerakan naik turun per. Karena ban tidak selembut per, untuk meredam hentakan ban mungkin dibutuhkan shock yang kaku yang lebih ideal untuk kendaraan.

Peredam kejut pneumatik dan hidraulik umumnya mengambil bentuk sebuah silinder dengan piston yang bergerak di dalamnya. Silinder harus diisi dengan cairan kental, seperti minyak hidraulik atau udara. Cairan ini diisikan ke dalam dashpot. Peredam kejut berbasis per umumnya menggunakan per keong atau per daun. Per ideal itu sendiri, bukanlah peredam kejut seperti per yang hanya menyimpan dan tidak menghilangkan atau menyerap energi. Kendaraan biasanya menggunakan dua per atau palang torsi yang berfungsi sebagaimana peredam kejut hidraulik. Dalam kombinasi ini, peredam kejut secara khusus menyediakan piston hidraulik yang menyerap dan menghilangkan getaran. Per tidak dianggap sebagai peredam kejut.

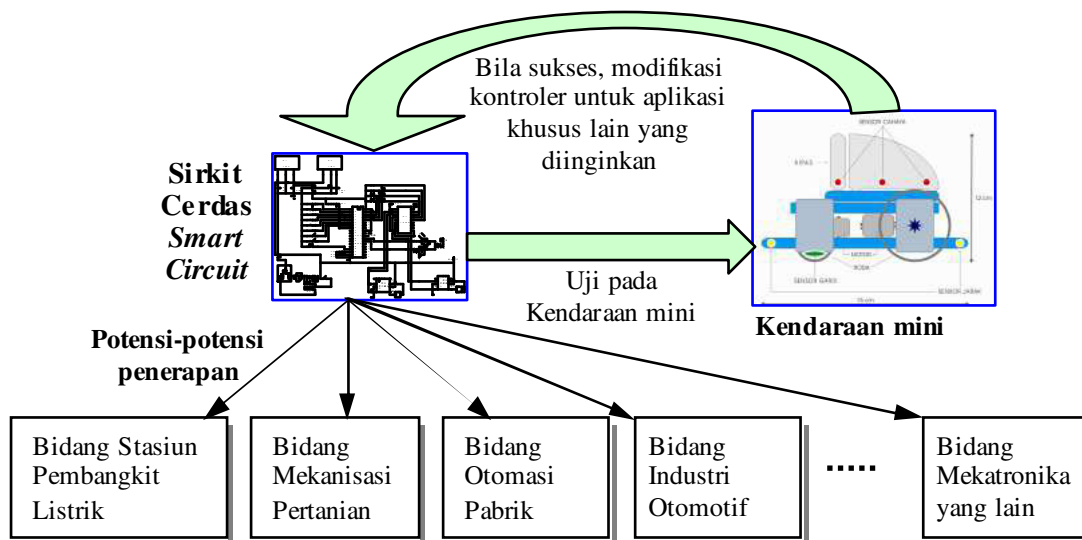
Peredam kejut harus menyerap atau menghilangkan energi. Desainnya harus dipertimbangkan, oleh karena itu harus dibuat ketika mendesain atau memilih sebuah peredam kejut adalah ke mana energi akan pergi. Umumnya, dalam kebanyakan dashpot, energi diubah ke dalam panas di dalam cairan kental. Dalam silinder hidraulik, minyak hidraulik akan memanas. Dalam silinder udara, udara panas selalu dilepaskan ke atmosfer. Dalam tipe dashpot yang lain, seperti elektromagnetik, energi yang hilang dapat disimpan dan bisa digunakan kemudian jika diperlukan.

Sistem transmisi, dalam otomotif, adalah sistem yang menjadi penghantar energi dari mesin ke diferensial dan as. Dengan memutar as, roda dapat berputar dan menggerakkan mobil. Transmisi diperlukan karena mesin pembakaran yang umumnya digunakan dalam mobil merupakan mesin pembakaran internal yang menghasilkan putaran (rotasi) antara 600 sampai 6000 rpm. Sedangkan, roda berputar pada kecepatan rotasi antara 0 sampai 2500 rpm.

Sekarang ini, terdapat dua sistem transmisi yang umum, yaitu manual dan otomatis. Terdapat juga sistem-sistem transmisi yang merupakan gabungan antara kedua sistem tersebut, namun ini merupakan perkembangan terakhir yang baru dapat ditemukan pada mobil-mobil berteknologi tinggi. Transmisi manual merupakan sistem transmisi yang banyak dipergunakan dengan alasan perawatan yang lebih mudah dan biasanya terdiri dari 3 sampai dengan 7 speed.

Transmisi otomatis terdiri dari 3 bagian utama, yaitu : Torque converter, Planetary gear unit, dan Hydraulic control unit. Torque converter berfungsi sebagai kopling otomatis dan dapat memperbesar momen mesin. Sedangkan Torque converter terdiri dari Pump impeller, Turbine runner, dan Stator. Stator terletak diantara impeller dan turbine. Torque converter diisi dengan ATF (Automatic Transmission Fluid). Momen mesin dipindahkan dengan adanya aliran fluida.

Begitu banyaknya penggunaan sistem mekatronika dalam kehidupan kita memperkuat salah satu sifatnya yang multiguna (aplikatif). Sebagai contoh sistem mekatronika pada kendaraan bermotor adalah sistem rem ABS (Anti-lock Breaking system) atau sistem pengereman yang menghindari terkuncinya roda sehingga mobil tetap dapat dikendalikan dalam pengereman mendadak, ESP (Elektronik Stability Programm), ABC (Active Body Control) dan Motor-Management-System. Contoh pada Teknologi Penerbangan : Dalam teknologi penerbangan moderen digunakan Comfort-In-Turbulence System sehingga dapat meningkatkan kenyamanan penumpang walau ketika terjadi turbulensi. Gust Load Alleviation serta banyak contoh lainnya. Pada Teknik Produksi : Contoh dalam teknik produksi adalah penggunaan sensor pada robot. Sistem kendali umpan balik pada elektromotor berkecepatan rotasi tinggi dengan ‘pemegang as’ tenaga magnet.



Gambar 1.6. Potensi-potensi penerapan Sistem Cerdas dalam bidang mekatronika.

1.4 Penutup

1.4.1. Bahan diskusi dan tugas

1. Apakah perbedaan sistem kendali loop terbuka dengan loop tertutup?
2. Apakah sesungguhnya a PLC itu?
3. Sebutkan masing-masing 3 contoh :

- a. Alat input
- b. Alat output
- c. Alat penunjang

1.5 Daftar pustaka

1. J. Webb, *Industrial Control Electronics*, Macmillan Publish,. New York, 1992
2. J. Webb, *Programmable Logic Control*, Macmillan Publish,. New York, 1992
3. Frank D Petruzella, *Elektronik Industri* (terjemahan), Penerbit Andi, Yogyakarta, 2001
4. William Bolton, *Programmable Logic Controller (PLC)*, Sebuah Penngantar, Edisi ketiga, Penerbit Erlan gga, 2004
5. M. Budiyanto dan A. Wijaya, *Pengenalan Dasar-dasar PLC (Programmable Logic Controller)*, Penerbit Gaya Media, Yogyakarta, 2003
6. Endra Ditawarna, *Robotika, Desain, Kontrol dan Kecerdasan Buatan*, Penerbit Andi, Yogyakarta, 2006
7. David G. Alciator dan Michael B Histan d, *Introduction to Mechatronics and Measurement Systems*, Mc Graw Hill, New York, 2003

1.6. Senarai

Pada mobil juga terdapat berbagai sistem lain yang saat ini menerapkan sistem mekatronika, yaitu sistem transmisi otomatis, sistem suspensi aktif, sistem anti-lock braking system (ABS), sistem pengkondisi udara, serta display kecepatan, putaran mesin dan level bahan bakar.

Selain pada kendaraan bermotor, mekatronika juga diterapkan pada berbagai hal antara lain :

- perancangan sensor/transduser.
- peralatan rumah tangga dan perkantoran : mesin cuci, mesin isap debu, timbangan digital, microwave, remote control, pembuat kopi, sistem HVAC, kamera, mesin foto kopi dan masih banyak lagi.
- berbagai peranti pada komputer : mouse, printer, disk drive, CD ROM drive, keyboard.
- dunia penerbangan : pengendalian pesawat terbang secara Fly By Wir (FBW)
- peralatan medis dan laboratorium.
- bidang industri : monitoring dan kendali berbagai peralatan industri.
- bidang robotika.

BAB 2

SENSOR

2.1. Deskripsi Singkat

Tujuan dari mempelajari sensor adalah untuk dapat menerapkan dalam kehidupan sehari-hari terutama bidang industri dalam hal monitoring, controlling dan proteksi sistem.

2.2. Tujuan Instruksional Khusus

Setelah materi ini diajarkan mahasiswa dapat mengetahui tentang sensor dan mampu menjelaskan jenis-jenis sensor.

2.3. 2.3 Penyajian

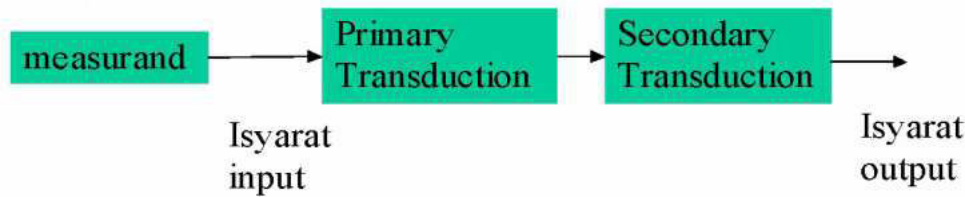
2.3.1. Pengantar

Sensor adalah piranti yang mentransformasikan (mengubah) suatu nilai (isyarat/energi) fisik ke nilai fisik yang lain serta menghubungkan antara fisik nyata dan industri listrik dan piranti elektronika. Dalam dunia industri berguna untuk monitoring, controlling, dan proteksi. Sering disebut juga dengan transducer. Sensor umumnya dikategorikan menurut apa yang diukur dan sangat berperan penting dalam proses pengendali manufaktur modern.

Ada 6 tipe isyarat Sensor :

- Mechanical, contoh: panjang, luas, mass flow, gaya, torque, tekanan, kecepatan, percepatan, panjang gel acoustic.
- Thermal, contoh: temperature, panas, entropy, heat flow
- Electrical, contoh: tegangan, arus, muatan, resistance, frekuensi.
- Magnetic, contoh: intensitas medan, flux density.
- Radiant, contoh: intensitas, panjang gelombang, polarisasi.
- Chemical, contoh: komposisi, konsentrasi, pH, kecepatan reaksi.

Sensor mengkonversi dari suatu isyarat input ke suatu isyarat output. Sensor bisa saja menggunakan satu atau lebih pengkonversian untuk menghasilkan suatu isyarat keluaran.



Gambar 2.1. Diagram Block Sensor

2.3.2. Sensor Cahaya

Sensor cahaya adalah alat yang digunakan untuk merubah besaran cahaya menjadi besaran listrik. Prinsip kerja dari alat ini adalah mengubah energi dari foton menjadi elektron. Idealnya satu foton dapat membangkitkan satu elektron. Sensor cahaya sangat luas penggunaannya, salah satu yang paling populer adalah kamera digital. Pada saat ini sudah ada alat yang digunakan untuk mengukur cahaya yang mempunyai 1 buah photon saja. Di bawah ini adalah jenis-jenis sensor cahaya, di antaranya :

- Detektor kimiawi, seperti pelat fotografis, dimana mmolekul silver halida dibagi menjadi sebuah atom perak metalik dan atom halogen. Pengembangan fotografis menyebabkan terbaginya molekul yang berdekatan secara sama.
- Fotoresistor atau Light Dependent Resistor (LDR) yang berubah resistansinya ketika dikenai cahaya.
- Sel fotovoltaik atau sel matahari yang menghasilkan tegangan dan memberikan arus listrik ketika dikenai cahaya.
- Fotodiode yang dapat beroperasi pada mode fotovoltaik maupun fotokonduktif
- Tabung fotomultiplier yang mengandung fotokatoda yang memancarkan elektron ketika dikenai cahaya, kemudian elektron-elektron tersebut akan dikuatkan dengan rantai dynode.
- Tabung cahaya yang mengandung fotokatoda yang memancarkan elektron ketika dikenai cahaya, dan umumnya bersifat sebagai fotoresistor.
- Fototransistor menggabungkan salahsatu dari metode penyensoran di atas.
- Detektor optis yang berlaku seperti termometer, secara murni tanggap terhadap pengaruh panas dari radiasi yang masuk, seperti detektor piroelektrik, sel Golay, termokopel dan termistor, tapi kedua yang terakhir kurang sensitif.
- Detektor cryogenic cuku tanggap untuk mengukur energi dari sinar-x tunggal, serta foton cahaya terlihat dan dekat dengan inframerah (Enss 2005).

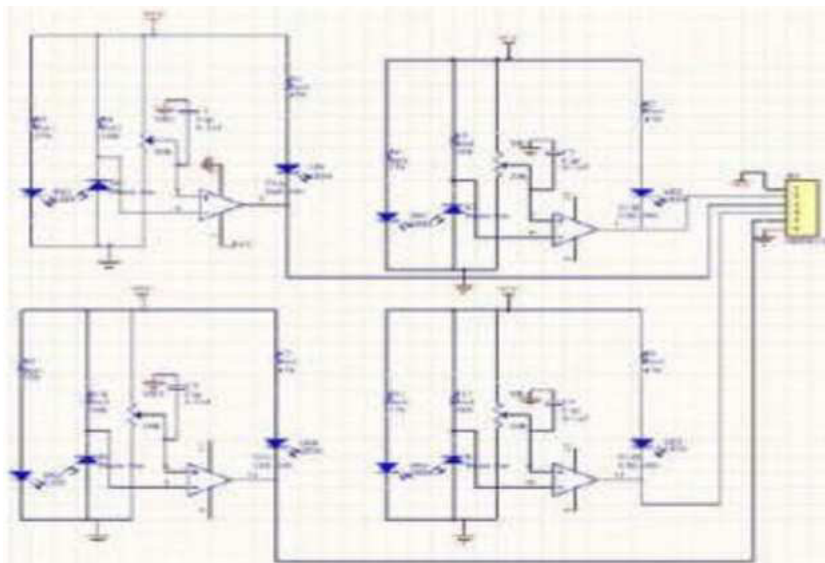
2.3.3. Sensor suhu

Sensor Suhu adalah alat yang digunakan untuk merubah besaran panas menjadi besaran listrik yang dapat dengan mudah dianalisis besarnya. Ada beberapa metode yang digunakan untuk membuat sensor ini, salah satunya dengan cara menggunakan material yang berubah hambatannya terhadap arus listrik sesuai dengan suhunya. Menggunakan bahan logam Logam akan bertambah besa hambatannya terhadap arus listrik jika panasnya bertambah. Hal ini dapat dijelaskan dari sisi komponen penyusun logam. Logam dapat dikatakan sebagai muatan positif yang

berada di dalam elektron yang bergerak bebas. Jika suhu bertambah, elektron-elektron tersebut akan bergetar dan getarannya semakin besar seiring dengan naiknya suhu. Dengan besarnya getaran tersebut, maka gerakan elektron akan terhambat dan menyebabkan nilai hambatan dari logam tersebut bertambah. Menggunakan bahan semikonduktor Bahan semikonduktor mempunyai sifat terbalik dari logam, semakin besar suhu, nilai hambatan akan semakin turun. Hal ini dikarenakan pada suhu yang semakin tinggi, elektron dari semikonduktor akan berpindah ke tingkat yang paling atas dan dapat bergerak dengan bebas. Seiring dengan kenaikan suhu, semakin banyak elektron dari semikonduktor tersebut yang bergerak bebas, sehingga nilai hambatan tersebut berkurang. Untuk mendapatkan sinyal listrik yang baik dengan sedikit kegaduhan, dapat digunakan jembatan Wheatstone dan rangkaian Lock in Amplifier.

2.3.4. Sensor proximity

Merupakan sensor yang mendeteksi keberadaan dari suatu objek tanpa melakukan kontak fisik. Sensor proximity adalah sensor untuk mendeteksi ada atau tidaknya suatu obyek. Dalam dunia robotika, sensor proximity seringkali digunakan untuk mendeteksi ada atau tidaknya suatu garis pembimbing gerak robot atau lebih dikenal dengan istilah “Line Follower Robot” atau “Line Tracer Robot”, juga biasa digunakan untuk mendeteksi penghalang berupa dinding atau penghalang lain pada Robot A voider. Jenis sensor proximity meliputi limit switch (saklar mekanik), ultrasonic proximity, proximity (infra merah), kamera dan lain sebagainya.

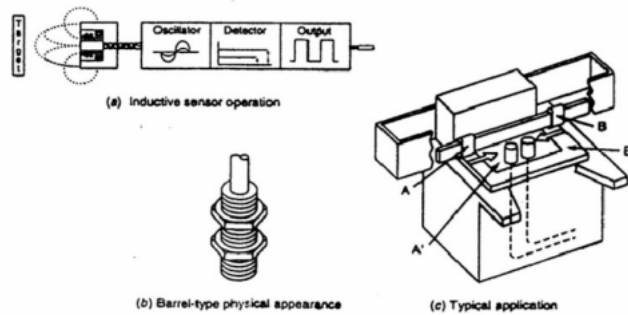


Gambar 2.2. Sensor proximity

Sensor proximity yang digunakan untuk line follower robot dibuat menggunakan pasangan LED/infrared dan photo transistor.

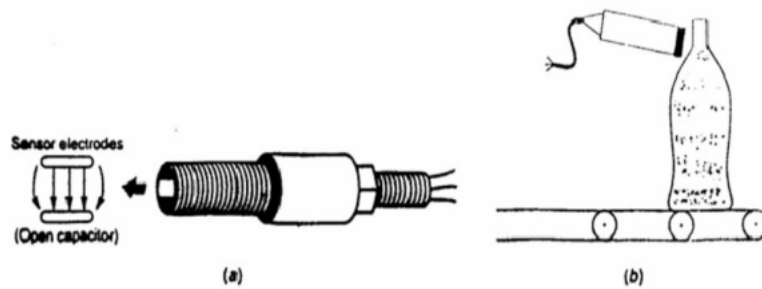
- Lingkungan yang berbahaya, dimana sistem terproteksi dengan baik, dan tidak diharapkan adanya kontak mekanik.
- Jika dibutuhkan saklar yang panjang umur dan andal
- Sistem pengendali elektronik cepat yang membutuhkan signal masukan bebas pantulan.

Inductive Proximity Sensor adalah peralatan sensor yang diaktifkan oleh objek logam. Inductive. Sensor ini dapat diaktifkan dalam posisi aksial ataupun radial.



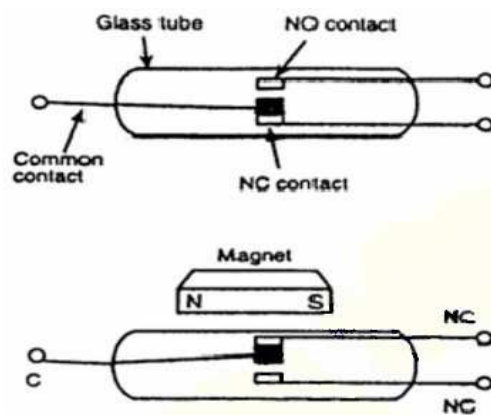
Gambar 2.3. Inductive Proximity Sensor

Capacitive Proximity Sensor adalah sensor yang diaktifkan oleh material konduktif ataupun non- konduktif, seperti kayu, plastik, cairan, gula, tepung, ataupun gandum.



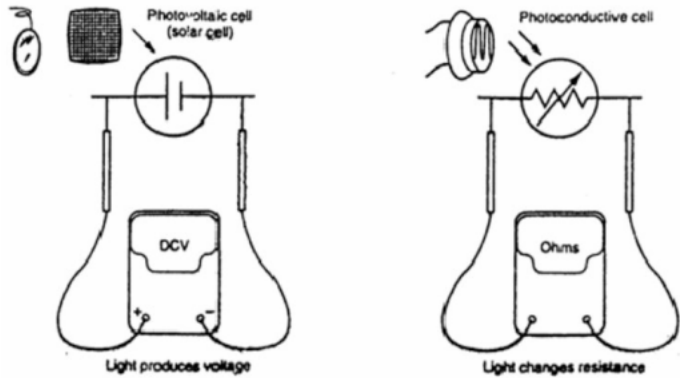
Gambar 2.4. Capacitive Proximity Sensor

Magnetic switch (reed relay) Sensor ini terdiri dari dua plat kontak yang terproteksi penuh dalam ruang kaca yang mempunyai gas proteksi. Kontak akan berpindah ketika pengaruh medan magnet dilingkungan sekitarnya secara kuat menariknya.



Gambar 2.5. Magnetic switch

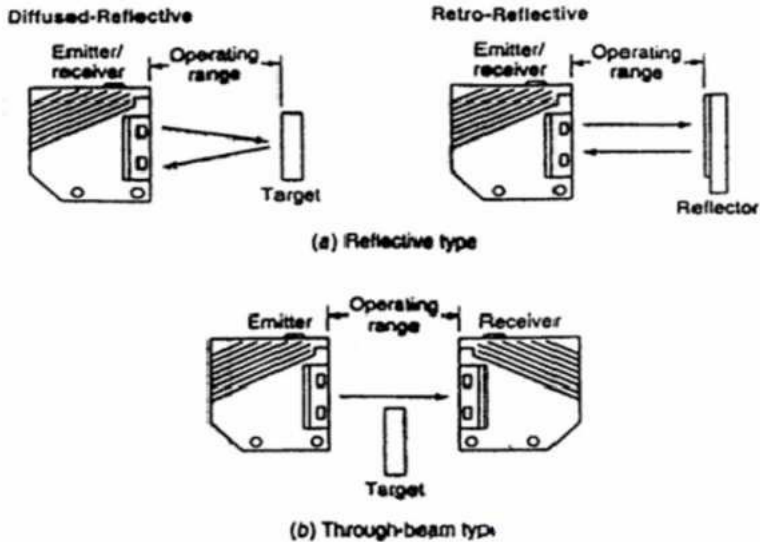
- *Photoconductive (photoresistive) cell* merupakan salah satu transducers cahaya yang sering dipergunakan. Energi cahaya yang jatuh pada sel photoconductive akan mengakibatkan perubahan resistansi pada sel, jika cahaya sedang gelap, maka resistansi akan tinggi, demikian juga sebaliknya.



Gambar 2.6. Pengukuran *photoconductive*

Terdapat dua metode penerapan sensor photoelektrik yaitu :

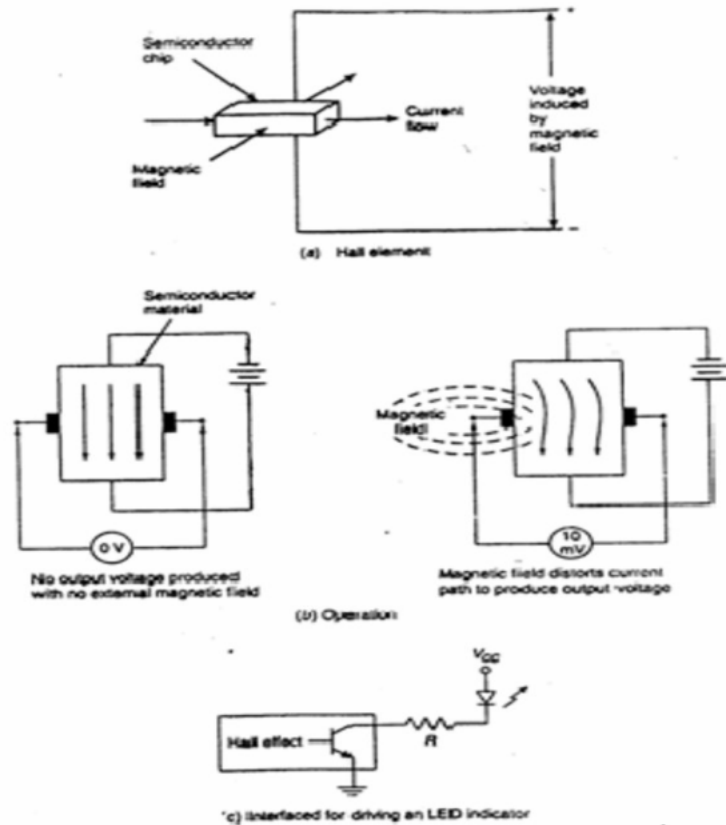
- Reflektive-type photoelectric sensor, dipergunakan untuk mendeteksi cahaya yang dipantulan dari obyek.
- Through-beam photoelectric sensor, dipergunakan untuk mengukur perubahan kuantitas cahaya yang diakibatkan dari adanya obyek yang melewati aksis optiknya.



Gambar 2.7. Mode penerapan sensor photoelektrik

Hall-Effect Sensor

Sensor ini didesain untuk mendeteksi obyek magnetik, biasanya digunakan untuk mensinyalir posisi dari komponen.

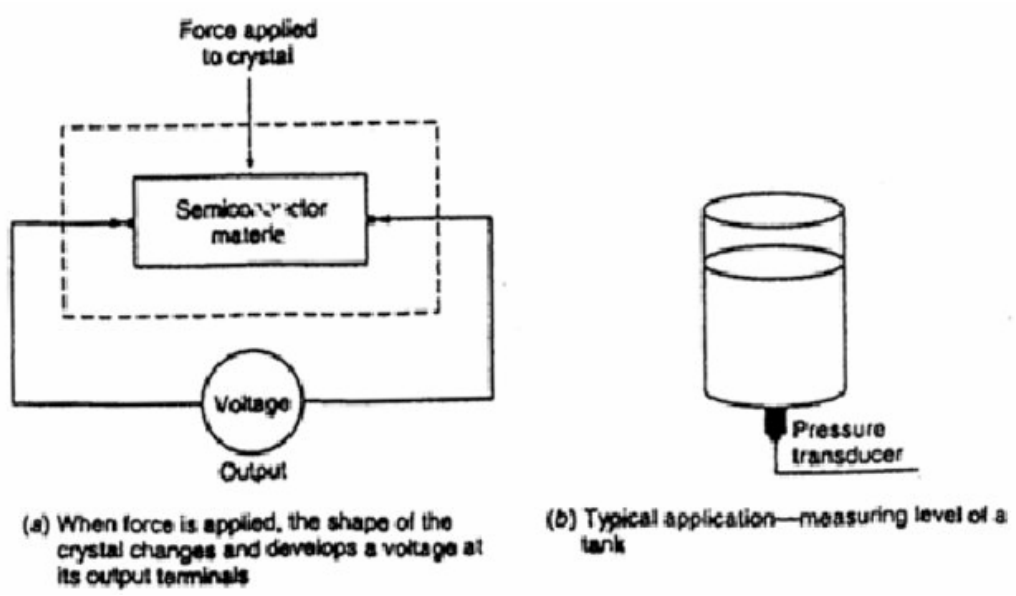


Gambar 2.8. Prinsip Kerja Hall Effect Sensor

2.3.5. Pressure sensors

Strain Wire Gauge Transducer akan mengkonversi tegangan mekanik menjadi signal elektrik.

