

BAB II

LANDASAN TEORI

2.1. Pendahuluan

Pada bab ini akan dibahas mengenai teori penunjang dari sistem yang akan dirancang. Beberapa pokok pembahasan yang akan di jabarkan adalah “Mikrokontroler AT89S52, peralatan pendeteksi (sensor *limit switch* dan sensor photodiode), motor penggerak atau *aktuator* (motor DC dan motor AC) serta rangkaian driver motor (driver motor DC dan driver motor AC)”.

2.2. Mikrokontroler AT89S52

Perangkat ini adalah perangkat inti yang merupakan pusat dari sistem. Mikrokontroler AT89S52 adalah keluarga mikrokontroler MCS-51. Dalam penggunaannya IC ini dapat digunakan sebagai pemroses penghitungan matematis, komunikasi, dan banyak lagi hal lainnya. Mikrokontroler AT89S52 adalah mikrokontroler keluaran Atmel dengan 8K *byte Flash Programable and Erasable Read Only Memory* (PEROM). AT89S52 merupakan memori dengan teknologi *nonvolatile memory*, yaitu memori yang dapat ditulis dan dihapus berulang-ulang.

Memori ini biasa digunakan untuk menyimpan instruksi atau perintah berstandar MCS-51, sehingga memungkinkan Mikrokontroler AT89S52 ini bekerja tanpa memerlukan external memory untuk menyimpan source code tersebut. Selain itu AT89S52 memiliki *Random Access Memory* (RAM) sebesar 256 Byte, dan mempunyai 32 jalur atau bit *input* dan *output* (Nalwan, 2003: 1).

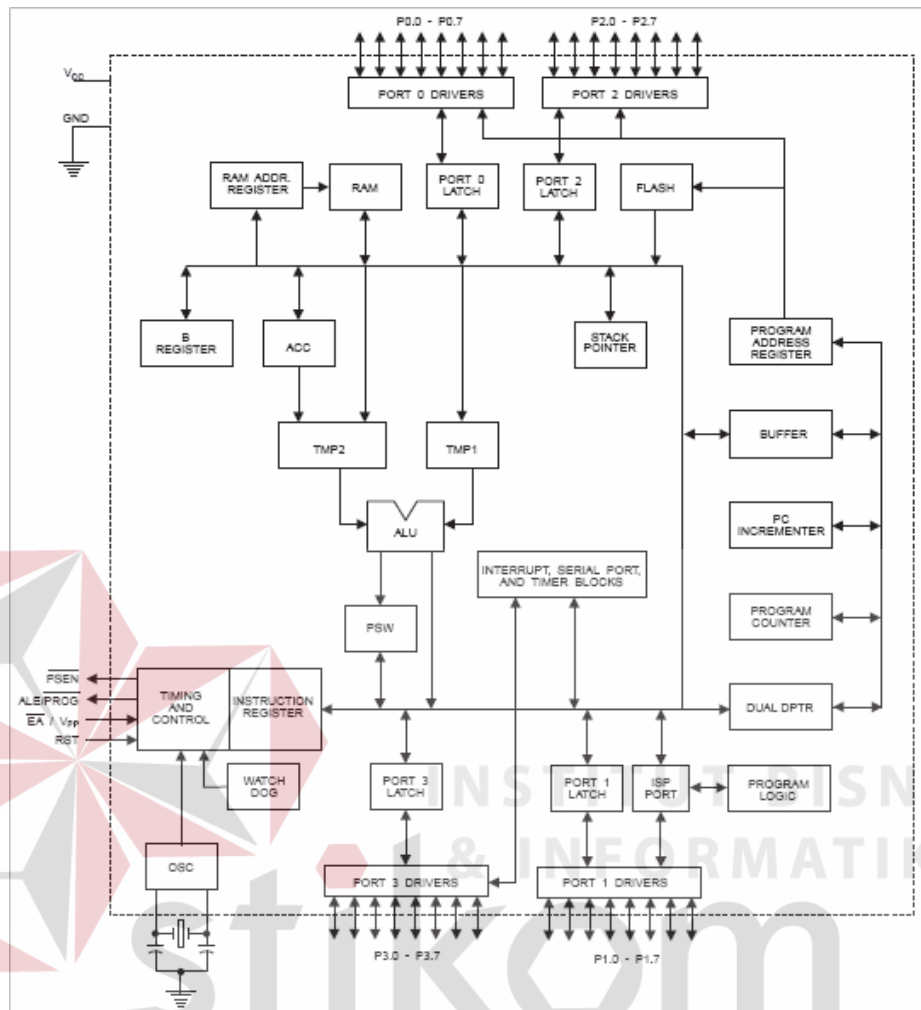
2.2.1. Arsitektur AT89S52

Pada bagian ini dibahas tentang hal-hal yang berhubungan dengan mikrokontroler AT89S52 secara *hardware*. Diskripsi *internal* mikrokontroler, fungsi *pin-pin*, *register-register*, dan lain-lain.

A. Diagram Blok AT89S52

Untuk masukan atau *inputan* data dari luar dan mengontrol peralatan-peralatan *eksternal*, dapat menggunakan fasilitas I/O yang bersifat *Parallel Port*. Mikrokontroler ini mempunyai *On Chip Serial Port* yang dapat digunakan untuk komunikasi data secara *Full Duplex* sehingga *Port Serial* ini masih dapat menerima data pada saat proses pengiriman data terjadi. Fasilitas *Timer* atau *Counter* memungkinkan kita untuk menghitung jumlah kejadian yang terjadi dalam periode tertentu dan lamanya suatu kejadian yang terjadi, ataupun sekedar sebagai *timer* biasa.

Sistem *Interrupt* merupakan fasilitas yang dapat digunakan untuk melakukan penyelaan sejenak terhadap proses yang dilakukan oleh mikrokontroler. Struktur *memory* terdiri dari RAM *Internal* yaitu memori yang mempunyai kapasitas penyimpanan sebesar 128 *byte*, *Special Function Register* atau memori dengan *register-register* dengan fungsi khusus, dan *Flash ROM* yang berfungsi untuk menyimpan instruksi-instruksi. Diagram blok AT89S52 dapat dilihat pada Gambar 2.1. (Atmel Corporation, 2005: 3).

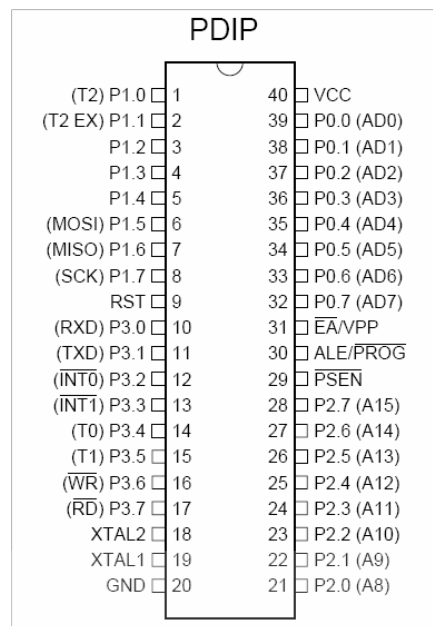


Gambar 2.1. Blok Diagram AT89S52

Sumber: ATMEL Corporation, 2005. 8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52. (Online). (<http://www.atmel.com>, diakses 17 Desember 2008).

B. Fungsi Pin-pin AT89S52

Jumlah *pin* pada mikrokontroler sebanyak empat puluh buah. Masing-masing dari *pin* tersebut mempunyai fungsi tersendiri. Ada yang sebagai *input* atau *output*, kontrol, *supply*, dan lain-lain. Gambar 2.2. merupakan susunan *pin* dari mikrokontroler AT89S52 (Atmel Corporation, 2005: 2).



Gambar 2.2. Susunan *Pin* Mikrokontroler AT89S52

Sumber: ATMEL Corporation, 2005. 8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52. (Online). (<http://www.atmel.com>, diakses 17 Desember 2008).

Keterangan *pin* mikrokontroler AT89S52:

a. *Pin* 40 (Vcc)

Power Supply dihubungkan dengan tegangan +5 Volt.

b. *Pin* 20 (GND)

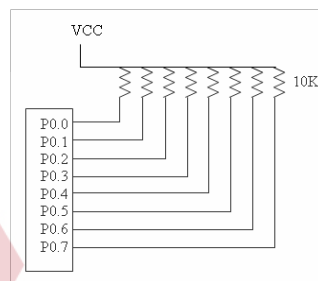
Ground.

c. *Pin* P0.0-P0.7 (*Port* 0) (32...39)

Port 0 dapat difungsikan sebagai port keluaran atau masukan (I/O) bertipe *open drain bidirectional*. Pada saat masing-masing *pin* pada *port* 0 diberikan nilai logika 1, maka *pin-pin Port* 0 dapat digunakan sebagai masukan berimpedansi tinggi.

Jika *Port* 0 dapat dikonfigurasi sebagai *bus* alamat atau data bagian rendah (*low byte*) selama proses pengaksesan memori data dan program

eksternal. Jika digunakan dalam mode ini *Port 0* memiliki *pull up internal*. *Port 0* juga menerima kode–kode yang dikirim kepadanya selama proses pemrograman dan mengeluarkan kode–kode selama proses verifikasi program yang telah tersimpan dalam *flash*. Dalam hal ini dibutuhkan *pullup eksternal* selama proses verifikasi program. Gambar 2.3. Tahapan *pull-up eksternal* (Putra, 2002: 208).



Gambar 2.3. Tahapan *pull-up*

Sumber: Putra, A. E, 2002. Belajar Mikrokontroler AT89C51/52/55 Teori dan Aplikasi. Yogyakarta : Gava Media.

d. *Pin P1.0-P1.7 (Port 1) (1...8)*

Port 1 merupakan I/O dua arah yang dilengkapi dengan *pull up internal*. Penyangga keluaran *Port 1* mampu memberikan atau menyerap arus empat masukan TTL (sekitar 1,6 mA). Jika '1' dituliskan ke kaki–kaki *Port 1*, maka masing–masing kaki akan di-*pulled high* dengan *pull up internal* sehingga dapat digunakan sebagai masukan. Sebagai masukan, jika *pin-pin Port 1* dihubungkan ke ground (di-*low pulled*), maka masing–masing *pin* akan memberikan arus (*source*) karena di-*pulled high* secara *internal*. *Port 1* juga menerima alamat bagian rendah (*low byte*) selama pemrograman dan verifikasi flash.

e. *Pin P2.0-P2.7 (Port 2) (21...28)*

Port 2 merupakan I/O dwi-arah yang dilengkapi dengan *pull up internal*. Penyangga keluaran *port 1* mampu memberikan atau menyerap arus empat masukan TTL (sekitar 1,6 mA). Jika '1' dituliskan ke kaki-kaki *Port 2*, maka masing-masing kaki akan di-pulled high dengan *pull up internal* sehingga dapat digunakan sebagai masukan. Sebagai masukan, jika *pin-pin Port 2* dihubungkan ke *ground* (di-pulled low), maka masing-masing *pin* akan memberikan arus (*source*) karena di-pulled high secara *internal*.

Port 2 akan memberikan byte alamat bagian tinggi (*high byte*) selama pengambilan instruksi dari memori program *eksternal* dan selama pengaksesan memori data *eksternal* yang menggunakan perintah dengan alamat 16-bit. Dalam aplikasi ini, jika ingin mengirimkan '1', maka digunakan *pull up internal* yang sudah disediakan. Selama pengaksesan memori data *eksternal* yang menggunakan perintah dengan alamat 8-bit, *Port 2* akan mengirimkan isi dari SFR P2. *Port 2* juga menerima alamat bagian tinggi selama pemrograman dan verifikasi flash.

f. *Pin P3.0-P3.7 (Port 3) (10...17)*

Port 3 adalah 8 bit I/O *port* dua arah dengan dilengkapi *pull up internal*, sama seperti *Port 1* dan *Port 2* jika difungsikan sebagai *port*. Sedangkan sebagai fungsi special, *port-port* ini mempunyai keterangan yang dijelaskan pada Tabel 2.1. (Putra, 2002: 75) (Mazidi, 200: 5).

Tabel 2.1. Keterangan *Pin* Pada *Port 3*

<i>Port Pin</i>	Fungsi Pengganti
P3.0	RXD (<i>Port Serial Input</i>)
P3.1	TXD (<i>Port Serial Output</i>)
P3.2	INT 0 (<i>Port External Interupt 0</i>)
P3.3	INT 1 (<i>Port External Interupt 1</i>)
P3.4	T0 (<i>Port External Timer 0</i>)
P3.5	T1 (<i>Port External Timer 1</i>)
P3.6	WR (<i>External Data Memory Write Strobe</i>)
P3.7	RD (<i>External Data Memory Read Strobe</i>)

Sumber: ATMEL Corporation, 2005. 8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52. (Online). (<http://www.atmel.com>, diakses 17 Desember 2008).

g. *Pin 9* (Reset)

Reset *input*, aktif *high*. Reset akan aktif dengan memberikan *input high* selama 2 *cycle*. *Pin* ini digunakan untuk mengatur kerja dari perangkat lunak, yaitu menentukan aliran program dalam prosesor, dalam hal ini mikrokontrol. Program yang dijalankan dengan cara *reset*, adalah program utama.

h. *Pin 30* (ALE)

Pin ini dapat berfungsi sebagai *Address Latch Enable* (ALE) yang memberikan sinyal *latch low byte address* pada saat mengakses memori *eksternal*. ALE hanya akan aktif pada saat mengakses memori *eksternal* (MOVX dan MOVC). Ketika *pin* ini bernilai 1 (*high*), maka jalur *address data* berisi alamat I/O.

i. *Pin 29* (PSEN)

Program Store Enable, *pin* ini berfungsi pada saat mengeksekusi program yang terletak pada memori *eksternal*. PSEN akan aktif dua kali setiap *cycle*.

j. *Pin 31 (EA/VPP)*

External Access Enable, pin ini membuat mikrokontroler menjalankan program yang ada pada memori *eksternal* setelah sistem di-*reset*, pada kondisi *low*. Pada kondisi *high*, membuat mikrokontroler menjalankan program yang ada pada memori *internal*.

k. *Pin 19 (XTAL1)*

Input Osilator Eksternal.

