BAB II
DASAR TEORI

2.1 Optocoupler

Optocoupler merupakan salah satu jenis komponen yang banyak dimanfaatkan oleh para penggemar *microcontroler*. Sensor yang digunakan pada kali ini yaitu sensor *optocoupler* sebagai sensor untuk menentukan seberapa besar kemiringan terhadap suatu benda. Benda yang diukur oleh sensor ini adalah ban sepeda motor.

Sensor *optocoupler* ini sangat mudah digunakannya hanya dengan memberikan sumber catudaya +5V dan ground. Pada saat sensor dipasang pada alat yang akan diukur, maka sensor akan memberikan suatu tegangan output sesuai dengan posisi kemiringan alat yang diukur.



Gambar 2.1 Optocoupler

2.2 Mikrokontroler AT89S51

Mikrokontroler sebagai salah satu terobosan teknologi *microprocessor* dan *microcomputer* hadir untuk memenuhi kebutuhan pasar. Sebagai salah satu teknologi baru, yaitu teknologi semikonduktor dengan kandungan transistor yang lebih banyak namun hanya membutuhkan ruang yang kecil. Tidak seperti sistem komputer yang mampu menangani berbagai macam aplikasi, mikrokontroler hanya dapat digunakan untuk aplikasi tertentu saja artinya hanya satu program yang dapat disimpan dalam *microcontroler* ini. *Microcontroler* telah dikombinasikan dengan *input*/*output* (I/O) dan memori sebagai komponen utamanya dalam bentuk *single chip*. *Microprocessor* telah dikombinasikan dengan *input*/ *output* (I/O) dan memori baik *RAM* maupun *ROM*

dengan kompleksitas tinggi dan mempunyai pemrosesan pusat (*Central Processing Unit)* sehingga disebut sebagai mikrokomputer.

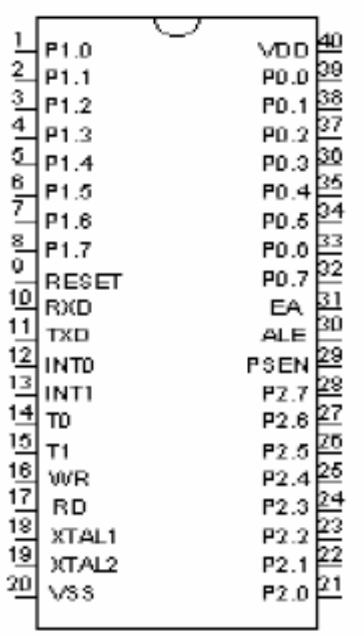
Microcontroler AT89S51 adalah *microcontroler* buatan ATMEL *Corporation. Mikrokontroler* ini termasuk kedalam bagian keluarga mikrokontroler MCS51, varian mikrokontroler 8051 yang memiliki ciri-ciri sebagai berikut:

- Kompatibel dengan produk MCS-51
- 4K byte In System Programmable Flas Memory
  Dapat dilakukan pemrograman 1000 tulis dan hapus
- Range catu daya 4,0V s/d 5,0V
- Operasi statis: 0 Hz s/d 33 MHz
- Tiga Tingkat Program memory lock
- 128 x 8 bit RAM internal
- 32 Programmable Jalur I/O
- Dua 16 bit Timer/ Counter
- Enam Sumber Interupsi
- Full Duplex Serial Channel
- Low Power Idle dan Mode Power Down
- Watcht Dog Timer
- Data Pointer
- Power Off Flag
- Fast Programming Time
- Fleksibel ISP programming

AT89S51 mempunyai konsumsi daya rendah, mikrokontroller 8-bit CMOS dengan 4K byte memori Flash ISP (in system programmable/ dapat diprogram didalam sistem).Divais ini dibuat dengan teknologi memori nonvolatile kerapatan tinggi dan kompatibel dengan standart industri 8051, set instruksi dan pin keluaran. Flash yang berada didalam chip memungkinkan memori program untuk diprogram ulang pada saat chip didalam sistem atau dengan menggunakan Programmer memori nonvolatile konvensional. Dengan mengkombinasikan CPU 8 bit yang serbaguna dengan flash ISP pada chip, ATMEL 89S51 merupakan *microcontroler* yang luarbiasa yang memberikan fleksibelitas yang tinggi dan penyelesaian biaya yang efektif untuk beberapa aplikasi kontrol., selain itu AT89S51 didesain dengan dan didukung dengan mode penghematan daya. Pada mode *idle* akan menghentikan CPU sementara RAM, timer/ counter, serial port dan sistem interupsi tetap berfungsi. Mode *Power Down* akan tetap menyimpan isi

dari RAM tetapi akan membekukan osilator, menggagalkan semua fungsi chip sampai interupsi eksternal atau reset hardware ditemui.