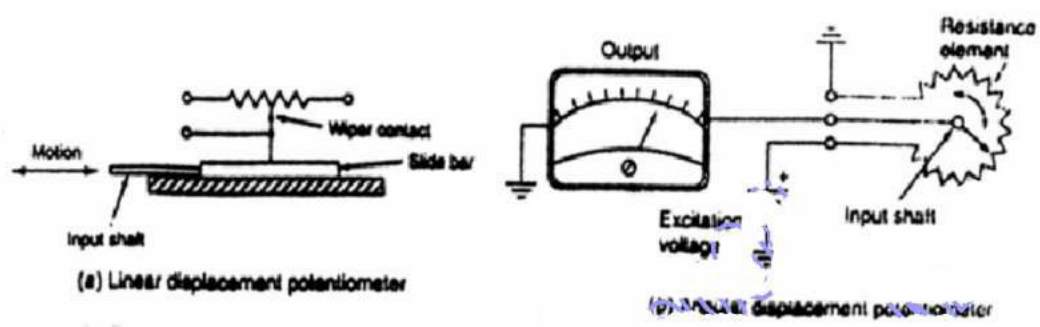
- Prinsip yang diterapkan adalah mengikuti prinsip bahwa konduktor akan memiliki resistansi yang berbeda pada sisi panjang ataupun mendatarnya.
- Gaya yang dikenakan pada bidang ukur akan menyebabkan bidang membengkok, bengkokan ini akan mendistorsi ukuran bidang, dan akan terjadi perubahan resistansi.
- Perubahan resistansi ini akan disinyalir oleh sirkuit yang ada. Strain Gauge Load Cells biasanya terbuat dari baja dan bidang ukur tegangan yang sensitif.
- Semiconductor strain gauge
 - Menggunakan piezoelectric crystal sebagai elemen pengindraannya. Jika kristal diberi gaya, maka bentuk kristal akan berubah dan menghasilkan tegangan pada terminal keluaran dari kristal, jenis strain gauge ini memiliki sensitivitas yang lebih tinggi dibandingkan dengan strain wire gauge



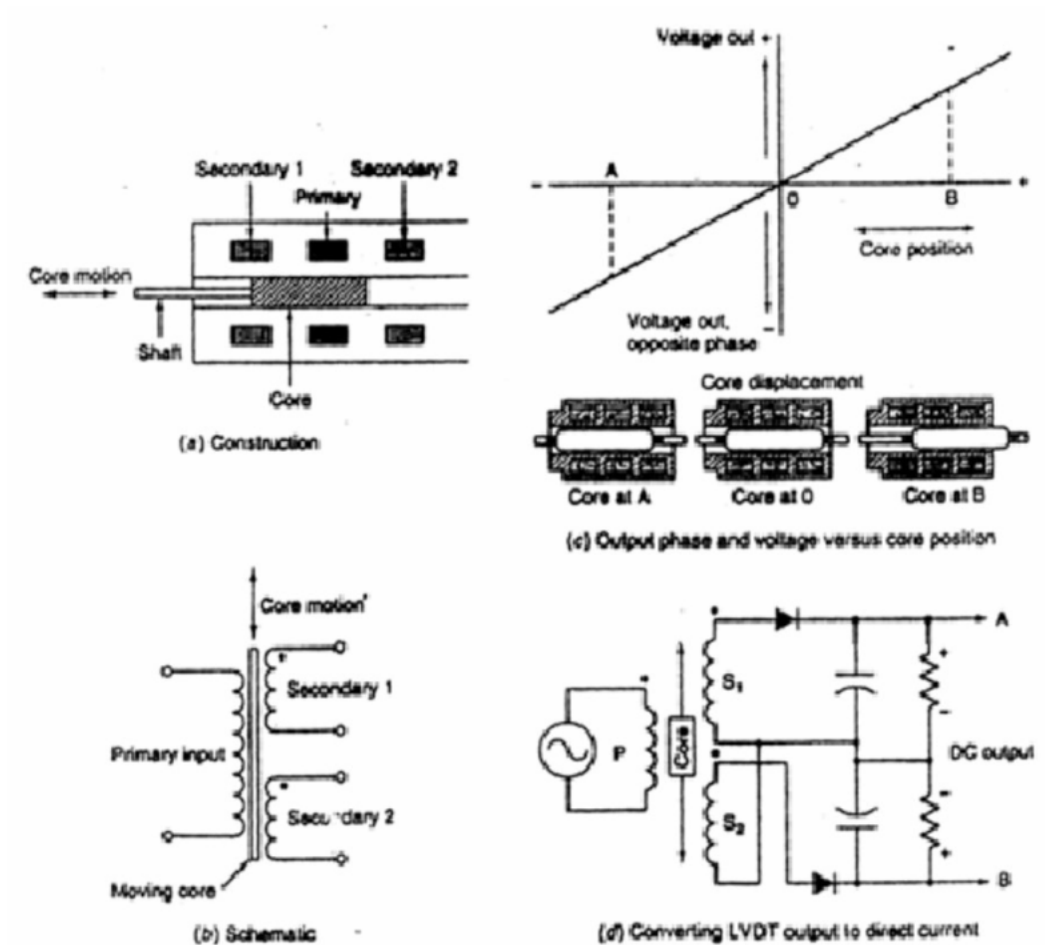
Gambar 2.9. Semikonduktor strain gate

2.3.6. Displacement Transducer

Displacement Transducers yang paling umum dipakai industri adalah LVDT (Linear Variable Differential Transformer). LVDT ini digunakan untuk mengendalikan level air di dalam tangki.



Gambar 2.10. Displacement Transducer menggunakan potensiometer



Gambar 2.11. Sensor LVDT

2.3.7. Sensor Putaran/Velocity (RPM Sensor)

Sensor ini dikenal dengan nama Tachometer. Tachometer biasanya merupakan magnet permanen DC generator kecil. Jika generator berotasi, akan menghasilkan tegangan DC yang proporsional langsung terhadap kecepatan. Tachometer seringkali dipasang ke motor untuk mengindikasikan putaran sebagai masukan pengendali (Controller). Dewasa ini dikembangkan pengukur kecepatan sistem digital menggunakan piringan bercelah yang disambung para poros motor. Putaran celah yang disensor menggunakan sensor cahaya akan menghasilkan pulsa yang dapat diproses lebih lanjut oleh pengolah digital.

2.4 Penutup

2.4.1 Bahan diskusi dan tugas

1. Sebutkan isyarat sensor yang anda ketahui?
2. Sebutkan dan jelaskan lima jenis sensor?
3. Berikan tiga contoh penggunaan sensor pada sistem peralatan?

2.5 Daftar Pustaka

8. J. Webb, *Industrial Control Electronics*, Macmillan Publish,. New York, 1992
9. J. Webb, *Programmable Logic Control*, Macmillan Publish,. New York, 1992
10. Frank D Petruzella, *Elektronik Industri* (terjemahan), Penerbit Andi, Yogyakarta, 2001
11. William Bolton, *Programmable Logic Controller (PLC)*, Sebuah Pennantar, Edisi ketiga, Penerbit Erlangga, 2004
12. M. Budiyanto dan A. Wijaya, *Pengenalan Dasar-dasar PLC (Programmable Logic Controller)*, Penerbit Gaya Media, Yogyakarta, 2003
13. Endra Ditawarna, *Robotika, Desain, Kontrol dan Kecerdasan Buatan*, Penerbit Andi, Yogyakarta, 2006
14. David G. Alciator dan Michael B Histan, *Introduction to Mechatronics and Measurement Systems*, Mc Graw Hill, New York, 2003

2.6. Senarai

Sensor adalah alat untuk mendeteksi adanya perubahan fisik pada suatu objek dan merubahnya menjadi Signal yang dapat diukur atau disimpan. Transducer adalah alat untuk mentransfer Signal dari satu sistem ke sistem lainnya baik dalam bentuk Signal yang sama atau bentuk Signal lainnya. Sensor merupakan alat perabanya sedangkan Transducer merupakan Sensor plus sirkit yang digunakannya. Ada sebagian Sensor yang juga menghasilkan Signal dan berperilaku sebagai Transducer. Semua Transducer mengandung Sensor, dan tidak semua Sensor berfungsi sebagai Transducer. Signal merupakan istilah yang dipakai menggambarkan adanya satu bentuk energi yang berubah sebagai fungsi dari ruang dan waktu.

Ada beberapa media kerja (bentuk energi) yaitu elektrik, hidrolik dan pneumatik. Persaingan antara peralatan pneumatik dengan peralatan mekanik, hidrolik atau elektrik makin menjadi besar. Dalam penggunaannya sistem pneumatik diutamakan karena beberapa hal yaitu : paling banyak dipertimbangkan untuk beberapa mekanisasi, dapat bertahan lebih baik terhadap keadaan-keadaan tertentu. Seringkali suatu proses tertentu dengan cara pneumatik, berjalan lebih rapi (efisien) dibandingkan dengan cara lainnya.

BAB 3

AKTUATOR ELEKTRIK

3.1. Deskripsi Singkat

Tujuan dari mempelajari aktuator elektrik adalah untuk dapat memahami dan implementasi setiap alat yang mengubah energi listrik menjadi gerakan mekanis.

3.2. Tujuan Intruksional Khusus

Setelah materi ini diajarkan mahasiswa dapat mengetahui tentang aktuator dan mampu menjelaskan bagian-bagian aktuator serta penggunaannya.

3.3. Penyajian

3.3.1. Pengantar

Penggerak, dalam pengertian listrik adalah setiap alat yang mengubah energi listrik menjadi gerakan mekanis. Biasa digunakan sebagai proses lanjutan dari keluaran suatu proses olah data yang dihasilkan oleh suatu sensor atau kontroler. Aktuator Adalah perangkat elektromekanik yang menghasilkan daya gerakan. Dapat dibuat dari sistem motor listrik/motor DC (permanent magnet, brushless, motor DC servo, motor DC stepper, solenoid, dsb.), sistem pneumatic (perangkat kompresi berbasis udara atau gas nitrogen), dan perangkat hidrolis (berbasis bahan cair seperti oli). Untuk meningkatkan tenaga mekanik aktuator atau torsi gerakan dapat dipasang sistem gearbox, baik sistem direct-gear (sistem lurus, sistem ohmic/worm-gear dsb.), sprocket-chain (gir-rantai, gir-belt, ataupun sistem wire-roller, dsb.)

Beberapa jenis aktuator berdasarkan kategori tenaganya antara lain :

- Aktuator tenaga listrik, biasanya digunakan solenoid, motor arus searah. sifat mudah diatur dengan torsi kecil sampai sedang.
- Aktuator tenaga hidrolis. torsi yang besar konstruksinya sukar.
- Aktuator tenaga pneumatik. sukar dikendalikan
- Aktuator Lainnya : piezoelectric, magnetic, ultra sound, SMA

Keunggulan Aktuator Elektrik :

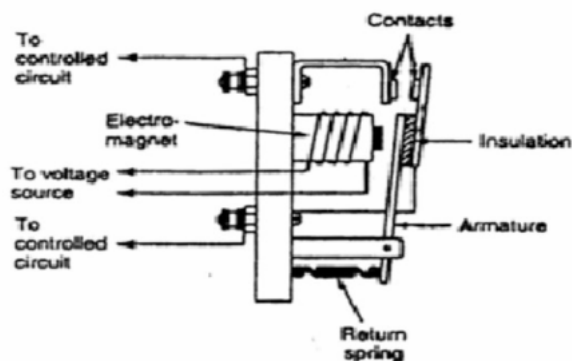
- Mudah dalam pengontrolan
- Mulai dari mW sampai MW
- Berkecepatan tinggi 1000 - 10000 rpm
- Banyak macamnya
- A kurasi tinggi
- Torsi ideal untuk pergerakan
- Efisiensi tinggi



Gambar 3.1. Aktuator Elektrik

3.3.2. Relay

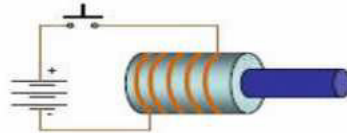
Relay adalah peralatan yang dioperasikan secara elektrik yang secara mekanik akan men-switch sirkuit elektrik. Relay merupakan bagian yang penting dalam sistem kontrol, karena kegunaannya dalam kendali jarak jauh, dan mengendalikan listrik tegangan tinggi dengan menggunakan listrik tegangan rendah. Ketika tegangan mengalir ke dalam elektromagnet pada sistem kontrol relay, maka magnet akan menarik lengan logam pada arah magnet, dengan demikian kontak terjadi. Relay bisa memiliki jenis NO atau NC ataupun dua-duanya.



Gambar 3.2. Relay

3.3.3. Solenoida

Solenoida adalah alat yang digunakan untuk mengubah sinyal listrik atau arus listrik menjadi gerakan mekanis linear. Terbentuk dari kumparan dengan inti besi yang dapat bergerak, besarnya gaya tarikan atau dorongan yang dihasilkan adalah ditentukan dengan jumlah lilitan kumparan tembaga dan besar arus yang mengalir melalui kumparan



Gambar 3.3. Prinsip Kerja Solenoid

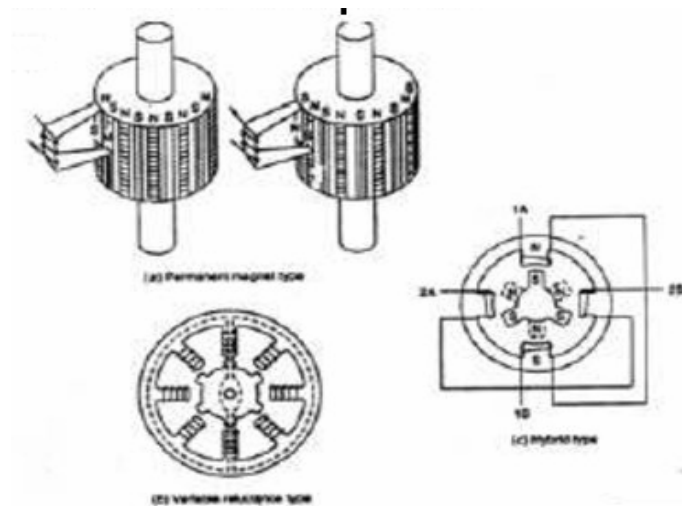
Aspek penting pada solenoid adalah sentakan. Sentakan kecil akan dihasilkan tingkat operasi yang tinggi, dan daya yang dibutuhkan juga lebih sedikit.



Gambar 3.4. Contoh Solenoid

3.3.4. Motor Stepper

Stepper adalah alat yang mengubah pulsa listrik yang diberikan menjadi gerakan rotor diskrit (tidak kontinyu) yang disebut step (langkah). Satu putaran motor memerlukan 360 derajat dengan jumlah langkah yang tertentu perderajatnya. Ukuran kerja dari stepper biasanya diberikan dalam jumlah langkah per-putaran per-detik. Motor stepper mempunyai kecepatan dan torsi yang rendah namun memiliki kontrol gerakan posisi yang cermat, hal ini dikarenakan memiliki beberapa segmen kutub kumparan.



Gambar 3.5. Stepper Motor

Motor stepper banyak digunakan untuk aplikasi-aplikasi yang biasanya cukup menggunakan torsi yang kecil, seperti untuk penggerak piringan disket atau piringan CD. Motor stepper merupakan motor DC yang tidak memiliki komutator. Pada umumnya motor stepper hanya mempunyai kumparan pada statornya sedangkan pada bagian rotornya merupakan magnet permanen. Dengan model motor seperti ini maka motor stepper dapat diatur posisinya pada posisi tertentu dan/atau berputar ke arah yang diinginkan, searah jarum jam atau sebaliknya. Kecepatan motor stepper pada dasarnya ditentukan oleh kecepatan pemberian data pada komutatornya. Semakin cepat data yang diberikan maka motor stepper akan semakin cepat pula berputarnya.



Gambar 3.6. Stepper motor 4 phase

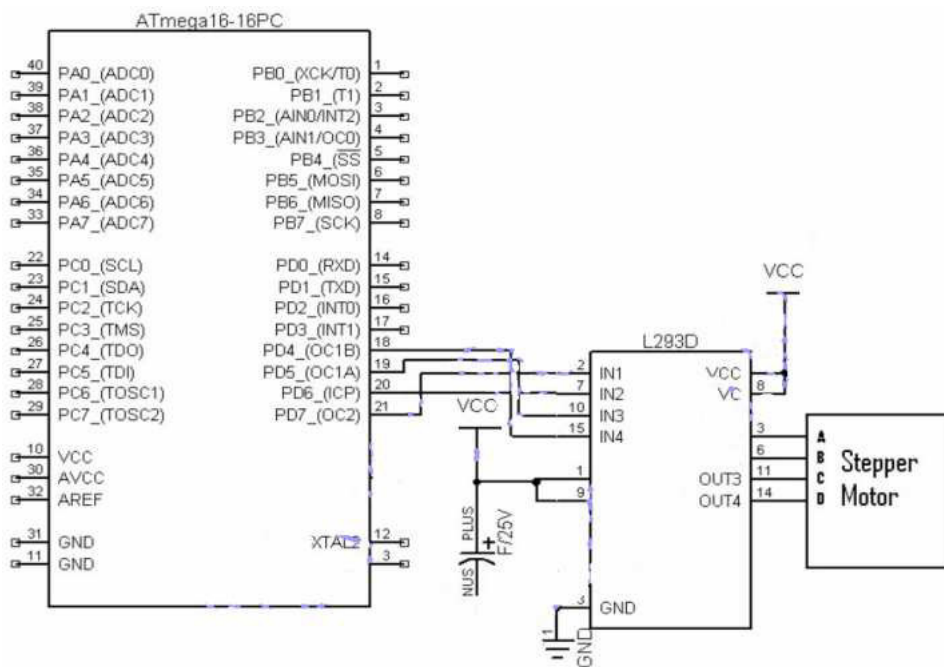
Berikut ini tabel logika untuk menggerakkan stepper motor bipolar yang dapat diterapkan pada motor stepper merek Mitsumi 9V standar.

Tabel 3.1. Logika Motor Stepper.

No	Half step(8 siklus) / Full step(4 siklus)			
	A (PD7)	B (PD6)	C (PD5)	D (PD4)
1	1	0	0	0
2	1	1	0	0
3	0	1	0	0
4	0	1	1	0
5	0	0	1	0
6	0	0	1	1
7	0	0	0	1
8	1	0	0	1

Keterangan
 Urutan data ke bawah, arah : CCW
 Urutan data ke atas, arah : CW

Gambar di bawah ini menampilkan penggunaan driver L293D untuk dapat menggerakkan motor stepper / motor DC karena membutuhkan arus yang cukup besar. 4 jalur output pada IC tersebut dapat mengendalikan 1 buah stepper motor atau 2 buah motor DC. Untuk motorDC, cukup berikan logika high atau low di salah satu pin dari 2 pin yang digunakan oleh tiap motor, dimana logika high /low tersebut menentukan arah putar motor DC.



Gambar 3.7. Dri ver L293D untuk Motor Stepper/Motor DC

3.3.5. Servo Motor

Servo motor banyak digunakan sebagai aktuator pada mobile robot atau lengan robot. Servo motor umumnya terdiri dari servo continuous dan servo standar. Servo motor continuous dapat berputar sebesar 360 derajat.. Sedangkan servo motor tipe standar hanya mampu berputar 180 derajat. Servo motor yang umum digunakan ialah Continuous Parallax. Namun jika Anda ingin servo motor yang berkekuatan

besar dan cepat, idealnya Anda memiliki servo HS-311 (continuous) dan servo HS-322HD(standar). Untuk menggerakkan motor servo ke kanan atau ke kiri, tergantung dari nilai delay yang kita berikan. Untuk membuat servo pada posisi center, berikan pulsa 1.5ms. Untuk memutar servoke kanan, berikan pulsa $\leq 1.3\text{ms}$, dan pulsa $\geq 1.7\text{ms}$ untuk berputar ke kiri dengan delay 20ms.

Spesifikasi dari servo continuous parallax ialah :

- Power 6vdc max
- Average Speed 60 rpm - Note: with 5vdc and no torque
- Weight 45.0 grams/1.59oz
- Torque 3.40 kg-cm/47oz-in
- Size mm (L x W x H) 40.5x20.0x38.0
- Size in (L x W x H) 1.60x.79x1.50
- Manual adjustment port

3.3.6. Motor DC

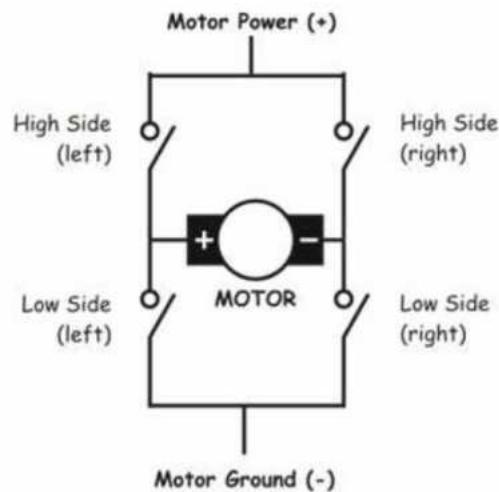
Motor DC adalah alat yang mengubah energi listrik menjadi gerak, mempunyai prinsip dasar yang sama dengan motor stepper namun gerakannya bersifat kontinyu atau berkelanjutan. Motor DC dibagi menjadi 2 jenis yaitu ; Motor DC dengan sikat (mekanis komutasi), yaitu motor yang memiliki sikat karbon berfungsi sebagai pengubah arus pada kumparan sedemikian rupa sehingga arah tenaga putaran motor akan selalu sama. Motor DC tanpa sikat, menggunakan semi konduktor untuk merubah maupun membalik arus sehingga layaknya pulsa yang menggerakkan motor tersebut.

Biasa digunakan pada sistem servo, karena mempunyai efisiensi tinggi, umur pemakaian lama, tingkat kebisingan suara listrik rendah, karena putarannya halus seperti stepper namun putarannya terus menerus tanpa adanya step.

- Sederhana dan murah
- Mudah mengontrol ($I = T$)
- Daya mulai dari 1W - 1kW
- Kecepatan tinggi

Kendali motor DC

- Kontrol + H-bridge
- Kontrol PWM
- Pengaturan motor dengan mengatur variasi tegangan/ arus- akan mengatur torsi motor/kecepatan.
- Efisien
- Kontrol PID



Gambar 3.8. Kendali motor DC menggunakan H Bridge

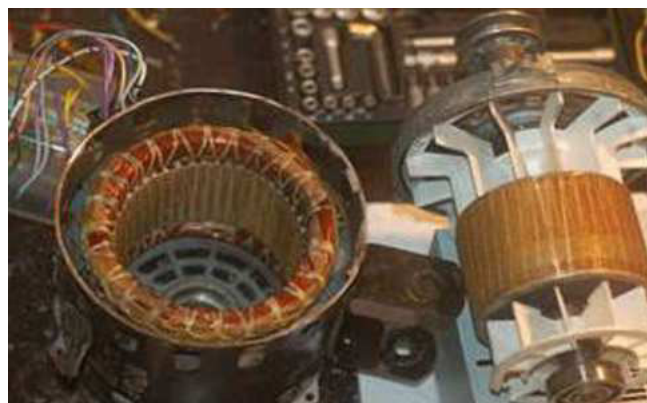
3.3.7. Motor AC

Sebuah motor AC digerakkan oleh sebuah arus bolak-balik (AC) dan terdiri dari dua bagian dasar yaitu:

- Sebuah stator yang diam memiliki lilitan (koil) yang disuplai arus AC untuk menghasilkan medan magnet berputar, dan
- Sebuah rotor di bagian dalam yang disambungkan ke poros keluaran yang diberi torsi putar oleh medan magnet yang berputar.

Ada dua jenis motor AC, tergantung pada tipe rotor yang digunakan :

- Motor sinkron (serempak), yang berputar persis sesuai dengan frekuensi yang disuplai atau sepersekian kali dari frekuensi suplainya. Medan magnet pada rotor dihasilkan dari arus yang dilalukan melalui slip ring atau sebuah magnet permanen.
- Motor induksi, yang berputar sedikit lebih lambat dari frekuensi yang disuplai Medan magnet pada rotor dari motor jenis ini dihasilkan dengan sebuah arus induksi.



Gambar 3.9. Motor listrik dari mesin cuci yang diurai, di mana 12 lilitan dalam rumah motor (kiri) dan sangkar rotor dengan porosnya.

Bila sumber listrik tiga phase ada, maka pada umumnya motor induksi AC tiga phase digunakan, khususnya untuk motor bertenaga besar. Perbedaan phase pada listrik tiga-phase memberikan medan elektromagnetik berputar pada motor. Melalui induksi elektromagnetik, medan magnet berputar menginduksikan arus dalam konduktor dari rotor, yang pada gilirannya menset sebuah counterbalancing medan magnet yang menyebabkan rotor berputar pada arah putaran medan magnet. Rotor harus selalu berputar lebih lambat dari medan magnet yang berputar yang dihasilkan oleh suplai listrik tiga phase; bila tidak, maka tidak akan dihasilkan medan counterballance pada rotor. Motor induksi merupakan motor penggerak di industri, dan motor dengan daya sekitar 500 kW (670 Hp) diproduksi dengan ukuran frame berstandar tinggi. Motor bertenaga lebih besar dari 10 hingga ribuan kW, untuk compressor pipeline, penggerak terowongan angin dan sistem konveyor. Ada dua jenis rotor yang digunakan pada motor induksi yaitu: 1) rotor sangkar (squirrel cage rotor) dan 2) rotor belitan (wound rotor).

Rotor Sangkar

Kebanyakan motor AC menggunakan rotor sangkar, yang sering ditemukan pada motor- motor AC untuk keperluan rumah tangga dan industri kecil. Rotor sangkar tersusun oleh sebuah ring pada ujung-ujung rotor, dengan batang-batang penghubung ring sepanjang rotor. Biasanya coran alumunium atau tembaga di antara lapisan besi dari rotor, dan biasanya hanya ring-ring ujungnya yang nampak. Motor dengan efisiensi tinggi biasanya menggunakan tembaga cor untuk mengurangi tahanan pada rotor. Dalam pengoperasiannya, motor sangkar dapat dilihat sebagai sebuah transformer dengan sebuah putaran sekunder - bila rotor tidak berputar serempak dengan medan magnet, arus rotor yang tinggi diinduksikan; arus rotor yang besar memagnetkan rotor dan berinteraksi dengan medan magnet stator untuk membawa rotor pada keserempakan dengan medan stator. Sebuah motor sangkar tanpa beban pada kecepatan serempak akan mengkonsumsi daya listrik hanya untuk menjaga kecepatan rotor melawan gesekan dan kehilangan tahanan. Saat beban mekanis meningkat, sehingga ada beban elektrik, beban elektrik berhubungan erat dengan beban mekanis. Hal ini mirip dengan sebuah transformer, di mana beban listrik primer berhubungan dengan beban listrik sekunder.

Rotor Belitan

Bila diperlukan kecepatan bervariasi, digunakan rotor belitan. Dalam jenis ini, rotor memiliki jumlah kutub yang sama dengan stator dan belitannya dibuat dari kawat, dihubungkan ke slip rings pada poros. Sikat karbon menghubungkan slip ring ke sebuah kontroller eksternal seperti sebuah resister variabel yang memungkinkan perubahan tingkat slip motor. Dibandingkan dengan rotor sangkar, rotor belitan lebih mahal dan memerlukan pemeliharaan dari slip ring dan sikatnya, tetapi motor jenis ini merupakan motor standar untuk kontrol kecepatan yang bervariasi. Saat ini, untuk mengontrol kecepatan dapat digunakan inverter bertransistor dengan variable-frequency drive dan motor dengan rotor belitan menjadi tidak umum digunakan. Beberapa metode untuk starting motor tiga phase dapat digunakan. Bila arus besar dan torsi start yang tinggi dapat diijinkan, maka motor dapat distart melalui line, dengan menggunakan tegangan penuh pada terminal-terminalnya (Direct-on-line, DOL). Bila diperlukan pembatasan arus start (di mana motor lebih besar

dibandingkan dengan kapasitas jaringan dari supply), tegangan start dikurangi dengan induktor series, sebuah autotransformer, thyristor, atau alat lain. Sebuah cara yang kadang-kadang digunakan adalah konfigurasi “Bintang-Delta” (Y D), di mana koil motor pertama-tama dihubungkan dalam wye untuk mengkaselerasi beban, kemudian diswitch ke delta saat beban mencapai kecepatannya. Motor jenis ini menjadi lebih umum dalam aplikasi traksi seperti lokomotif, di mana dikenal dengan motor traksi tak serempak. Kecepatan motor AC ditentukan dengan frekuensi dari sumber arus AC-nya dan jumlah kutub dari belitan stator, dengan hubungan.