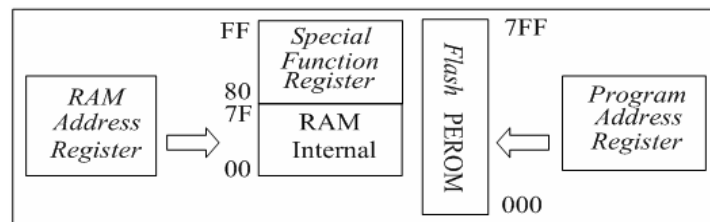
l. *Pin 18 (XTAL2)*

Output Osilator Eksternal.

(Atmel Corporation, 2005: 4-5).

2.2.2. Struktur Memori

Mikrokontroler AT89S52 mempunyai struktur memori yang terpisah antara RAM *Internal* dan *Flash* PEROM-nya. Seperti yang tampak pada Gambar 2.4. RAM *Internal* dialamatkan oleh *RAM Address Register (Register alamat RAM)*. Sedangkan *Flash* PEROM yang menyimpan perintah-perintah MCS-51 dialamatkan oleh *Program Address Register (Register alamat Program)*. Melihat struktur memori tersebut jelas antara RAM *Internal* dan *Flash* PEROM secara fisik keduanya tidak saling berhubungan (Nalwan, 2003: 4).



Gambar 2.4. Alamat RAM *Internal* dan *Flash* PEROM

Sumber: Nalwan, P. A, 2003. Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51, Elex Media Komputindo.

Struktur memori yang terdapat pada AT89S52 terdiri dari:

A. RAM *Internal*

RAM *Internal* mempunyai memori sebesar 256 *byte* yang biasanya digunakan untuk menyimpan variabel atau data yang bersifat sementara terdiri atas tiga bagian utama, yaitu *register banks*, *bit addressable RAM* dan *general purpose* RAM. Berikut ini adalah uraian singkat dari bagian-bagian RAM tersebut:

a. *Register Banks*

Mikrokontroler AT89S52 mempunyai delapan buah *register* yang terdiri atas R0 hingga R7. Kedelapan *register* tersebut terletak pada alamat 00H hingga 07H pada setiap kali sistem di-*reset*. Posisi *register* tersebut dapat dipindah dengan mengatur kombinasi *bit* RS0 dan RS1 (*Register Bank Select Bit*) ke *Bank* 1 hingga *Bank* 0.

b. *Bit Addressable* RAM

Pada alamat 20H hingga 2FH dapat diakses secara pengalamatan bit (*bit addressable*). Jadi dengan sebuah intruksi saja setia bit dalam area ini dapat diset, *clear*, AND dan OR secara langsung. Lokasi yang dapat diakses dengan pengalamatan bit ini dapat juga digunakan untuk menandai suatu lokasi bit

tertentu baik berupa SFR yang dapat dialamati secara bit (termasuk Register I/O) ataupun lokasi-lokasi tertentu yang dapat dialamati secara bit.

c. *General Purpose RAM*

Mempunyai posisi alamat 30H hingga 7FH dan dapat diakses dengan pengalamatan langsung maupun tak langsung. Pengalamatan langsung dilakukan ketika salah satu *operand* merupakan bilangan yang menunjukan lokasi yang dialamati.

Sedangkan pengalamatan secara tak langsung pada lokasi dari RAM *Internal* ini adalah akses data dari memori ketika alamat memori tersebut tersimpan dalam suatu *register* R0 dan R1. Dua buah *register* tersebut pada arsitektur MCS51 yang dapat digunakan sebagai *pointer* dari sebuah lokasi memori pada RAM *Internal*. (Nalwan, 2003: 5-7)

B. *Special Function Register (SFR)*

SFR adalah memori *internal* yang berisi register–register yang mempunyai fungsi khusus seperti *timer*, serial dan lain sebagainya. SFR pada AT89S52 sebanyak 21 buah yang terletak pada alamat antara 80H hingga FFH. Pada SFR ini juga dapat dialamati dengan menggunakan cara pengalamatan *bit* sehingga dapat dioperasikan seperti yang ada pada RAM. Gambar 2.5. menunjukkan peta memori dari *Special Function Register* mikrokontroler AT89S52 (Nalwan, 2003: 9).

Byte	Address	Bit address	
FF			
F0	F7 F6 F5 F4 F3 F2 F1 F0		B
E0	E7 E6 E5 E4 E3 E2 E1 E0		A,C,C
D0	D7 D6 D5 D4 D3 D2 - D0		PSW
B5	- - - BC BB BA B9 B8		IP
B0	B7 B6 B5 B4 B3 B2 B1 B0		P3
A5	A7 - - AC AB AA A9 A8		IE
A0	A7 A6 A5 A4 A3 A2 A1 A0		P2
99	not bit addressable		SBUF
98	9F 9E 9D 9C 9B 9A 99 98		SCON
90	97 96 95 94 93 92 91 90		P1
8D	not bit addressable		TH1
8C	not bit addressable		TH0
8B	not bit addressable		TL1
8A	not bit addressable		TL0
89	not bit addressable		TMOD
88	8F 8E 8D 8C 8B 8A 89 88		TCON
87	not bit addressable		PCON
83	not bit addressable		DPH
82	not bit addressable		DPL
81	not bit addressable		SP
80	87 86 85 84 83 82 81 80		P0

SPECIAL FUNCTION REGISTERS

Gambar 2.5. Peta Memori dari SFR

Sumber: Nalwan, P. A, 2003. Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51, Elex Media Komputindo.

Register-register ini dapat diakses dengan memanggil langsung nama registernya atau dengan memanggil alamatnya. Adapun register–register tersebut adalah:

a. *Accumulator*

Accumulator mempunyai posisi alamat pada E0H. Hampir semua operasi aritmatik dan operasi logika selalu menggunakan *register* ini. Pada proses pengambilan dan pengiriman data ke memori *eksternal*, *register* ini juga diperlukan.

b. *Port*

Seperti yang telah diuraikan diatas bahwa AT89S52 mempunyai empat buah *port* antara lain: *Port 0*, *Port 1*, *Port 2*, dan *Port 3*; yang masing-masing terletak pada alamat 80H, 90H, A0H, dan B0. Apabila kita menggunakan

eksternal memori atau fungsi-fungsi yang spesial (misalnya : *serial*, *external interrupt*, *external timer*), maka *Port 0*, *Port 2*, dan *Port 3* tidak dapat digunakan sebagai *port* dengan fungsi umum. Semua *port* dapat diakses dengan pengalamatan secara *bit*, sehingga perubahan keluaran pada tiap *pin* tidak mempengaruhi *pin* yang lainnya.

c. *Program Status Word*

PSW terletak pada alamat memori D0H.

d. *Register B*

Selain bersifat *addressable*, register ini juga digunakan bersama *Accumulator* untuk proses aritmatik dan logika. Register ini terletak pada alamat 0F0H.

e. *Stack Pointer*

Stack Pointer merupakan *register 8 bit* yang terletak pada alamat 81H. Isi dari *Stack Pointer* ini merupakan alamat dari data yang disimpan di *stack*. *Register* ini dapat diedit atau dibiarkan saja mengikuti standar setelah terjadi *reset*. Jika tidak dilakukan perubahan pada isi *Stack Pointer*, isi dari *register* ini akan selalu berisi 07H sehingga penyimpanan data ke *stack* yang pertama kali adalah pada alamat 08H. Proses yang berhubungan dengan *stack* ini biasa dilakukan oleh instruksi-instruksi *Push*, *Pop*, *Acall*, dan *Lcall*.

f. *Data Pointer*

Data Pointer atau DPTR adalah register 16 bit yang terletak pada alamat 82H untuk *Data Pointer Low* (DPL), dan 83H untuk *Data Pointer High*

(DPH). DPTR biasanya digunakan sebagai pengakses *source code* ataupun data yang terletak pada memori *eksternal*.

g. *Register Timer*

Keluarga mikrokontroler MCS-51 dilengkapi dengan tiga perangkat Timer atau Counter, masing-masing dinamakan sebagai Timer/Counter 0, Timer atau Counter 1, dan Timer 2. Untuk mengakses Timer atau Counter tersebut digunakan register khusus yang tersimpan dalam *Special Function Register* (SFR). Pencacah biner Timer 0 diakses melalui register TL0 (*Timer 0 Low Byte*, memori *internal* alamat 6AH) dan register TH0 (*Timer 0 High Byte*, memori *internal* alamat 6CH). Pencacah biner Timer 1 diakses melalui register TL1 (*Timer 1 Low Byte*, memori *internal* alamat 6BH) dan register TH1 (*Timer 1 High Byte*, memori *internal* alamat 6DH).

Pencacah biner Timer atau Counter pada MCS-51 merupakan pencacah biner 16 bit naik (*count up binary counter*) yang mencacah 0000H sampai FFFFh, saat kondisi pencacah berubah dari FFFFh kembali ke 0000H akan timbul sinyal berlebihan (*overflow*).

Untuk mengatur kerja Timer/Counter tersebut digunakan 2 register tambahan, yaitu register *Timer Control Register* (TCON). Lihat Tabel 2.2. memori data *internal* alamat 88h, dapat diberi alamat per bit. Dan register *Timer Mode Register* (TMOD) Lihat Tabel 2.3. Memori data *internal* alamat 89h, tidak bisa diberi alamat per bit (Nalwan, 2003: 9-12).

Tabel 2.2. Susunan Bit Dalam Register TCON

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

←Timer 1→ ←Timer 0→

Sumber: ATMEL Corporation, 2005. 8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52. (Online). (<http://www.atmel.com>, diakses 17 Desember 2008).

Keterangan Tabel 2.2. susunan register TCON:

- TF1 : *overflow flag* Timer 1/Counter 1 (1 = overflow)
- TR1 : Enable Timer 1/Counter 1
- TF0 : *overflow flag* Timer 0/Counter 0 (1 = overflow)
- TR0 : Enable Timer 0/Counter 0
- IE1 : *External Interrupt 1 edge flag*
- IT1 : *Interrupt 1 type control bit*. Set/clear oleh program untuk menspesifikasi sisi turun/level rendah trigger dari interupsi *eksternal*.
- IE0 : *External Interrupt 0 edge flag*
- IT0 : *Interrupt 0 type control bit*. Set/clear oleh program untuk menspesifikasi sisi turun/level rendah trigger dari interupsi *eksternal*

Tabel 2.3. Susunan Bit Dalam Register TMOD

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
GATE	C/T	M1	M0	GATE	C/T	M1	M0

←Timer 1→ ←Timer 0→

Sumber: ATMEL Corporation, 2005. 8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52. (Online). (<http://www.atmel.com>, diakses 17 Desember 2008).

Keterangan Tabel 2.3. susunan bit register TMOD:

- **GATE** : merupakan bit pengatur sinyal detak. Jika $GATE = 0$, Timer/Counter akan berjalan saat $TR0$ atau $TR1$ pada register TCON (TRx) = 1. Jika $GATE = 1$, Timer/Counter akan berjalan saat $TRx = 1$ atau $INT1$ untuk Timer 1 dan $INT0$ untuk Timer 0 ($INTx$) = 1.
- **C/T** : dipakai untuk mengatur sumber sinyal detak yang diberikan kepada pencacah biner. Jika $C/T = 0$, maka Timer akan aktif dengan sinyal detak diperoleh dari osilator kristal yang frekuensinya sudah dibagi 12. Jika $C/T = 1$, maka Counter akan aktif dengan sinyal detak diperoleh dari kaki $T0$ (untuk Timer 0) dan kaki $T1$ (untuk Timer 1). **M0** dan **M1** : dipakai untuk menentukan Mode Timer/Counter. Operasi Timer/Counter dapat dilihat pada Tabel 2.4. berikut:

Tabel 2.4. Mode Operasi Timer/Counter

M1	M0	Mode	Operasi
0	0	0	Timer/ Counter 13 bit
0	1	1	Timer/Counter 16 bit
1	0	2	Timer auto reload 8 bit
1	1	3	TL0 adalah timer/Counter 8 bit yang dikontrol oleh kontrol bit Timer 0 (TF0). TH0 adalah Timer/Counter 8 bit yang dikontrol oleh kontrol bit Timer 1 (TF1).

Sumber: ATMEL Corporation, 2005. 8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52. (Online). (<http://www.atmel.com>, diakses 17 Desember 2008).