2.2.1 Fungsi Pin Mikrokontroler AT89S51



Gambar 2.2. Konfigurasi Pin Mikrokontroler AT89S51

1. **VCC** Tegangan Supply
2. **GND** Ground
3. **Port 0** Port 0, merupakan port I/O 8 bit open drain dua arah. Sebagai sebuah port, setiap pin dapat mengendalikan 8 input TTL. Ketika logika “1” dituliskan ke port 0, maka port dapat digunakan sebagai input dengan high impedansi. Port 0 dapat juga dikonfigurasikan untuk multipleksing dengan address/ data bus selama mengakses memori program atau data eksternal. Pada mode ini P0 harus mempunyai pull up
4. **Port 1** Port 1 merupakan port I/0 8 bit dua arah dengan internal pull up. Buffer output port 1 dapat mengendalikan empat TTL input. Ketika logika “1” dituliskan ke port 1, maka port ini akan mendapatkan internal pull up dan dapat digunakan sebagai input. Port 1 juga menerima alamat byte rendah selama pemrograman dan verifikasi Flash.

Tabel 2.1. Fungsi khusus dari port 1:

Port Pin	Fungsi Alternatif
P1.5	MOSI (digunakan untuk In System Programming)
P1.6	MISO (digunakan untuk In System Programming)
P1.7	SCK (digunakan untuk In System Programming)

5. Port 2 Port 2 merupakan port I/O 8 bit dua arah dengan internal pull up. Buffer output port 2 dapat mengendalikan empat TTL input. Ketika logika "1" dituliskan ke port 2, maka port ini akan mendapatkan internal pull up dan dapat digunakan sebagai input.

6. Port 3 Port 3 merupakan port I/O 8 bit dua arah dengan internal pull up. Buffer output port 3 dapat mengendalikan empat TTL input. Ketika logika "1"dituliskan ke port 3, maka port ini akan mendapatkan internal pull up dan dapat digunakan sebagai input.Port 3 juga melayani berbagai macam fitur khusus, sebagaimana yang ditunjukkan pada tabel berikut:

Tabel 2.2.Fungsi khusus dari port 3:

Port Pin	Fungsi Alternatif
P3.0	RXD (Port Serial Input)
P3.1	TXD (Port Serial Output)
P3.2	INT0 (Interupsi Eksternal 0)
P3.3	INT1 (Interupsi Eksternal 1)
P3.4	T0 (Input Eksternal Timer 0)
P3.5	T1 (Input Eksternal Timer 1)
P3.6	WR (Write Strobe Memori Data Eksternal)
P3.7	WR (Write Strobe Memori Program Eksternal)

7. RST Input Reset. Logika high "1" pada pin ini untuk dua siklus mesin sementara oscilator bekerja maka akan mereset devais.

8. ALE/ PROG Address Latch Enale (ALE) merupakan suatu pulsa output untuk mengunci byte low dari alamat selama mengakses memori eksternal. Pin ini juga merupakan input pulsa pemrograman selama pemrograman flash (paralel) Pada operasi normal, ALE

mengeluarkan suatu laju konstan 1/6 dari frekuensi oscilator dan dapat digunakan untuk pewaktu eksternal.

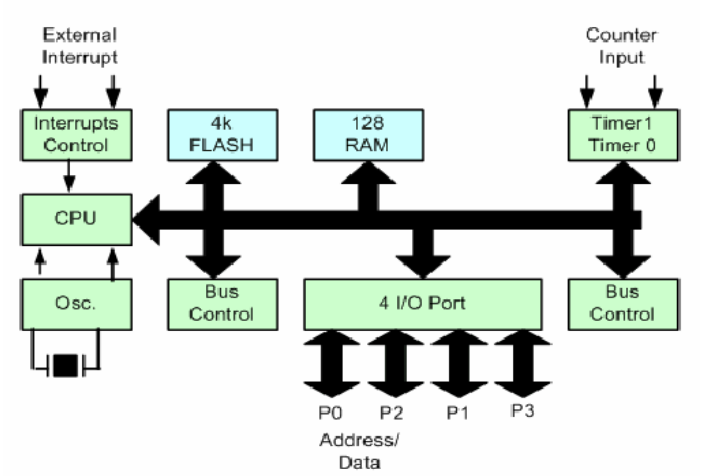
9.	**PSEN**	Program Store Enable merupakan strobe read untu memori Program eksternal.
10.	**EA/ VPP**	Eksternal Access Enable. EA harus di hubungkan ke GND untuk Enable devais, untuk memasuki memori program eksternal mulai alamat 0000H s/d FFFFH. EA harus dihubungkan ke VCC untuk akses memori program internal Pin ini juga menerima tegangan pemrogramman (VPP) selama pemrograman Flash
11.	**XTAL1**	Input untuk penguat oscilator inverting dan input untuk rangkaian internal clock
12.	**XTAL2**	Output dari penguat oscilator inverting.