Kecepatan putar aktual dari motor induksi akan lebih kecil dari perhitungan kecepatan serempak dengan sejumlah slip, yang meningkat dengan torsi yang dihasilkan. Tanpa beban, kecepatan akan mendekati kecepatan serempak. Bila diberi beban, motor standar memiliki slip 2-3%, motor khusus memiliki sampai 7 % slip, dan motor torsi memiliki 100% slip. Slip dari motor AC dihitung dengan
Sebagai contoh, sebuah motor dengan empat kutub beroperasi pada 60 Hz bisa memiliki plat nama 1725 RPM pada beban penuh, sedangkan bila dihitung kecepatannya 1800 RPM. Motor Serempak Tiga-phase

Bila sambungan ke lilitan rotor dilakukan pada slip ring dan pengumpulan arus medan yang terpisah untuk menciptakan medan magnet kontinu (atau bila rotor terdiri dari sebuah magnet permanen), hasilnya disebut motor serempak karena rotor akan berputar serempak dengan medan putar yang dihasilkan oleh sumber arus tiga phase. Motor serempak dapat juga digunakan sebagai sebuah alternator.

Motor Induksi AC Satu-phase

Pada sumber arus satu-phase, medan magnet putar (medan putar) harus dibuat menggunakan cara lain selain perbedaan phase (pada motor dua atau tiga phase). Beberapa cara yang umum digunakan adalah sebagai berikut.

Motor Shaded-pole

shaded-pole motor, digunakan pada peralatan yang memerlukan torsi start yang kecil seperti pada kipas listrik atau beberapa peralatan rumah lainnya. Pada motor jenis ini, “shading coils” tembaga putaran tunggal yang kecil menciptakan medan magnet berputar. Bagian dari kutub dikitari oleh sebuah koil tembaga atau strap; arus induksi pada strap berhadapan dengan perubahan dari fluks melalui koil (Hukum Lenz), sehingga intensitas medan maksimum bergerak melintasi muka kutub pada setiap putaran, menghasilkan sebuah medan putar berlevel rendah yang cukup besar untuk memutar rotor dan bebannya. Rotor mempercepat torsi mencapai level penuh saat medan magnet utama berputar relatif terhadap rotor putar.

Motor Induksi Split-phase

Jenis lain dari motor AC satu phase adalah split-phase induction motor, yang umum digunakan pada peralatan seperti mesin cuci. Dibandingkan dengan motor shaded-pole, motor-motor tersebut dapat menghasilkan torsi start yang lebih tinggi menggunakan sebuah lilitan start yang khusus sebagai pengganti sebuah switch sentrifugal. Pada motor split-phase, lilitan start didesain dengan sebuah tanahan yang lebih tinggi dari lilitan putar. Hal ini menciptakan sebuah rangkaian LR yang sedikit menggeser phase dari arus di lilitan start. Saat motor distart, lilitan start dihubungkan ke sumber daya via satu set kontak beban-pegas yang ditekan oleh swtich sentrifugal yang belum berputar. Lilitan start dibelit dengan lilitan yang lebih sedikit dengan

kawat yang lebih kecil dari lilitan utama, sehingga memiliki induktansi (L) lebih kecil dan resistance (R) lebih tinggi. Rasio L/R yang lebih rendah menghasilkan sebuah geseran phase yang kecil, tidak lebih dari 30 o, di antara fluks akibat lilitan utama dan fluks dari lilitan start. Arah putaran dapat dengan mudah dipindahkan dengan menukar hubungan-hubungan dari lilitan start relatif terhadap lilitan putar. Phase dari medan magnet pada lilitan start ini digeser dari phase daya utama, mengakibatkan terciptanya medan putar yang menstart motor. Sekali motor mencapai kecepatan operasi yang didesain, switch sentrifugal aktif, membuka kontak dan memutuskan hubungan lilitan start dari sumber daya. Motor kemudian beroperasi dengan lilitan putar.

Motor dengan start Kapasitor

Pada sebuah motor dengan start kapasitor, sebuah kapasitor start dimasukkan dalam seri dengan lilitan start, menciptakan sebuah rangkaian LC yang dapat memberikan pergeseran phase yang lebih besar, dan torsi start yang lebih tinggi.

3.4. Penutup

3.4.1 Bahan diskusi dan tugas

1. Jelaskan apa yang anda ketahui tentang aktuator?
2. Sebutkan lima jenis aktuator dan jelaskan cara kerjanya?
3. Beri tiga contoh penggunaan aktuator?

3.5 Daftar Pustaka

15. J. Webb, *Industrial Control Electronics*, Macmillan Publish., New York, 1992.
16. J. Webb, *Programmable Logic Control*, Macmillan Publish., New York, 1992.
17. Frank D Petruzella, *Elektronik Industri* (terjemahan), Penerbit Andi, Yogyakarta, 2001.
18. William Bolton, *Programmable Logic Controller (PLC)*, Sebuah Pengantar, Edisi ketiga, Penerbit Erlangga, 2004.
19. M. Budiyanto dan A. Wijaya, *Pengenalan Dasar-dasar PLC (Programmable Logic Controller)*, Penerbit Gaya Media, Yogyakarta, 2003.
20. Endra Ditawarna, *Robotika, Desain, Kontrol dan Kecerdasan Buatan*, Penerbit Andi, Yogyakarta, 2006.
21. David G. Alciator dan Michael B Histan, *Introduction to Mechatronics and Measurement Systems*, Mc Graw Hill, New York, 2003.

3.6. Senarai

Aktuator dikategorikan berdasarkan jenis sumber penggerakannya meliputi : Elektrik, hidrolis, pneumatik yang masing-masing memiliki porsi penggunaan tertentu. Motor DC, stepper dan servo merupakan aktuator yang paling banyak digunakan pada sistem otomasi industri dan robotika.

BAB 4

SISTEM MEKANIK

4.1. Deskripsi Singkat

Tujuan dari mempelajari sistem mekanik adalah untuk dapat memahami sistem transmisi, kinematika dan hidrolika.

4.2. Tujuan Intruksional Khusus

Setelah materi ini diajarkan mahasiswa dapat mengetahui tentang sistem mekanik dan bagian-bagiannya.

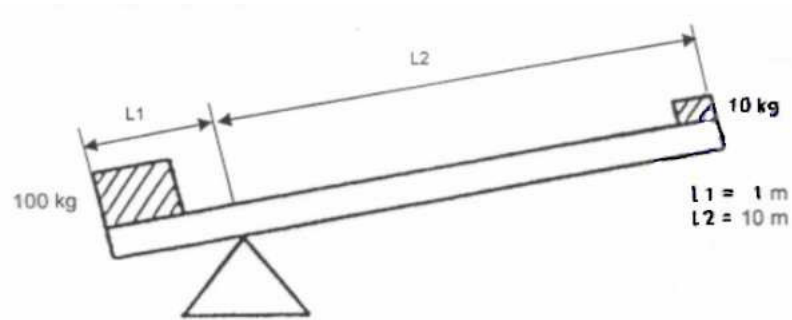
4.3. Penyajian

4.3.1. Pengantar

Mekanika mempelajari keadaan gerak dari suatu sistem fisis (benda). Mekanika dapat dipecah menjadi dua berdasarkan ada atau tidaknya gaya yang bekerja pada sistem yaitu : Kinematika dan Dinamika. Besar-besaran fisis yang menggambarkan keadaan gerak dari suatu benda (partikel, sistem partikel) secara umum dapat diwakili oleh koordinat posisi, kecepatan, percepatan, momenatum dan lain-lain. Lebih lanjut formulasi persamaan gerak dalam sistem gerak dapat ditelaah berdasarkan mekanika Newtonian, Mekanika Lagrangian atau Mekanika Hamiltonian. Dalam hal ini akan membahas transmisi, kinematika dan hidrolika.

4.3.2. Transmisi Mekanik

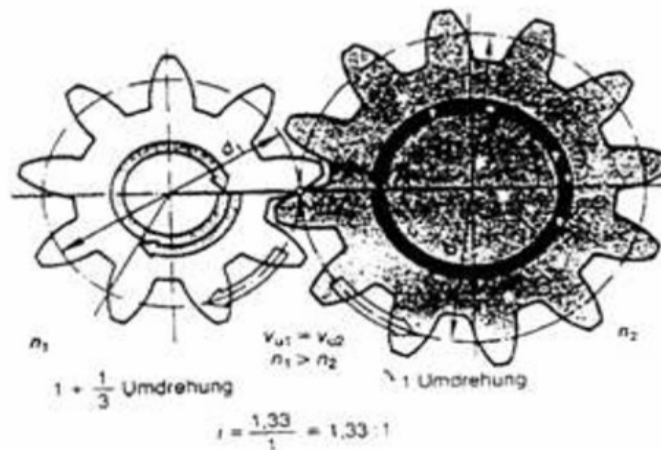
Sistem transmisi, dalam otomotif, adalah sistem yang menjadi penghantar energi dari mesin ke diferensial dan as. Dengan memutar as, roda dapat berputar dan menggerakkan mobil.



Gambar 4.2. Mekanisme pengungkit

Roda gigi

Pengaturan Moment putar dan putaran didasarkan pada prinsip kerja pasangan Roda gigi, seperti yang digambarkan dibawah ini.



Gambar 4.3. Roda Gigi

Pada Roda gigi kecil :

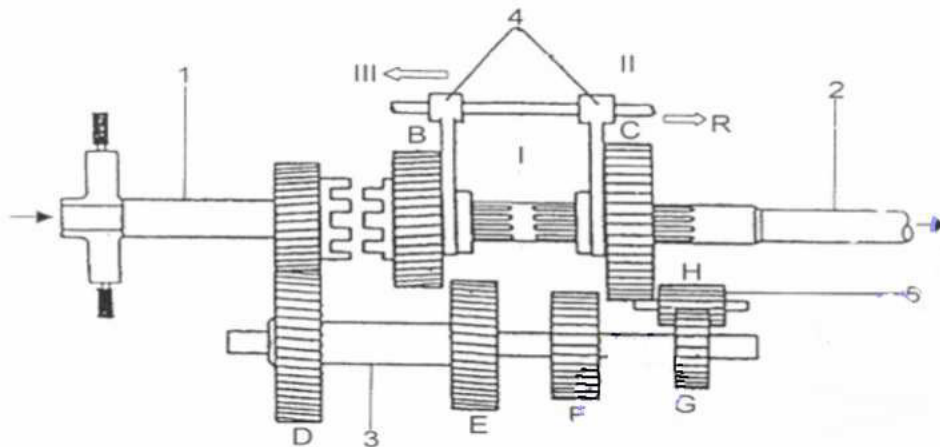
- Putaran cepat
- Moment putar kecil

Pada Roda gigi besar :

- Putaran lambat
- Moment putar besar

Bagian-bagian Utama Transmisi

Contoh : Transmisi Biasa dengan Roda Gigi Geser



- | | |
|------------------------------|--|
| 1. Poros kopling/Poros input | 4. Garpu pemindah |
| 2. Poros utama/Poros output | 5. Roda gigi balik (mundur) / Reverse Gear |
| 3. Poros bantu/Counter Gear | |

Gambar 4.4. Bagian-bagian Utama Sistem Transmisi

4.3.3. Kinematika

Kinematika adalah cabang dari mekanika yang membahas gerakan benda tanpa mempersoalkan gaya dinamika, yang mempersoalkan gaya yang mempengaruhi gerakan. penyebab gerakan. Hal terakhir ini berbeda dari dinamika, yang mempersoalkan gaya yang dipengaruhi oleh gerakan. Setiap hari kita selalu melihat sepeda motor, mobil, pesawat atau kendaraan beroda lainnya. Apa yang terjadi seandainya kendaraan tersebut tidak mempunyai roda ? yang pasti kendaraan tersebut tidak akan bergerak. Sepeda motor atau mobil dapat berpindah tempat dengan mudah karena rodanya berputar, demikian juga pesawat terbang tidak akan lepas landas jika terdapat kerusakan fungsi roda. Putaran roda merupakan salah satu contoh gerak melingkar yang selalu kita temui dalam kehidupan sehari-hari, walaupun sering luput dari perhatian kita. Permainan gasing merupakan contoh lainnya. Sangat banyak gerakan benda yang berbentuk melingkar yang dapat kita amati dalam kehidupan sehari-hari, termasuk gerakan mobil/sepeda motor pada tikungan jalan, gerakan planet kesayangan kita (bumi), planet-planet lainnya, satelit, bintang dan benda angkasa yang lain. Setiap benda yang bergerak membentuk suatu lingkaran dikatakan melakukan gerakan melingkar. Sebelum membahas lebih jauh mengenai gerak melingkar, terlebih dahulu kita pelajari besaran-besaran fisis dalam gerak melingkar.

Besaran-Besaran Fisis dalam Gerak Melingkar (Perpindahan Sudut, Kecepatan sudut dan Percepatan Sudut) Dalam gerak lurus kita mengenal tiga besaran utama yaitu perpindahan (linear), kecepatan (linear) dan Percepatan (linear). Gerak melingkar juga memiliki tiga komponen tersebut, yaitu perpindahan sudut, kecepatan sudut dan percepatan sudut. Pada gerak lurus kita juga mengenal Gerak Lurus Beraturan dan Gerak Lurus Berubah Beraturan. Dalam gerak melingkar juga terdapat Gerak Melingkar Beraturan (GMB) dan Gerak Melingkar Berubah Beraturan

(GMBB). Selengkapnya akan kita bahas satu persatu. Sekarang mari kita berkenalan dengan besaran-besaran dalam gerak melingkar dan melihat hubungannya dengan besaran fisis gerak lurus. Pada Benda yang Bergerak Melingkar Beraturan Dari modul "Kinematika Gerak Lurus" Anda telah mempelajari bahwa benda yang bergerak melingkar beraturan memiliki percepatan sentripetal (a_s) yang besarnya :

$$a_s = \frac{v^2}{R} = \omega^2 \cdot R$$

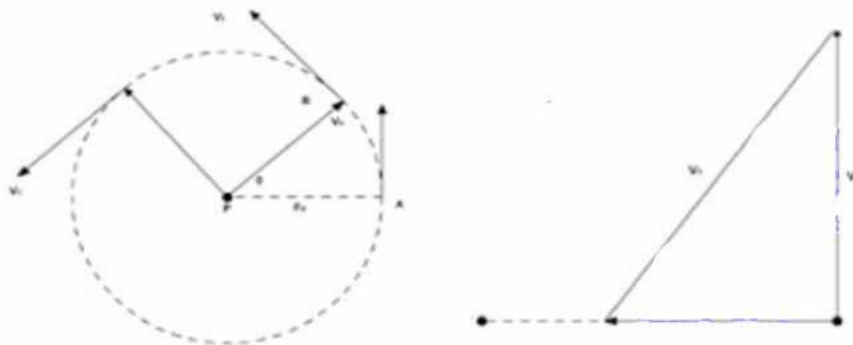
Dimana :

v = kecepatan linier

ω = kecepatan sudut (dibaca omega)

R = jari-jari lintasan

Untuk mengingatkan, Anda perhatikan gambar 4.5 berikut ini :



Gambar 4.5. Perubahan kecepatan pada benda GMB menuju pusat lintasan.

Sesuai dengan Hukum II Newton, percepatan sentripetal a_s disebabkan oleh gaya yang searah dengan a_s . Gaya ini dinamakan gaya sentripetal (F_s). Jadi :

$$SF = m \cdot a_s$$

di mana $SF = F_s$ adalah gaya sentripetal dapat ditulis bahwa :

$$a_s = \frac{v^2}{R} = \omega^2 \cdot R$$

Dari persamaan ini nampak bahwa besarnya gaya sentripetal bergantung pada :

- m = massa benda (kg)
- v = kecepatan linier (m/s)
- ω = kecepatan sudut (rad/s)
- R = jari-jari lintasan, m]

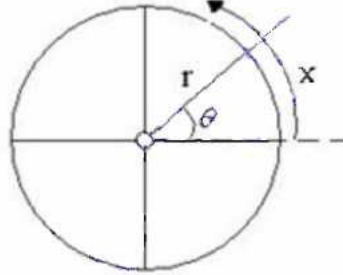
Gaya sentripetal, F_s berperan mempertahankan benda bergerak melingkar beraturan agar tetap pada lintasannya.

Perpindahan Sudut

Mari kita tinjau sebuah contoh gerak melingkar, misalnya gerak roda kendaraan yang berputar. Ketika roda berputar, tampak bahwa selain poros alias pusat roda, bagian lain roda lain selalu berpindah terhadap pusat roda sebagai kerangka

acuan. Perpindahan padagerak melingkar disebut perpindahan sudut. Bagaimana caranya kita mengukur perpindahan sudut ?

Ada tiga cara menghitung sudut. Cara pertama adalah menghitung sudut dalam derajat ($^{\circ}$). Satu lingkaran penuh sama dengan 360° . Cara kedua adalah mengukur sudut dalam putaran. Satu lingkaran penuh sama dengan satu putaran. Dengan demikian, satu putaran = 360° . Cara ketiga adalah dengan radian. Radian adalah satuan Sistem Internasional (SI) untuk perpindahan sudut, sehingga satuan ini akan sering kita gunakan dalam perhitungan. Bagaimana mengukur sudut dengan radian ? Mari kita amati gambar di bawah ini.



Nilai radian dalam sudut adalah perbandingan antara jarak linear x dengan jari-jari roda r. Jadi,

$$\theta \text{ (rad)} = \frac{x}{r}$$

Perhatikan bahwa satu putaran sama dengan keliling lingkaran, sehingga dari persamaan di atas, diperoleh :

$$\theta \text{ (rad)} = \frac{2\pi r}{r} = 2\pi \text{ rad}$$

Berikut ini konversi sudut yang perlu anda ketahui :

$$1 \text{ putaran} = 360^{\circ} = 2\pi \text{ rad}$$

$$1 \text{ rad} = \frac{180}{\pi} \text{ derajat} = 57,3^{\circ}$$

Derajat, putaran dan radian adalah besaran yang tidak memiliki dimensi. Jadi, jika ketiga satuan ini terlibat dalam suatu perhitungan, ketiganya tidak mengubah satuan yang lain.

Kecepatan Sudut

Dalam gerak lurus, kecepatan gerak benda umumnya dinyatakan dengan satuan km/jam atau m/s. Telah kita ketahui bahwa tiap bagian yang berbeda pada benda yang melakukan gerak lurus memiliki kecepatan yang sama, misalnya bagian depan mobil mempunyai kecepatan yang sama dengan bagian belakang mobil yang bergerak lurus.

Dalam gerak melingkar, bagian yang berbeda memiliki kecepatan yang berbeda. Misalnya gerak roda yang berputar. Bagian roda yang dekat dengan poros bergerak dengan kecepatan linear yang lebih kecil, sedangkan bagian yang jauh dari poros alias pusat roda bergerak dengan kecepatan linear yang lebih besar. Oleh karena

itu, bila kita menyatakan roda bergerak melingkar dengan kelajuan 10 m/s maka hal tersebut tidak bermakna, tetapi kita bisa mengatakan tepi roda bergerak dengan kelajuan 10 m/s. Pada gerak melingkar, kelajuan rotasi benda dinyatakan dengan putaran per menit (biasa disingkat rpm - revolution per minute). Kelajuan yang dinyatakan dengan satuan rpm adalah kelajuan sudut. Dalam gerak melingkar, kita juga dapat menyatakan arah putaran. Misalnya kita menggunakan arah putaran jarum jam sebagai patokan. Oleh karena itu, kita dapat menyatakan kecepatan sudut, di mana selain menyatakan kelajuan sudut, juga menyatakan arahnya (ingat perbedaan kelajuan dan kecepatan, mengenai hal ini sudah diterangkan pada pokok bahasan kinematika). Jika kecepatan pada gerak lurus disebut kecepatan linear (benda bergerak pada lintasan lurus), maka kecepatan pada gerak melingkar disebut kecepatan sudut, karena benda bergerak melalui sudut tertentu. Terdapat dua jenis kecepatan pada Gerak Lurus, yakni kecepatan rata-rata dan kecepatan sesaat. Kita dapat mengetahui kecepatan rata-rata pada Gerak Lurus dengan membandingkan besarnya perpindahan yang ditempuh oleh benda dan waktu yang dibutuhkan benda untuk bergerak. Pada gerak melingkar, kita dapat menghitung **kecepatan sudut rata-rata** dengan membandingkan perpindahan sudut dengan selang waktu yang dibutuhkan ketika benda berputar. Secara matematis kita tulis :

Kecepatan sudut rata-rata = perpindahan sudut/selang waktu :

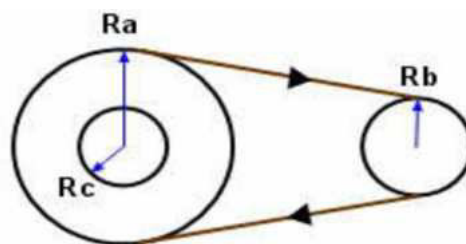
$$\bar{\omega} = \frac{\Delta\theta}{\Delta t} = \frac{\theta_2 - \theta_1}{t_2 - t_1}$$

Bagaimana dengan kecepatan sudut sesaat? Kecepatan sudut sesaat kita diperoleh dengan membandingkan perpindahan sudut dengan selang waktu yang sangat singkat. Secara matematis kita tulis :

$$\bar{\omega} = \frac{\Delta\theta}{\Delta t} \longrightarrow \text{Untuk } \Delta t \text{ sangat kecil}$$

Contoh persoalan hubungan roda-roda

Perhatikan gambar tiga roda yang di hubungan sebagai berikut :



Jika $R_c = 4$ cm, $R_b = 6$ cm dan $R_a = 8$ cm, dan kecepatan sudut roda b = 8 rad/s. Tentukan :

1. hubungan masing-masing roda;
2. kecepatan sudut roda a dan
3. kelajuan linier roda c

Pembahasan :

Diketahui :

$$R_a = 4 \text{ cm} = 4 \times 10^{-2} \text{ m}$$

$$R_b = 6 \text{ cm} = 6 \times 10^{-2} \text{ m}$$

$$R_c = 8 \text{ cm} = 8 \times 10^{-2} \text{ m}$$

$$\omega_b = 8 \text{ rad/s}$$

Ditanyakan :

1. Hubungan roda-roda
2. ω_a
3. v_c

Jawab :

1. - Roda a sepusat dengan roda c
- Roda a dan roda b dihubungkan dengan sabuk/tali
2. $v_a = v_b$
 $\omega_a R_a = \omega_c R_c$
 $\omega_a (4 \times 10^{-2}) = 8 (6 \times 10^{-2})$
 $\omega_a = 12 \text{ rad/s.}$
3. $\omega_a = \omega_c$

$$\frac{v_a}{R_a} = \frac{v_c}{R_c}$$

$$v_c = \omega_a R_c$$

$$v_c = 12 (8 \times 10^{-2})$$

$$v_c = 0,96 \text{ m/s}$$

4.3.4. Pneumatik

Pneumatik berasal dari bahasa Yunani “pneuma” yang berarti tiupan atau hembusan. Sistem pneumatik itu sendiri mempunyai beberapa kelebihan dan kekurangan yang dipengaruhi terutama oleh sifat udara terkompresi sebagai penggerakannya. Sifat-sifat udara yang mempengaruhi sifat-sifat pengontrolan sistem pneumatik antara lain :

- Udara tidak mempunyai bentuk khusus. Bentuknya selalu sesuai dengan tempatnya/wadahnya.
- Dapat dimampatkan /kompresible.
- Memenuhi semua ruang dengan sama rata.
- Dapat dikontrol baik laju alirannya maupun tekanan dan gaya yang bekerja.

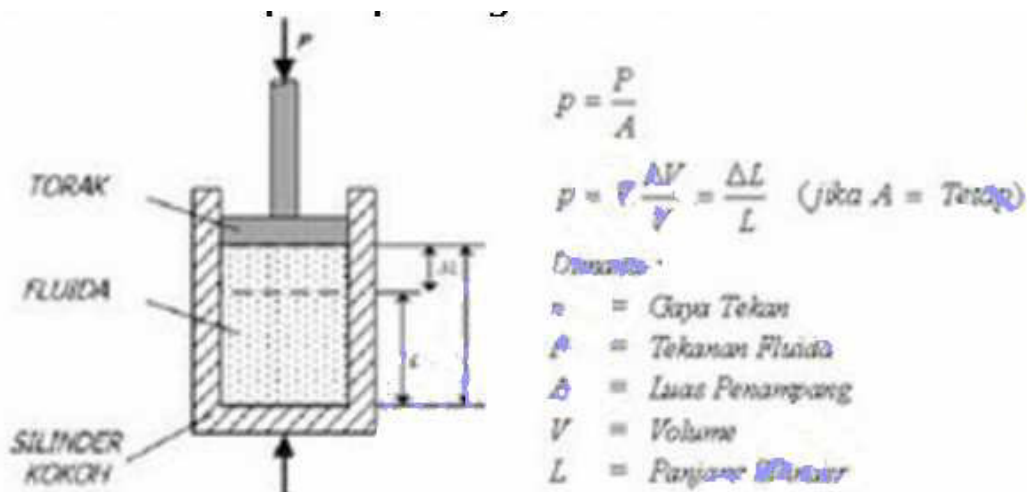
Keuntungan dan Kerugian Udara Bertekanan

Udara bertekanan memiliki banyak sekali keuntungan, tetapi dengan sendirinya juga terdapat segi-segi yang merugikan atau pembatasan-pembatasan pada penggunaannya. Hal-hal yang menguntungkan dari pneumatik pada mekanisasi yang sesuai dengan tujuan sudah diakui oleh cabang-cabang industri yang lebih banyak lagi. Pneumatik mulai digunakan untuk pengendalian maupun penggerakan mesin-mesin dan alat-alat.

Komponen sistem pneumatik

Kompresor digunakan untuk menghisap udara di atmosfer dan menyimpannya kedalam tangki penampung atau receiver. Kondisi udara dalam atmosfer dipengaruhi oleh suhu dan tekanan. Kompresor adalah mesin untuk memampatkan udara atau gas. Secara umum biasanya menghisap udara dari atmosfer, yang secara fisika merupakan campuran beberapa gas dengan susunan 78% Nitrogen, 21% Oksigen dan 1% Campuran Argon, Carbon Dioksida, Uap Air, Minyak, dan lainnya. Namun ada juga kompresor yang menghisap udara/ gas dengan tekanan lebih tinggi dari tekanan atmosfer dan biasa disebut penguat (booster). Sebaliknya ada pula kompresor yang menghisap udara/ gas bertekanan lebih rendah dari tekanan atmosfer dan biasanya disebut pompa vakum. Jika suatu gas/ udara didalam sebuah ruangan tertutup diperkecil volumenya, maka gas/ udara tersebut akan mengalami kompresi. Kompresor yang menggunakan azas ini disebut kompresor jenis displacement dan prinsip kerjanya dapat dilukiskan seperti pada gambar 4.6.

Disini digunakan torak yang bergerak bolak balik oleh sebuah penggerak mula (prime mover) didalam sebuah silinder untuk menghisap, menekan dan mengeluarkan udara secara berulang-ulang. Dalam hal ini udara tidak boleh bocor melalui celah antara dinding torak dengan dinding silinder yang saling bergesekan. Untuk itu digunakan cincin torak sebagai perapat.