Untuk menghitung *clock frequency* adalah menggunakan perhitungan dengan cara seperti persamaan dibawah ini:

$$T = \frac{1}{\frac{1}{12} \times \text{frekuensi osilator}} \quad (2.1)$$

Sehingga, dengan menggunakan Mode 1 dapat dihitung waktu tunda yang diperlukan dengan perhitungan berikut :

$$\text{Delay} = (65536 - n) \times T \quad (2.2)$$

Dengan n adalah nilai desimal dari nilai heksadesimal THxTLx. (Mazidi, 2000: 159-160)

h. Register Port Serial

Pada mikrokontroler ini mempunyai sebuah *on chip serial port* yang dapat digunakan untuk komunikasi dengan peralatan lain yang menggunakan *serial port* juga seperti *modem*, *shift register*, dan lain-lain. *Buffer* (penyangga) untuk proses pengiriman maupun pengambilan data terletak pada *register* SBUF dengan alamat 99H. Sedangkan pengaturan mode *serial* dapat dilakukan dengan mengubah isi dari SCON yang ada pada alamat 98H.

i. Register Interupsi

Interupsi adalah suatu keadaan *eksternal* atau *internal* yang mengakibatkan mikrokontroler menunda proses yang sedang berjalan untuk menjalankan proses lain. Mikrokontroler MCS-51 menyediakan enam sumber *interupsi*, yaitu dua *interupsi eksternal*, tiga *interupsi timer*, dan satu *interupsi serial*. Masing-masing sumber *interupsi* tersebut dapat diaktifkan dan

dinonaktifkan sendiri-sendiri dengan mengatur bit-bit yang terkait dalam register *Interrupt Enable* (IE) di alamat memori *internal* A8h (Putra, 2002: 156) (Mazidi, 2000: 211). Tabel 2.5 berikut adalah susunan bit yang terdapat dalam *Register Interrupt Enable*.

Tabel 2.5. Susunan Bit Dalam Register IE

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EA	-	ET2	ES	ET1	EX1	ET0	EX0

Sumber: ATMEL Corporation, 2005. 8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52. (Online). (<http://www.atmel.com>, diakses 17 Desember 2008).

Keterangan Tabel 2.5. Susunan bit Register IE:

- EA : Enable semua *interupsi* (0 = non aktif, 1 = aktif).
- ET2 : *Enable Timer 2 overflow* (0 = non aktif, 1 = aktif).
- ES : Enable *interupsi* serial (0 = non aktif, 1 = aktif).
- ET1 : Enable Timer 1 overflow (0 = non aktif, 1 = aktif).
- EX1 : Enable *eksternal interupsi* 1 (0 = non aktif, 1 = aktif).
- ET0 : Enable Timer 0 overflow (0 = non aktif, 1 = aktif).
- EX0 : Enable *eksternal interupsi* 0 (0 = non aktif, 1 = aktif).

Program yang tergabung dalam sistem *interupsi* disebut *Interrupt Service Routine* (ISR) atau *interrupt handler*. Jika terjadi *interupsi*, mikrokontroler akan mengeksekusi ISR. Setiap program *interupsi* berada pada lokasi memori *internal* tertentu dan tidak berpindah-pindah. Pada Tabel 2.6. menunjukkan alamat ISR pada memori *internal* mikrokontroler.

Tabel 2.6. Alamat ISR

No.	Nama	ROM Address	Alat <i>Interupsi</i>
1.	Reset	0000h	Power on Reset (<i>pin</i> 9)
2.	INT0	0003h	<i>Interupsi</i> 0 hardware <i>eksternal</i> (<i>pin</i> 12)
3.	Timer 0	000Bh	Overflow Timer 0 (TF0)
4.	INT1	0013h	<i>Interupsi</i> 1 hardware <i>eksternal</i> (<i>pin</i> 13)
5.	Timer 1	001Bh	Overflow Timer 1 (TF1)
6.	Serial COM	0023h	<i>Port</i> I/O Serial

Sumber: ATMEL Corporation, 2005. 8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52. (Online). (<http://www.atmel.com>, diakses 17 Desember 2008).

Secara default, apabila dalam suatu proses yang sedang berjalan terdapat beberapa *interupsi* yang datang bersamaan, maka urutan prioritas *interupsi* yang dikerjakan sesuai dengan urutan pada Tabel 2.7. Jika ingin mengatur prioritas dari suatu *interupsi*, dapat diatur dalam register *Interrupt Priority* (IP) (Mazidi, 2000: 228).

Tabel 2.7. Susunan Bit Dalam Register IP

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
-	-	PT2	PS	PT1	PX1	PT0	PX0

Sumber: ATMEL Corporation, 2005. 8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52. (Online). (<http://www.atmel.com>, diakses 17 Desember 2008).

Keterangan Tabel 2.7. Susunan bit register IP:

- PT2: prioritas *interupsi* Timer 2 (0= prioritas rendah, 1= prioritas tinggi).
- PS: prioritas *interupsi* Serial (0= prioritas rendah, 1= prioritas tinggi).
- PT1: prioritas *interupsi* Timer 1 (0= prioritas rendah, 1= prioritas tinggi).
- PX1: prioritas *interupsi* INT1 (0= prioritas rendah, 1= prioritas tinggi).

- PT0: prioritas *interupsi* Timer 0 (0= prioritas rendah, 1= prioritas tinggi).
- PX0: prioritas *interupsi* INT0 (0= prioritas rendah, 1= prioritas tinggi).

C. **Flash PROM**

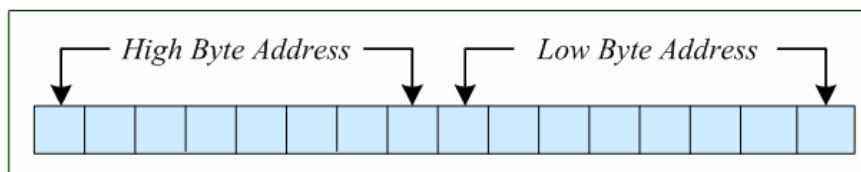
Flash PEROM adalah memori *internal* yang digunakan untuk menyimpan instruksi-instruksi MCS51. *Flash* PEROM pada AT89S52 mempunyai 8K Byte, ROM yang dapat ditulis dan dihapus secara berulang-ulang hingga kurang lebih sampai 1000 kali (Nalwan, 2003: 17).

2.2.3. Memori *Eksternal*

Mikrokontroler AT89S52 tidak hanya mempunyai kemampuan dalam memaksimalkan kemampuannya sendiri, tetapi juga dapat melakukan interaksi dengan peralatan lain seperti memori tambahan yang dapat diakses mikrokontroler walaupun memori tambahan tersebut berada di luar dirinya.

Memori tambahan biasanya diperlukan ketika kita membuat aplikasi yang membutuhkan kapasitas memori yang cukup besar, sedangkan memori pada mikrokontroler tidak mencukupi.

Pengalamatan memori pada AT89S52 dapat dilakukan dengan cara *multiplex addressing* (pengalamatan bergantian) yaitu memultiplex data dan *low byte address*. Dengan sistem ini, *port 2* dan *port 0* dapat melakukan pengalamatan untuk 64 K *byte* alamat memori. Gambar 2.6. adalah sistem pengalamatan memori pada AT89S52 (Nalwan, 2003: 18).



Gambar 2.6. 16 bit Pengalamatan Memori

Sumber: Nalwan, P. A, 2003. Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51, Elex Media Komputindo.

Pengaksesan memori mempunyai dua jenis pengaksesan yaitu pembacaan (*read*) atau penulisan (*write*). Untuk proses pembacaan data pada alamat memori tambahan (*eksternal*), terdiri dari beberapa bagian penting yaitu:

- Dibutuhkan alamat dari lokasi memori yang akan diambil/dibaca.
- Perintah untuk pembacaan/pengambilan data dari memori *eksternal*.

Sedangkan untuk proses penulisan data pada alamat memori tambahan (*eksternal*), terdiri dari beberapa bagian penting yaitu:

- Dibutuhkan alamat dari lokasi yang akan ditulis pada memori *eksternal*.
- Harus ditentukan data yang akan dikirim.
- Perintah untuk penulisan/pengiriman data ke memori *eksternal*.

(Nalwan, 2003: 19).

2.2.4. Operasi Serial Port AT89S52

AT89C51 mempunyai *on chip serial port* yang dapat digunakan untuk melakukan komunikasi data serial secara *full duplex* sehingga *port serial* ini masih dapat menerima data pada saat proses pengiriman data terjadi. Untuk menampung data yang diterima atau data yang akan dikirimkan, 89C51 mempunyai sebuah register, yaitu SBUF yang terletak pada alamat 99H (Nalwan, 2003: 39).

Komunikasi data *port* serial ada dua jenis, yaitu komunikasi data sinkron dan asinkron. Kedua jenis komunikasi tersebut dibedakan oleh sinyal denyut (*clock*) yang dipakai untuk mendorong data serial, jika *clock* dikirim bersama dengan data serial, cara tersebut dikatakan sebagai transmisi data serial secara sinkron. Sedangkan dalam transmisi data seri secara asinkron, *clock* tidak dikirim bersama data serial, rangkaian penerima data harus membangkitkan sendiri *clock* pendorong data serial. Umumnya orang selalu menganggap bahwa *port* serial pada MCS51 adalah UART yang bekerja secara asinkron, jarang yang menyadari *port* serial tersebut bisa pula bekerja secara sinkron, padahal sebagai *port* serial yang bekerja secara sinkron merupakan sarana yang baik sekali untuk menambah masukan/keluaran bagi mikrokontroler. Saat ini yang akan kita bahas adalah komunikasi data asinkron (Nalwan, 2003: 40).

Komunikasi data asinkron adalah komunikasi data yang tidak memerlukan sinyal *clock* dalam pengiriman datanya (sebagai sinkronisasi), namun pengiriman data ini harus diawali dengan *start* bit dan diakhiri dengan *stop* bit (Nalwan, 2003: 42).

A. Pengaturan *Baud Rate Port Serial*

Baud rate dari *port* serial AT89S51 dapat diatur pada Mode 1 dan Mode 3, namun pada Mode 0 dan Mode 2, *baud rate* tersebut mempunyai kecepatan yang tetap. Kecepatan pada Mode 0 yaitu $1/12$ frekuensi osilator, sedangkan kecepatan pada Mode 2 adalah $1/64$ frekuensi osilator (Nalwan, 2003: 46).

Dengan mengubah bit SMOD yang terletak pada Register PCON menjadi bernilai 1 (kondisi awal pada saat sistem di-reset nilai bit SMOD adalah 0), baud rate pada Mode 1, 2, dan 3 akan berubah menjadi dua kali lipat (Nalwan, 2003: 46).

Pada Mode 1 dan 3, baud rate dapat diatur dengan menggunakan Timer 1. Cara yang biasa digunakan adalah Timer Mode 2 (*8 bit auto reload*) yang hanya menggunakan register TH1 saja. Pengiriman setiap bit data terjadi setiap Timer 1 overflow sebanyak 32 kali sehingga dapat disimpulkan bahwa (Nalwan, 2003 : 46).

$$\text{Lama pengiriman setiap bit data} = \text{Timer 1 overflow} \times 32 \quad (2.3)$$

$$\text{Baud rate (Jumlah bit data yang terkirim tiap detik)} = \frac{1}{\text{Timer1_overflow} \times 32} \text{ detik} \quad (2.4)$$

Apabila diinginkan baud rate 9600 bit per detik, maka Timer 1 harus diatur agar overflow setiap $\frac{1}{9600 \times 32}$ detik.

Timer 1 overflow setiap kali TH1 mencapai nilai overflow dengan frekuensi sebesar $f_{osc}/12$ atau periode sebesar $12/f_{osc}$. Dari sini akan ditemukan formula sebagai berikut (Nalwan, 2003: 46):

$$\frac{12 \times (256 - TH1)}{f_{osc}} = \frac{1}{9600 \times 32} \quad (2.5)$$

$$9600 = \frac{f_{osc}}{12 \times (256 - TH1) \times 32} \quad (2.6)$$

Dengan frekuensi osilator sebesar 11,0592 MHz, nilai desimal TH1 adalah 253 atau 0xFD dalam heksa desimal. Selain variabel-variabel diatas, masih terdapat sebuah variabel lagi yang menjadi pengatur baud rate serial, yaitu bit SMOD pada Register SCON. Jika bit ini diset (bernilai 1), faktor pengali 32 pada formula 3.1 akan berubah menjadi 16. Oleh karena itu dapat disimpulkan formula untuk baudrate serial untuk Mode 1 dan Mode 3 adalah (Nalwan, 2003: 47).:

$$\text{Baud rate} = \frac{f_{osc}}{12 \times (256 - TH1) \times K} \quad (2.7)$$

Tabel 2.8. Tabel Mode Serial Vs Baud Rate

Mode	Baud rate	
0	$1/12 f_{osc}$	
	SMOD = 0	SMOD = 1
1	$\text{Baud rate} = \frac{f_{osc}}{12 \times (256 - TH1) \times 32}$	$\text{Baud rate} = \frac{f_{osc}}{12 \times (256 - TH1) \times 16}$
2	$1/32 f_{osc}$	$1/32 f_{osc}$
3	$\text{Baud rate} = \frac{f_{osc}}{12 \times (256 - TH1) \times 32}$	$\text{Baud rate} = \frac{f_{osc}}{12 \times (256 - TH1) \times 16}$

Sumber: Nalwan, P. A, 2003. Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51, Elex Media Komputindo.