2.2.2. Organisasi Memori

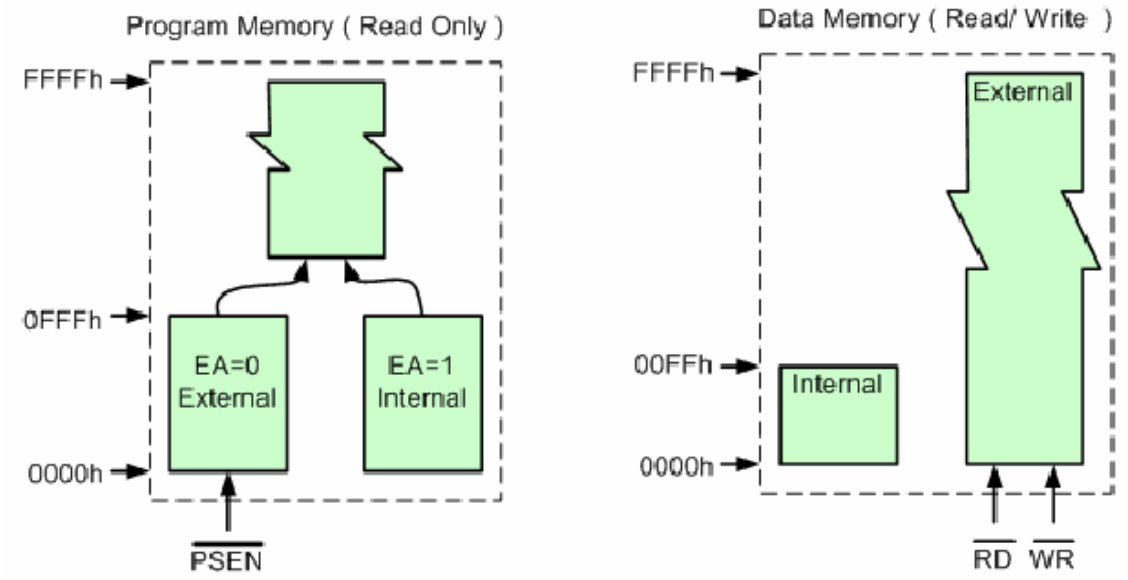
Semua *device* 8051 mempunyai ruang alamat yang terpisah untuk memori program dan memori data.Pemisahan secara logika dari memori program dan data, mengijinkan memori data untuk diakses dengan pengalamatan 8 bit, yang dengan cepat dapat disimpan dan dimanipulasi dengan CPU 8 bit. Selain itu, pengalamatan memori data 16 bit dapat juga dibangkitkan melalui register DPTR. Memori program (*ROM*, *EPROM* dan *FLASH*) hanya dapat dibaca, tidak ditulis. Memori program dapat mencapai sampai 64K byte. Pada 89S51, 4K byte memori program terdapat didalam chip. Untuk membaca memori program eksternal mikrokontroller mengirim sinyal PSEN (program store enable)

Memori data (RAM) menempati ruang alamat yang terpisah dari memori program. Pada keluarga 8051, 128 byte terendah dari memori data, berada didalam chip. RAM eksternal (maksimal 64K byte). Dalam pengaksesan RAM Eksternal, mikrokontroller mingirimkan sinyal RD (baca) dan WR (tulis).

Di bawah ini dapat dilihat gambar masing-masing perangkat keras (*hardware*) yang ada pada mikrokontroler MCS51. Didalamnya terdapat *CPU, Memory*, dan perangkat keras lain yang akan menjalankan instruksi sesuai keinginan *programmer*.



Gambar 2.3. Diagram blok mikrokontroller 8051



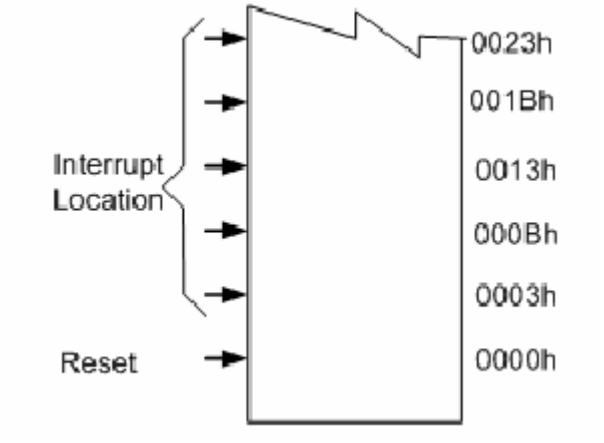
Gambar 2.4. Arsitektur Memori Mikrokontroller 8051