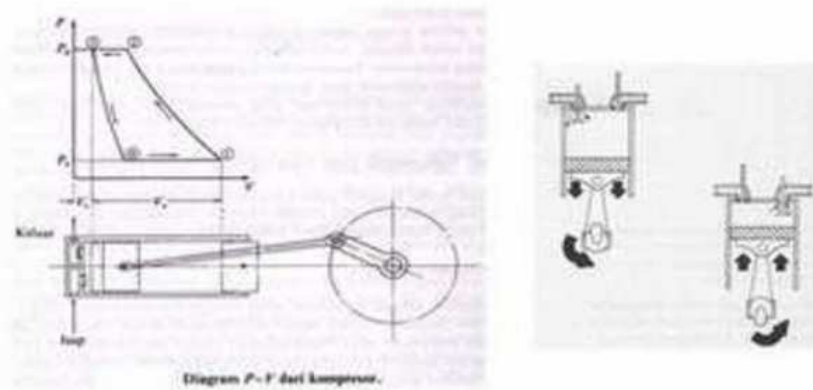
Gambar 4.6. Kompresi Fluida

Jika torak ditarik keatas, tekanan dalam silinder dibawah torak akan menjadi negatif (kecil dari tekanan atmosfer) sehingga udara akan masuk melalui celah katup isap. Kemudian bila torak ditekan kebawah, volume udara yang terkurung dibawah torak akan mengecil sehingga tekanan akan naik.

Berdasarkan prinsip kerjanya, kompresor terdiri dari 2 (dua) jenis yaitu : Displacement (torak) seperti dijelaskan diatas dan Dynamic (rotary) yang mengalirkan udara melalui putaran sudu berkecepatan tinggi.

Kompresi Udara

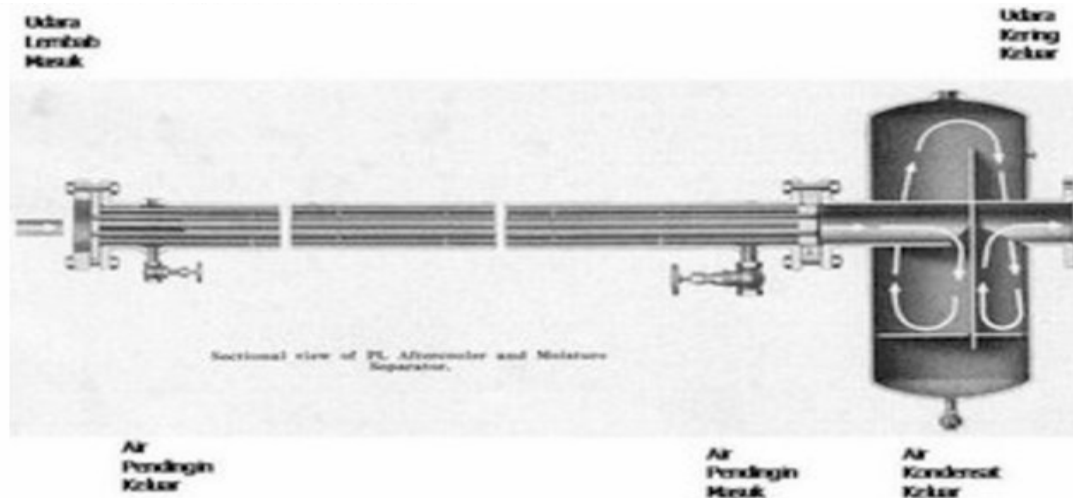
Proses kompresi udara yang terjadi pada kompresor torak dapat dijelaskan dengan menggunakan pendekatan seperti terlihat pada gambar 4.7. Torak memulai langkah kompresinya pada titik (1) diagram P-V, kemudian bergerak ke kiri dan udara dimampatkan hingga tekanan naik ke titik (2). Pada titik ini tekanan dalam silinder mencapai harga tekanan P_d yang lebih tinggi dari pada tekanan dalam pipa keluar (atau tangki tekan) sehingga katup keluar pada kepala silinder akan terbuka. Jika torak terus bergerak ke kiri, udara akan didorong keluar silinder pada tekanan tetap sebesar P_d . Di titik (3) torak mencapai titik mati atas, yaitu titik akhir gerakan torak pada langkah kompresi dan pengeluaran.



Gambar 4.7. Diagram P-V dari Kompresor

Kondensasi Uap air

Udara yang dihisap dan dimampatkan didalam kompresor akan mengandung uap air dalam jumlah cukup besar. Jika uap ini didinginkan udara yang keluar dari kompresor maka uap akan mengembun menjadi air. Air ini akan terbawa ke mesin/ peralatan yang menggunakannya dan mengakibatkan gangguan pada pelumasan, korosi dan peristiwa water hammer pada piping system. Aftercooler adalah heat-exchanger yang berguna untuk mendinginkan udara/ gas keluaran kompresor untuk membuang uap air yang tidak diinginkan sebelum dikirim ke alat lain. Uap air dipisahkan dari udara dengan cara pendinginan dengan air atau oli pendingin. Sumber Ingersoll-Rand. Dapat dilihat pada gambar dibawah ini :



Gambar 4.9. Aftercooler Kompresor Multi Stage

Condensate Drain Valve

Condensate drain valve ialah bagian dari kompresor yang berfungsi membuang kondensat (uap air) yang terjadi saat kompresor bekerja dengan mengambil udara dari luar, sehingga udara yang masuk ke dalam sistem udara tekan menjadi bersih dan tidak menimbulkan adanya endapan air. Manfaat lainnya pada sistem hidrolis adalah :

- Oli tetap bersih karena kontaminasi dari air telah dibuang melalui Condensate Drain Valve.
- Fungsi dari Oil and Water Trap adalah sebagai pemisah oli dan air dari udara yang masuk dari kompresor. Jumlah air persentasenya sangat kecil dalam udara yang masuk ke dalam sistem Pneumatik, tetapi dapat menjadi penyebab serius dari tidak berfungsinya sistem.
- Fungsi unit ini adalah sebagai pemisah kimia untuk memisahkan sisa uap lembab yang mana boleh jadi tertinggal waktu udara melewati unit Oil and Water Trap.
- Setelah udara yang dikompresi melewati unit Oil and Water Trap dan unit Dehydrator, akhirnya udara yang dikompresi akan melewati Filter untuk memisahkan udara dari kemungkinan adanya debu dan kotoran yang mana mungkin terdapat dalam udara.
- Sistem tekanan udara siap masuk pada tekanan tinggi menambah tekanan pada bilik dan mendesak beban pada piston.
- Restrictor adalah tipe dari pengontrol klep yang digunakan dalam sistem Pneumatik, Restrictor yang biasa digunakan ada dua (2) tipe, yaitu tipe Orifice dan Variable Restrictor.

Perawatan sistem Pneumatik terdiri dari memperbaiki, mencari gangguan, pembersihan dan pemasangan komponen, dan uji coba pengoperasian. Tindakan pencegahan untuk menjaga udara dalam sistem selalu terjaga kebersihannya. Saringan dalam komponen harus selalu dibersihkan dari partikel-partikel metal yang mana hal tersebut dapat menyebabkan keausan pada komponen. Setiap memasang komponen Pneumatik harus dijaga kebersihannya dan diproteksi dengan pita penutup atau

penutup debu dengan segera setelah pembersihan. Memastikan ketika memasang kembali komponen tidak ada partikel metal yang masuk kedalam sistem. Sangat penting mencegah masuknya air, karena dapat menjadi penyebab sistem tidak dapat memberikan tekanan. Operasi dalam temperatur rendah, walaupun terdapat jumlah air yang sangat kecil dapat menjadi penyebab serius tidak berfungsinya sistem. Setiap tahap perawatan harus memperhatikan masuknya air kedalam sistem. Kebocoran bagian dalam komponen, selama kebocoran pada O-Ring atau posisinya, yang mana ketika pemasangan tidak sempurna atau tergores oleh partikel metal atau sudah batas pemakaian.

4.3.5. Sistem Hidrolik

Bertahun-tahun lalu manusia telah menemukan kekuatan dari perpindahan air, meskipun mereka tidak mengetahui hal tersebut merupakan prinsip hidrolik. Sejak pertama digunakan prinsip ini, mereka terus menerus mengaplikasikan prinsip ini untuk banyak hal untuk kemajuan dan kemudahan umat manusia. Hidrolik adalah ilmu pergerakan fluida, tidak terbatas hanya pada fluida air. Jarang dalam keseharian kita tidak menggunakan prinsip hidrolik, tiap kali kita minum air, tiap kali kita menginjak rem kita mengaplikasikan prinsip hidrolik. Sistem hidrolik banyak memiliki keuntungan.

Sebagai sumber kekuatan untuk banyak variasi pengoperasian. Keuntungan sistem hidrolik antara lain :

- Ringan
- Mudah dalam pemasangan
- Sedikit perawatan
- Sistem hidrolik hampir 100 % efisien, bukan berarti mengabaikan terjadinya gesekan fluida.

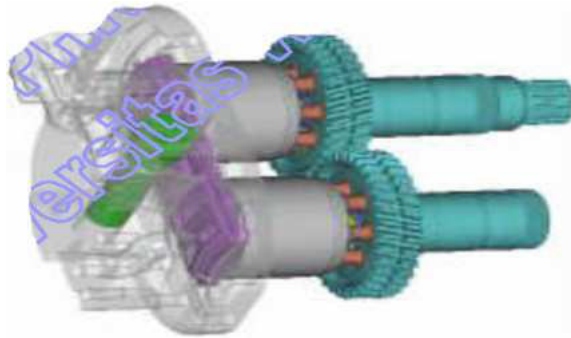
Untuk mengerti prinsip hidrolik kita harus mengetahui perhitungan dan beberapa hukum yang berhubungan dengan prinsip hidrolik.

- Area adalah ukuran permukaan (in^2 , m^2)
- Force adalah jumlah dorongan atau tarikan pada objek (lb, kg)
- Unit pressure adalah jumlah kerkuatan dalam satu unit area (lb/in^2 , Psi)
- Stroke (panjang) adalah diukur berdasarkan jarak pergerakan piston dalam silinder (in, m)
- Volume diukur berdasarkan jumlah dalam in^3 , m^3 yang dihitung berdasarkan jumlah fluida dalam reservoir atau dalam pompa atau pergerakan silinder.

Fluida yang digunakan dalam bentuk liquid atau gas. Fluida yang digunakan dalam system hidrolik umumnya oli. Suatu aliran didalam silinder yang dilengkapi dengan sebuah penghisap yang mana kita dapat memakaikan sebuah tekanan luar p_o tekanan p disuatu titik P yang sebarang sejarak h dibawah permukaan yang sebelah atas dari cairan tersebut diberikan oleh persamaan. Prinsip Pascal, tekanan yang dipakaikan kepada suatu fluida tertutup diteruskan tanpa berkurang besarnya kepada setiap bagian fluida dan dinding-dinding yang berisi fluida tersebut. Hasil ini adalah suatu konsekuensi yang perlu dari hokum hukum mekanika fluida, dan bukan merupakan sebuah prinsip bebas.

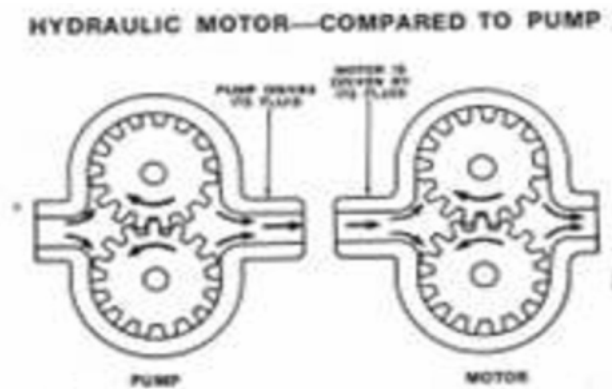
Komponen Sistem Hidrolik

Motor hidrolik berfungsi untuk mengubah energi tekanan cairan hidrolik menjadi energi mekanik. Pompa umumnya digunakan untuk memindahkan sejumlah volume cairan yang digunakan agar suatu cairan tersebut memiliki bentuk energi. Katup pada sistem dibedakan atas fungsi, disain dan cara kerja katup Pompa hidrolik berfungsi untuk mentransfer energi mekanik menjadi energi hidrolik. Pompa hidrolik bekerja dengan cara menghisap oli dari tangki hidrolik dan mendorongnya ke dalam sistem hidrolik dalam bentuk aliran (flow).



Gambar 4.11. Pompa Hidrolik

Aliran ini yang dimanfaatkan dengan cara merubahnya menjadi tekanan. Tekanan dihasilkan dengan cara menghambat aliran oli dalam sistem hidrolik. Hambatan ini dapat disebabkan oleh orifice, silinder, motor hidrolik, dan aktuator. Pompa hidrolik yang biasa digunakan ada dua macam yaitu Positive dan Non-positive Displacement Pump.



Gambar 4.12. Skema Motor Hidrolik

Cara Memanfaatkan Tenaga Pada Sistem hidrolik

Ada dua macam peralatan yang biasanya digunakan dalam merubah energi hidrolik menjadi energi mekanik yaitu motor hidrolik dan aktuator. Motor hidrolik mentransfer energi hidrolik menjadi energi mekanik dengan cara memanfaatkan aliran oli dalam sistem merubahnya menjadi energi putaran yang dimanfaatkan untuk menggerakkan roda, transmisi, pompa dll. Perawatan dari sistem hidrolik, memerlukan penggunaan fluida hidrolik yang layak, pemilihan tube dan seal yang layak. Dan kita harus dapat mengetahui bagaimana pengecekan untuk kebersihannya yang layak.

Perbaikan pada sistem hidrolik, adanya satu prosedur perawatan dilakukan pada mekanik hidrolik. Sebelum perbaikan dimulai, spesifikasi tipe fluida harus diketahui. Warna dari fluida pada sistem dapat juga digunakan sebagai penentu dari tipe fluida. Perawatan efektif dari sistem hidrolik yang diperlukan adalah melihat kelayakan seal, tube, selang yang digunakan. Untuk sistem hidrolik (3000psi) digunakan tube stainless steel, dan untuk sistem hidrolik tekanan rendah dapat digunakan tube dari aluminium alloy.

4.4. Penutup

4.4.1. Bahan diskusi dan tugas

1. Jelaskan apa yang anda ketahui tentang sistem mekanik dan berikan contohnya?
2. Jelaskan dan sebutkan bagian-bagian dari hidrolik dan pneumatik?
3. Apa yang anda ketahui tentang gerakan melingkar beraturan?

4.5. Daftar Pustaka

22. J. Webb, *Industrial Control Electronics*, Macmillan Publish,. New York, 1992.
23. J. Webb, *Programmable Logic Control*, Macmillan Publish,. New York, 1992.
24. Frank D Petruzella, *Elektronik Industri* (terjemahan), Penerbit Andi, Yogyakarta, 2001.
25. William Bolton, *Programmable Logic Controller (PLC)*, Sebuah Pengantar, Edisi ketiga, Penerbit Erlangga, 2004.
26. M. Budiyanto dan A. Wijaya, *Pengenalan Dasar-dasar PLC (Programmable Logic Controller)*, Penerbit Gaya Media, Yogyakarta, 2003.
27. Endra Ditawarna, *Robotika, Desain, Kontrol dan Kecerdasan Buatan*, Penerbit Andi, Yogyakarta, 2006.
28. David G. Alciator dan Michael B Histan, *Introduction to Mechatronics and Measurement Systems*, Mc Graw Hill, New York, 2003.

4.6. Senarai

Ada beberapa media kerja (bentuk energi) yaitu elektrik, hidrolik dan pneumatik. Persaingan antara peralatan pneumatik dengan peralatan mekanik, hidrolik atau elektrik makin menjadi besar. Dalam penggunaannya sistem pneumatik diutamakan karena beberapa hal yaitu : paling banyak dipertimbangkan untuk beberapa mekanisasi, dapat bertahan lebih baik terhadap keadaan-keadaan tertentu. Seringkali suatu proses tertentu dengan cara pneumatik, berjalan lebih rapi (efisien) dibandingkan dengan cara lainnya.

BAB 5

PENGENDALIAN BERBASIS MIKROPROSESSOR

5.1. Deskripsi Singkat

Tujuan dari mempelajari pengendalian berbasis mikroprosesor adalah untuk dapat memahami *internal hardware design (architecture)*.

5.2. Tujuan Intruksional Khusus

Setelah materi ini diajarkan mahasiswa dapat menjelaskan arsitektur mikroprosesor, mengidentifikasi komponen dasar mikroprosesor dan dapat menjelaskan mekanisme pengendalian peralatan luar menggunakan mikroprosesor.

5.3. Penyajian

5.3.1. Pengantar

Gambaran atau Features dari sebuah Mikroprosesor dapat dipelajari dengan baik melalui pemahaman dan pengkajian Internal Hardware Design, yang disebut juga dengan istilah Architecture. Internal Hardware design berkaitan dengan masalah-masalah Jenis, Jumlah, dan Ukuran Register serta komponen lainnya. Sedangkan untuk dapat menginstalasikan sebuah mikroprosesor dengan komponen lainnya seperti RAM, ROM, dan I/O sebagai komponen utama dan rangkaian Clock, Reset, Buffer, dan lain-lain sebagai komponen pendukung diperlukan pemahaman sistem bus yang dimiliki oleh setiap Mikroprosesor.

5.3.2. Arsitektur Mikroprosesor

Ada tiga jenis arsitektur Mikroprosesor :

1. Arsitektur I/O Terisolasi

Mikroprosesor dengan arsitektur I/O Terisolasi menggunakan disain pengalamatan atau pemetaan I/O terpisah atau terisolasi dengan pengalamatan atau

pemetaan memori. Pengalamatan I/O menggunakan sebagian dari jumlah saluran alamat (Address Buss) sedangkan pengalamatan memori menggunakan semua saluran alamat (Address Buss). Metode I/O terisolasi menggunakan akumulator pada CPU untuk menerima informasi dari I/O atau mengeluarkan informasi ke bus I/O selama operasi Input Output. Tidak ada Register lain selain akumulator yang terpakai untuk akses I/O. Metode I/O Terisolasi disebut juga dengan I/O akumulator.

Konsep ini memiliki pengaruh penting pada program computer yaitu :

- Instruksi yang digunakan hanya dua kode operasi yaitu IN dan OUT
- Informasi/data yang ada pada akumulator harus dialihkan pada suatu lokasi penyimpanan sementara sebelum ada operasi I/O berikutnya
- Perlu ada tambahan instruksi pada program pengalihan data/informasi pada akumulator.

Keuntungan metode I/O terisolasi:

- Komputer dapat mengalihkan informasi/data ke atau dari CPU tanpa menggunakan memori. Alamat atau lokasi memori untuk rangkaian memori buka untuk operasi I/O
- Lokasi memori tidak terkurangi oleh sel-sel I/O Instruksi I/O lebih pendek sehingga dapat dengan mudah dibedakan dari instruksi memori
- Pengalamatan I/O menjadi lebih pendek dan perangkat keras untuk pengkodean alamat lebih sederhana.

Kerugian metode I/O terisolasi :

Lebih banyak menggunakan penyemat pengendalian pada Mikroprosesornya. Mikroprosesor buatan Intel dan Mikroprosesor buatan Zilog menggunakan arsitektur I/O Terisolasi.

2. Arsitektur I/O Terpetakan dalam Memori

Mikroprosesor dengan arsitektur I/O terpetakan dalam memori menyatukan sel-sel I/O dalam pengalamatan yang bersama dengan sel-sel memori. I/O yang terpetakan dalam memori menunjukkan penggunaan instruksi tipe memori untuk mengakses alat-alat I/O. I/O yang dipetakan dalam memori memungkinkan CPU menggunakan instruksi yang sama untuk alih memori seperti yang digunakan untuk alih I/O. Sebuah pintu I/O diperlakukan seperti sebuah lokasi memori. Keuntungan sistim ini adalah instruksi yang dipakai untuk pembacaan dan penulisan memori dapat digunakan untuk memasukkan dan mengeluarkan data pada I/O. Kerugiannya pertama tiap satu pintu I/O mengurangi satu lokasi memori yang tersedia. Kedua alamat lokasi I/O memerlukan 16 bit saluran. Ketiga instruksi I/O yang dipetakan dalam memori lebih lama dari instruksi I/O terisolasi.

3. Arsitektur Harvard

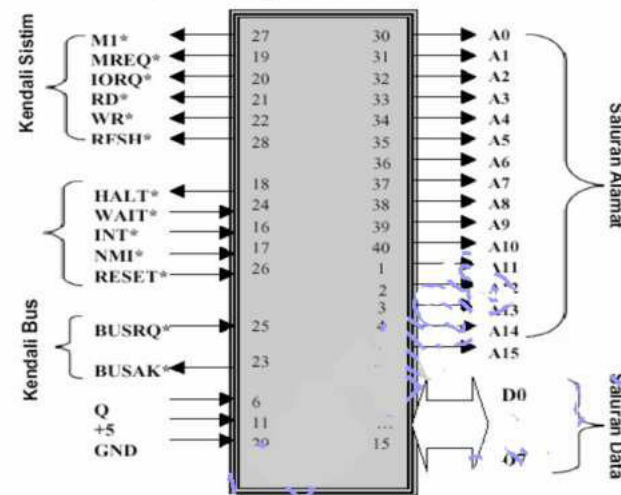
Arsitektur Harvard menggunakan disain yang hampir sama dengan arsitektur I/O terisolasi. Perbedaannya pada arsitektur harvard antara memori program dan memori data dipisahkan atau diisolasi. Pemisahan antara memori program dan memori data menggunakan perintah akses memori yang berbeda. Harvard arsitektur ditinjau dari kemampuan jumlah memori lebih menguntungkan.

Kemasan Mikroprosesor

Ada empat jenis bentuk kemasan Mikroprosesor :

- PDIP : Pastic Dual Inline Package
- PLCC : Plastic J-Lieded Chip Carrier
- TQFP : Plastic Gull Wing Quad Flat Package
- SOIC : Plastic Gull-wing Small Outline.

Feature Kasus pada Zilog Z-80 CPU



Gambar 5.1. Susunan dan Konfigurasi Pin Z-80 CPU

Keterangan Gambar diatas adalah sebagai berikut :

- Mikroprosesor 8 bit dengan arsitektur I/O Terisolasi
- 16 bit Address Bus dengan kemampuan: pengalaman memori 64 Kbyte
Pengalaman I/O 256 byte
- 148 instruksi
- 8 buah Register 8 bit sebagai Register utama, buah register 8 bit sebagai Register alternatif, 4 buah Register 16 bit, 2 buah Register 8 bit fungsi khusus.
- Frekuensi Clock 2,5 MHz - 4 Mhz
- Komsumsi Daya: A ktif 150 mA
- Kemasan PDIP

Kendali CPU menjalankan fungsi-fungsi sebagai berikut :

- M1* (Machin Cycle One: satu siklus mesin) merupakan pin keluaran aktrendah jika CPU sedang mengambil sandi operasi instruksi dari memori.Pada saat ini bus alamat berisi alamat memori seperti data yang ada pada Register PC, dan data bus mengarah masuk.
- MREQ* (Memori Request: pesan memori) merupakan pin Keluaran aktifrendah pada waktu saluran alamat berisi alamat memori
- IORQ* (Input Output Request: pesan Input Output) Keluaran aktif rendahpada waktu saluran alamat A0 s/d A7 berisi alamat I/O
- RD* (Read: Baca) Keluaran aktif rendah pada waktu CPU melakukan operasi baca/memasukkan data
- WR* (Write: Tulis) Keluaran aktif rendah pada waktu CPU melakukan operasi

tulis/mengeluarkan data

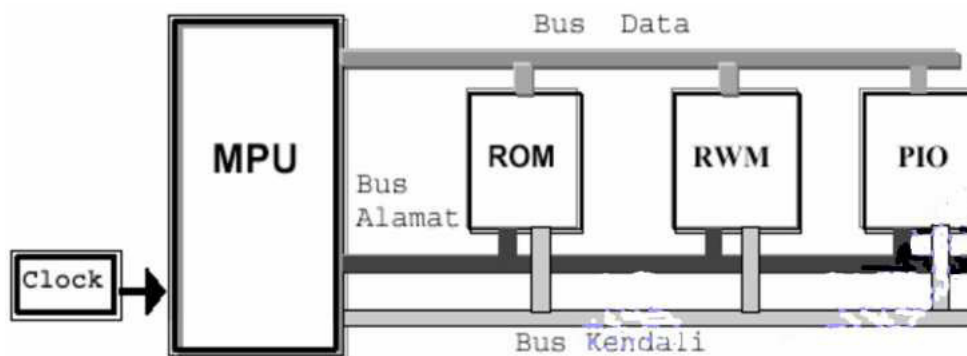
- RFSH* (Refresh: Penyegaran) Keluaran aktif rendah jika CP mengeluarkan alamat memori untuk menyegarkan memori mekanik
- HA LT* Keluaran aktif rendah pada saat CPU melaksanakan instruksi Halt/berhenti
- WA IT* Masukan dibuat aktif rendah oleh alat luar yang menyela kerja CPU
- INT* (Interrupt: interupsi) Masukan aktif rendah jika ada luar yang meminta layanan interupsi
- NMI* (Non Mascable Interrupt: interupsi yang tidak bisa dihalangi) Masukan aktif rendah jika ada selaan yang yang tak dapat dihalangi
- RESET* Masukan dibuat aktif rendah oleh alat luar untuk membuat CPU ada dalam keadaan awal
- BUSRQ* (Buss Request: pesan bus) Sinyal masukan yang dibuat aktif rendah jika ada alat luar yang meminjam bus sistem
- BUSAK* (Bus Acknowledge) Keluaran aktif rendah yang menandakan CPU mengijinkan peminjaman bus sistem.

Z-80 CPU dalam mengendalikan sistem menggunakan enam pin kendali dan empat diantaranya digunakan untuk berkomunikasi dengan Memori dan I/O. Cara berkomunikasi menggunakan status bit seperti tabel berikut :

Tabel 5.1. Operasi Komunikasi Memori

Pin Kendali				Operasi	Arah Data Bus
MREQ*	IORQ*	RD*	WR*		
0	1	0	1	Baca data dan memory	Input
0	1	1	0	Tulis data ke memory	output
1	0	0	1	Baca data dari I/O	Input
1	0	1	0	Tulis data ke I/O	output

Komponen utama sebuah sistem Mikroprosesor tersusun dari lima unit pokok: unit mikroprosesor atau Microprocessor Unit (MPU) atau CPU, unit memori baca atau Read Only Memory (ROM), unit memori baca tulis atau Read Write Memory (RWM), unit masukan keluaran terprogram atau Programmable Input Output (PIO) dan unit detak/Clock.



Gambar 5.2. Blok Diagram Sistem Mikroprosesor

MPU adalah sebuah CPU yang tersusun dari tiga bagian pokok yaitu :

- Control Unit (CU)
- Arithmetic Logic Unit (A LU)
- Register Unit (RU)

Sebagai CPU, MPU bekerja dan melakukan fungsi dasar yaitu fungsi Logika dan Aritmetika. Fungsi Logika antara lain fungsi AND, OR, XOR, CPL, dan NEG. Sedangkan fungsi Aritmetika antara lain: ADD, SUB, ADC, SBC, INC, dan DEC. Disamping fungsi pengolahan Aritmetika dan Logika MPU juga melakukan fungsi pengalihan data dengan menggunakan perintah MOV, atau LOAD, EXCHANGE, PUSH, dan POP. Untuk menyimpan program dan data yang digunakan pada sistem Mikroprosesor harus dilengkapi dengan Memori.