B. Register-register *Port Serial* MCS51

MCS51 dilengkapi dengan 2 register dan beberapa bit tambahan untuk keperluan pemakai *port serial*.

B.1. *Serial Data Buffer (SBUF)*

SBUF adalah sebuah register 8-bit yang selalu digunakan untuk komunikasi serial dalam MCS-51. Data-data yang dikirimkan lewat jalur TxD, akan selalu disimpan didalam register SBUF demikian juga ketika menerima data,

SBUF menahan data yang diterima pada RxD. SBUF dapat diakses seperti halnya register-register yang lain. SBUF merupakan *Special Function Register* (SFR) yang terletak pada memori-data *internal* dengan alamat 99H (0x99) (Nalwan, 2003: 39).

B.2. *Serial Port Control Register* (SCON)

Register SCON adalah register 8 bit yang digunakan program untuk menentukan *start* bit, *stop* bit, dan yang lainnya. Gambar 2.7. berikut menunjukkan berbagai macam bit pada register SCON (Nalwan, 2003: 43).

	SCON.7	SCON.6	SCON.5	SCON.4	SCON.3	SCON.2	SCON.1	SCON.0
98 H	SM0	SM1	SM2	REN	TB8	RB8	TI	RI

Gambar 2.7. Pengalamatan Register SCON

Sumber: Nalwan, P. A, 2003. Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51, Elex Media Komputindo.

SCON merupakan *Special Function Register* (SFR) yang terletak pada memori-data *internal* dengan alamat 98H (0x98). Dari gambar diatas terdapat bit bit dengan fungsi-fungsi tertentu antarlain:

- Bit SM0 dan bit SM1

SM0 dan SM1 adalah data bit ke 7 dan ke 6 dari register SCON. Kedua bit ini akan menentukan mode kerja dari *port* serial. Tabel 2.9. menunjukkan kombinasi dari kedua bit SM0 dan SM1 dari register SCON.

Tabel 2.9. Kombinasi SM0 dan SM1 register SCON

SM0	SM1	Mode	Keterangan
-----	-----	------	------------

0	0	0	Shif register 8 bit
0	1	1	UART 8 bit dengan baud rate yang dapat diatur
1	0	2	UART 9 bit dengan baud rate permanen
1	1	3	UART 9 bit dengan baud rate yang dapat diatur

Sumber: Nalwan, P. A, 2003. Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51, Elex Media Komputindo.

Dari keempat mode diatas, mode serial yang sangat kompatibel dengan komputer, khususnya IBM-PC adalah Serial Mode 1, ditambah lagi Serial Mode 1 dapat diakses dengan baud rate yang bervariasi.

- SM2

SM2 digunakan untuk komunikasi multiprosesor. Biasanya bit ini diberikan nilai 0.

- REN

Receive Enable (REN) adalah bit ke 4 dari Register SCON. Ketika bit ini mempunyai nilai 1, akan mengizinkan mikrokontrol untuk menerima data dari jalur RxD. Demikian juga apabila ingin dikirimkan data lewat jalur TxD, nilai yang diberikan pada bit REN adalah 1. Apabila nilai REN 0, mengakibatkan tidak diaktifkannya penerima.

- TB8

Transfer bit 8 (TB8) adalah bit ke 3 dari Register SCON. Bit ini digunakan untuk Mode Serial 2 dan 3. Biasanya nilai bit ini dibuat 0, meskipun bit ini tidak digunakan.

- RB8

Receive bit (RB8) adalah bit ke 2 dari Register SCON. Pada serial mode 1, bit ini mendapatkan salinan dari nilai stop bit ketika 8-bit data telah

dikirimkan. Sama halnya dengan bit TB8, bit ini juga jarang digunakan dan nilai bit ini dibuat menjadi 0. bit RB8 juga digunakan pada Mode Serial 2 dan 3.

- TI

Transmit Interrupt (TI) adalah bit ke 1 dari Register SCON, bit ini merupakan tanda yang setara dengan tanda *Transmitter Data Register Empty* (TDRE) yang umum dijumpai pada UART standard. Setelah *port* serial selesai mengirimkan data yang disimpan kedalam SBUF, bit TI akan bernilai 1 dengan sendirinya, bit ini harus di-nol-kan dengan program agar bisa dipakai untuk memantau keadaan SBUF dalam pengiriman data berikutnya.

- RI

Receive Interrupt (RI) adalah bit ke 0 dari Register SCON, bit ini merupakan tanda yang setara dengan tanda *Receiver Data Register Full* (RDRF) yang umum dijumpai pada UART standard. Setelah SBUF menerima data dari *port* seri, bit RI akan bernilai 1 dengan sendirinya, bit ini harus di-nol-kan dengan program agar bisa dipakai untuk memantau keadaan SBUF dalam penerimaan data berikutnya.

(Nalwan, 2003: 43-45).

2.3. Peralatan Pendeteksi (Sensor)

Sensor adalah alat untuk mendeteksi atau mengukur sesuatu, yang digunakan untuk mengubah variasi mekanis, magnetis, panas, sinar dan kimia menjadi tegangan dan arus listrik. Dalam lingkungan sistem pengendali dan robotika, sensor memberikan kesamaan yang menyerupai mata, pendengaran, hidung, lidah yang kemudian akan diolah oleh kontroler sebagai otaknya (Petruzella, 2001).

Sensor dalam teknik pengukuran dan pengaturan secara elektronik berfungsi mengubah besaran fisik (misalnya: temperatur, gaya, kecepatan putaran) menjadi besaran listrik yang proposional. Sensor dalam teknik pengukuran dan pengaturan ini harus memenuhi persyaratan-persyaratan kualitas yakni (<http://www.indomicron.co.cc/elektronika/analog/pengertian-sensor/>):

- Linieritas
Konversi harus benar-benar proposional, jadi karakteristik konversi harus linier
- Tidak tergantung temperatur.
Keluaran konverter tidak boleh tergantung pada temperatur di sekelilingnya, kecuali sensor suhu.
- Kepekaan
Kepekaan sensor harus dipilih sedemikian, sehingga pada nilai-nilai masukan yang ada dapat diperoleh tegangan listrik keluaran yang cukup besar).
- Waktu tanggapan
Waktu tanggapan adalah waktu yang diperlukan keluaran sensor untuk mencapai nilai akhirnya pada nilai masukan yang berubah secara mendadak.

Sensor harus dapat berubah cepat bila nilai masukan pada sistem tempat sensor tersebut berubah.

- Batas frekuensi terendah dan tertinggi

Batas-batas tersebut adalah nilai frekuensi masukan periodik terendah dan tertinggi yang masih dapat dikonversi oleh sensor secara benar. Pada kebanyakan aplikasi disyaratkan bahwa frekuensi terendah adalah 0Hz..

- Stabilitas waktu

Untuk nilai masukan (*input*) tertentu sensor harus dapat memberikan keluaran (*output*) yang tetap nilainya dalam waktu yang lama.

- Histerisis

Gejala histerisis yang ada pada magnetisasi besi dapat pula dijumpai pada sensor. Misalnya, pada suatu temperatur tertentu sebuah sensor dapat memberikan keluaran yang berlainan.

Empat sifat diantara syarat-syarat di atas, yaitu linieritas, ketergantungan pada temperatur, stabilitas waktu dan histerisis menentukan ketelitian sensor (<http://www.indomicron.co.cc/elektronika/analog/pengertian-sensor/>).

2.3.1. Sensor *Limit switch*

Limit switch digunakan pada semua instalasi automasi dan juga aplikasi yang beragam. *Limit switch* digunakan bila objek yang akan dideteksi dapat disentuh. *Limit switch* ini bekerja berdasarkan perubahan kondisi kontak yang terdapat didalamnya, dari tertutup menjadi terbuka atau sebaliknya. Pemilihan *limit switch* dilakukan berdasarkan ukuran, material dan jenis aktuator dari *limit*

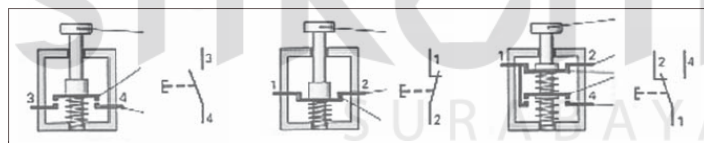
switch yang sesuai dengan fungsinya. Gambar 2.8. berikut adalah contoh model sensor *limit switch*.



Gambar 2.8. Sensor *Limit switch*

Sumber: (<http://www.inotek.com/Catalog/omron2cn.html>).

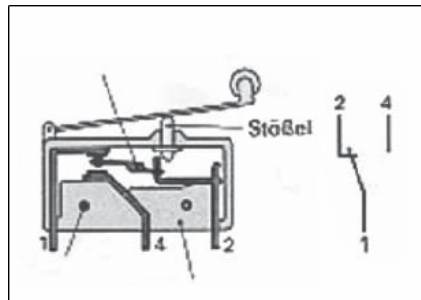
Bagian ini biasanya berupa rangkaian tertutup dan mempunyai bagian *output* yang digunakan untuk menyambungkannya dengan komponen atau bagian lain sesuai dengan kebutuhan. Dipasaran biasanya tersedia dalam bentuk saklar tekan (*pushbutton*) atau *selector switch*, *normally open*, *normally closed* atau dalam bentuk *toggle* Gambar 2.9. (Teknik pengaturan otomatis.pdf : 333-334).



Gambar 2.9. Tombol NO, NC dan Toggle

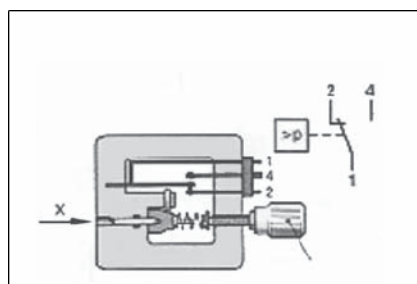
Sumber: (<http://125.163.203.113/buku/TEKNIK%20LISTRIK%20INDUSTRI%203/BAB%2012%20-%20Teknik%20Pengaturan%20Otomts.pdf>).

Sebuah *limit switch* mekanik Gambar 2.10 dapat diset pada posisi tertentu, sehingga ketika ada benda kerja yang menyentuh *limit switch* tersebut, maka dia akan mengeluarkan sinyal untuk mengontrol kerja mesin atau bagian dari mesin. *Limit switch* biasanya berfungsi sebagai pembuka atau penyambung dan pengubah aliran arus. (Teknik pengaturan otomatis.pdf : 334).

Gambar 2.10. *Limit switch*

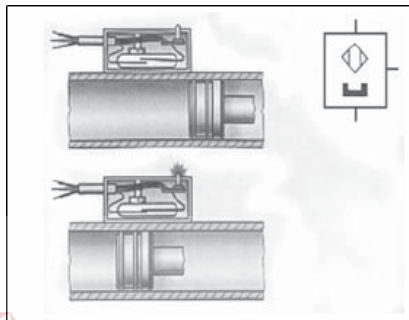
Sumber: (<http://125.163.203.113/buku/TEKNIK%20LISTRIK%20INDUSTRI%203/BAB%2012%20-%20Teknik%20Pengaturan%20Otomts.pdf>).

Saklar tekan biasanya berfungsi sebagai penyambung, pemutus atau pengubah aliran arus dengan cara mengeset saklar pada tekanan tertentu. Ketika tekanan mencapai nilai seting yang ditetapkan, maka saklar akan terbuka atau tertutup, atau mengalihkan arah arus. Tekanan *input* didapat dari sebuah piston yang akan menghasilkan daya tekan. Daya tekan tersebut dapat diatur melalui sebuah tombol putar Gambar 2.11. Ketika ada tekanan melebihi nilai setingnya, maka *limit switch* akan bekerja. Saat ini banyak digunakan saklar tekan yang bekerja secara elektronis. Saklar tekan elektronis bekerja melalui tekanan yang terjadi pada membran. (Teknik pengaturan otomatis.pdf : 334).

Gambar 2.11. *Limit switch Tekan*

Sumber: (<http://125.163.203.113/buku/TEKNIK%20LISTRIK%20INDUSTRI%203/BAB%2012%20-%20Teknik%20Pengaturan%20Otomts.pdf>).

Saklar magnet jenis proximity juga dapat diset pada posisi tertentu dalam silinder Gambar 2.12. Biasanya rumah kontak saklar ini berupa diode jenis LED yang akan langsung menyala saat terjadi kontak (saklar tersambung). (Teknik pengaturan otomatis.pdf : 334).



Gambar 2.12. Proximity Switch Terpasang Pada Silinder

Sumber: (<http://125.163.203.113/buku/TEKNIK%20LISTRIK%20INDUSTRI%203/BAB%2012%20-%20Teknik%20Pengaturan%20Otomatis.pdf>).

2.3.2. Sensor Photodiode

Sensor photodiode adalah termasuk jenis sensor optik (cahaya), yaitu sensor yang bekerja berdasarkan ada atau tidaknya cahaya yang diterima oleh bagian penerima dari sensor tersebut. Sensor cahaya atau sensor sinar dibagi menjadi tiga kategori yaitu (Mengenal sensor dan aktuator, <http://www.caltron.co.id>):

- *Fotovoltaic* atau sel solar adalah alat sensor sinar yang mengubah energi sinar langsung menjadi energi listrik, dengan adanya penyinaran cahaya akan menyebabkan pergerakan elektron dan menghasilkan tegangan.
- *Fotokonduktif* (fotoresistif) adalah sensor yang bekerja dengan cara memberikan perubahan tahanan (resistansi) pada sel-selnya, semakin tinggi

intensitas cahaya yang diterima, maka akan semakin kecil pula nilai tahanannya.