2.2.3. Program Memory

Pada gambar 2.4. menunjukkan suatu peta bagian bawah dari memori program. Setelah reset CPU mulai melakukan eksekusi dari lokasi 0000H. Sebagaimana yang ditunjukkan pada gambar 2.3. setiap interupsi ditempatkan pada suatu lokasi tertentu pada memori program. Interupsi menyebabkan CPU untuk melompat ke lokasi dimana harus dilakukan suatu layanan tertentu.Interupsi Eksternal 0, sebagi contoh, menempati lokasi 0003H. Jika Interupsi Eksternal 0 akan digunakan, maka layanan rutin harus dimulai pada

lokasi 0003H.Jika interupsi ini tidak digunakan, lokasi layanan ini dapat digunakan untuk berbagai keperluan sebagai Memori Program.

.

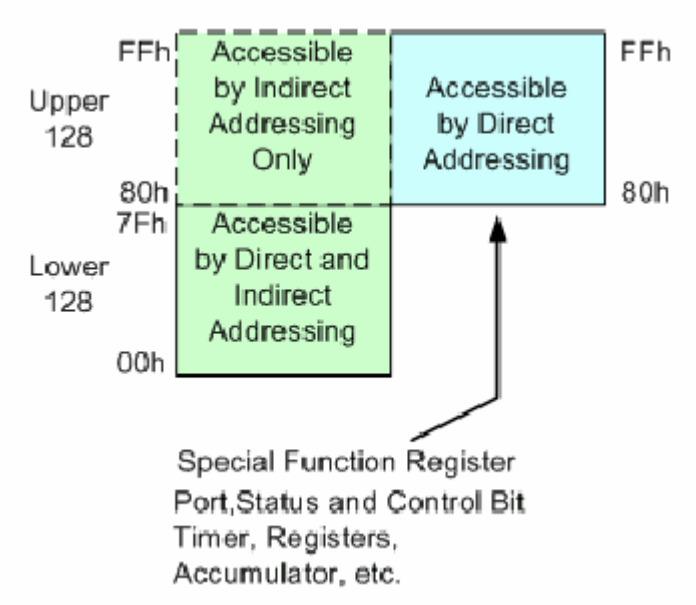


Gambar 2.5. Peta Interupsi mikrokontroller 8051

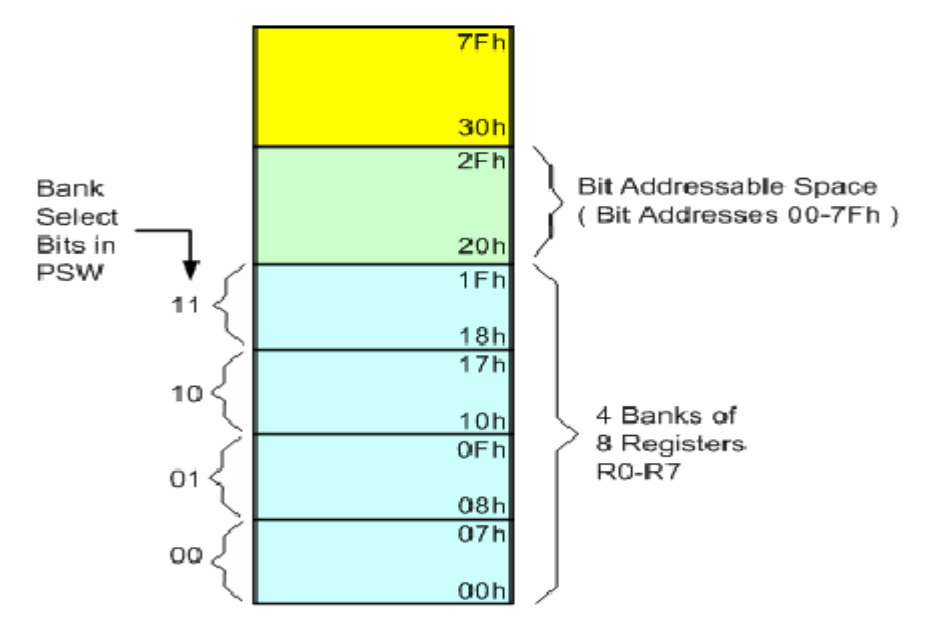
2.2.4. Memory Data

Memori data internal terpetakan seperti pada gambar 2.5. Ruang memori dibagi menjadi tiga blok, yang diacukan sebagai 128 byte lower, 128 byte upper dan ruang SFR.