Jadi memori mutlak diperlukan dalam Sistem Mikroprosesor. Tanpa ada memori Sistem Mikroprosesor tidak dapat bekerja terutama memori program dalam ROM. I/O unit dipersiapkan untuk menghubungkan MPU dengan alat-alat input-output luar seperti Keyboard, Monitor, Printer, Mouse, dan sebagainya.

Sistem Bus

Mikroprosesor berkomunikasi dengan unit memori, unit I/O menggunakan saluran yang disebut dengan BUSS. Setiap Mikroprosesor dilengkapi dengan tiga bus sebagai berikut :

Tabel 5.2. Sistem Bus

Nama Buss	Sifat	Arah Data dari CPU	Jumlah Saluran
Bus Data	Dua arah	Masuk dan Keluar	8 bit
Bus Alamat	Satu arah	Keluar	16 bit
Bus Kendali	Satu arah	Masuk dan Keluar	10-12 bit

Alih data diantara MPU dengan komponen luar berlangsung pada Bus Data. Mikroprosesor standar memiliki saluran bus data 8 bit dua arah artinya alih data atau informasi berlangsung pada 8 saluran paralel dari MPU ke unit lain diluar MPU atau dari unit lain di luar ke MPU. Untuk menetapkan kemana data itu dikirim atau dari mana data itu diambil di gunakan bus alamat. Bus alamat bertugas menetapkan dan memilih satu lokasi memori atau satu lokasi I/O yang hendak di akses. Bus Kendali adalah seperangkat bit pengendali yang berfungsi mengatur: (1) Penyerempakan memori, (2) Penyerempakan I/O, (3) Penjadualan MPU, Interupsi, DMA , (4) Pembentuk Clock, dan Reset.

Clock

Merupakan bagian dari Sistem Mikroprosesor yang mengatur denyut kerja MPU. Sehingga Frekuensi Clock berkaitan dengan kecepatan kerja komputer. Beberapa jenis MPU ada yang menggunakan detak sistim tunggal dan ada juga sistim ganda (dual fase). Detak dapat dibangkitkan menggunakan sistim diskrit atau IC khusus. Intel memperkenalkan IC 8224 untuk penggerak detak.

Pengendalian Sistem Mikroprosesor

MPU dalam suatu sistem mikroprosesor dalam fungsinya sebagai pengendali system bekerja sebagai :

- Pengendali sistim
- Pengendali bus/saluran
- Dikendalikan oleh alat luar.

Pada Tabel berikut digambarkan Ekuivalensi sinyal-sinyal kendali beberapa jenis Mikroprosesor. Penyerempakan memori dan penyerempakan I/O pada pokoknya analogis. Digunakan prosedur jabat tangan. Dalam operasi “baca” suatu status sinyal “siap”(Ready) akan menunjukkan tersedianya data. Kemudian data dialihkan ke bus data. Pada beberapa alat I/O dibangkitkan suatu sinyal “pengakuan” (acknowledge) untuk memberitahukan penerimaan data. menggunakan sistim tak serempak (A sinkron). Pada sistem sinkron tidak diperlukan adanya pembangkitan sinyal pengakuan.

Tabel 5.3. Sinyal Kendali Mikroprosesor

Sinyal Pengendali	Jenis Mikroprosesor				
	8080/8228	8085	Z-80	6800	6502
PENGENDALI SISTIM	SYNC	-	M1	-	SYNC
	MEMR	RD&IO/M	RD&MEMRQ	R/WQ2	R/W&Q2
	MEMW	WR&IO/M	WR&MEMRO	R/W&Q2	R/W&Q2
	I/ORD	RD&IO/M	RD&IORQ	R/W&Q2	R/W&Q2
	I/OWR	WR&IO/M	WR&IORQ	R/W&Q2	R/W&Q2
PENGENDALI MPU	-	-	HALT	-	-
	READY	READY	WAIT	-	RDY
	INT	INTR	INT	I/O	I/O
	-	TRAP	NMI	NMI	NMI
	RESET	RESET	RESET	RESET	RESET
PENGENDALI BUS	HOLD	HOLD	BUSRQ	HALT	RDY
	HLDA	HLDA	BUSAK	BA&VMA	-

Ciri dari sistem sinkron adalah :

- Kecepatan yang lebih tinggi
- Jumlah saluran bus pengendali lebih sedikit
- Pembatasan kecepatan pada alat-alat I/O.

Pada sistem asinkron tercirikan adanya :

- Jumlah saluran bus pengendali lebih banyak
- Memungkinkan penggunaan piranti berkecepatan berbeda dalam satu system yang sama.

5.3.3. Transfer Data Paralel (PPI 8255)

Salah satu serpih perantara yang digunakan untuk pengantarmukaan paralel (paraleln interfacing) adalah Programmable Peripheral Interface (PPI) 8255. Serpih ini diproduksi oleh Intel Corporation dan dikemas dalam bentuk 40 pin dual in line package dan dirancang untuk berbagai fungsi pengantarmukaan dalam mikroprosesor. PPI 8255 memiliki 24 pin I/O yang dibagi menjadi tiga port masing-masing 8 bit. Port - port tersebut adalah port A (PA0- PA7), port B (PB0-PB7) dan port C (PC0-PC7). Sebagai jalur untuk transfer data dari dan ke PPI 8255 disediakan saluran 8 bit bus

data (D0-D7). Bus data dari PPI ini dapat dihubungkan langsung dengan bus data dari mikrokomputer.

Proses pembacaan dan penulisan data dari dan ke PPI 8255 dapat dilakukan dengan program. Salah satu register yang akan dituju dari ketiga port dan register control ditentukan dengan kombinasi penyemat A0 dan A1. Tabel 5.4 menyatakan format operasi dasar pembacaan atau penulisan dan pengalamatan dari 3 port I/O dan register kendali PPI 8255.

Kontrol Group A dan Group B

Konfigurasi fungsional setiap port diprogram oleh sistem perangkat lunak, yaitu melalui output CPU (control word) ke PPI 8255. Control word berisikan mode, bit set, bit reset, dan sebagainya. Ini merupakan suatu proses inisialisasi konfigurasi fungsional dari PPI 8255. Setiap blok kontrol (Group A dan Group B) menerima perintah dari kontrol logika read/write, menerima control word dari bus data untuk keperluan perintah pada setiap port. Kontrol group A adalah port A dan Port C upper (C7..C4) dan control group B adalah port B dan port C lower (C3..C0), sedangkan register kontrol hanya dapat dioperasikan untuk output (operasi write). Port A, B dan C Port A. Output 8 bit data latch/buffer dan input 8 bit data latch. Port B. Input/output 8 bit data latch/buffer dan input 8 bit data buffer. Port C. Output 8 bit data latch/buffer dan input 8 bit data buffer. Dan port ini dapat dibagi atas dua port (setiap port 4 bit data latch) yang digunakan secara bersama dengan port A dan port B untuk mengontrol sinyal output dan status sinyal input.

Deskripsi Operasional PPI 8255

Ada tiga mode operasi yang dapat dipilih oleh sistem perangkat lunak untuk mengoperasikan PPI 8255 yaitu :

Mode 0 - Basic Input/Output

Mode ini digunakan untuk input/output sederhana langsung ke port I/O. Peralatan luar yang dihubungkan selalu siap untuk mengirimkan/menerima data, sehingga mode ini tidak tergantung pada waktu. Semua port A, B dan C bisa bekerja pada mode ini. Port-port PPI hanya bisa digunakan sebagai port input atau port output dari sistem mikroprosesor. Port A dan port B masing-masing dapat digunakan sebagai 8 bit masukan saja atau 8 bit keluaran saja. Sedangkan port C dapat digunakan sebagai empat (4) bit masukan atau empat (4) bit keluaran seperti port A dan port B.

Mode 1 - Strobe Input/Output

Mode ini digunakan untuk peralatan luar yang mempunyai data valid pada saat – saat tertentu, sehingga diperlukan sinyal-sinyal pemicu (strobe) pada I/O agar data segera dapat dikirim, sehingga mode ini tergantung pada waktu. Pada mode ini port A dan port B bisa ditentukan sebagai port masukan atau keluaran data, sedangkan port C berfungsi sebagai pembawa sinyal status. Transfer data mode ini merupakan sinyal terprogram bersyarat.

Mode 2 - Bidirectional Bus

Mode ini mampu mengirim/menerima data dalam dua arah (bidirectional handshake data transfer). Mode ini menyebabkan port A bisa berfungsi sebagai masukan sekaligus keluaran yang dilengkapi dengan sinyal jabat tangan 5 bit dari port C

sebagai kontrol port A . Mode ini tidak tersedia untuk port B.

5.4. Sistem Interface Input/Output antara Sistem Digital dan Sistem Analog

Penggunaan komputer saat ini tidak lagi terbatas pada pengolahan dan manipulasi data saja tetapi sudah digunakan untuk mengontrol berbagai peralatan seperti penghitung pulsa telepon, menyalakan/mematikan lampu secara otomatis, dan lain sebagainya. Dengan penggunaan komputer seperti yang telah disebutkan di atas maka seolah-olah computer berperan sebagai manusia yang dapat diprogram untuk menjalankan apa yang dikehendaki oleh programernya. Antara sistem digital (sebagai pengontrol) dan sistem analog (sebagai peralatan yang dikontrol) harus terdapat suatu jembatan yang menghubungkan kedua sistem tersebut. Jembatan ini selanjutnya disebut sistem interface IO. Jadi untuk sistem kontrol secara digital ini selalu terdiri dari 3 bagian yaitu : sistem digital, sistem interface IO dan sistem analog. Sistem digital merupakan sistem yang menjadi otak dari sistem secara keseluruhan. Sistem digital ini membaca kondisi dari sistem analog melalui sistem interface IO dan mengontrol sistem analog melalui sistem interface IO. Sistem kontrol secara digital ini menggantikan sistem kontrol manual yang menggunakan switch mekanik dan diatur secara manual pula. Selain itu dengan sistem kontrol secara digital ini, kondisi sistem analog yang dikontrol dapat pula dimonitor. Keadaannya. Sistem analog merupakan bagian dari peralatan analog yang aktivitasnya dikontrol oleh sistem digitalnya melalui sistem interface IO. Sistem analog dapat berupa lampu bolam 220 volt, motor AC, bahkan sampai ke peralatan industri yang menggunakan arus besar.

Disini terlihat bahwa sistem interface IO sangat penting peranannya yaitu untuk menginterfacekan sistem digital yang hanya mengenal kondisi 'H', yang ekuivalen dengan tegangan 4.5 volt sampai 5 volt dan kondisi 'L' yang setara dengan tegangan dibawah 1.2volt dengan sistem analog dengan tegangan 220 VAC dengan konsumsi arus yang paling tidak 1A ke atas. Dari kondisi seperti di atas maka perlulah bagian digital dan bagian analog ini dilewatkan sistem interface yang secara elektronik terisolasi antar bagiannya. Teknik interface IO disini ada beberapa teknik dan tiap teknik tersebut mempunyai keistimewaan pada aplikasi tertentu.

Contoh Aplikasi

Dengan menggunakan sebuah PC diharapkan dapat mengontrol 10 buah titik lampu yang menyala/mati pada jam-jam tertentu. Melalui sebuah PPI card (dengan menggunakan chip PPI 8255) dapat dikontrol 24 buah beban. Output PPI adalah TTL level sedangkan untuk lampu yang digunakan adalah lampu TL biasa. Untuk menginterfacekan antara PPI (sistem digital) dengan lampu (sistem analog) digunakan relay 5volt. Contoh aplikasi ini adalah salah satu contoh penggunaan relay sebagai interface antara sistem digital dan sistem analog.

Sistem Interface I/O

Sistem interface I/O yang paling baik adalah sistem interface dimana sistem digital dan sistem analognya terisolasi, terpisah. Biasanya digunakan relay atau optocoupler. Penggunaan relay lebih mudah namun lebih sering menimbulkan masalah karena relay dapat menghasilkan noise pada sistem digital pada saat relay berubah keadaan. Selain itu penggunaan relay membutuhkan daya yang lebih besar jika

dibandingkan dengan penggunaan optoisolator. Sistem interface yang baik pada umumnya menggunakan optoisolator atau yang lebih dikenal dengan optocoupler seperti 4N31 atau 4N35. Dengan menggunakan optocoupler arus yang digunakan lebih sedikit paling tidak 10 mA -15 mA. Penggunaan optocoupler seperti 4N35 lebih disukai daripada penggunaan relay secara langsung.

Optoisolator

Optoisolator merupakan komponen yang digunakan sebagai komponen kontrol I/O untuk peralatan yang beroperasi dengan tegangan DC atau AC. Sebuah optocoupler terdiri dari GaAs LED dan phototransistor NPN yang terbuat dari silikon. Untuk rangkaian penggunaan optoisolator dapat dilihat pada gambar 5.6. Pada gambar 5.7a. optoisolator mendapat input TTL berbentuk sinyal kotak sehingga outputnya juga berupa sinyal kotak namun level tegangan berubah menjadi 0-+24 volt.

5.4.1. Konversi Analog dan Digital

Perangkat elektronika modern kebanyakan melakukan pengolahan data secara digital. Karena sinyal pada umumnya secara alamiah merupakan sinyal analog, maka keberadaan peranti pengubah (konversi) data dari analog ke digital, dan sebaliknya menjadi sangat vital. Berikut ini membahas spesifikasi peranti konversi data yang lazim dicantumkan dalam lembar data yang disediakan oleh produsen.

1. Fungsi Transfer Ideal Konverter Analog-ke-Digital

Secara teoritis, fungsi transfer ideal untuk konverter analog-ke-digital (A DC, analog-to-digital converter) berbentuk garis lurus. Bentuk ideal garis lurus hanya dapat dicapai dengan konverter data beresolusi tak-hingga. Karena tidak mungkin mendapatkan resolusi tak hingga, maka secara praktis fungsi transfer ideal tersebut berbentuk gelombang tangga seragam.

ADC ideal secara unik dapat merepresentasikan seluruh rentang masukan analog tertentu dengan sejumlah kode keluaran digital. Pada gambar 1 ditunjukkan bahwa setiap kode digital merepresentasikan sebagian dari rentang masukan analog total. Oleh karena skala analog bersifat kontinyu sedangkan kode digital bersifat diskrit, maka ada proses kuantisasi yang menimbulkan kekeliruan (galat). Apabila jumlah kode diskritnya (yang mewakili rentang masukan analog) ditambah, maka lebar undak (step width) akan semakin kecil dan fungsi transfer akan mendekati garis lurus ideal. Lebar satu undak (step) didefinisikan sebagai 1 LSB (least significant bit) dan unit ini digunakan sebagai unit rujukan untuk besaran-besaran lain dalam spesifikasi peranti konversi data. Unit 1 LSB itu juga digunakan untuk mengukur resolusi konverter karena ia juga menggambarkan jumlah bagian atau unit dalam rentang analog penuh.

Resolusi ADC selalu dinyatakan sebagai jumlah bit-bit dalam kode keluaran digitalnya. Misalnya, ADC dengan resolusi n -bit memiliki 2^n kode digital yang mungkin dan berarti juga memiliki 2^n tingkat undak (step level). Meskipun demikian, karena undak pertama dan undak terakhir hanya setengah dari lebar penuh, maka rentang skala-penuh (FSR, full-scale range) dibagi dalam $(2^n - 1)$ lebar undak. Karenanya, untuk konverter n -bit.

2. Fungsi Transfer Ideal Konverter Digital-ke-Analog

Konverter digital-ke-analog (DAC, digital-to-analog converter) merepresentasikan sejumlah kode masukan digital diskrit dengan sejumlah nilai keluaran analog diskrit. Karenanya, fungsi transfer DAC adalah sederet titik-titik diskrit. Untuk DAC, 1 LSB adalah tinggi undak (step height) antara dua keluaran analog yang berdekatan. Secara sederhana, DAC dapat dibayangkan sebagai potensiometer terkendali secara digital yang keluarannya merupakan bagian dari tegangan analog skala-penuh, bergantung pada kode masukan digitalnya.

3. Kesalahan (Galat) Statik

Kesalahan statik adalah kesalahan (galat) yang mempengaruhi akurasi konverter bila konverter tersebut mengkonversi sinyal statik (DC, direct current). Yang termasuk dalam jenis galat statik adalah galat offset, galat bati (gain error), non-linieritas integral, dan non-linieritas diferensial. Masing-masing galat dapat diekspresikan dalam unit LSB atau kadang-kadang sebagai prosentase dari FSR.

4. Efek Kuantisasi

Masukan analog ADC biasanya berupa sinyal kontinyu yang memiliki kemungkinan nilai tak terhingga banyaknya, sedangkan keluaran digital merupakan fungsi diskrit dengan kemungkinan nilai yang dibatasi jumlahnya oleh resolusi konverter. Oleh karena itu dapat dipahami apabila dalam pengubahan bentuk analog ke bentuk digital, beberapa sinyal analog berbeda nilai yang direpresentasikan dengan tegangan berbeda pada masukannya, direpresentasikan dengan kode digital yang sama pada keluarannya. Beberapa informasi hilang dan distorsi tertambahkan pada sinyal. Untuk fungsi transfer gelombang-tangga ideal pada ADC, kesalahan antara masukan digital dengan bentuk digital keluarannya memiliki fungsi rapat probabilitas yang seragam (uniform probability density function) jika sinyal masukannya diasumsikan acak. Kesalahan itu dapat bervariasi dalam rentang $\pm \frac{1}{2}$ LSB atau $\pm q/2$, di mana q adalah lebar satu undak.

Waktu Penetapan (settling time)

Waktu penetapan adalah waktu yang diperlukan DAC untuk mencapai nilai akhir sesudah terjadi perubahan kode digital masukan. Spesifikasi untuk waktu penetapan ini biasanya dicantumkan bersama dengan laju slew (slew rate).

Slew Rate

Slew rate merupakan keterbatasan yang melekat (inherent) pada penguat keluaran yang ada pada DAC yang membatasi laju perubahan tegangan keluaran sesudah terjadi perubahan kode digital masukan. Besaran slew rate dinyatakan dalam satuan nvolt/m s , dan pada umumnya bernilai antara 0,2 sampai beberapa ratus V/m s .

Koefisien Temperatur

Aus atau usangnya komponen pembentuk peranti konversi data karena umur akan menghasilkan atau memperparah beberapa jenis kesalahan (galat) pada temperatur operasi yang berubah-ubah. Galat offset dapat berubah akibat koefisien temperatur penguat dan komparator. Kesalahan dapat juga terjadi karena bergesernya tegangan rujukan atau berubahnya nilai resistor tangga (akibat panas). Pada dasarnya, hampir semua kesalahan, kecuali resolusi dan galat kuantisasi, terpengaruh oleh

koefisien temperatur komponen dalam konverter data.

Overshoot dan Glitch

Overshoot dan glitch muncul pada saat terjadi perubahan kode digital masukan pada DAC. Glitch adalah lonjakan tegangan sangat singkat (sehingga berbentuk seperti paku) yang terjadi akibat ketakserempakan pensaklaran tiap-tiap bit. Jika masukan DAC berubah dari 01111 menjadi 10000, misalnya, maka 4 buah saklar pada DAC membuka dan sebuah saklar menutup. Jika kecepatan membuka/menutup tiap-tiap saklar tidak sama maka terdapat saat sangat singkat dimana keluaran menunjukkan nilai tak-sebenarnya baru kemudian mencapai kondisi mapan (settle). Glitch seperti ini dapat diredam pada tegangan keluaran DAC karena penguat keluaran umumnya tidak dapat mengikuti laju perubahan yang sangat cepat. Penguat keluaran menghasilkan overshoot atau ayunan tegangan yang dapat diminimalkan tetapi tidak dapat dihilangkan sama sekali.

Kemelesetan Jangka Panjang

Dalam jangka panjang, akibat usia komponen terutama resistor semikonduktor, karakteristik peranti konverter data menjadi berubah. Karakteristik peranti yang paling terpengaruh adalah linieritas dan galat offset.

Laju Konversi Data

Laju konversi data adalah kecepatan ADC atau DAC melakukan konversi data berulang. Hal ini dipengaruhi oleh waktu tunda perambatan pada rangkaian pencacah, tangga saklar dan komparator, tangga RC dan waktu penetapan penguat, serta laju slew penguat dan kompartor. Laju konversi didefinisikan sebagai jumlah konversi per detik atau dapat juga dinyatakan sebagai waktu konversi, yakni lamanya waktu yang diperlukan untuk menyelesaikan satu proses konversi (termasuk efek waktu penetapan).

Laju Detak

Laju detak dispesifikasikan sebagai laju pulsa minimum dan maksimum yang harus dipasang pada peranti konverter data. Terdapat hubungan tetap antara laju konversi minimum dengan laju detak, tergantung pada tipe dan akurasi konverter. Semua yang mempengaruhi laju konversi dari suatu ADC membatasi laju detak.

5.5. Penutup

5.5.1. Bahan diskusi dan tugas

1. Sebutkan tiga jenis arsitektur mikroprosesor?
2. Jelaskan apa yang dimaksud dengan kemasan PDIP, PLCC, TQFP dan SOIC?
3. Apa fungsi PPI 8255 dan bagaimana menginisiasi port-portnya?
4. Mengapa peralatan luar perlu diisolasi menggunakan optoisolator?
5. Jelaskan prinsip kerja dari ADC dan DAC?

5.6. Daftar Pustaka

29. J. Webb, *Industrial Control Electronics*, Macmillan Publish., New York, 1992.
30. J. Webb, *Programmable Logic Control*, Macmillan Publish., New York, 1992.
31. Frank D Petruzella, *Elektronik Industri* (terjemahan), Penerbit Andi, Yogyakarta, 2001.
32. William Bolton, *Programmable Logic Controller (PLC)*, Sebuah Pengantar, Edisi ketiga, Penerbit Erlangga, 2004.
33. M. Budiyanto dan A. Wijaya, *Pengenalan Dasar-dasar PLC (Programmable Logic Controller)*, Penerbit Gaya Media, Yogyakarta, 2003.
34. Endra Ditawarna, *Robotika, Desain, Kontrol dan Kecerdasan Buatan*, Penerbit Andi, Yogyakarta, 2006.
35. David G. Alciator dan Michael B Histan, *Introduction to Mechatronics and Measurement Systems*, Mc Graw Hill, New York, 2003.

5.7. Senarai

Ada tiga jenis arsitektur Mikroprosesor :

1. Arsitektur I/O Terisolasi
2. Arsitektur I/O Terpetakan dalam Memori
3. Arsitektur Harvard

Ada empat jenis bentuk kemasan Mikroprosesor:

- PDIP: Pastic Dual Inline Package
- PLCC: Plastic J-Lieded Chip Carrier
- TQFP: Plastic Gull Wing Quad Flat Package
- SOIC: Plastic Gull-wing Small Outline.

MPU adalah sebuah CPU yang tersusun dari tiga bagian pokok yaitu :

- Control Unit (CU)
- Arithmetic Logic Unit (A LU)
- Register Unit (RU)

Bus Kendali adalah seperangkat bit pengendali yang berfungsi mengatur :
(1) Penyerempakan memori, (2) Penyerempakan I/O, (3) Penjadualan MPU, Interupsi, DMA , (4) Pembentuk Clock, dan Reset.

Mikroprosesor sebagai komponen utama dalam sistem mikroprosesor dapat mdikelompokkan menurut : (a) Teknologi yang digunakan; (b) Jumlah Bit Data; (c) Kemampuan atau Karakteristik Mikroprosesor. Agar Mikroprosesor dapat mengendalikan peralatan maka diperlukan perantaraan. PPI 8255 merupakan serpih perantara paralel yang telah banyak digunakan. Mengendalikan peralatanluar perlu memperhatikan level tegangan antar keduanya agar peralatan dapat bekerja dengan baik dan aman. Antarmuka analog memerlukan pengonversi agar mikroprosesor dapat memahami informasi data secara digital.

BAB 6

MIKRO KONTROLER

6.1. Deskripsi Singkat

Tujuan dari mempelajari mikro kontroler adalah untuk dapat memahami teknologi semikonduktor yang sangat membantu perkembangan dunia elektronika.

6.2. Tujuan Intruksional Khusus

Setelah materi ini diajarkan mahasiswa dapat memahami arsitektur dan bagian-bagian mikrokontroler serta mampu membuat program mikrokontroler.

6.3. Penyajian

6.3.1. Pengantar

Mikrokontroler sebagai teknologi baru yaitu teknologi semikonduktor kehadirannya sangat membantu perkembangan dunia elektronika. Dengan arsitektur yang praktis tetapi memuat banyak kandungan transistor yang terintegrasi, sehingga mendukung dibuatnya rangkaian elektronika yang lebih portable. Mikrokontroler dapat diproduksi secara massal sehingga harganya menjadi lebih murah dibandingkan dengan mikroprosesor, tetapi tetap memiliki kelebihan yang bisa diandalkan. Mikrokontroler memiliki perbandingan ROM dan RAM-nya yang besar, artinya program kontrol disimpan dalam ROM (bisa Masked ROM atau Flash PEROM) yang ukurannya relative lebih besar, sedangkan RAM digunakan sebagai tempat penyimpanan sementara, termasuk register-register yang digunakan pada mikrokontroler yang bersangkutan.

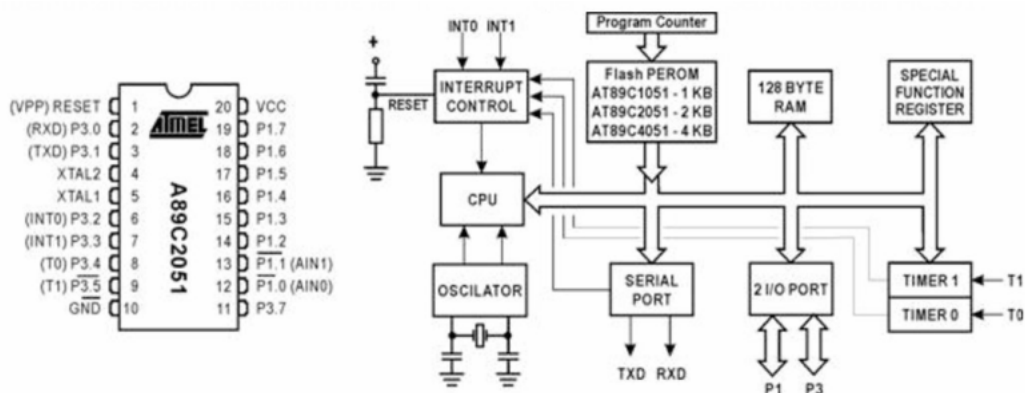
Penggunaan utama dari mikrokontroler adalah untuk mengontrol operasi dari mesin. Strategi kendali untuk mesin tertentu dimodelkan dalam program algoritma pengaturan yang ditulis dalam bahasa rakitan (assembly language). Program tersebut selanjutnya ditranslasi ke kode mesin digital yang selanjutnya disimpan di dalam media penyimpanan digital yang disebut ROM (Lihat Gambar . Pendekatan disain dari mikrokontroler dan mikroprosesor adalah sama. Jadi mikroprosesor merupakan rumpun dari suatu mikrokontroler. Mikrokontroler terdiri dari fitur-fitur yang terdapat dalam suatu mikroprosesor yaitu ALU, SP, PC dan register-register termasuk fitur dari

ROM, RAM, input/output paralel dan input/output pencacah (counter seri). Dalam penyajian mikrokontroler pembahasan akan ditekankan pada mikrokontroler.

6.3.2. Konstruksi dasar AT89Cx051

Mikrokontroler MCS51 adalah Mikrokontroler yang paling populer saat ini. Keluarga ini diawali oleh Intel yang mengenalkan IC Mikrokontroler type 8051 pada awal tahun 1980-an, 8051 termasuk sederhana dan harganya murah sehingga banyak digemari, banyak pabrik IC besar lain yang ikut memproduksinya, tentu saja masing-masing pabrik menambahkan kemampuan pada mikrokontroler buatannya meskipun semuanya masih dibuat berdasarkan 8051. Sampai kini sudah ada lebih 100 macam mikrokontroler turunan 8051, sehingga terbentuklah sebuah ‘keluarga besar mikrokontroler’ dan biasa disebut sebagai MCS51.