- *Fotolistrik* adalah sensor yang berprinsip kerja berdasarkan pantulan karena perubahan posisi atau jarak suatu sumber sinar (inframerah atau laser) ataupun target pemantulnya, yang terdiri dari pasangan sumber cahaya dan penerima. (<http://www.caltron.co.id/modules.php?name=News&file=article&sid=11>).

2.4. Motor Penggerak

Melalui medium medan magnet, bentuk energi mekanik dapat diubah menjadi energi listrik. Hal tersebut berlangsung melalui alat konversi yang disebut generator. Sebaliknya dengan alat konversi yang disebut motor, energi listrik dapat diubah menjadi energi mekanik (Zuhail, 1991: 1).

2.4.1. Motor DC

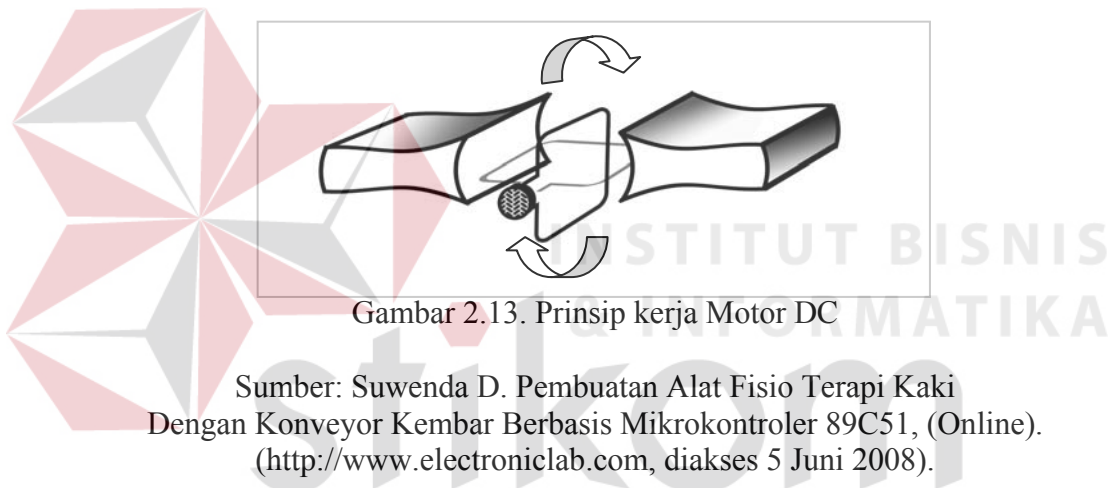
A. Dasar Motor DC

Motor DC atau mesin arus searah pada dasarnya sama dengan mesin arus bolak-balik, perbedaannya adalah bahwa mesin arus searah mempunyai suatu komutator yang berfungsi mengubah tegangan bolak-balik menjadi tegangan searah (Zuhail, 1991: 134).

Pada percobaan *Maxwell* dihasilkan, bilamana arus listrik yang mengalir dalam kawat yang arahnya menjauhi kita (maju), maka medan-medan yang terbentuk di sekitar kawat arahnya searah dengan putaran jarum jam. Begitu pula sebaliknya, bila mana arus listrik yang mengalir dalam kawat arahnya mendekati

kita (mundur) maka medan-medan magnet yang terbentuk di sekitar kawat arahnya berlawanan dengan arah putaran jarum jam (Sumanto, 1984 : 107).

Prinsip kerja dari Motor DC yaitu, apabila suatu kawat berarus diletakkan diantara kutub-kutub magnet kutub U (utara) dan kutub S (selatan), maka pada kawat itu akan bekerja suatu gaya yang menggerakkan kawat tersebut. Gambar 2.13. berikut menggambarkan prinsip dasar kerja Motor DC (Suwenda D. http://www_electroniclab_com%20-%20main%20page.html).



Gambar 2.13. Prinsip kerja Motor DC

Sumber: Suwenda D. Pembuatan Alat Fisio Terapi Kaki Dengan Konveyor Kembar Berbasis Mikrokontroler 89C51, (Online). (<http://www.electroniclab.com>, diakses 5 Juni 2008).

Arah gerak kawat ditentukan dengan “Kaidah Tangan Kiri”, yang berbunyi:

Apabila tangan kiri terbuka diletakkan diantara kutub U dan kutub S, sehingga garis-garis gaya yang keluar dari kutub U menembus telapak tangan kiri dan arus di dalam kawat mengalir searah dengan arah keempat jari, maka kawat itu mendapat gaya yang arahnya sesuai dengan arah ibu jari (Sumanto, 1984 : 107) besarnya gaya tersebut adalah:

$$F = BIl \cdot 10^{-7} \text{ dyne}$$

(2.8)

Dimana: B adalah kepadatan flux magnet (gauss)

l panjang penghantar (cm)

I arus listrik (Ampere)

atau

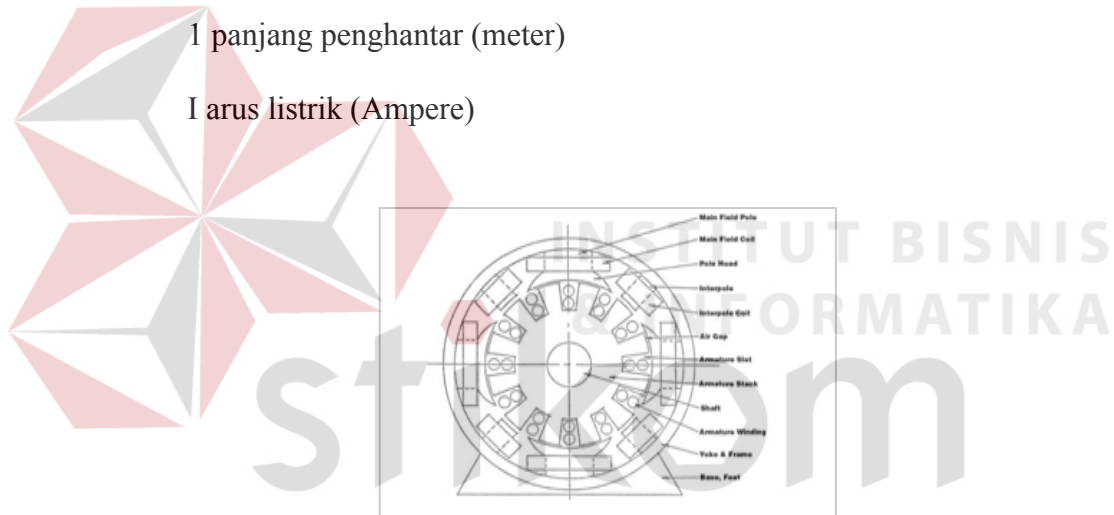
$$F = BIl \text{ Newton}$$

(2.9)

Dimana: B kepadatan flux magnet (Weber)

l panjang penghantar (meter)

I arus listrik (Ampere)



Gambar 2.14. Kontruksi Motor DC

Sumber: Suwenda D. Pembuatan Alat Fisio Terapi Kaki
 Dengan Konveyor Kembar Berbasis Mikrokontroler 89C51, (Online).
 (<http://www.electroniclab.com>, diakses 5 Juni 2008).

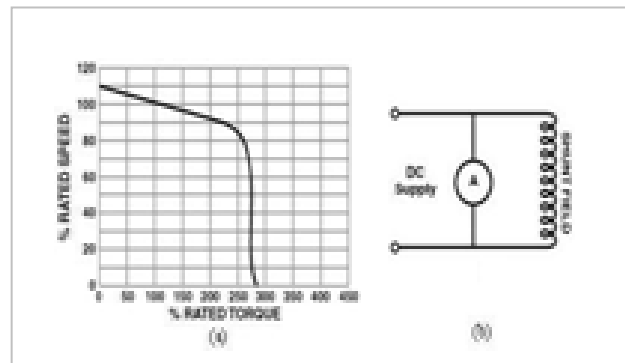
Motor DC memiliki rotor yang terdiri dari kumparan dan komutator yang banyak untuk mendapatkan torsi terus menerus, seperti yang ditunjukkan pada Gambar 2.14 (Suwenda D. http://www_electroniclab_com%20-%20main%20page.html).

B. Jenis-jenis Motor DC

Ada beberapa jenis motor DC yang dapat kita jumpai dipasaran. Motor DC dibedakan atas dua jenis, yaitu (Suwenda D. http://www_electroniclab_com%20-%20main%20page.html):

- Berdasarkan sumber arus penguat magnet, motor DC dibedakan atas:
 - Motor DC permanen magnet
 - Motor DC penguat terpisah, bila arus penguatan medan diperoleh dari sumber DC diluar motor.
 - Motor DC dengan penguat sendiri, bila arus penguatan magnet berasal dari motor itu sendiri.
- Berdasarkan hubungan lilitan penguat magnet terhadap liitan jangkar, motor DC dibedakan atas tiga jenis, yaitu:
 - Motor DC Shunt

Motor jenis ini memiliki kumparan medan yang dihubungkan secara paralel dengan kumparan jangkar sehingga akan menghasilkan kecepatan yang relatif konstan meskipun terjadi perubahan, perubahan beban hanya sekitar 10 % dapat dilihat pada Gambar 2.15. Berikut.

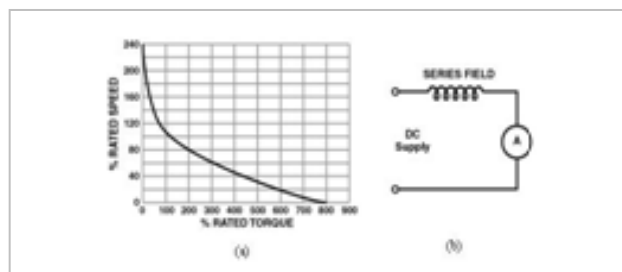


Gambar 2.15. Motor DC Shunt

Sumber: Suwenda D. Pembuatan Alat Fisio Terapi Kaki Dengan Konveyor Kembar Berbasis Mikrokontroler 89C51, (Online). (<http://www.electroniclab.com>, diakses 5 Juni 2008).

- Motor DC Seri

Motor DC ini memiliki medan penguat yang dihubungkan seri dengan medan jangkar. Arus jangkar untuk motor jenis ini lebih besar dari pada kumparan jangkar untuk motor jenis shunt, selain itu kumparan (N_s) lebih sedikit. Tahanan (R_f) lebih kecil, ini disebabkan tahanan tersebut merupakan bagian dari jumlah lilitan yang sedikit. Kecepatan dari motor DC jenis ini dapat diatur melalui tegangan suplai. Gambar 2.16. Jenis Motor DC Seri.

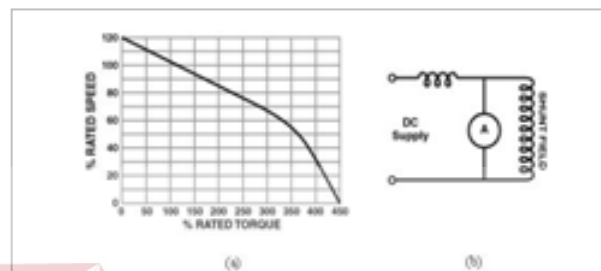


Gambar 2.16. Motor DC Seri

Sumber: Suwenda D. Pembuatan Alat Fisio Terapi Kaki Dengan Konveyor Kembar Berbasis Mikrokontroler 89C51, (Online). (<http://www.electroniclab.com>, diakses 5 Juni 2008).

- Motor DC Kompond

Motor ini merupakan gabungan dari sifat-sifat pada motor shun dan seri, tergantung mana lebih kuat lilitannya. Umumnya motor jenis ini memiliki momen start yang besar sekitar 25 % terhadap kecepatan tanpa beban. Gambar 2.17 Motor DC Jenis Kompond.



Gambar 2.17. Motor DC Kompond

Sumber: Suwenda D. Pembuatan Alat Fisio Terapi Kaki Dengan Konveyor Kembar Berbasis Mikrokontroler 89C51, (Online). (<http://www.electroniclab.com>, diakses 5 Juni 2008).

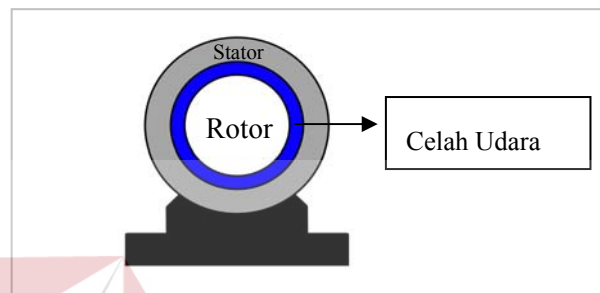
2.4.2. Motor AC

Perputaran motor pada mesin bolak-balik (motor AC) ditimbulkan oleh adanya medan putar (fluks yang berputar) yang dihasilkan dalam kumparan statornya. Medan putar ini terjadi apabila kumparan stator dihubungkan dalam fasa banyak (Zuhal, 1991: 65).