Alamat memori data internal selalu mempunyai lebar data satu byte. Pengalamatan langsung diatas 7Fh akan mengakses satu alamat memori, dan pengalamatan tak langsung diatas 7Fh akan mengakses satu alamat yang berbeda. Demikianlah pada gambar 2.6 menunjukkan 128 byte bagian atas dan ruang SFR menempati blok alamat yang sama, yaitu 80h sampai dengan FFh, yang sebenarnya mereka terpisah secara fisik 128 byte RAM bagian bawah dikelompokkan lagi menjadi beberapa blok. 32 byte RAM paling bawah, dikelompokkan menjadi 4 bank yang masing-masing terdiri dari 8 register. Instruksi program untuk memanggil register-register ini dinamai sebagai R0 sampai dengan R7. Dua bit pada Program Status Word (PSW) dapat memilih register bank mana yang akan digunakan. Penggunaan register R0 sampai dengan R7 ini akan membuat pemrograman lebih efisien dan singkat, bila dibandingkan pengalamatan secara langsung.



Gambar 2.6. Memori data internal



Gambar 2.7. RAM internal 128 byte paling bawah

Semua pada lokasi RAM 128 byte paling bawah dapat diakses baik dengan menggunakan pengalamatan langsung dan tak langsung. 128 byte paling atas hanya dapat diakses dengan cara tak langsung.

2.2.5. Special Function Register

Sebuah peta memori yang disebut ruang *special function register* (SFR) ditunjukkan pada gambar 2.8. Perhatikan bahwa tidak semua alamat-alamat tersebut ditempati, dan alamat-alamat yang tak ditempati tidak diperkenankan untuk

diimplementasikan. Akses baca untuk alamat ini akan menghasilkan data random, dan akses tulis akan menghasilkan efek yang tak jelas.

Accumulator

ACC adalah register akumulator. Mnemonik untuk instruksi spesifik akumulator ini secara sederhana dapat disingkat sebagai A.

Register B

Register B digunakan pada saat opersi perkalian dan pembagian. Selain untuk keperluan tersebut diatas, register ini dapat digunakan untuk register bebas.

Program Status Word

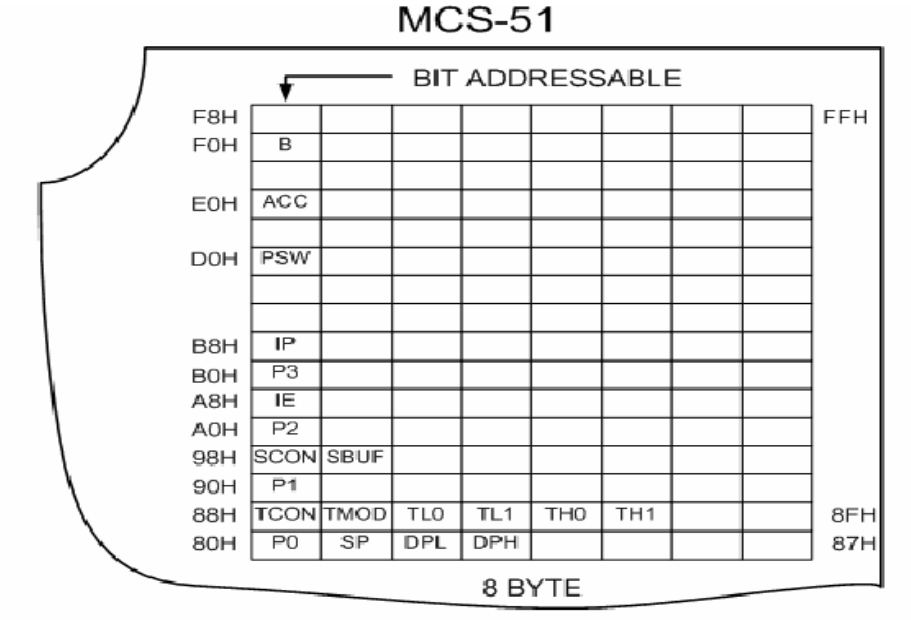
Register PSW terdiri dari informasi status dari program yang secara detail ditunjukkan pada Tabel 2.3.

Stack Pointer

Register Pointer stack mempunyai lebar data 8 bit. Register ini akan bertambah sebelum data disimpan selama eksekusi push dan call. Sementara stack dapat berada disembarang tempat RAM. Pointer stack diawali di alamat 07h setelah reset. Hal ini menyebabkan stack untuk memulai pada lokasi 08h.

Data Pointer

Pointer Data (DPTR) terdiri dari byte atas (DPH) dan byte bawah (DPL). Fungsi ini ditujukan untuk menyimpan data 16 bit. Dapat dimanipulasi sebagai register 16 bit atau dua 8 bit register yang berdiri sendiri.