Gambar 6.1 memperlihatkan susunan kaki AT89C2051, susunan kaki ini sama persis dengan AT89C1051 dan AT89C4051. Demikian pula ketiga IC mempunyai konstruksi di dalam chip yang sama persis, perbedaannya hanya terletak pada kapasitas Flash PEROM. IC AT89Cx051 hanya memerlukan tambahan 3 kapasitor, 1 resistor dan 1 Xtal serta catu daya 5 Volt. Kapasitor 10 mikro-Farad dan resistor 10 Kilo Ohm dipakai untuk membentuk rangkaian reset, dengan adanya rangkaian reset ini AT89Cx051 otomatis di-reset begitu rangkaian menerima catu daya. Xtal dengan frekuensi maksimum 24 MHz dan 2 kapasitor 30 pico-Farad dipakai untuk melengkapi rangkaian oscilator pembentuk clock yang menentukan kecepatan kerja mikrokontroler.



Gambar 6.1. Susunan kaki AT89Cx051 dan kontruksi dalam chip

6.3.3. Memori Pada AT89Cx051

Memori merupakan bagian yang sangat penting bagi mikrokontroler, diperlukan 2 macam memori yang sifatnya berbeda. Read Only Memory (ROM) yang isinya tidak berubah meskipun IC kehilangan catu daya, dipakai untuk menyimpan program, begitu di-reset mikrokontroler akan langsung bekerja dengan program dalam ROM tersebut. Sesuai dengan keperluannya, dalam susunan MCS51 memori penyimpan program ini dinamakan sebagai MEMORI PROGRAM. Random Access Memory (RAM) isinya akan sirna begitu IC kehilangan catu daya, dipakai untuk menyimpan data pada saat program bekerja. Di samping untuk data, RAM dipakai

pula untuk Stack. RAM yang dipakai untuk menyimpan data ini disebut pula sebagai MEMORI DATA.

Memori Data yang disediakan dalam chip AT89Cx051 sebesar 128 byte, meskipun hanya kecil saja tapi untuk banyak keperluan memori kapasitas itu sudah mencukupi. Sarana Input/Output yang disediakan cukup banyak dan bervariasi. AT89Cx051 mempunyai 17 jalur Input/Output paralel. Jalur Input/Output paralel ini dikenal sebagai Port 1 (P1.0..P1.7) dan Port 3 (P3.0..P3.5 dan P3.7). AT89Cx051 dilengkapi UART (Universal Asynchronous Receiver/Transmitter) yang biasa dipakai untuk komunikasi data secara seri. Jalur untuk komunikasi data seri (RXD dan TXD) diletakkan berhimpitan dengan P1.0 dan P1.1 di kaki nomor 2 dan 3, sehingga kalau sarana UART ini dipakai maka P1.0 dan P1.1 tidak lagi bisa dipakai untuk jalur input/output paralel. Timer 0 dan Timer 1 masing-masing adalah untaian pencacah biner 16 bit (16 bit binary counter) di dalam chip yang dipakai sebagai sarana input/output yang bekerja menurut fungsi waktu. Clock penggerak untaian pencacah ini bisa berasal dari oscillator kristal atau clock yang diumpan dari luar lewat T0 dan T1. T0 dan T1 berhimpitan dengan P3.4 dan P3.5, sehingga P3.4 dan P3.5 tidak bisa dipakai untuk jalur input/output paralel kalau T0 dan T1 dipakai.

AT89Cx051 mempunyai 5 sumber pembangkit interupsi, 2 diantaranya adalah sinyal interupsi yang diumpankan ke kaki INT0 dan INT1, kedua kaki ini berhimpitan dengan P3.2 dan P3.3 sehingga tidak bisa dipakai sebagai jalur input/output paralel kalau INT0 dan INT1 dipakai untuk menerima sinyal interupsi. 3 sumber interupsi yang lain berasal dari sarana komunikasi data seri dan dari sistem Timer 0 dan Timer 1. Port 1, Port 2, UART, Timer 0, Timer 1 dan sarana lainnya merupakan register yang secara fisik merupakan RAM khusus, yang ditempatkan di Special Function Register (SFR).

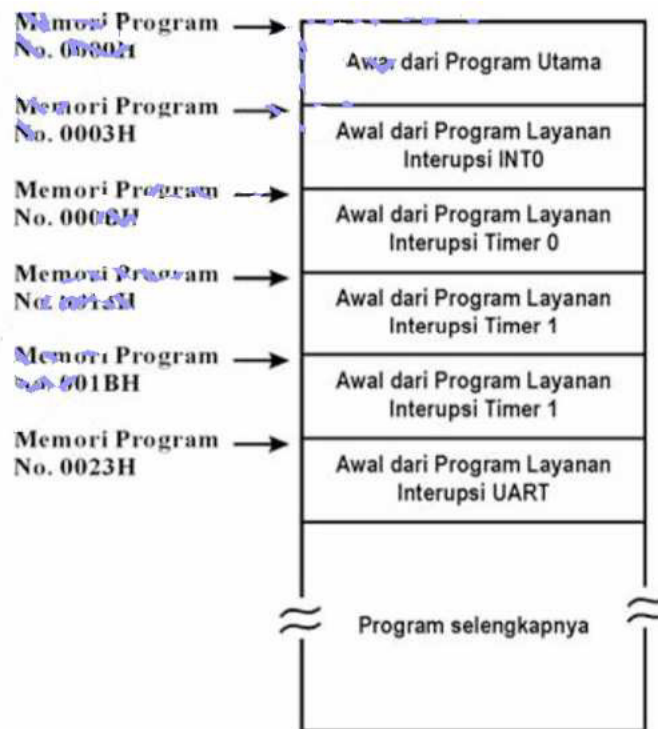
Dasar kerja program :

Program untuk mengendalikan kerja dari mikrokontroler disimpan di dalam memori program. Program pengendali tersebut merupakan kumpulan dari instruksi kerja mikrokontroler, 1 instruksi MCS51 merupakan kode yang panjangnya bisa satu sampai empat byte. Sepanjang mikrokontroler bekerja, instruksi tersebut byte demi byte diambil ke CPU dan selanjutnya dipakai untuk mengatur kerja mikrokontroler. Proses pengambilan instruksi dari memori program dikatakan sebagai 'fetch cycles' dan saat-saat CPU melaksanakan instruksi disebut sebagai 'execute cycles'. Semua mikrokontroler maupun mikroprosesor dilengkapi sebuah register yang berfungsi khusus untuk mengatur 'fetch cycles', register tersebut dinamakan sebagai Program Counter. Nilai Program Counter secara otomatis bertambah satu setiap kali selesai mengambil 1 byte isi memori program, dengan demikian isi memori program bisa berurutan diumpankan ke CPU. Saat MCS51 di-reset, isi Program Counter di-reset menjadi 0000. Artinya sesaat setelah reset isi dari memori program nomor 0 dan seterusnya akan diambil ke CPU dan diperlakukan sebagai instruksi yang akan mengatur kerja mikrokontroler.

Dengan demikian, awal dari program pengendali MCS51 harus ditempatkan di memori nomor 0, setelah reset MCS51 menjalankan program mulai dari memori-program nomor 0000, dengan melakukan proses 'fetch cycles' dan 'execute cycles' terus menerus tanpa henti. Jika sarana interupsi diaktifkan, dan tegangan di kaki INT0 (kaki nomor 6) merubah dari '1' menjadi '0', maka proses menjalankan program di atas akan dihentikan sebentar, mikrokontroler melayani dulu permintaan interupsi,

selesai melayani permintaan interupsi CPU akan melanjutkan mengerjakan program utama lagi. Untuk melaksanakan hal tersebut, pertama-tama CPU menyimpan nilai Program Counter ke mStack (Stack merupakan satu bagian kecil dari data memori-RAM), kemudian mengganti isi Program Counter dengan 0003. Artinya MCS51 akan melaksanakan program yang ditempatkan di memori program mulai byte ke 3 untuk melayani interupsi yang diterima dari kaki INT0.

Adalah tugas programmer untuk mengatur agar program yang dipakai untuk melayani interupsi lewat INT0 diletakkan disitu. Selesai melayani interupsi, nilai Program Counter yang tadi disimpan ke dalam Stack akan dikembalikan ke Program Counter, dengan demikian CPU bisa melanjutkan pekerjaan di] program Utama. Selain INT0, AT89Cx051 bisa menerima interupsi dari INT1, dari UART dan dari Timer. Agar permintaan interupsi itu bisa dilayani dengan program yang berlainan, maka masing-masing sumber interupsi itu mempunyai nomor awal program untuk layanan interupsi yang berlainan. Nomor-nomor awal tersebut digambarkan dalam Gambar 6.2.



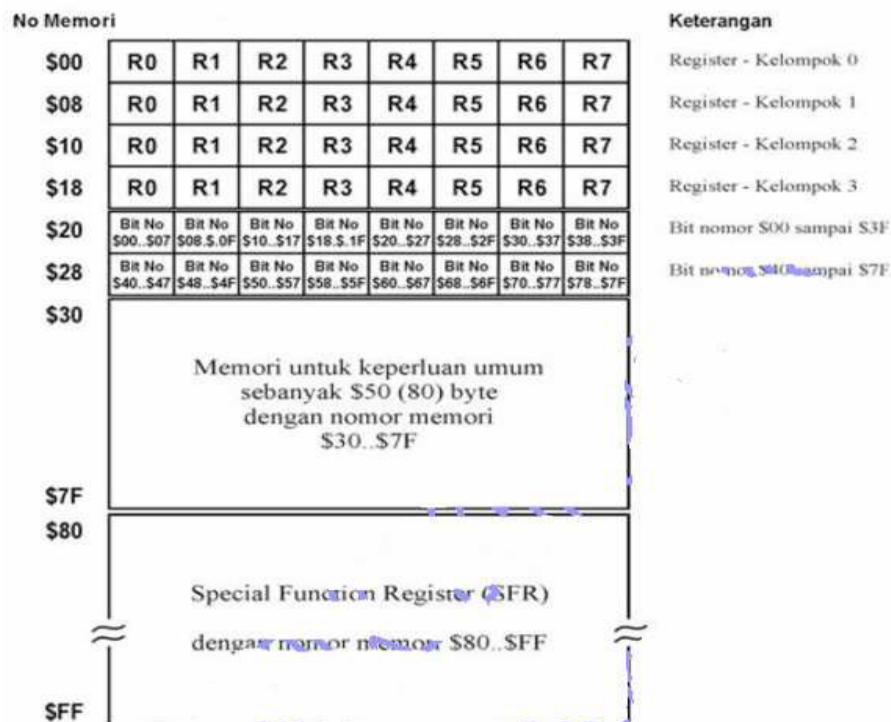
Gambar 6.2. Peta Memori Program

RAM dan Register dalam AT89Cx051

Bagi mereka yang sudah terbiasa memakai komputer, kapasitas Random Access Memory (RAM) yang dimiliki AT89Cx051 ‘mengerikan’ karena sangat sedikit, hanya 256 byte! Itupun tidak semuanya bisa dipakai sebagai memori penyimpanan biasa, lebih dari setengahnya merupakan memori dengan keperluan khusus yang biasa dikenal sebagai register. Meskipun demikian bagi mikrokontroler kapasitas itu sudah mencukupi. Dalam pengertian MCS51, Random Access Memory dalam chip AT89Cx051 adalah memori-data, yaitu memori yang dipakai untuk menyimpan data, sedangkan Flash PEROM merupakan memori penampung program pengendali AT89Cx051, dikenal sebagai memori-program. Karena kedua memori itu

memang dibedakan dengan tegas, maka kedua memori itu mempunyai penomoran yang terpisah. Memori-program dinomori sendiri, pada AT89C2051 mulai dari nomor \$0000 sampai \$07FF. Sedangkan memori-data yang hanya 256 byte dinomori dari nomor \$00 sampai \$FF.

Seperti terlihat dalam denah memori-data Gambar 6.3, memori-data dibagi menjadi dua bagian, memori nomor \$00 sampai \$7F merupakan memori seperti RAM selayaknya meskipun beberapa bagian mempunyai kegunaan khusus, sedangkan memori nomor \$80 sampai \$FF dipakai sangat khusus yang dinamakan sebagai Special Function Register (akan dibahas tersendiri dibagian lain). Memori-data nomor \$00 sampai \$7F bisa dipakai sebagai memori penyimpan data biasa, dibagi menjadi 3 bagian :



Gambar 6.3. Denah Memori-data

1. Memori nomor \$00 sampai \$18 selain sebagai memori-data biasa, bisa pula dipakai sebagai Register Serba Guna (General Purpose Register).
2. Memori nomor \$20 sampai \$2F selain sebagai memori-data biasa, bisa dipakai untuk menyimpan informasi dalam level bit.
3. Memori nomor \$30 sampai \$7F (sebanyak 80 byte) merupakan memori-data biasa, bisa dipakai untuk menyimpan data maupun dipakai sebagai Stack.

Register Serba Guna

Register Serba Guna (General Purpose Register) menempati memori-data nomor \$00 sampai \$18, memori sebanyak 32 byte ini dikelompokkan menjadi 4 Kelompok Register (Register Bank), 8 byte memori dari masing-masing Kelompok itu dikenali sebagai Register 0, Register 1 .. Register 7 (R0, R1, R2, R3, R4, R5, R6 dan R7). Dalam penulisan program memori-memori ini bisa langsung disebut sebagai

R0, R1, R2, R3, R4, R5, R6 dan R7, tidak lagi dengan nomor memori. Dengan cara ini instruksi yang terbentuk bisa lebih sederhana dan bekerja lebih cepat. Pengertian ini bisa diperjelas dengan contoh 2 instruksi berikut :

```
MOV A,$04
MOV A,R4
```

Instruksi pertama mempunyai makna isi memori-data nomor 4 di-copy-kan ke A kumulator A, sedangkan instruksi kedua artinya isi R4 di-copy-kan ke A kumulator A. Karena R4 menempati memori-data nomor 4, jadi kedua instruksi itu berakibat sama bagi Akumulatorm A. Tapi saat diterjemahkan ke kode mesin, intruksi pertama dirubah menjadi E5 04 (heksadesimal) dan instruksi kedua menjadi E6 (heksadesimal), jadi instruksi kedua lebih sederhana dari instruksi pertama.

Memori level Bit

Memori-data nomor \$20 sampai \$2F bisa dipakai menampung informasi dalam level bit. Setiap byte memori di daerah ini bisa dipakai menampung 8 bit informasi yang masing- masing dinomori tersendiri, dengan demikian dari 16 byte memori yang ada bisa dipakai untuk menyimpan 128 bit (16 x 8 bit) yang dinomori dengan bit nomor \$00 sampai \$7F. Informasi dalam level bit tersebut masing-masing dapat di-‘1’-kan, di-‘0’-kan dengan instruksi. Instruksi SETB \$00 mengakibatkan memori-bit nomor 0 menjadi ‘1’, atau sama dengan membuat bit nomor 0 dari memori-data nomor \$20 menjadi ‘1’, sedangkan bit-bit lainnya dalam memori nomor \$20 tidak berubah nilai. Sedangkan instruksi CLR \$7F mengakibatkan memori-bit nomor \$7F menjadi ‘0’, satau sama dengan membuat bit nomor 7 dari memori-data nomor \$2F menjadi ‘0’, sedangkan bit-bit lainnya dalam memori nomor \$2F tidak berubah nilai. Pengertian ini dipertegas dengan instruksi-instruksi berikut: MOV \$21,#\$0F Sama dengan hasil kerja instruksi-instruksi berikut :

- SETB \$08
- SETB \$09
- SETB \$0A
- SETB \$0B
- CLR \$0C
- CLR \$0D
- CLR \$0E
- CLR \$0F

Instruksi MOV \$21,#\$0F mempunyai makna mengisi memori-data nomor \$21 dengan nilai \$0F (atau bilangan biner 00001111), berarti mengisi memori-bit nomor \$0F sampai \$08 dengan bilangan biner 00001111 yang bisa dinyatakan dengan 8 baris instruksi berikutnya.

Special Function Register (SFR)

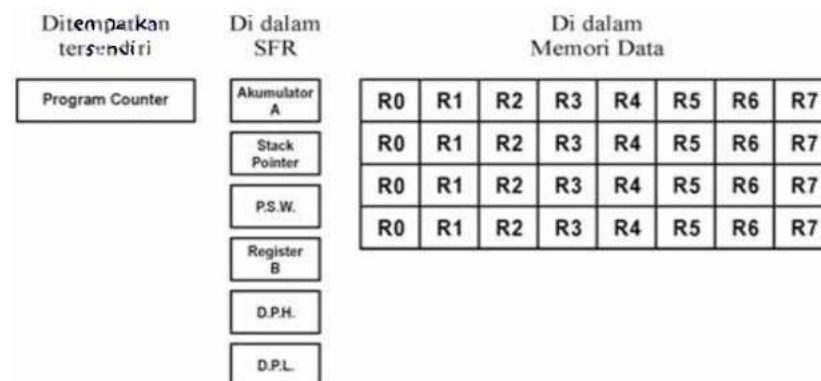
Register Khusus (SFR - Special Function Register) adalah satu daerah RA M dalam IC keluarga MCS51 yang dipakai untuk mengatur perilaku MCS51 dalam hal-hal khusus, misalnya tempat untuk berhubungan dengan port paralel P1 atau P3, dan sarana input/output lainnya, tapi tidak umum dipakai untuk menyimpan data seperti layaknya memori-data. Meskipun demikian, dalam hal penulisan program SFR diperlakukan persis sama dengan memori-data. Untuk mengisi memori-data nomor \$60 dengan bilangan \$0F, instruksi yang dipergunakan adalah : MOV \$60,#\$0F.

Sedangkan untuk menyimpan \$0F ke Port 1 yang di SFR menempati memori-data nomor \$90, instruksi yang dipergunakan adalah : MOV \$90,#\$0F Membandingkan kedua instruksi di atas bisa dimengerti dalam segi penulisan program SFR diperlakukan persis sama dengan memori-data.

Register Dasar MCS51

Untuk keperluan penulisan program, setiap mikroprosesor/mikrokontroler selalu dilengkapi dengan Register Dasar. Ada beberapa macam register merupakan register baku yang bisa dijumpai di semua jenis mikroprosesor/ mikrokontroler, ada register yang spesifik pada masing-masing prosesor. Yang termasuk Register Baku antara lain Program Counter, Akumulator, Stack Pointer Register, Program Status Register. MCS51 mempunyai semua register baku ini. Sebagai register yang khas MCS51, antara lain adalah Register B, Data Pointer High Byte dan Data Pointer Low Byte. Semua ini digambarkan dalam Gambar 6.5. Di samping itu MCS51 masih mempunyai Register Serba Guna R0..R7 yang sudah disebut di bagian atas. Dalam mikroprosesor/mikrokontroler yang lain, register-register dasar biasanya ditempatkan di tempat tersendiri dalam inti prosesor, tapi dalam MCS51 register-register itu ditempatkan secara terpisah.

1. Program Counter ditempatkan di tempat tersendiri di dalam inti prosesor
2. Register Serba Guna R0..R7 ditempatkan di salah satu bagian dari memori-data
3. Register lainnya ditempatkan dalam Special Function Register (SFR).



Gambar 6.5. Susunan Register Dasar MCS51

6.3.4. Dasar Pemrograman MCS51

Program pengendali mikrokontroler disusun dari kumpulan instruksi, instruksi tersebut setara dengan kalimat perintah bahasa manusia yang hanya terdiri atas predikat dan objek. Dengan demikian tahap pertama pembuatan program pengendali mikrokontroler dimulai dengan pengenalan dan pemahaman predikat (kata kerja) dan objek apa saja yang dimiliki mikrokontroler.

Objek dalam pemrograman mikrokontroler adalah data yang tersimpan di dalam memori, register dan input/output. Sedangkan 'kata kerja' yang dikenal pun secara umum dikelompokkan menjadi perintah untuk perpindahan data, arithmetik, operasi logika, pengaturan alur program dan beberapa hal khusus. Kombinasi dari 'kata kerja' dan objek itulah yang membentuk perintah pengatur kerja mikrokontroler.

Intruksi MOV A,\$7F merupakan contoh sebuah intruksi dasar yang sangat spesifik, MOV merupakan ‘kata kerja’ yang memerintahkan peng-copy-an data, merupakan predikat dalam kalimat perintah ini. Sedangkan objeknya adalah data yang di-copy-kan, dalam hal ini adalah data yang ada di dalam memori nomor \$7F di-copy-kan ke A kumulator A.

Penyebutan data dalam MCS51

Data bisa berada diberbagai tempat yang berlainan, dengan demikian dikenal beberapa cara untuk menyebut data (dalam bahasa Inggris sering disebut sebagai ‘Addressing Mode’), antara lain sebagai berikut :

1. Penyebutan data konstan (immediate addressing mode): MOV A,#\$20. Data konstan merupakan data yang berada di dalam intruksi. Contoh intruksi ini mempunyai makna data konstan \$20 (sebagai data konstan ditandai dengan ‘#’) di-copy-kan ke Akumulator A. Yang perlu benar-benar diperhatikan dalam perintah ini adalah bilangan \$20 merupakan bagian dari intruksi.
2. Penyebutan data secara langsung (direct addressing mode), cara ini dipakai untuk menunjuk data yang berada di dalam memori dengan cara menyebut nomor memoritempat data tersebut berada : MOV A,\$30. Contoh intruksi ini mempunyai makna data yang berada di dalam memori nomor \$30 di-copy-kan ke A kumulator. Sekilas intruksi ini sama dengan intruksi data konstan di atas, perbedaannya intruksi diatas memakai tanda ‘#’ yang menandai \$20 adalah data konstan, sedangkan dalam intruksi ini karena tidak ada tanda ‘#’ maka \$30 adalah nomor dari memori.
3. Penyebutan data secara tidak langsung (indirect addressing mode), cara ini dipakai untuk menunjuk data yang berada di dalam memori, kalau memori penyimpan data ini letaknya berubah-ubah sehingga nomor memori tidak disebut secara langsung tapi di-‘titip’-kan ke register lain : MOV A,@R0.

Dalam intruksi ini register serba guna R0 dipakai untuk mencatat nomor memori, sehingga intruksi ini mempunyai makna memori yang nomornya tercatat dalam R0 isinya di-copy-kan ke A kumulator A .Tanda ‘@’ dipakai untuk menandai nomor memori disimpan di dalam R0. Dalam intruksi ini register serba guna R0 berfungsi dengan register penampung alamat (indirect address register), selain R0 register serba guna R1 juga bisa dipakai sebagai register penampung alamat.

1. Penyebutan data dalam register (register addressing mode): MOV A,R5. Intruksi ini mempunyai makna data dalam register serba guna R5 di-copy-kan ke A kumulator A .Intruksi ini membuat register serba guna R0 sampai R7 sebagai tempat penyimpanan.data yang sangat praktis yang kerjanya sangat cepat.
2. Data yang dimaksud dalam bahasan di atas semuanya berada di dalam memori data (termasuk register serba guna letaknya juga di dalam memori data). Dalam penulisan program, sering-sering diperlukan tabel baku yang disimpan bersama dengan program tersebut. Tabel semacam ini sesungguhnya merupakan data yang berada di dalam memori program!

Kata kerja dalam AT89Cx051

Secara keseluruhan AT89Cx051 mempunyai sebanyak 255 macam intruksi, yang dibentuk dengan mengkombinasikan ‘kata kerja’ dan objek. “Kata kerja’ tersebut secara kelompok dibahas sebagai berikut :

KELOMPOK PENG-COPY-AN DATA

Kode dasar untuk kelompok ini adalah MOV, singkatan dari MOVE yang artinya memindahkan, meskipun demikian lebih tepat dikatakan perintah ini mempunyai makna peng-copy-an data. Hal ini bisa dijelaskan berikut : setelah instruksi MOV A,R7 dikerjakan, Akumulator A dan register serba guna R7 berisikan data yang sama, yang asalnya tersimpan di dalam R7. Perintah MOV dibedakan sesuai dengan jenis memori AT89Cx051. Perintah ini pada memori data dituliskan menjadi MOV, misalkan :

```
MOV A,$20
MOV A,@R1
MOV A,P1
MOV P3,A
```

Untuk pemakaian pada memori program, perintah ini dituliskan menjadi MOVC, hanya ada 2 jenis instruksi yang memakai MOVC, yakni:

```
MOVC A,@A+DPTR ; DPTR sebagai register indirect ,
MOVC A,@A+PC; PC sebagai register indirect
```

Selain itu, masih dikenal pula perintah MOVX, yakni perintah yang dipakai untuk memori data eksternal (X singkatan dari External). Perintah ini hanya dimiliki oleh anggota keluarga MCS51 yang mempunyai memori data eksternal, misalnya AT89C51 dan lain sebagainya, dan jelas tidak dikenal oleh kelompok AT89Cx051 yang tidak punya memori data eksternal. Hanya ada 6 macam instruksi yang memakai MOVX, instruksi-instruksi tersebut adalah :

```
MOVX A,@DPTR
MOVX A,@R0
MOVX A,@R1
MOVX @DPTR,A
MOVX @R0,A
MOVX @R1,A
```

KELOMPOK ARITMATIK (ADD, ADDC, SUBB, DA, MUL dan DIV)

Perintah ADD dan ADDC isi A kumulator A ditambah dengan bilangan 1 byte, hasil penjumlahan akan ditampung kembali dalam Akumulator. Dalam operasi ini bit Carry (C flag dalam PSW – Program Status Word) berfungsi sebagai penampung limpahan hasil penjumlahan. Jika hasil penjumlahan tersebut melimpah (nilainya lebih besar dari 255) bit Carry akan bernilai '1', kalau tidak bit Carry bernilai '0'. ADDC sama dengan ADD, hanya saja dalam ADDC nilai bit Carry dalam proses sebelumnya ikut dijumlahkan bersama.

Bilangan 1 byte yang ditambahkan ke A kumulator, bisa berasal dari bilangan konstan, dari register serba guna, dari memori data yang nomor memorinya disebut secara langsung maupun tidak langsung, seperti terlihat dalam contoh berikut :

```
ADD A,R0; register serba guna
ADD A,#$23; bilangan konstan
ADD A,@R0; no memori tak langsung
ADD A,P1 ; no memori langsung (port 1)
```

Perintah SUBB

Isi A kumulator A dikurangi dengan bilangan 1 byte berikut dengan nilai bit Carry, hasil pengurangan akan ditampung kembali dalam Akumulator. Dalam operasi ini bit Carry juga berfungsi sebagai penampung limpahan hasil pengurangan. Jika hasil pengurangan tersebut melimpah (nilainya kurang dari 0) bit Carry akan bernilai '1', kalau tidak bit Carry bernilai '0'.

SUBB A,R0; $A = A - R0 - C$

SUBB A,#\$23; $A = A - \$23$

SUBB A,@R1

SUBB A,P0

Perintah DA

Perintah DA (Decimal Adjust) dipakai setelah perintah ADD; ADDC atau SUBB, dipakai untuk merubah nilai biner 8 bit yang tersimpan dalam A kumulator menjadi 2 buah bilangan desimal yang masing-masing terdiri dari nilai biner 4 bit.

Perintah MUL AB

Bilangan biner 8 bit dalam Akumulator A dikalikan dengan bilangan biner 8 bit dalam register B. Hasil perkalian berupa bilangan biner 16 bit, 8 bit bilangan biner yang bobotnya lebih besar ditampung di register B, sedangkan 8 bit lainnya yang bobotnya lebih kecil ditampung di A kumulator A. Bit OV dalam PSW (Program Status Word) dipakai untuk menandai nilai hasil perkalian yang ada dalam register B. Bit OV akan bernilai '0' jika register B bernilai \$00, kalau tidak bit OV bernilai '1'.