A. Konstruksi Motor AC

Konstruksi Motor AC terdiri atas dua komponen, yaitu stator dan rotor. Stator adalah bagian dari motor yang tidak bergerak dan rotor adalah bagian dari motor yang bergerak yang bertumpu pada bantalan poros terhadap stator. Motor AC terdiri atas kumparan-kumparan stator dan rotor yang berfungsi

membangkitkan gaya gerak listrik akibat dari adanya arus listrik bolak-balik yang melewati kumparan-kumparan tersebut, sehingga terjadi suatu interaksi induksi medan magnet antara stator dan rotor. Bentuk dan konstruksi motor tersebut digambarkan pada gambar dibawah ini. Gambar 2.18. Konstruksi Motor AC (Sofwan dan Baqo, 2004: 709).

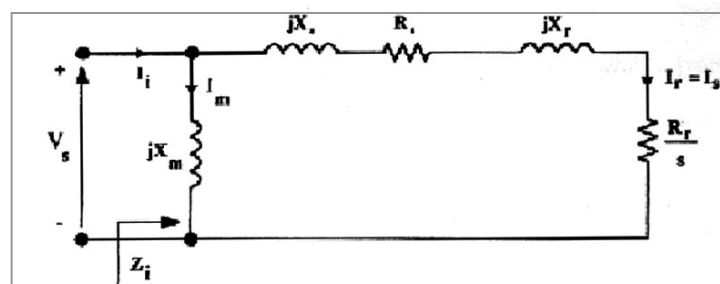


Gambar 2.18. Konstruksi Motor AC

Sumber: Sofwan A dan Baqo S, Sistem Pengendalian Kecepatan Putaran Motor AC Phasa Satu Menggunakan Mikrokontroler AT89S8252, Motor AC.pdf

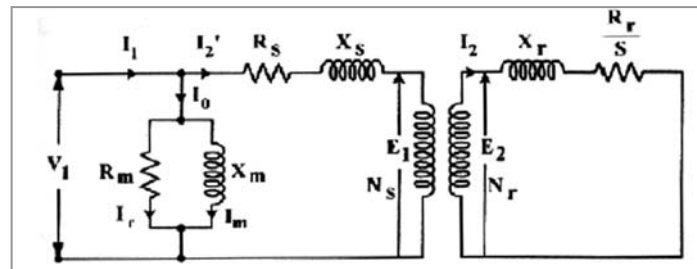
B. Rangkaian Ekuivalen Motor AC

Motor AC terdiri dari kumparan stator dan rotor. Kumparan stator dan rotor tersusun dari parameter resistansi (R), reaktansi (jX) dan lilitan penguat (N) (Sofwan dan Baqo, 2004: 709).



Gambar 2.19. Rangkaian Ekuivalen Motor AC Sederhana

Sumber: Sofwan A dan Baqo S, Sistem Pengendalian Kecepatan Putaran Motor AC Phasa Satu Menggunakan Mikrokontroler AT89S8252, Motor AC.pdf



Gambar 2.20. Rangkaian Pengganti Motor AC

Sumber: Sofwan A dan Baqo S, Sistem Pengendalian Kecepatan Putaran Motor AC Phasa Satu Menggunakan Mikrokontroler AT89S8252, Motor AC.pdf

Nilai arus suber motor AC dapat dirumuskan sebagai berikut:

$$I_1 = I_0 + I_2 \quad (2.10)$$

Besarnya arus pemagnitan I_0 yang timbul akibat adanya induksi yang terjadi antara medan stator dan rotor adalah:

$$I_0 = I_r + I_m \quad (2.11)$$

Ggl yang dihasilkan akibat interaksi induksi medan magnet antara stator dan rotor yang masing-masing sebesar E_1 dan E_2 adalah:

$$E_1 = I_2' (R_s + jX_s) \quad (2.12)$$

$$E_2 = I_2 \left(\frac{R_r}{S} + jX_r \right) \quad (2.13)$$

Impedansi pada kumparan motor stator dan rotor masing-masing adalah:

$$jX_s = j\omega_s L_s \quad (2.14)$$

$$jX_r = j\omega_r L_r \quad (2.15)$$

(Sofwan dan Baqo, 2004: 709-710).

C. Prinsip Kerja Motor AC

Apabila kumparan-kumparan motor AC dialiri arus bolak-balik, maka pada celah udara akan dibangkitkan medan yang berputar dengan kecepatan putaran sebesar dengan menggunakan rumus (Sofwan dan Baqo, 2004: 710):

$$n_s = \frac{120 \cdot f}{p} \frac{\text{putaran}}{\text{menit}} \text{ [ppm]} \quad (2.16)$$

atau

$$\omega_s = \frac{2\pi f}{p} \quad (2.17)$$

Medan magnet berputar bergerak memotong lilitan rotor sehingga menginduksikan tegangan listrik pada kumparan-kumparan tersebut. Biasanya lilitan rotor berada dalam hubungan singkat, akibatnya lilitan rotor akan mengalir arus listrik yang besarnya tergantung pada besarnya tegangan induksi dan impedansi rotor. Arus listrik yang mengalir pada rotor akan mengakibatkan medan magnet rotor dengan kecepatan sama dengan kecepatan medan putar stator (n_s).

Interaksi medan stator dan rotor akan membangkitkan torsi yang menggerakkan rotor berputar searah dengan arah medan putar stator. Interaksi medan stator dan rotor juga menyebabkan terjadinya ggl induksi yang disebabkan oleh kumparan-kumparan stator dan rotor. Rumusan matematis gaya gerak listrik yang terjadi pada motor AC adalah sebagai berikut (Sofwan dan Baqo, 2004: 710):

$$\varepsilon = -N \frac{d\phi(t)}{dt} \quad (2.18)$$

Dimana nilai dari $\Phi(t)$ untuk fluksi maksimum akibat dari penyebaran kerapatan fluks yang melewati lilitan dengan rumus (Sofwan dan Baqo, 2004: 710):

$$\Phi(t) = \Phi_{\max} \cdot \cos\omega t \quad (2.19)$$

Adanya perbedaan medan putar stator dan medan putar rotor atau yang disebut slip pada motor induksi satu fasa pada rumus sebagai berikut (Sofwan dan Baqo, 2004: 710):

$$s = \frac{\omega_s - \omega_r}{\omega_s} \text{ atau } s = \frac{n_s - n_r}{n_s} \quad (2.20)$$

D. Hubungan Torsi dan Slip Pada Motor AC

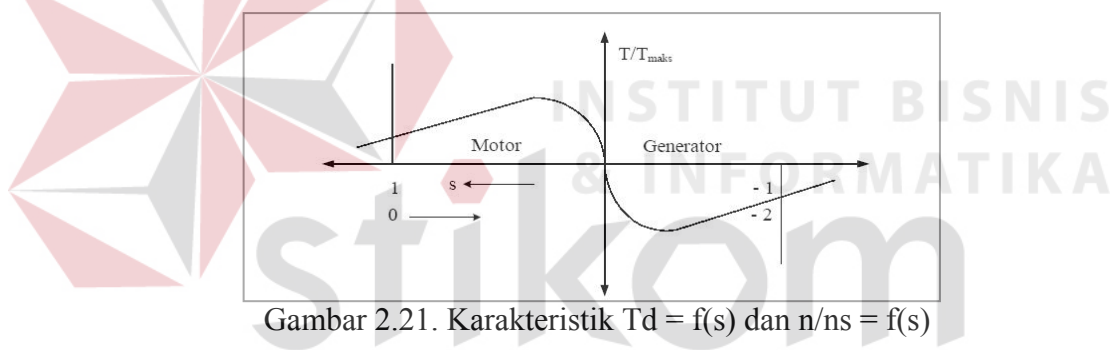
Berubah-ubahnya kecepatan motor induksi (n_s) akan mengakibatkan harga slip dari 100% pada saat start hingga 0% pada saat motor diam ($n_r - n_s$). torsi yang dihasilkan selama motor induksi satu fasa berputar tergantung pada perubahan slip dan perubahan dalam Newton.meter. Perubahan pembebanan dapat terjadi dengan naiknya nilai tegangan dan arus pada rotor. Hubungan torsi (T_d) terhadap parameter impedansi stator, impedansi rotor, arus rotor, tegangan sumber dan kecepatan sudut secara umum dapat dirumuskan sebagai berikut (Sofwan dan Baqo, 2004: 711):

Arus rotor (I_r) yang dihasilkan pada motor AC:

$$I_r = \frac{V_s}{\left[(R_s + R_r / S) + (X_s + X_r)^2 \right]^{1/2}} \quad (2.21)$$

$$I_r = \frac{R_r \cdot V_s^2}{s \omega_s \left[(R_s + R_r / S) + (X_s + X_r)^2 \right]^{3/2}} \quad (2.22)$$

Karakteristik torsi terhadap perubahan slip saat 100% pada saat start hingga 0% pada saat motor diam ($n_r = n_s$) pada motor induksi satu fasa dapat dilihat pada gambar dibawah ini (Sofwan dan Baqo, 2004: 711):



Gambar 2.21. Karakteristik $T_d = f(s)$ dan $n/n_s = f(s)$

Sumber: Sofwan A dan Baqo S, Sistem Pengendalian Kecepatan Putaran Motor AC Fasa Satu Menggunakan Mikrokontroler AT89S8252, Motor AC.pdf

2.5. Driver Motor

Driver motor merupakan sebuah rangkaian yang berfungsi untuk mengatur dan mengontrol sebuah aktuator (alat penggerak). Dengan rangkaian driver motor inilah sebuah aktuator (alat penggerak) dapat berhubungan dan dapat dikendalikan oleh mikrokontroler.

2.5.1. Driver Motor DC

Ada banyak macam komponen dan cara untuk membuat sebuah driver motor DC, seperti halnya kita bisa menggunakan relay yang diaktifkan dengan transistor sebagai saklar, namun yang demikian ini sangat menyulitkan dan tidak efisien dalam pengerjaan hardware-nya.

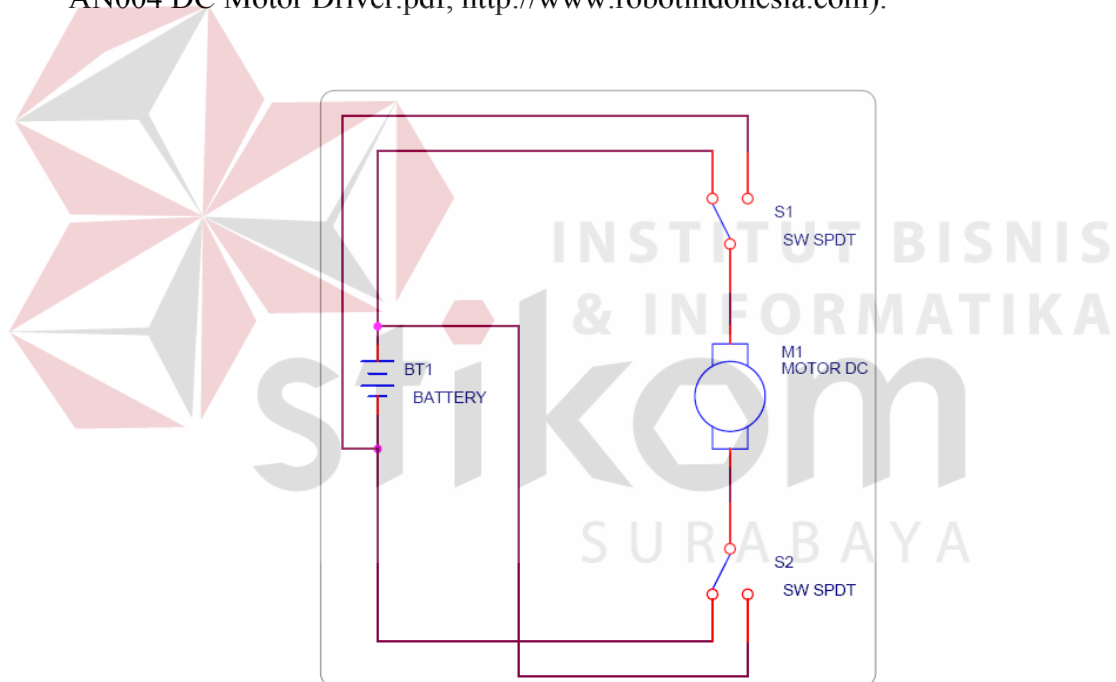
Dengan berkembangnya dunia digital saat ini terutama dibidang IC, sekarang sudah ada driver motor DC yang menggunakan teknologi H Bridge yang dikemas dalam suatu IC seperti IC L293 atau L298, dengan adanya IC ini kita dapat mengatur putaran motor secara dua arah sehingga akan memberikan kemudahan bagi kita untuk mengaplikasikannya kedalam suatu aplikasi pengendalian dan kontrol (<http://www.sunarto-ok.blogspot.com/2007/11/driver-motor-dc-dengan-l298-l293.html>).

A. IC L298 Driver Motor DC

IC L298 adalah salah satu jenis IC H Bridge yang memiliki kemampuan menggerakkan motor DC sampai arus 4A dengan tegangan mulai dari 5 Volt DC sampai dengan 46 Volt DC untuk satu kanalnya. Satu buah IC dapat digunakan untuk mengontrol sebanyak dua buah motor DC secara dua arah (<http://www.sunarto-ok.blogspot.com/2007/11/driver-motor-dc-dengan-l298-l293.html>).