Gambar 2.8. Peta SFR

Ports 0, 1, 2 dan 3

P0, P1, P2 dan P3 adalah SFR yang ditempati oleh Port 0, 1, 2 dan 3. Menulis suatu logika 1 terhadap sebuah bit dari sebuah port SFR (P0, P1, P2 atau P3) menyebabkan pin output port yang bersesesuaian akan berada dalam kondisi logika high ‘1’. Dan sebaliknya

Buffer Data Serial

Buffer serial sesungguhnya merupakan dua buah register yang terpisah, buffer pemancar dan buffer penerima. Ketika data diisikan ke SBUF, maka akan menuju ke buffer pemancar dan ditahan untuk proses transmisi. Ketika data diambil dari SBUF, maka akan berasal dari buffer penerima.

Registers Timer

Pasangan register (TH0, TL0) dan (TH1, TL1) adalah register pencacah 16 bit untuk Timer/ Counter 0 dan 1, masing-masing.

Register Control

Registers IP, IE, TMOD, TCON, SCON, dan PCON terdiri dari bit control dan status.

Program Status Word

PSW atau Program Status Word berisi bit-bit status yang berkaitan dengan kondisi atau keadaan CPU mikrokontroler pada saat tersebut. PSW berada dalam lokasi ruang SFR (perhatikan pada gambar 2.8. dengan lokasi alamat D0h). Pada PSW ini kita dapat memantau beberapa status yang meliputi: carry bit, auxiliary carry (untuk operasi BCD), dua bit pemilih bank register, flag overflow, sebuah bit paritas dan dua flag status yang bisa didifinisikan sendiri. Bit carry dapat juga anda guakan pada keperluan operasi aritmatika, juga bisa digunakan sebagai universal akumulator untuk beberapa operasi boolean.

Bit RS0 dan RS1 dapat digunakan untuk memilih satu dari empat bank register sebagaimana ditunjukkan pada tabel 2.3 Bit paritas dapat digunakan untuk mengetahuai jumlah logika '1' pada akumulator: P=1 bila pada akumulator mempunyai logika '1' yang jumlahn ya ganjil, dan P=0 jika akumulator mempunyai logika '1' yang jumlahnya genap. Dua bit yang lain PSW1 dan PSW5 dapat digunakan untuk berbagai macam tujuan

Tabel 2.3. Program Status Word

MSB							LSB
CY	AC	F0	RS1	RS0	OV	-	P

BIT	SYMBOL	FUNCTION
PSW.7	CY	Carry flag.
PSW.6	AC	Auxilliary Carry flag. (For BCD operations.)
PSW.5	F0	Flag 0. (Available to the user for general purposes.)
PSW.4	RS1	Register bank select control bit 1. Set/cleared by software to determine working register bank. (See Note.)
PSW.3	RS0	Register bank select control bit 0. Set/cleared by software todetermine working register bank. (See Note.)
PSW.2	OV	Overflow flag.
PSW.1	-	User-definable flag.
PSW.0	P	Parity flag. Set/cleared by hardware each instruction cycle to indicate an odd/even number of "one" bits in the Accumulator, i.e., even parity.

Tabel 2. 4. Alamat rekening bank

RS1	RS0	Bank	Address RAM
0	0	0	00 h - 07 h
0	1	1	08 h - 0F h
1	0	2	10 h - 17 h
1	1	3	18 h - 1F h

2.2.6. Sumber interupsi MCS51

Mikrokontroler AT89S51 mempunyai 2 buah *timer*, yaitu *timer* 0 dan *timer* 1, setiap *timer* terdiri dari 16 bit *timer* yang tersimpan dalam 2 buah register yaitu THx untuk *timer High Byte* dan TLx untuk *timer Low Byte* yang keduanya dapat berfungsi sebagai *counter* maupun sebagai *timer*. *Timer* mempunyai sumber *clock* dengan frekuensi tertentu yang sudah pasti sedangkan *counter* mendapat sumber *clock* dari pulsa yang hendak di hitung jumlahnya. Aplikasi dari *timer* atau pewaktu dapat digunakan untuk aplikasi mengitung lamanya suatu kejadian yang terjadi sedangkan *counter* atau penghitung biasa digunakan untuk aplikasi menghitung jumlah kejadian yang terjadi dalam periode tertentu. Register *THx* dan *TLx* diatur oleh register *TMOD* dan register *TCON*. *Timer* dapat diaktifkan melalui perangkat keras maupun perangkat lunak.