MOV A,#10

MOV B,#20

MUL AB

KELOMPOK LOGIKA (ANL, ORL dan XRL)

Kelompok perintah ini dipakai untuk melakukan operasi logika mikrokontroler MCS51, operasi logika yang bisa dilakukan adalah operasi AND (kode operasi ANL), operasi OR (kode operasi ORL) dan operasi Exclusive-OR (kode operasi XRL). Data yang dipakai dalam operasi ini bisa berupa data yang berada dalam A kumulator atau data yang berada dalam memori-data, hal ini sedikit berlainan dengan operasi aritmatik yang harus melihatkan A kumulator secara aktif. Hasil operasi ditampung di sumber data yang pertama.

1. Operasi logika AND banyak dipakai untuk me-'0'-kan beberapa bit tertentu dari sebuah bilangan biner 8 bit, caranya dengan membentuk sebuah bilangan biner bit sebagai data konstan yang di-ANL-kan bilangan asal. Bit yang ingin di-'0'-kan diwakili dengan '0' pada data konstan, sedangkan bit lainnya diberi nilai '1', misalnya: Instruksi ANL P1,#%01111110 akan mengakibatkan bit 0 dan bit 7 dari Port 1 (P1) bernilai '0' sedangkan bit-bit lainnya tetap tidak berubah nilai.
2. Operasi logika OR banyak dipakai untuk me-'1'-kan beberapa bit tertentu dari sebuah bilangan biner 8 bit, caranya dengan membentuk sebuah bilangan biner 8 bit sebagai data konstan yang di-ORL-kan bilangan asal. Bit yang ingin di-'1'-kan diwakili dengan '1' pada data konstan, sedangkan bit lainnya diberi nilai '0',misalnya: Instruksi ORL A,#%01111110 akan mengakibatkan bit 1 sampai dengan bit 6 dari A kumulator bernilai '1'

sedangkan bit-bit lainnya tetap tidak berubah nilai.

3. Operasi logika Exclusive-OR banyak dipakai untuk membalik nilai (complement) beberapa bit tertentu dari sebuah bilangan biner 8 bit, caranya dengan membentuk sebuah bilangan biner 8 bit sebagai data konstan yang di-XOR-kan bilangan asal. Bit yang ingin dibalik-nilai diwakili dengan '1' pada data konstan, sedangkan bit lainnya diberi nilai '0', misalnya: Instruksi XOR A,#%01111110 akan mengakibatkan bit 1 sampai dengan bit 6 dari A kumulator berbalik nilai, sedangkan bit-bit lainnya tetap tidak berubah nilai.

Mengatur Alur Program MCS51

Secara umum kelompok instruksi yang dipakai untuk mengatur alur program terdiri atas instruksi-instruksi JUMP (setara dengan statemen GOTO dalam Pascal), instruksi-instruksi untuk membuat dan memakai sub-rutin/modul (setara dengan PROCEDURE dalam Pascal), instruksi-instruksi JUMP bersyarat (conditional Jump, setara dengan statemen IF .. THEN dalam Pascal). Di samping itu ada pula instruksi PUSH dan POP yang bisa memengaruhi alur program. Karena Program Counter adalah satu-satunya register dalam mikrokontroler yang mengatur alur program, maka kelompok instruksi pengatur program yang dibicarakan di atas, semuanya merubah nilai Program Counter, sehingga pada saat kelompok instruksi ini dijalankan, nilai Program Counter akan tidak akan runtun dari nilai instruksi sebelumnya. Selain karena instruksi-instruksi di atas, nilai Program Counter bisa pula berubah karena pengaruh perangkat keras, yaitu saat mikrokontroler di-reset atau menerima sinyal interupsi dari perangkat input/output.

Operasi Bit dengan MCS51

Pada umumnya mikrokontroler mengolah data 8 bit sekali gus, misalnya mengisi akumulator dengan data 8 bit sekali gus, isi akumulator yang 8 bit dijumlahkan dengan isi memori yang 8 bit dan lain sebagainya. MCS51 dilengkapi kemampuan mengolah data per bit, untuk keperluan ini bit Carry dalam PSW diperlakukan sebagai 'akumulator bit', dan dilengkapi dengan beberapa instruksi khusus untuk operasi Boolean.

Objek operasi Bit

Seperti sudah dibahas dibagian depan, memori-data nomor \$20 sampai \$2F bisa dipakai menampung informasi dalam level bit. Setiap byte memori di daerah ini bisa menampung 8 bit informasi yang masing-masing dinomori tersendiri, misalkan bit 0 dari memori-data nomor \$20 bisa disebut sebagai bit nomor 0, bit 1 memori-data nomor \$20 disebut sebagai bit nomor 1... seterusnya bit 0 dari memori-data nomor \$21 disebut sebagai bit nomor 8, bit 7 memori-data nomor \$21 disebut sebagai bit nomor 15 dan seterusnya. Secara keseluruhan operasi bit bisa diberlakukan pada 256 lokasi bit.

Operasi bit yang bisa ditangani oleh MCS51 antara lain mencakup : pemberian nilai pada data biner 1 bit, perpindahan data 1 bit, operasi logika 1 bit meliputi operasi AND, OR dan NOT, pengujian nilai data biner 1 bit. Pemberian nilai data biner Untuk keperluan ini disediakan 2 instruksi, yakni SETB (Set Bit) dipakai memberi nilai '1' pada data biner 1 bit, dan CLR (Clear Bit) dipakai memberi nilai '0' pada data biner 1 bit.

Contoh pemakaian instruksi ini sebagai berikut :

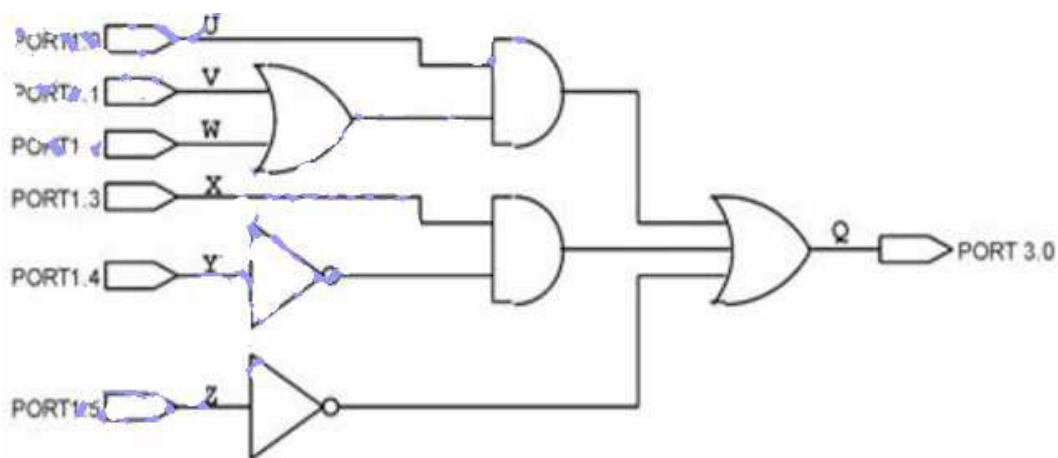
```
SETB ACC.0
SETB $E0
CLR P1.1
CLR $90
```

Instruksi SETB ACC.0 di atas membuat bit 0 dari akumulator (ACC.0) bernilai '1', tapi mengingat nomor bit dari bit 0 akumulator adalah \$E0 (lihat Gambar 6), maka hasil kerja kedua instruksi SETB di atas adalah sama. Demikian pula dengan kedua instruksi CLR berikutnya, instruksi-instruksi ini akan mengakibatkan P1.1 bernilai '0'. Di samping itu perlu pula diingat, P1.1 terhubung ke kaki IC MCS51, jadi hasil kerja operasi SETB maupun CLR pada P0, P1, P2 dan P3 bisa langsung terukur dengan volt meter, atau operasi-operasi ini bisa langsung dipakai men-on/off-kan rangkaian di luar IC MCS51.

Pemakaian Instruksi operasi bit

Dengan instruksi-instruksi operasi bit yang ada, MCS51 bisa dipakai untuk mengimplementasi fungsi Boolean secara langsung, sebagai contoh persamaan Boolean berikut diimplementasikan dengan instruksi-instruksi MCS51.

$Q = (U \cdot (V + W)) + (X \cdot Y) + Z$ (Catatan : /Y artinya not Y dan /Z artinya not Z). Misalkan U V W X Y dan Z masing-masing adalah besaran Boolean yang dimasukkan ke kaki Port 1 bit 0 sampai dengan 5, sedangkan Q merupakan besaran Boolean yang ditampilkan di Port 3 bit 0, seperti terlihat dalam Gambar 6.7 berikut :



Gambar 6.7. Rangkaian $Q = (U \cdot (V + W)) + (X \cdot Y) + Z$

Dalam Potongan Program baris 1 sampai dengan 7, P1.0 P1.1 dan lainnya dinyatakan sebagai data bit dengan nama U, V dan selanjutnya, penamaan ini menggunakan perintah khusus untuk assembler (Assembler Directive) BIT, hanya nama-nama yang dibentuk dengan assembler-directive BIT yang bisa dipakai untuk operasi bit.

Potongan Program - Fungsi Boolean

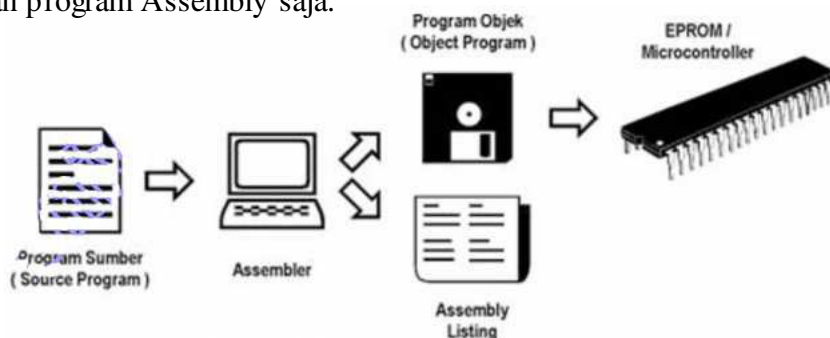
```

01:U   bit P1.0 ; P1.0 : bit bernama U
02:V   bit P1.1 ; P1.1 : bit bernama V
03:W   bit P1.2 ; P1.2 : bit bernama W
04:X   bit P1.3 ; P1.3 : bit bernama X
05:Y   bit P1.4 ; P1.4 : bit bernama Y
06:Z   bit P1.5 ; P1.5 : bit bernama Z
07:Q   bit P3.1 ; P3.1 : bit bernama Q
08:
09:    MOV C,V ; V dipindahkan ke C
10:    ORL C,W ; di-OR-kan dengan W
11:    ANL C,U ; di-AND-kan dengan U
12:    MOV FO,C ; simpan dulu ke FO (PSW)
13:    MOV C,X ; X dipindahkan ke C
14:    ANL C,/Y ; di-AND-kan dgn not Y
15:    ORL C,FO ; di-OR-kan dgn simpanan
16:    ORL C,/Z ; di-OR-kan dgn not Z
17:    MOV Q,C ; hasilnya disimpan ke Q

```

6.3.5. Pengetahuan Dasar Program Assembly

Bahasa Assembly adalah bahasa komputer yang kedudukannya di antara bahasa mesin dan bahasa level tinggi misalnya bahasa C atau Pascal. Bahasa C atau Pascal dikatakan sebagai bahasa level tinggi karena memakai kata-kata dan pernyataan yang mudah dimengerti manusia, meskipun masih jauh berbeda dengan bahasa manusia sesungguhnya. Bahasa mesin adalah kumpulan kode biner yang merupakan instruksi yang bisa dijalankan oleh komputer. Sedangkan bahasa Assembly memakai kode Mnemonic untuk menggantikan kode biner, agar lebih mudah diingat sehingga lebih memudahkan penulisan program. Program yang ditulis dengan bahasa Assembly terdiri dari label; kode mnemonic dan lain sebagainya, pada umumnya dinamakan sebagai program sumber (Source Code) yang belum bisa diterima oleh prosesor untuk dijalankan sebagai program, tapi harus diterjemahkan dulu menjadi bahasa mesin dalam bentuk kode biner. Yang perlu diperhatikan adalah setiap prosesor mempunyai konstruksi yang berlainan, instruksi untuk mengendalikan masing-masing prosesor juga berlainan, dengan demikian bahasa Assembly untuk masing-masing prosesor juga berlainan, yang sama hanyalah pola dasar cara penulisan program Assembly saja.

**Gambar 6.8. Bagan kerja proses assembly****Konstruksi Program Assembly**

Program sumber dalam bahasa Assembly menganut prinsip 1 baris untuk satu perintah, setiap baris perintah tersebut bisa terdiri atas beberapa bagian (field), yakni bagian Label, bagian mnemonic, bagian operand yang bisa lebih dari satu dan terakhir bagian komentar. Untuk membedakan masing-masing bagian tersebut dibuat ketentuan sebagian berikut :

1. Masing-masing bagian dipisahkan dengan spasi atau TAB, khusus untuk operand yang lebih dari satu masing-masing operand dipisahkan dengan koma.
2. Bagian-bagian tersebut tidak harus semuanya ada dalam sebuah baris, jika ada satu bagian yang tidak ada maka spasi atau TAB sebagai pemisah bagian tetap harus ditulis.
3. Bagian Label ditulis mulai huruf pertama dari baris, jika baris bersangkutan tidak mengandung Label maka label tersebut digantikan dengan spasi atau TAB, yakni sebagai tanda pemisah antara bagian Label dan bagian mnemonic.

Label mewakili nomor memori-program dari instruksi pada baris bersangkutan, pada saat menulis instruksi JUMP, Label ini ditulis dalam bagian operand untuk menyatakan nomor memori-program yang dituju. Dengan demikian Label selalu mewakili nomor memori- program dan harus ditulis dibagian awal baris instruksi. Disamping Label dikenal pula Symbol, yakni satu nama untuk mewakili satu nilai tertentu dan nilai yang diwakili bisa apa saja tidak harus nomor memori-program. Cara penulisan Symbol sama dengan cara penulisan Label, harus dimulai di huruf pertama dari baris instruksi.

Mnemonic (artinya sesuatu yang memudahkan diingat) merupakan singkatan perintah, dikenal dua macam mnemonic, yakni mnemonic yang dipakai sebagai instruksi mengendalikan prosesor, misalnya ADD, MOV, DJNZ dan lain sebagainya. Ada pula mnemonic yang dipakai untuk mengatur kerja dari program Assembler misalnya ORG, EQU atau DB, mnemonic untuk mengatur kerja dari program Assembler ini dinamakan sebagai 'Assembler Directive'.

Operand adalah bagian yang letaknya di belakang bagian mnemonic, merupakan pelengkap bagi mnemonic. Kalau sebuah instruksi di-ibaratkan sebagai kalimat perintah, maka mnemonic merupakan subjek (kata kerja) dan operand merupakan objek (kata benda) dari kalimat perintah tersebut.

Tergantung pada jenis instruksinya, operand bisa berupa berbagai macam hal. Pada instruksi JUMP operand berupa Label yang mewakili nomor memori-program yang dituju misalnya LJMP Start, pada instruksi untuk pemindahan/pengolahan data, operand bisa berupa Symbol yang mewakili data tersebut, misalnya A DD A,#Offset. Banyak instruksi yang operandnya adalah register dari prosesor, misalnya MOV A,R1. Bahkan ada pula instruksi yang tidak mempunyai operand, misalnya RET.

Komentar merupakan bagian yang sekedar sebagai catatan, tidak berpengaruh pada prosesor juga tidak berpengaruh pada kerja program Assembler, tapi bagian ini sangat penting untuk keperluan dokumentasi. Assembler Directive Seperti sudah dibahas di atas, bagian Mnemonic dari sebuah baris perintah bisa merupakan instruksi untuk prosesor, maupun berupa Assembler Directive untuk mengatur kerja dari program Assembler. Mnemonic untuk instruksi prosesor, sangat tergantung pada prosesor yang dipakai, sedangkan mnemonic untuk Assembler Directive tergantung pada program Assembler yang dipakai. Meskipun demikian, terdapat beberapa Assembler Directive yang umum, yang sama untuk banyak macam program Assembler. Assembler Directive yang bersifat umum tersebut, antara lain adalah ORG - singkatan dari ORIGIN, untuk menyatakan nomor memori yang dipakai setelah perintah itu, misalnya ORG \$1000 maka memori berikutnya yang dipakai Assembler adalah \$1000. ORG berlaku untuk memori program maupun memori-data.

Dalam penomoran memori, dikenal \$ sebagai awalan untuk menyatakan nomor memori dari baris bersangkutan. Misalnya : ORG 1000 dan LJMP \$+1000. Operand \$+\$500 mempunyai arti nomor memori-program bersangkutan ditambah

dengan \$500, karena instruksi LJMP ini terletak persis di bawah ORG \$1000 maka nomor memori- program baris ini adalah \$1000, sehingga operand \$+\$500 bernilai \$1500 dan instruksi ini indentik dengan LJMP \$1500.

1. EQU - singkatan dari EQUATE, dipakai untuk menentukan nilai sebuah Symbol. Misalnya Angka 88 EQU 88 memberi nilai 88 pada Symbol Angka 88, atau CR EQU\$0D mempunyai makna kode ASCII dari CR adalah \$08.
2. DB - singkatan dari DEFINE BYTE, dipakai untuk memberi nilai tertentu pada memori-program. Nilai tersebut merupakan nilai 1 byte, bisa berupa angka ataupun kode ASCII. DB merupakan A ssembler Directive yang dipakai untuk membentuk teks maupun tabel.

```
ORG $0200
```

```
STRING DB 'Atmel AT89C2051'
```

```
PANJANG EQU $-STRING
```

ORG \$0200 memerintahkan program Assembler agar bekerja mulai dari memori-program nomor \$0200, instruksi selanjutnya memerintahkan program A ssembler agar mengisi memori-program nomor \$0200 dan berikutnya dengan tulisan "Atmel AT89C2051" (yang diisikan adalah kode ASCII dari 'A', 't' dan seterusnya), PANJANG dari STRING bisa dihitung dengan cara PANJANG EQU \$-STRING, yakni selisih dari nomor memori-program baris bersangkutan dikurangi dengan nomor awal memori-program yang diisi STRING.

1. DW - singkatan dari DEFINE WORD, dipakai untuk memberi nilai 2 byte ke memori-program pada baris bersangkutan. A ssembler Directive ini biasa dipakai untuk membentuk suatu tabel yang isinya adalah nomor-nomor memori-program.
2. DS - singkatan dari Define Storage, A ssembler Directive ini dipakai untuk membentuk variable. Sebagai variabel tentu saja memori yang dipakai adalah memori-data (RAM) bukan memori-program (ROM). Hal ini harus benar-benar dibedakan dengan Assembler Directive DB dan DW yang membentuk kode dimemori-program. Dan karena DS bekerja di RAM, maka DS hanya sekedar menyediakan tempat di memori, tapi tidak mengisi nilai pada memori bersangkutan.

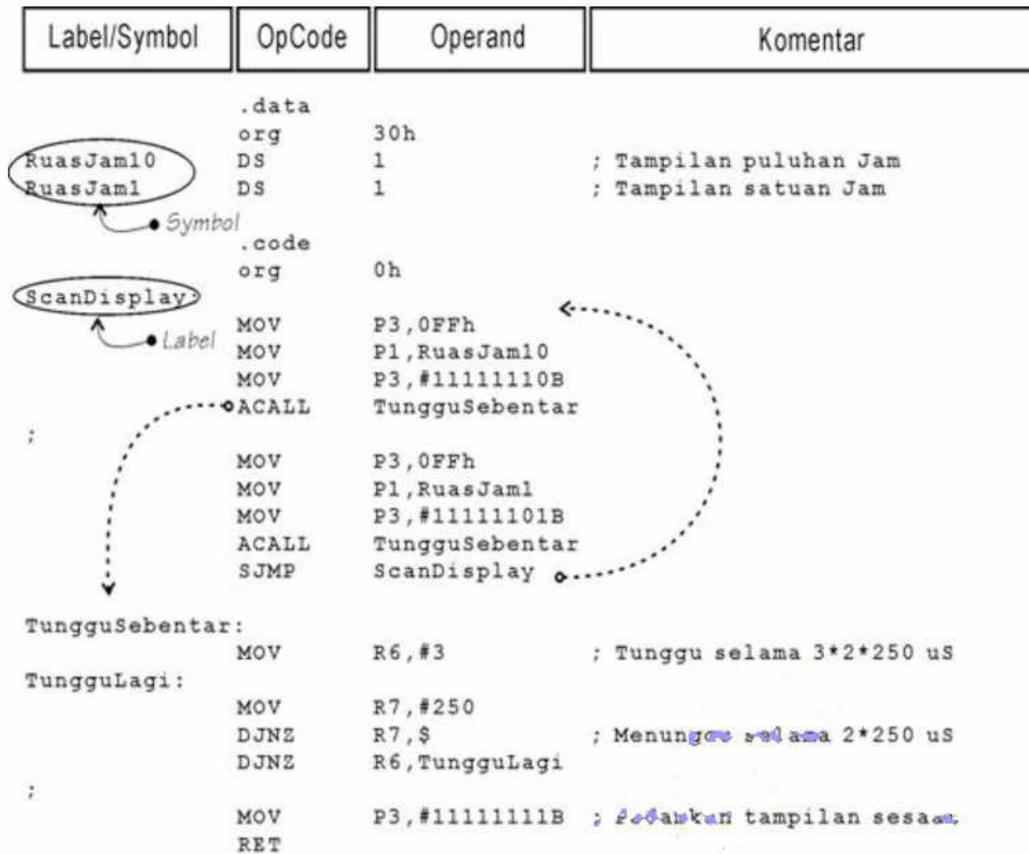
Struktur Program Assembly

Sarana yang ada dalam program assembly sangat minim, tidak seperti dalam bahasa pemrograman tingkat atas (high level language programming) semuanya sudah siap pakai. Penulis program assembly harus menentukan segalanya, menentukan letak program yang dituliskannya dalam memori-program, membuat data konstan dan tabel konstan dalam memori-program, membuat variabel yang dipakai kerja dalam memori-data dan lain sebagainya.

Program sumber assembly

Program-sumber assembly (assembly source program) merupakan kumpulan dari baris-baris perintah yang ditulis dengan program penyunting-teks (text editor) sederhana, misalnya program EDIT.COM dalam DOS, atau program NOTEPAD dalam Windows. Kumpulan baris- perintah tersebut biasanya disimpan ke dalam file dengan nama ekstensi *.ASM atau nama lain misalnya *.A51 dan lain sebagainya, tergantung pada program Assembler yang akan dipakai untuk mengolah program-sumber assembly tersebut. Setiap baris-perintah merupakan sebuah perintah yang

utih, artinya sebuah perintah tidak mungkin dipecah menjadi lebih dari satu baris. Satu baris perintah bisa terdiri atas 4 bagian, bagian pertama dikenali sebagai label atau sering juga disebut sebagai symbol, bagian kedua dikenali sebagai kode operasi, bagian ketiga adalah operand dan bagian terakhir adalah komentar. Pembahasan di atas diringkas dalam gambar 6.9.



Gambar 6.9. Program-sumber assembly

Assembly Listing

Program-sumber assembly di atas, setelah selesai ditulis diserahkan ke program Assembler untuk diterjemahkan. Setiap prosesor mempunyai program assembler tersendiri, bahkan satu macam prosesor bisa memiliki beberapa macam program Assembler buatan pabrik perangkat lunak yang berlainan. Hasil utama pengolahan program Assembler adalah program-obyek. Program-obyek ini bisa berupa sebuah file tersendiri, berisikan kode-kode yang siap dikirimkan ke memori-program mikroprosesor/mikrokontroler, tapi ada juga program-obyek yang disisipkan pada program- sumber assembly seperti terlihat dalam assembly listing di gambar 6.10. Bagian kanan gambar 6.10 merupakan program-sumber assembly karya asli penulis program, setelah diterjemahkan oleh program assembler kode-kode yang dihasilkan berikut dengan nomor-nomor memori tempat penyimpanan kode-kode tadi, disisipkan pada bagian kiri setiap baris perintah, sehingga bentuk program ini tidak lagi dikatakan sebagai program-sumber assembly tapi dikatakan sebagai assembly listing.

Program Obyek	Program-sumber assembly
0000	.DATA ←•Pakai memori-data
0030	ORG 30h ←•mulai nomor 30h
0030 RuasJam10:	DS 1 ; Tampilan puluhan Jam
0031 RuasJam1:	DS 1 ; Tampilan satuan Jam
0000	.CODE ←•Pakai memori-program
0000	ORG 0h ←•mulai nomor 0h
0000	ScanDisplay:
0000 85 FF B0	MOV P3,0FFh
0003 85 30 90	MOV P1,RuasJam10
0006 75 B0 FE	MOV P3,#11111110B
0009 11 18	ACALL TungguSebentar
	; TungguSebentar ada di 0018
000B 85 FF B0	MOV P3,0FFh
000E 85 31 90	MOV P1,RuasJam1
0011 75 B0 FD	MOV P3,#11111101B
0014 11 18	=0018 ACALL TungguSebentar
0016 80 E8	=0000 SJMP ScanDisplay
0018	TungguSebentar:
0018 7E 03	MOV R6,#3 ; Tunggu selama 3*2*250 us
001A	TungguLagi:
001A 7F FA	MOV R7,#250
001C DF FE	=001C DJNE R7,\$; Menunggu disini selama 2*250 us
001E DE FA	=001A DJNE R6,TungguLagi
	; TungguLagi ada di 001A
0020 75 B0 FF	MOV P3,#11111111B ; PAMAHAN tampilan sesaat
0023 22	RET

Gambar 6.10. Assembly Listing

Membaca assembly listing bisa memberikan gambaran yang lebih jelas bagi program yang ditulis, bagi pemula assembly listing memberi pengertian yang lebih mendalam tentang isi memori-program, sehingga bisa lebih dibayangkan bagaimana kerja dari sebuah program.

Program Obyek format HEX

Selain Assembly Listing, hasil kerja program Assembler lainnya adalah program obyek yang dipakai untuk mengendalikan sebuah mikroprosesor/mikrokontroler, program obyek disimpan dalam file. Terdapat dua macam bentuk file penyimpanan program obyek, yang pertama adalah file yang berisikan kode biner murni, dan yang satu lagi adalah file biner yang sudah diolah menjadi file teks. File jenis pertama biasanya dinamakan sebagai binary object file, biasanya memakai ekstensi *.BIN. File semacam ini hanya berisikan angka-angka biner yang akan diisikan ke dalam memori tanpa informasi lain, sehingga selalu dianggap bahwa file tersebut berisikan kode-kode biner yang nantinya ditempatkan mulai dari memori nomor 0. Kalau ternyata kode-kode biner diisikan mulai dari memori nomor 8000h, maka mulai posisi 0 sampai 7FFFh akan diisi dengan bilangan biner 00h, baru setelah itu menyusul kode biner yang sesungguhnya. File semacam ini banyak dipakai untuk EPROM Programmer model lama. File jenis kedua dinamakan Hexadecimal format object file, biasanya memakai ekstensi *.HEX. Data biner dirubah ke dalam bentuk heksadesimal dan yang disimpan ke dalam file adalah kode ASCII dari bilangan heksadesimal tersebut. Misalnya data biner 00111010, atau heksadesimal 3Ah, dituliskan ke dalam file menjadi 33h (kode ASCIInya angka 3) dan 41h (kode ASCIInya huruf A). Dengan cara ini isi dari file

tersebut bisa dengan mudah dibaca dengan program penyunting teks (text editor) biasa, bahkan bisa di-cetak di atas kertas seperti terlihat dalam Gambar 6.11, file semacam itu bisa dibaca dengan text editor biasa, misalnya EDIT.COM dalam DOS, atau NOTEPAD dalam Windows.