B. H Bridge (Half Bridge)

H Bridge (Half Bridge) adalah sebuah rangkaian yang digunakan untuk mengendalikan sebuah motor DC sehingga dapat berputar searah ataupun berlawanan dengan arah jarum jam. Prinsip kerja H Bridge adalah mengatur aliran arus pada motor DC, apabila aliran arus dibalik maka motor DC akan berputar ke arah sebaliknya. Rangkaian H Bridge sederhana dapat dibangun oleh dua buah saklar SPDT seperti yang terlihat pada Gambar 2.22. berikut (Delta Robot, AN004 DC Motor Driver.pdf, <http://www.robotindonesia.com>).

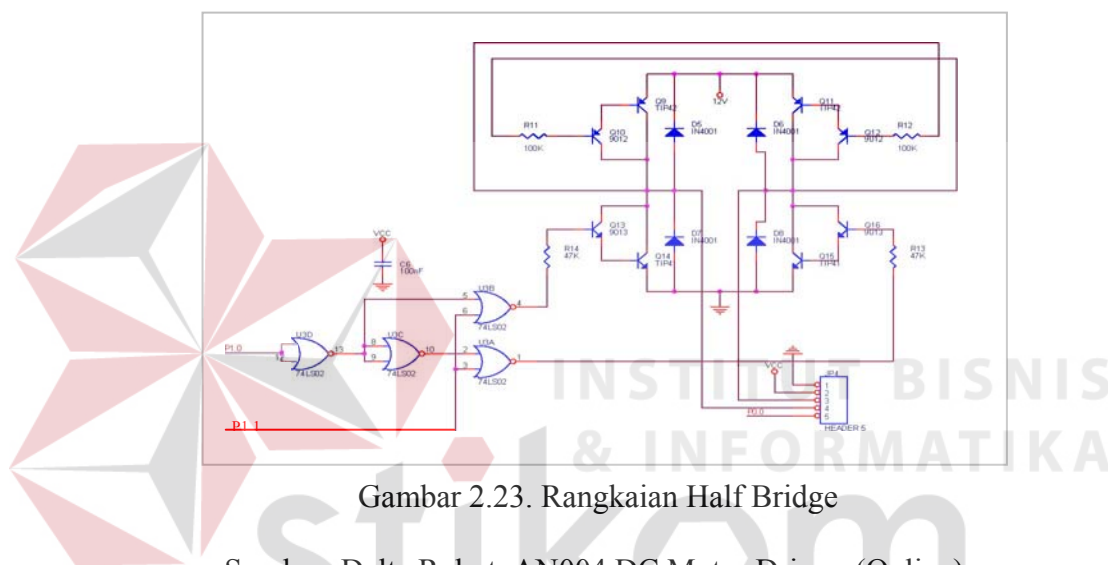


Gambar 2.22. Pengaturan Arah Motor DC

Sumber: Delta Robot, AN004 DC Motor Driver, (Online).
(<http://www.robotindonesia.com>, diakses 06 Juni 2008).

Saklar S1 menghubungkan kutub di bagian atas motor ke kutub positif sumber daya atau batrey, dan S2 menghubungkan kutub di bagian bawah motor ke kutub negatif sumber daya atau batrey sehingga arus pada motor mengalir dari atas ke bawah. Untuk mengubah putaran motor dengan melakukan perubahan arah

aliran arus dilakukan dengan memindahkan posisi S1 dan S2 bersamaan sehingga bagian atas motor terhubung pada kutub negatif batrey dan bagian bawah ke kutub positif batrey. Gambar 2.23. menunjukkan rangkaian Half Bridge yang lebih kompleks (Delta Robot, AN004 DC Motor Driver.pdf, <http://www.robotindonesia.com>).



Gambar 2.23. Rangkaian Half Bridge

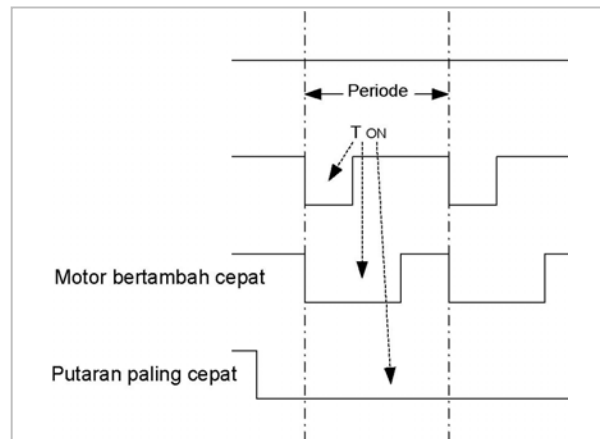
Sumber: Delta Robot, AN004 DC Motor Driver, (Online).
(<http://www.robotindonesia.com>, diakses 06 Juni 2008).

Saklar S2 digantikan dengan rangkaian transistor type NPN yaitu transistor TIP41 dan C9014 sedangkan saklar S1 digantikan dengan rangkaian transistor type PNP yaitu TIP42 dan C9012. Posisi saklar S1 diwakili dengan posisi transistor PNP mana yang aktif sedangkan posisi saklar S2 diwakili dengan posisi transistor NPN mana yang aktif (Delta Robot, AN004 DC Motor Driver.pdf, <http://www.robotindonesia.com>).

Apabila transistor PNP yang kiri aktif, maka transistor NPN kanan juga aktif sedangkan PNP kanan harus non aktif dan NPN kiri non aktif. Arus akan mengalir dari sumber daya positif ke kaki 3 JP4 yang terhubung pada kutub motor

dan diteruskan hingga kaki 4 JP4 ke sumber daya negatif atau ground. Sebaliknya pada saat transistor PNP kanan yang aktif, maka transistor NPN kiri juga aktif sedangkan PNP kiri harus non aktif dan NPN kanan juga non aktif. Arus akan mengalir dari sumber daya ke kaki 4 JP4 dan terus menuju ke sumber daya negatif melalui kaki 3. Dioda berfungsi untuk mencegah adanya tegangan reverse akibat induksi motor.

Pada rangkaian Half Bridge, hal yang tidak boleh terjadi adalah keempat bagian transistor yaitu NPN kiri, NPN kanan, PNP kiri dan PNP kanan aktif bersamaan. Hal ini akan menghubungkan singkatkan sumber daya positif dan negatif. Untuk mencegah kondisi ini, rangkaian gerbang logika yang dibentuk oleh IC 74LS02 diatur sehingga NPN kiri dan NPN kanan aktif bergantian. Hal ini ditentukan oleh kondisi logika pada P1.0 sebagai penentu arah gerakan motor. Sedangkan P1.1 berfungsi untuk mengatur apakah motor dalam keadaan aktif atau tidak. Bila kondisi logika P1.0 adalah logika 0, maka keluaran U3B akan berlogika 0 pula selama P1.1 aktif (berlogika 0). Hal ini akan mengakibatkan transistor NPN kiri non aktif. Sedangkan keluaran U3C akan berlogika 1 yang mengakibatkan keluaran U3A juga berlogika 1 dan transistor NPN kanan aktif. P1.1 juga berfungsi untuk mengatur kecepatan gerak motor. Dengan membangkitkan PWM pada P1.1 maka kecepatan gerak motor akan dapat diatur melalui bagian ini (Delta Robot, AN004 DC Motor Driver.pdf, <http://www.robotindonesia.com>).

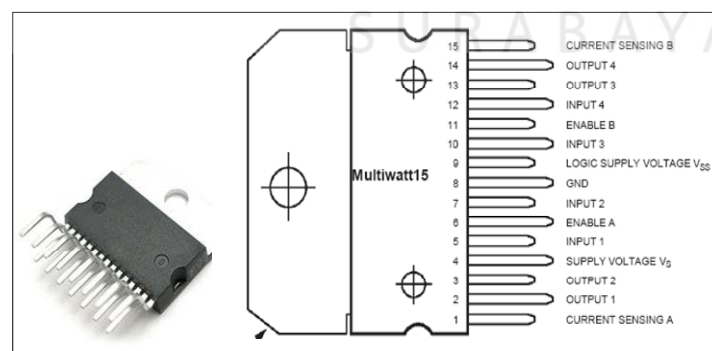


Gambar 2.24. Gambar Sinyal PWM

Sumber: Delta Robot, AN004 DC Motor Driver, (Online).
(<http://www.robotindonesia.com>, diakses 06 Juni 2008).

C. Fungsi Pin-pin L298

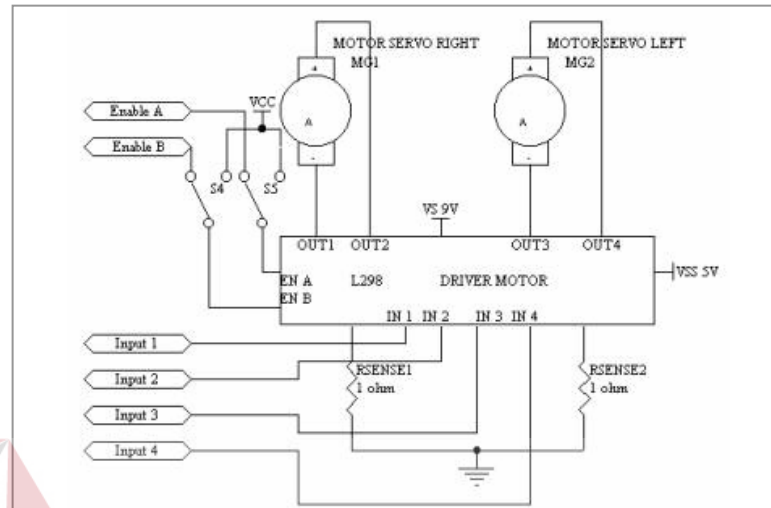
Jumlah *pin* pada L298 terdiri dari lima belas *pin*. Masing-masing dari *pin* tersebut mempunyai fungsi tersendiri. Gambar 2.25. Merupakan gambar *pin* dan fungsi *pin* dari IC driver motor DC L298 (STMicroelectronics GROUP OF COMPANIES, 2000: 2).



Gambar 2.25. Fungsi Pin L298

Sumber: STMicroelectronics GROUP OF COMPANIES, 2000. L298 Dual Full-Bridge Driver. (Online). (<http://www.st.com>, diakses 06 Juli 2008).

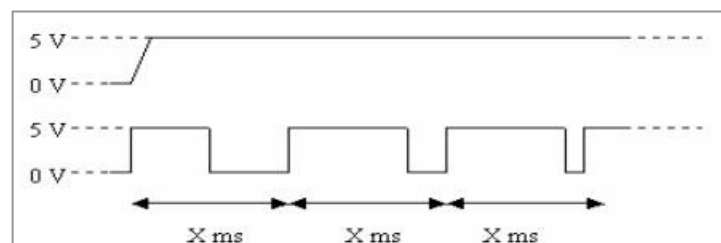
D. Rangkaian Driver Motor DC L298



Gambar 2.26. Rangkaian Driver Motor DC L298

Sumber: Susilo D. Robot Avoider. (Online).
(<http://www.electroniclab.com>, diakses 07 Juni 2008).

Pin Enable A dan *B* digunakan untuk mengendalikan jalan atau kecepatan motor, *pin Input 1* sampai 4 untuk mengendalikan arah putaran. *Pin Enable* diberi VCC 5 Volt untuk kecepatan penuh dan PWM (*Pulse Width Modulation*) untuk kecepatan rotasi yang bervariasi tergantung dari level highnya. Ilustrasinya ditunjukkan pada Gambar 2.27. berikut (Susilo D, 2008. <http://www.electroniclab.com>):



Gambar 2.27. Ilustrasi Pulse Width Modulation

Sumber: Susilo D. Robot Avoider. (Online).
(<http://www.electroniclab.com>, diakses 07 Juni 2008).

Jika dikehendaki kecepatan penuh maka diberikan 5 Volt konstan, jika dikehendaki kecepatan bervariasi maka diberikan pulsa yang lebar *high* dan *low*-nya bervariasi. Satu periode pulsa memiliki waktu yang sama sehingga dalam contoh diatas, kecepatan motor akan berubah dari setengah kecepatan penuh menjadi mendekati kecepatan penuh. Biasanya digunakan lebar pulsa dalam beberapa milisekon misalnya 2 ms. *Input* untuk motor servo kanan adalah *input* 1 (C) dan 2 (D), *direction*-nya dapat dilihat pada Tabel 2.10. berikut (Susilo D, 2008. <http://www.electroniclab.com>):

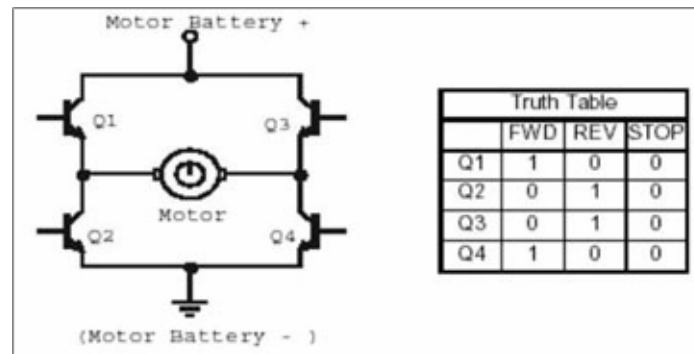
Tabel 2.10. Pengaturan IC Driver Motor

	Input	Function
Enable = H	Input 1 = H Input 2 = L	Maju
	Input 1 = L Input 2 = H	Mundur
	Input 1 = Input 2	Motor Berhenti Cepat
Enable = L	Input 1 = X Input 2 = X	Motor Bebas dan Berhenti

H = high L = low X = don't care

Sumber: Susilo D. Robot Avoider. (Online).
(<http://www.electroniclab.com>, diakses 07 Juni 2008).