Periode waktu *timer/counter* dapat dirumuskan sebagai berikut:

1. Sebagai *timer/ counter* 8 bit

T = (255 – TLx) * 1 µs

TLx merupakan isi register TL0 atau TL1

2. Sebagai *timer/ counter* 16 bit

T = (65535 – THx TLx) * 1 µs

THx merupakan isi register TH0 atau TH1 dan TLx merupakan isi register TL0 atau TL1

Register Timer				**Register Interupsi**			
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0
TCON 7	TCON 6	TCON 5	TCON 4	TCON 3	TCON2	TCON 1	TCON 0

(a)

Timer 1				**Timer 0**			
GATE	C/T	M1	M0	GATE	C/T	M1	M0

(b)

Gambar 2.9 (a) Register TCON, (b) TMOD

Pengontrolan kerja *timer/ counter* di atur oleh register TCON. Register ini bersifat *bit addressable* sehingga bit TF1 dapat disebut TCON.7 dan seterusnya hingga bit IT0 sebagai TCON 0. Register ini hanya mempunyai 4 bit saja yang berhubungan dengan *timer* seperti akan dijelaskan pada tabel 2.5.

.

Tabel 2.5 Fungsi Bit Register TCON yang Berhubungan dengan Timer

Nama	**Bit Fungsi**
TF1	*Timer* 1 *Overflow Flag* yang akan diset jika *timer overflow*
TR1	Membuat *timer* 1 aktif (*set*) dan non aktif (*clear*)
TR0	*Timer* 0 *Overflow Flag* yang akan di-*set* jika *timer overflow*
TF0	Membuat *timer* 0 aktif *(set*)dan nonaktif (*clear*)

Register TMOD berfungsi untuk pemilihan mode operasi *timer/ counter* dengan fungsi tiap bitnya adalah sebagai berikut:

- Gate berfungsi pada saat TRx =1, *timer* akan berjalan tanpa memperhatikan nilai pada *Gate* (*timer* di kontrol *software*)
- C/T berfungsi sebagai pemilihan fungsi *timer* (0) atau *counter* (1).
- M1 & M0 berfungsi untuk memilih mode *timer* dengan variasi

Tabel 2.6 Mode Timer

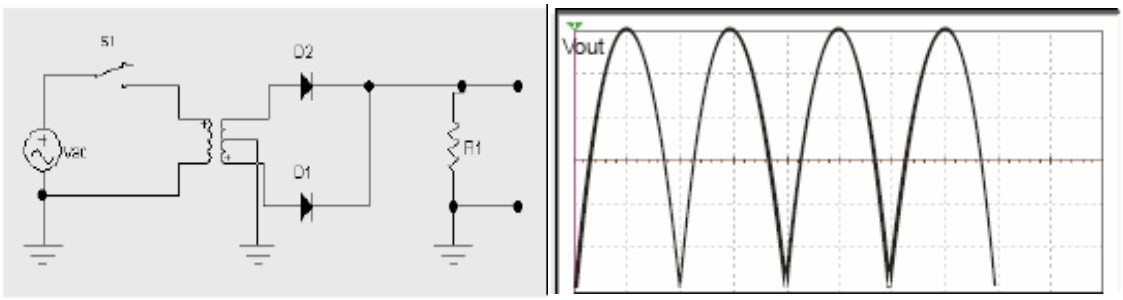
M1	M0	Mode	Operasi
0	0	0	*Timer* 13 bit
0	1	1	*Timer* / *counter* 16 bit
1	0	2	Register THx berisi nilai isi ulang yang akan dikirim ke TLx setiap overflow
1	1	3	Pada mode ini, AT89C51 seperti memiliki 3 buah *timer*. *Timer* 0 terpisah menjadi 2 buah *timer* 8 bit (TL0 – TF0 dan TH0 –TF1) dan *timer* 1 tetap 16 bit

2.3 Catudaya

Catudaya merupakan hal yang paling penting dalam setiap pembuatan suatu perangkat keras.kita perhatikan berapa tegangan yang diperlukan untuk tiap-tiap perangkat yang akan digunakan. Penyearahan adalah proses menyearahkan arus bolak-balik menjadi arus searah. Komponen utama yang diperlukan dalam penyearahan adalah transformator (Trafo), dioda dan kapasitor elektrolit.