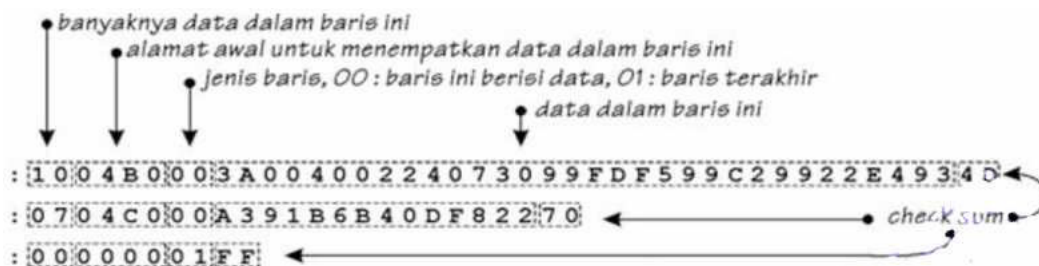
Dalam file format HEX semacam ini, selain disimpan data biner yang akan diisikan ke ROM, berisikan pula nomor-nomor memori tempat penyimpanan data biner tersebut. EPROM programmer baru umumnya memakai format file obyek semacam ini.

```
:2000000085FFB085309075B0FE111885FFB085319075B0FD111880E87E037FFADFFFEDEFA3F
:0400200075B0FF2296
:00000001FF
```

Gambar 6.11. Program obyek format HEX

Format HEX dari Intel

Beberapa macam format untuk membentuk file program obyek format HEX (Hexadecimal format object file), meskipun hanya 2 yang banyak dipakai, yakni



Gambar 6.12. Anatomi baris-baris dalam file format HEX

Format buatan Motorola yang dinamakan sebagai format S19 dan format buatan Intel yang disebut sebagai format HEX dari Intel. Berikut ini adalah pembahasan file program obyek dengan format HEX dari Intel yang dipakai MCS51, format ini didefinisikan dalam artikel dari Intel dengan judul Hexadecimal Object File Format Specification : (<http://alds.sts.edu.appnote/#MCS51>).

File program obyek dengan format HEX dari Intel berisikan baris-baris tulisan seperti terlihat dalam Gambar 6.12. Setiap baris mengandung informasi tentang berapa banyak data dalam baris tersebut, alamat awal tempat penyimpanan data dalam baris tersebut, jenis baris dan sarana untuk memastikan kebenaran data yang dinamakan sebagai check sum. Dalam baris tersebut, setiap huruf (kecuali huruf pertama) mewakili satu bilangan heksa-desimal, dengan demikian setiap 2 huruf membentuk data satu byte yang terdiri dari 2 bilangan heksadesimal. Rincian dari format tersebut sebagai berikut :

1. Huruf pertama dalam baris, selalu berisi tanda “:”, merupakan kode identitas yang menyatakan baris tersebut berisikan kode-kode biner yang disimpan dalam format HEX dari Intel.
2. Huruf ke-2 dan ke-3 dipakai untuk menyatakan banyaknya data dalam baris yang dinyatakan dengan 2 angka heksa-desimal, sehingga banyaknya data dalam 1 baris maksimal adalah 255 (atau heksa-desimal FF).
3. Huruf ke 4 sampai 7, merupakan 4 angka heksa-desimal yang dipakai untuk

- menyatakan alamat awal tempat penyimpanan kode-kode dalam baris teksbersangkutan.
4. Huruf 8 dan 9 dipakai untuk menyatakan jenis teks data. Nilai 00 dipakai untuk menyatakan baris tersebut berisikan data biasa, 01 menyatakan baris tersebut merupakan baris terakhir.
 5. Huruf ke 10 dan seterusnya adalah data. Setiap 2 huruf mewakili data 1 byte, sehingga jumlah huruf pada bagian ini adalah dua kali banyaknya data yang disebut pada butir 2 di atas.
 6. 2 huruf terakhir dalam baris merupakan check sum. Byte-byte yang disebut dalam butir 2 sampai 5 di atas dijumlahkan, hasil penjumlahan di-balik (inverted) sebagai bilangan check sum. (hasil penjumlahan bisa menghasilkan nilai yang lebih besar dari 2 bilangan heksadesimal, namun hanya 2 bilangan heksa-desimal yang bobotnya terkecil yang dipakai).

6.4. Penutup

6.4.1. Bahan diskusi dan tugas

1. Jelaskan maksud dari Random Acces Memory dalam chip AT89C/051?
2. Apa yang anda ketahui tentang SFR (Special Function Register)?
3. Sebutkan macam-macam dari register baku?
4. Jelaskan dan sebutkan kelompok instruksi aritmatik?
5. Buat program sederhana MCS51 untuk membaca data pada port 0 dan mengeluarkannya pada port 21?

6.5. Daftar Pustaka

1. J. Webb, *Industrial Control Electronics*, Macmillan Publish,. New York, 1992.
2. J. Webb, *Programmable Logic Control*, Macmillan Publish,. New York, 1992.
3. Frank D Petruzella, *Elektronik Industri* (terjemahan), Penerbit Andi, Yogyakarta, 2001.
4. William Bolton, *Programmable Logic Controller (PLC)*, Sebuah Pengantar, Edisi ketiga, Penerbit Erlangga, 2004.
5. M. Budiyanto dan A. Wijaya, *Pengenalan Dasar-dasar PLC (Programmable Logic Controller)*, Penerbit Gaya Media, Yogyakarta, 2003.
6. Endra Ditawarna, *Robotika, Desain, Kontrol dan Kecerdasan Buatan*, Penerbit Andi, Yogyakarta, 2006.
7. David G. Alciator dan Michael B Histand, *Introduction to Mechatronics and Measurement Systems*, Mc Graw Hill, New York, 2003.

6.6. Senarai

Program untuk mengendalikan kerja dari mikrokontroler disimpan di dalam memori program. Program pengendali tersebut merupakan kumpulan dari instruksi kerja mikrokontroler, 1 instruksi MCS51 merupakan kode yang panjangnya bisa satu sampai empat byte. Sepanjang mikrokontroler bekerja, instruksi tersebut byte demi byte diambil ke CPU dan selanjutnya dipakai untuk mengatur kerja mikrokontroler.

Proses pengambilan instruksi dari memori program dikatakan sebagai 'fetch

'cycles' dan saat-saat CPU melaksanakan instruksi disebut sebagai 'execute cycles'. Semua mikrokontroler maupun mikroprosesor dilengkapi sebuah register yang berfungsi khusus untuk mengatur 'fetch cycles', register tersebut dinamakan sebagai Program Counter. Nilai Program Counter secara otomatis bertambah satu setiap kali selesai mengambil 1 byte isi memori program, dengan demikian isi memori program bisa berurutan diumpankan ke CPU.

BAB 7

TEKNIK

PEMROGRAMAN PLC

7.1. Deskripsi Singkat

Tujuan dari mempelajari teknik pemrograman PLC (*Programing Logic Control*) adalah untuk dapat memahami operasi proses dan melakukan tindakan sesuai dengan yang dibutuhkan.

7.2. Tujuan Intruksional Khusus

Setelah materi ini diajarkan mahasiswa dapat merancang program kendali PLC sederhana, memasukkan program ke dalam PLC dan dapat Mengecek kebenaran program.

7.3. Penyajian

7.3.1. Pengantar

Sebuah PLC (*Programmable Logic Control*) adalah sebuah alat yang digunakan untuk menggantikan rangkaian sederetan relay yang dijumpai pada sistem kontrol proses konvensional. PLC bekerja dengan cara mengamati masukan (melalui sensor-sensor terkait), kemudian melakukan proses dan melakukan tindakan sesuai yang dibutuhkan, yang berupa menghidupkan atau mematikan keluarannya (logika 0 atau 1, hidup atau mati). Pengguna membuat program (yang umumnya dinamakan diagram tangga atau ladder diagram) yang kemudian harus dijalankan oleh PLC yang bersangkutan, Dengan kata lain, PLC menentukan aksi apa yang harus dilakukan pada instrumen keluaran berkaitan dengan status suatu ukuran atau besaran yang diamati.

Pendekatan sistematis dalam perancangan sistem kontrol proses :

Pertama, Anda perlu memilih suatu instrumen atau sistem yang hendak dikontrol; Sistem yang terotomasi bisa berupa sebuah mesin atau suatu proses yang kemudian disebut sebagai sistem kontrol proses. Fungsi dari sistem kontrol proses ini

secara terus-menerus akan mengamati sinyal-sinyal yang berasal dari piranti-piranti masukan (sensor) dan tanggapannya berupa suatu sinyal yang diberikan ke piranti keluaran eksternal yang secara langsung mengontrol bagaimana suatu sistem beroperasi atau bekerja.

Kedua, Anda perlu menentukan semua masukan dan keluaran yang akan dihubungkan ke PLC; Piranti masukan dapat berupa saklar, sensor dan lain sebagainya. Sedangkan piranti keluaran dapat berupa selenoida, kran elektromagnet, motor, relai, starter magnet begitu juga dengan instrumen lain yang bisa menghasilkan suara atau cahaya (lampu) dan lain sebagainya. Setelah menentukan kebutuhan semua piranti masukan dan keluaran dilanjutkan dengan menentukan penggunaan jalur-jalur masukan dan keluaran pada PLC untuk piranti-piranti masukan dan keluaran yang sudah ditentukan tadi.

Ketiga, membuat program yang lebih dikenal dengan diagram tangga (untuk PLC) sesuai dengan jalannya proses yang diinginkan. Dalam hal ini bisa digunakan terminal konsol yang langsung berhubungan dengan PLC yang bersangkutan atau melalui komputer PC yang memiliki saluran komunikasi yang dibutuhkan untuk mentransfer program dari komputer PC ke PLC maupun sebaliknya.

Keempat, program disimpan ke dalam PLC; baik dilakukan secara langsung melalui terminal konsol maupun melalui komputer PC.

Program kendali PLC terdiri atas tiga unsur yaitu : alamat, instruksi, dan operand. Alamat adalah nomor yang menunjukkan lokasi, instruksi, atau data dalam daerah memori. Instruksi harus disusun secara berurutan dan menempatkannya dalam alamat yang tepat sehingga seluruh instruksi dilaksanakan mulai dari alamat terendah hingga alamat tertinggi dalam program. Instruksi adalah perintah yang harus dilaksanakan PLC. PLC hanya dapat melaksanakan instruksi yang ditulis menggunakan ejaan yang sesuai. Oleh karena itu, pembuat program harus memperhatikan tata cara penulisan instruksi. Operand adalah nilai berupa angka yang ditetapkan sebagai data yang digunakan untuk suatu instruksi. Operand dapat dimasukkan sebagai konstanta yang menyatakan nilai angka nyata atau merupakan alamat data dalam memori.

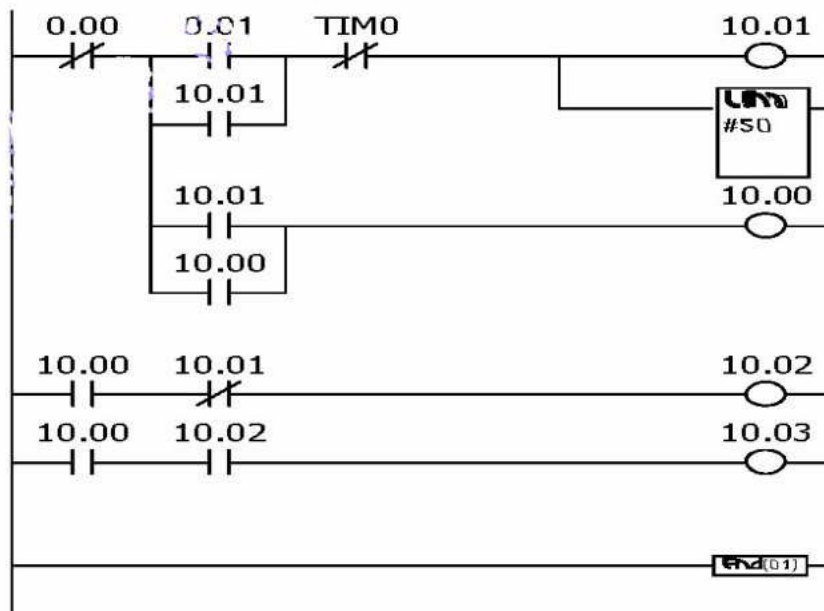
7.3.2. Bahasa Pemrograman

Program PLC dapat dibuat dengan menggunakan beberapa cara yang disebut bahasa pemrograman. Bentuk program berbeda-beda sesuai dengan bahasa pemrograman yang digunakan. Bahasa pemrograman tersebut antara lain: diagram ladder, kode mnemonic, diagram blok fungsi, dan teks terstruktur. Beberapa merk PLC hanya mengembangkan program diagram ladder dan kode mnemonic.

Diagram Ladder

Diagram ladder terdiri atas sebuah garis vertikal di sebelah kiri yang disebut bus bar, dengan garis bercabang ke kanan yang disebut rung. Sepanjang garis instruksi, ditempatkan kontak-kontak yang mengendalikan/ mengkondisikan instruksi lain di sebelah kanan. Kombinasi logika kontak-kontak ini menentukan kapan dan bagaimana instruksi di sebelah kanan dieksekusi. Contoh diagram ladder ditunjukkan pada gambar di bawah ini.

Terlihat dari gambar di atas bahwa garis instruksi dapat bercabang kemudian menyatu kembali. Sepasang garis vertikal disebut kontak (kondisi). Ada dua kontak, yaitu kontak NO (Normally Open) yang digambar tanpa garis diagonal dan kontak NC (Normally Closed) yang digambar dengan garis diagonal. Angka di atas kontak menunjukkan bit operand. Kode mneumonik memberikan informasi yang sama persis seperti halnya diagram ladder.



Gambar 7.1. Contoh Diagram Ladder

Sesungguhnya, program yang disimpan di dalam memori PLC dalam bentuk mneumonik, bahkan meskipun program dibuat dalam bentuk diagram ladder. Oleh karena itu, memahami kode mneumonik itu sangat

7.3.3. Struktur Daerah Memori

Program pada dasarnya adalah pemrosesan data dengan berbagai instruksi pemrograman. Data disimpan dalam daerah memori PLC. Pemahaman daerah data, disamping pemahaman terhadap berbagai jenis instruksi merupakan hal yang sangat penting, karena dari segi inilah intisari pemahaman terhadap program. Data yang merupakan operand suatu instruksi dialokasikan sesuai dengan jenis datanya.

7.3.4. Instruksi Pemrograman

Terdapat banyak instruksi untuk memrogram PLC, tetapi tidak semua instruksi dapat digunakan pada semua model PLC. Instruksi pemrograman dapat dikelompokkan sebagai berikut:

Klasifikasi menurut pengkodean mneumonik :

- Instruksi dasar
- Instruksi khusus

Klasifikasi menurut kelompok fungsi :

- Instruksi sisi kiri (ladder)

- Instruksi sisi kanan

Klasifikasi menurut kelompok fungsi :

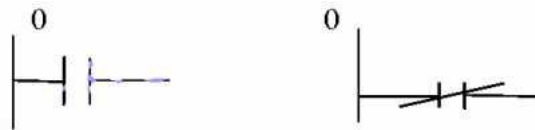
- Instruksi ladder
- Instruksi kendali bit
- Instruksi timer/counter
- Instruksi geser bit
- Instruksi sub routine
- Instruksi ekspansi

Pada dasarnya, tingkat pemahaman pemakai PLC ditentukan oleh seberapa banyak instruksi yang telah dipahaminya. Oleh karena itu, untuk pemula berikut ini hanya dijelaskan beberapa instruksi saja. Untuk pendalaman lebih lanjut dapat mempelajari manual pemrograman yang diterbitkan oleh pemilik merk PLC.

1. Instruksi Diagram Ladder

Instruksi diagram ladder adalah instruksi sisi kiri yang mengkondisikan instruksi lain di sisi kanan. Pada program diagram ladder instruksi ini disimbolkan dengan kontak-kontak seperti pada rangkaian kendali elektromagnet. Instruksi diagram ladder terdiri atas enam instruksi ladder dan dua instruksi blok logika. Instruksi blok logika adalah instruksi yang digunakan untuk menghubungkan bagian yang lebih kompleks. Instruksi LOD dimulai dengan barisan logic yang dapat diteruskan menjadi ladder diagram rung. Instruksi LOD digunakan setiap kali running baru dimulai.

- **Diagram Ladder (relay circuit)**



- **List Program**

Address	Instruction Word	Data
0	LOD	0
1	-	-

- **Key Operation**



2. Instruksi AND

- Instruksi AND digunakan untuk membuat program kontak sirkuit seri.
- Instruksi AND dimasukan sebelum set y ang kedua y ang berhubungan dan Selanjutnya.
- Dapat dilanjuti dengan instruksi NOT untuk contact normally closed.

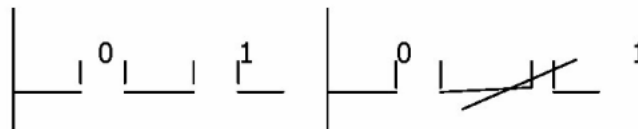
- Gerbang Logic AND



- Tabel Kebenaran gerbang AND

INPUT1	INPUT2	INPUT3
OFF	OFF	OFF
ON	OFF	OFF
OFF	ON	OFF
ON	ON	ON

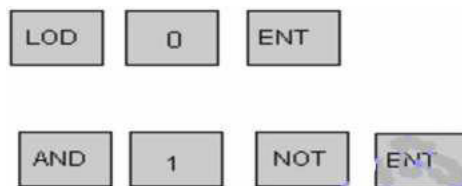
- Diagram Ladder (relay circuit)



- List Program

Address	Instruction Word	Data
0	LOD	0
1	AND	1

- Key Operation



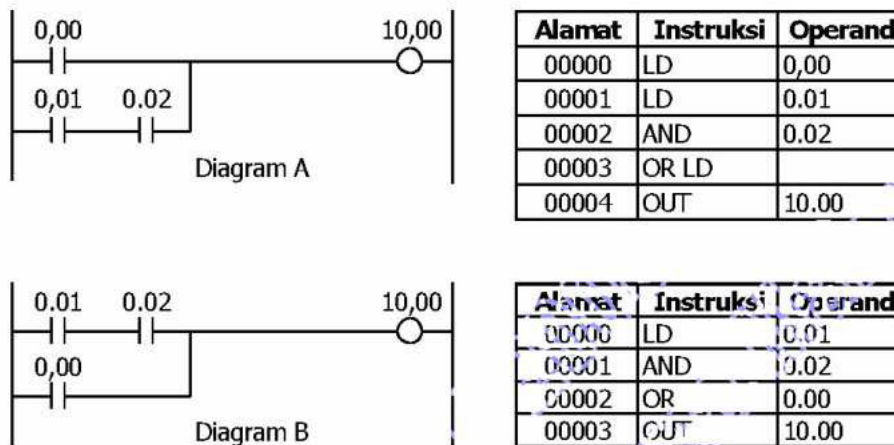
7.3.5. Eksekusi Pemrograman

Untuk mengurangi kemungkinan terjadinya kesalahan dalam merancang program kendali, perlu diingat hal-hal sebagai berikut :

- Jumlah kondisi (kontak) yang digunakan seri atau paralel dan juga banyaknya perulangan penggunaan suatu bit tak terbatas sepanjang kapasitas memori PLC tidak dilampaui.
- Diantara dua garis instruksi tidak boleh ada kondisi yang melintas secara vertikal.
- Tiap garis instruksi harus memiliki sedikitnya satu kondisi yang menentukan eksekusi instruksi sisi kanan, kecuali untuk instruksi END (01), ILC (03) dan JME (05).
- Dalam merancang diagram ladder harus memperhatikan kemungkinan instruksi

yang diperlukan untuk memasukannya. Misalnya, pada gambar A di bawah ini diperlukan instruksi OR LOAD. Hal ini dapat dihindari dengan menggambar ulang diagram ladder seperti gambar B.

Saat eksekusi program, PLC men-scan program dari atas ke bawah, mengecek semua kondisi, dan mengeksekusi semua instruksi. Instruksi harus ditempatkan dengan tepat, misalnya data yang dikehendaki dipindahkan ke words sebelum words tersebut digunakan sebagai operand instruksi.



Gambar 7.2. Penyederhanaan Program Logika

Ingat bahwa garis instruksi berakhir pada instruksi terminal sisi kanan, setelah itu baru mengeksekusi garis instruksi bercabang ke instruksi terminal yang lain. Eksekusi program semata-mata merupakan salah satu tugas yang dilakukan oleh PLC sebagai bagian dari waktu siklus.

7.3.6. Langkah-langkah pembuatan program

Untuk membuat program kendali PLC ditempuh melalui langkah-langkah sistematis sebagai berikut :

- Menguraikan urutan kendali, Pembuatan program diawali dengan penguraian urutan kendali. Ini dapat dibuat dengan menggunakan kalimat-kalimat logika, gambar-gambar, diagram waktu, atau bagan alir (flow chart).
- Menetapkan bit operand untuk peralatan input/output, Bit operand untuk peralatan input/output mengacu pada daerah memori PLC yang digunakan. Bit operand dapat dipilih secara bebas sejauh berada pada jangkah daerah memori yang dialokasikan. Tetapi, penggunaan secara bebas sering menjadikan ketidak-konsistenan sehingga menjadikan program kendali keliru. Oleh sebab itulah penggunaan bit operand harus ditetapkan sebelum program dibuat. Inventarisir semua peralatan input dan output yang akan disambung ke PLC, kemudian tetapkan bit operandnya.

Jumlah bit operand yang tersedia bergantung kepada tipe PLC yang spesifikasikan menurut jumlah input-outputnya. Perbandingan jumlah bit input dan output pada umumnya 3 : 2. Misalnya PLC dengan I/O 10 memiliki bit input sejumlah

6 dan bit output 4. Di bawah ini diberikan contoh daerah memori PLC OMRON CPM1A-10CDRA.

Daerah Data		Words	Bit
IR (Internal Relay)	Input	0	0.00 – 0.11
	Output	10	10.00 – 10.07
	Kerja (internal)	200 – 231	200.00 – 231.15
TR (Temporarily Relay)			TR0 – TR7
Timer/counter		TC0 – TC7	

7.4. Penutup

7.4.1 Bahan diskusi dan tugas

1. Jelaskan maksud dengan SV (set value)?
2. Apa tujuan suatu instruksi ditulis menggunakan kode fungsi?
3. Sebutkan contoh instruksi yang tidak memerlukan operand?
4. Sebutkan contoh instruksi yang tidak memerlukan kondisi?
5. Mengapa bits operand untuk peralatan I/O harus ditetapkan terlebih dahulu sebelum membuat diagram ladder?

7.5 Daftar Pustaka

6. J. Webb, *Industrial Control Electronics*, Macmillan Publish., New York, 1992.
7. J. Webb, *Programmable Logic Control*, Macmillan Publish., New York, 1992.
8. Frank D Petruzella, *Elektronik Industri* (terjemahan), Penerbit Andi, Yogyakarta, 2001.
9. William Bolton, *Programmable Logic Controller (PLC)*, Sebuah Pengantar, Edisi ketiga, Penerbit Erlangga, 2004.
10. M. Budiyanto dan A. Wijaya, *Pengenalan Dasar-dasar PLC (Programmable Logic Controller)*, Penerbit Gaya Media, Yogyakarta, 2003.
11. Endra Ditawarna, *Robotika, Desain, Kontrol dan Kecerdasan Buatan*, Penerbit Andi, Yogyakarta, 2006.
12. David G. Alciator dan Michael B Hestand, *Introduction to Mechatronics and Measurement Systems*, Mc Graw Hill, New York, 2003.

7.6 Senarai

Program kendali PLC terdiri atas tiga unsur yaitu alamat, instruksi dan operand. Program PLC dapat dibuat dengan diagram ladder atau kode mneumonik. Pemilihan tipe program ditentukan oleh alat pemrogram yang akan digunakan. Untuk dapat membuat program kendali PLC, pemrogram harus memahami struktur daerah memori PLC yang akan digunakan. Daerah memori PLC berbeda-beda sesuai dengan tipe PLC. Memahami instruksi pemrograman memegang peranan paling penting

dalam pembuatan program kendali. Terdapat banyak sekali instruksi pemrograman, tetapi tidak semua instruksi dapat diterapkan pada semua tipe PLC. Setiap program selalu diawali dengan instruksi LOAD dan diakhiri dengan instruksi END. Tanpa instruksi END program tidak dapat dieksekusi. Program dieksekusi dengan menscan mulai dari alamat terendah hingga ke alamat tertinggi yaitu instruksi END. Pada diagram ladder ini berarti program dieksekusi mulai dari atas ke bawah bila garis instruksi bercabang, dan kemudian ke kanan hingga mengeksekusi instruksi sisi kanan. Pembuatan program PLC harus dilakukan secara sistematis, yaitu mendeskripsikan sistem kendali, menetapkan operand untuk alat input/ output, baru membuat program.



universitas
MALIKUSSALEH

Mekatronika adalah kata baru yang lahir di Jepang pada awal tahun 1970-an yang merupakan gabungan antara 2 kata yaitu *mechanics* dan *electronics*. Sekarang kita sering melihat di sekeliling kita barang-barang mekatronik seperti robot, mesin bubut NC, kamera digital, printer dan lain sebagainya. Persamaan dari barang-barang mekatronik ini adalah objek yang dikendalikan adalah gerakan mesin. Jika dibandingkan dengan gerakan mesin konvensional maka gerakan mesin tersebut lebih bersifat fleksibel dan lebih memiliki kecerdasan. Hal ini dimungkinkan karena memanfaatkan kemajuan iptek micro-electronics. Artinya dengan bantuan micro-electronics mesin dapat bergerak dengan lebih cerdas. Jika seseorang memberikan sebuah perintah, lalu semua dapat dipasrahkan ke mesin yang dapat bergerak secara otomatis. Ini sangat membantu menciptakan mesin atau alat yang praktis dan mudah digunakan. Sehingga sumber daya pada manusia seperti waktu dan otak dapat dipakai untuk pekerjaan yang lain, sehingga dapat menciptakan nilai tambah.

Pada awalnya mekatronik diarahkan pada 3 target yaitu: penghematan energi (energy saving), pengecilan dimensi dan peringanan berat, dan peningkatan kehandalahan (reliability). Sekarang, setelah 30 tahun lebih berlalu dari kelahirannya, perlu dirumuskan kembali arah mekatronik sesuai dengan perkembangan jaman. Dan khususnya untuk Indonesia sebagai negara yang masih berkembang dengan segudang permasalahannya, rasanya arah mekatronik perlu ditentukan agar dapat membantu memecahkan masalah-masalah yang ada dengan tetap memperhatikan lingkungan regional dan global.

Mekatronik adalah teknologi atau rekayasa yang menggabungkan teknologi tentang mesin elektronika, dan informatika untuk merancang, memproduksi, mengoperasikan dan memelihara sistem untuk mencapai tujuan yang diamanatkan. Seperti diketahui dari definisi, mekatronika adalah gabungan disiplin teknik mesin, teknik elektro, teknik informatika, dan teknik kendali. Pada awalnya, secara khusus tidak ada disiplin mekatronika. Untuk menggabungkan beberapa disiplin iptek tersebut, mekatronika memerlukan teori kendali dan teori sistem.

**FAKULTAS TEKNIK
UNIVERSITAS MALIKUSSALEH**