Di dalam chip L298, untuk mengendalikan arah putaran motor digunakan metode H Bridge dari kombinasi transistor, jadi dengan metode demikian arus yang mengalir ke motor polaritasnya dapat diatur dengan memberikan logika ke transistor Q1 sampai Q4. Kondisi high untuk semua *input* tidak diijinkan sebab akan mengakibatkan semua transistor aktif dan akan merusakkan transistor karena secara otomatis arus dari kolektor Q1 dan Q2 langsung mengalir ke Q2 dan Q3 sehingga arus sangat besar tanpa melalui beban motor DC. Ilustrasi pengendalian motor oleh L298 dapat dilihat seperti Gambar 2.28. (Susilo D, 2008. <http://www.electroniclab.com>).



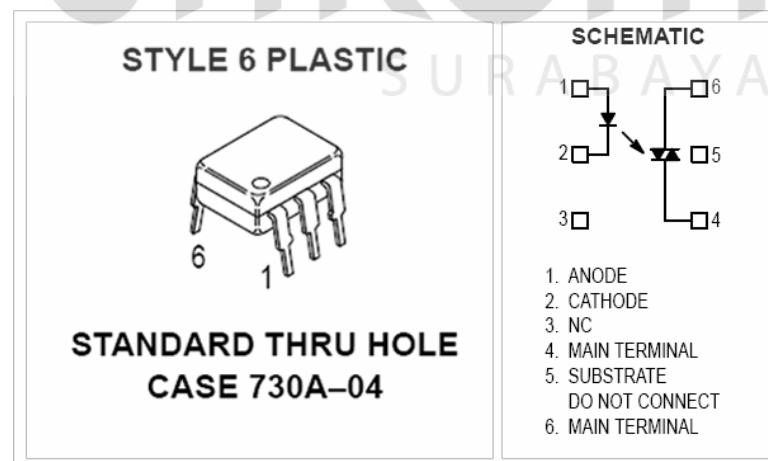
Gambar 2.28. Ilustrasi Pengendalian Motor di dalam IC L298

Sumber: Susilo D. Robot Avoider. (Online).
 (<http://www.electriniclab.com>, diakses 07 Juni 2008).

2.5.2. Driver Motor AC

A. Driver Motor AC (Triac MOC 3021)

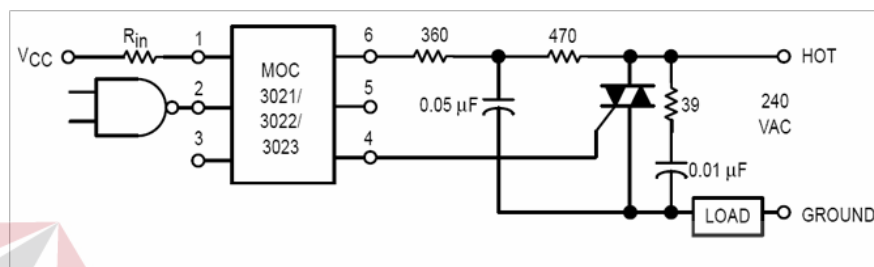
TRIAC merupakan saklar elektronik yang sangat ideal untuk mengatur daya arus bolak-balik. Kombinasi TRIAC dan mikrokontroler menghasilkan sistem pengaturan daya yang sangat fleksibel dan akurat (Motorola, Inc. 1995: 1).



Gambar 2.29. Skema IC MOC3021.

Sumber: Motorola, Inc, 1995. 6-Pin DIP Random-Phase Optoisolators Triac Driver Output MOC3021. (Online).
 (<http://www.Design-NET.com>, diakses 30 Mei 2008).

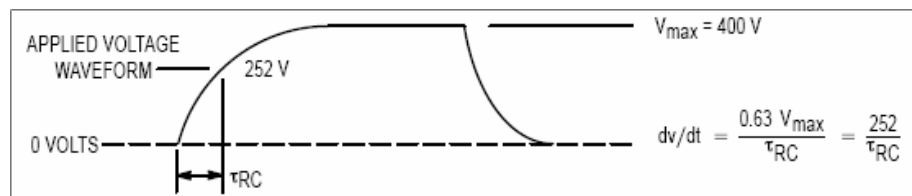
Rangkaian ini digunakan untuk mengontrol atau mengatur (mengaktifkan dan mematikan) motor AC yang menggerakkan mesin penggiling dan penyampur kacang dan bumbu. Driver triac ini terdiri dari IC MOC3021, capacitor dan triac. Seperti yang terlihat pada Gambar 2.30. berikut (Motorola, Inc. 1995: 4).



Gambar 2.30. Rangkaian Driver Triac MOC3021

Sumber: Motorola, Inc, 1995. 6-Pin DIP Random-Phase Optoisolators Triac Driver Output MOC3021. (Online).
(<http://www.Design-NET.com>, diakses 30 Mei 2008).

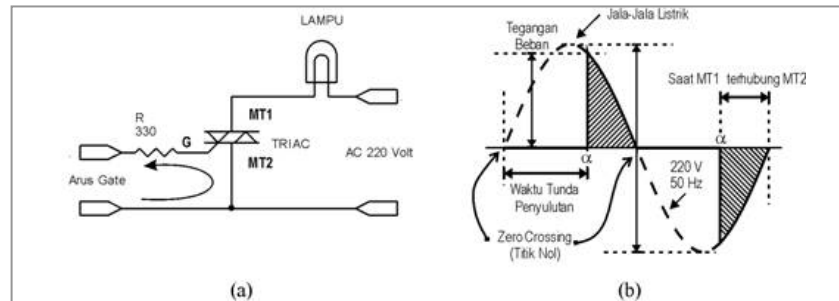
Tingkat yang kritis kenaikan off-state tegangan, dv/dt , terukur dengan masukan pada 0V. Frekwensi V_{in} ditingkatkan sampai phototriac menyala, frekwensi ini kemudian digunakan untuk menghitung dv/dt seperti pada contoh pengujian rangkaian dv/dt yang tidak berubah berikut ini (Motorola, Inc. 1995: 3):



Gambar 2.31. Pengujian Rangkaian untuk mencari dv/dt

Sumber: Motorola, Inc, 1995. 6-Pin DIP Random-Phase Optoisolators Triac Driver Output MOC3021. (Online).
(<http://www.Design-NET.com>, diakses 30 Mei 2008).

B. Cara Kerja Triac MOC 3021



Gambar 2.32. Dasar Pengaturan Daya dengan TRIAC

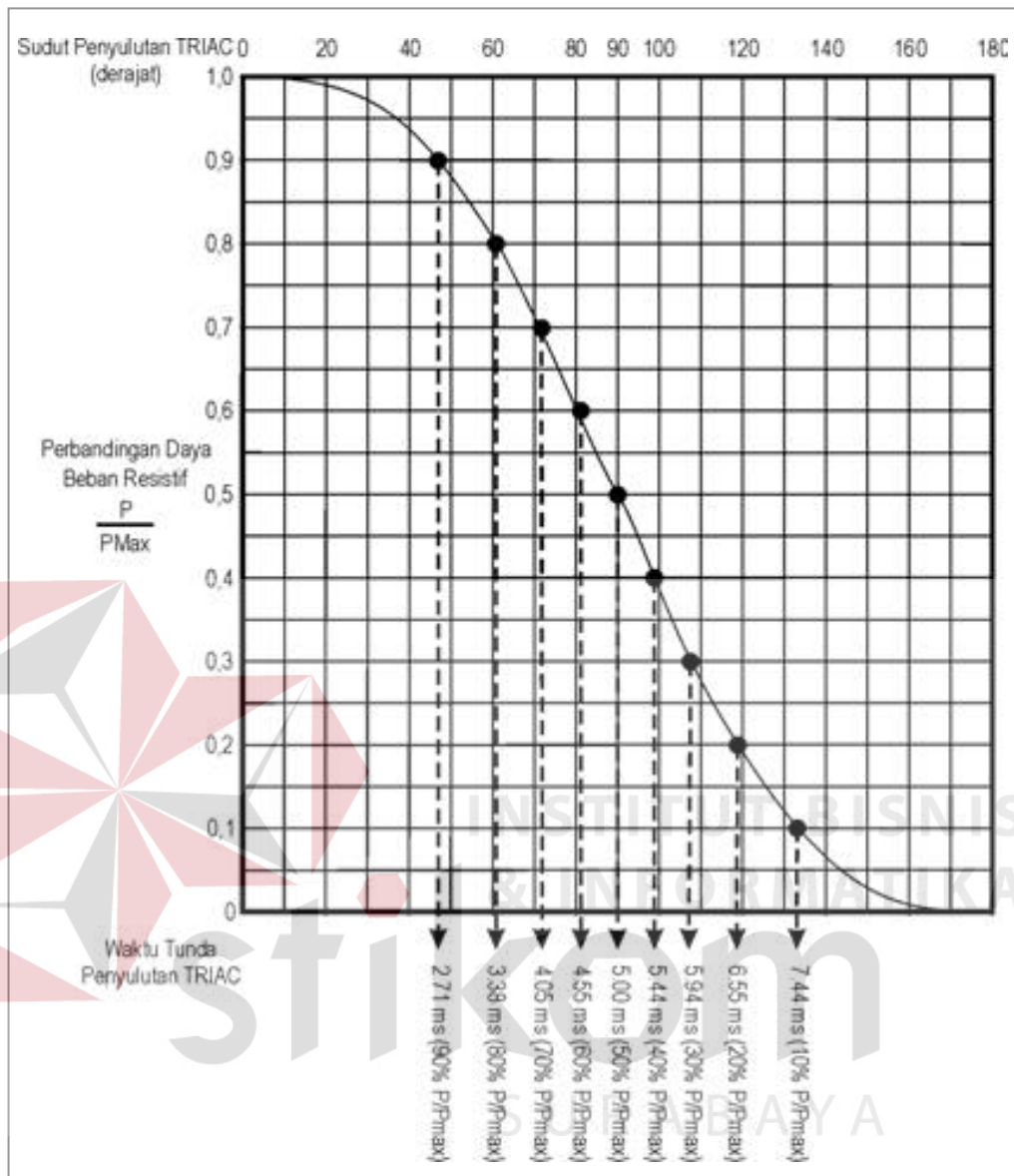
Sumber: Mengatur Daya Secara Phase Control dengan MCS51
(Phase Control.pdf).

Rangkaian dasar pemakaian TRIAC terlihat dalam Gambar 2.32. (a). Kaki MT1 dan MT2 merupakan saklar yang mengatur aliran arus beban yang berasal dari sumber tegangan bolak-balik (AC). Dalam keadaan normal kaki MT1 dan MT2 tidak terhubung, sehingga tidak ada arus beban yang mengalir. Saat ada arus gate mengalir, MT1 akan terhubung ke MT2 dan mengalirkan arus beban.

Arus Gate hanya diperlukan untuk menghubungkan MT1 dan MT2, setelah itu MT1 akan tetap terhubung ke MT2 meskipun sudah tidak ada arus gate lagi. Pemberian arus gate sesaat untuk menghubungkan MT1 dan MT2 dikatakan sebagai *menyulut* (men-*trigger*) TRIAC. MT1 terhubung terus ke MT2 selama arus beban yang mengalir lebih besar dari arus minimum (*holding current*) sesuai dengan karakteristik masing-masing TRIAC. Mengingat sumber daya yang dipakai berasal dari tegangan bolak-balik, pada daerah *titik nol* (*zero crossing*) dari tegangan bolak balik (lihat Gambar 2.32. (b)), arus beban yang mengalir akan mengecil sampai kurang dari arus minimum yang diperlukan, akibatnya hubungan antara MT1 dan MT2 akan terputus dengan sendirinya.

Daya yang disalurkan ke beban tergantung pada lamanya MT1 terhubung ke MT2 setiap setengah periode tegangan sinus dari jala-jala listrik, yakni bagian yang di-arsir dalam Gambar 2.32. (b), pada saat-saat itulah beban menerima daya. Dengan demikian, daya yang disalurkan ke beban bisa diatur dengan mengatur waktu tunda saat penyulutan TRIAC, terhitung mulai saat tegangan sinus jala-jala listrik mencapai titik nol. Teknik pengaturan daya semacam ini dikatakan sebagai teknik *phase control*.

Hubungan antara waktu tunda penyulutan dengan daya yang disalurkan (P) dibagi dengan daya maksimum (P_{max}). Dengan waktu tunda 0 mili-detik, P/P_{max} mencapai 1.0 artinya semua daya disalurkan ke beban. Untuk frekuensi jala-jala listrik 50 Hz, waktu tunda maksimum adalah 10 mili-detik, saat itu P/P_{max} bernilai 0 artinya tidak ada daya yang disalurkan. Hubungan waktu tunda dengan nilai P/P_{max} tidak linear, skala bagian bawah dari grafik Gambar 2.33. memperlihatkan nilai waktu tunda untuk memperoleh berbagai nilai P/P_{max} (Mengatur Daya Secara Phase Control dengan MCS51. Phase Control.pdf).



Gambar 2.33. Grafik Waktu Tunda Vs P/P_{max}

Sumber: Mengatur Daya Secara Phase Control dengan MCS51
(Phase Control.pdf)