2.3.1 Penyearah Gelombang Penuh Sistem CT

Penyearah Gelombang Penuh Sistem CT menggunakan transformator jenis CT dengan dua buah dioda sebagai penyearah. Dioda secara bergantian untuk mengambil *pulsa positif* dan *negatif* , sehingga keluaran berupa gelombang penuh

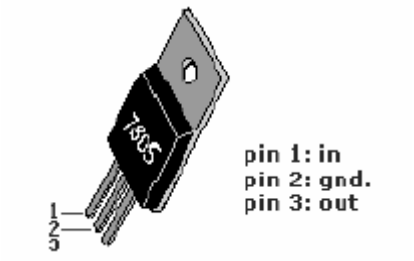


Gambar 2.10 Gelombang Penuh Sistem CT

2.4 LM 7805

IC ini mempunyai tiga kaki yang digunakan sebagai komponen pendukung dari Vcc untuk menghasilkan tegangan 5V. IC *regulator* ini berfungsi untuk menstabilkan tegangan 5V dan dapat bekerja dengan baik jika tegangan *input* (Vin) lebih besar dari pada tegangan *output* (Vout). Biasanya perbedaan tegangan *input* dengan *output* yang

direkomendasikan tertera pada *datasheet* komponen tersebut. Berikut ini Contoh IC LM7805 yang diperlihatkan pada gambar 2.11.



Gambar 2.11 IC LM7805

2.5 Resistor

Resistor merupakan komponen dasar elektronika yang berfungsi untuk membatasi arus pada rangkaian, bila tahanannya semakin besar maka semakin kecil pula arus dari keluaran tersebut, sebaliknya jika nilai hambatannya kecil maka akan besar pula arusnya. Resistor bersifat resistif dan umumnya terbuat dari bahan karbon. Dari hukum *Ohm* diketahui, resistansi berbanding terbalik dengan jumlah arus yang mengalir melaluinya. Satuan resistansi dari suatu resistor disebut *Ohm* (Ω). Tipe resistor yang sekarang banyak digunakan yaitu jenis resistor karbon dan metal film. Untuk mengetahui nilai resistansi dari suatu resistor caranya adalah dengan membaca warna gelang dari resistor atau membaca suatu nilai yang tertera pada badan resistor. Berikut ini Contoh Resistor yang diperlihatkan pada gambar 2.12.



Gambar 2.12 Resistor

2.6 Kapasitor

Kapasitor merupakan komponen pasif elektronika yang sering dipakai didalam merancang suatu sistem yang berfungsi untuk mengeblok arus DC, Filter, dan penyimpan energi listrik. Didalamnya 2 buah pelat elektroda yang saling berhadapan dan dipisahkan oleh sebuah *insulator*. Sedangkan bahan yang digunakan sebagai *insulator* dinamakan dielektrik. Ketika kapasitor diberikan tegangan DC maka energi

listrik disimpan pada tiap elektrodanya. Selama kapasitor melakukan pengisian, arus mengalir. Aliran arus tersebut akan berhenti bila kapasitor telah penuh. Yang membedakan tiap - tiap kapasitor adalah dielektriknya. Berikut ini salahsatu Contoh Kapasitor yang diperlihatkan pada gambar 2.13.



Gambar 2.13 Electrolytic Capacitor (Elco)

Kapasitor ELCO mempunyai kutub atau polaritas yaitu tanda positif dan tanda negatif. Jika dalam pemasangan kutub-kutub ELCO terbalik maka kapasitor akan rusak.

2.7 Dioda

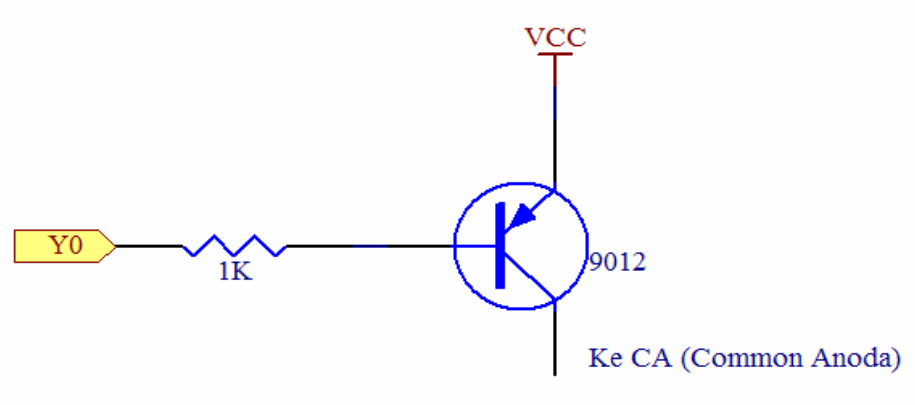
Dioda adalah suatu komponen elektronik yang dapat melewatkan arus pada satu arah saja. Ada berbagai macam dioda diantaranya dioda tabung, dioda sambungan P-N dan dioda kontak titik dan sebagainya. Fungsi dioda sangat berperan penting pada rangkaian elektronika, diantaranya mempunyai fungsi untuk menghasilkan tegangan searah dari tegangan bolak-balik. Bentuk dioda yang sering digunakan pada saat sekarang ini terdiri dari semikonduktor jenis P yang dibuat bersambung dengan semikonduktor jenis N.



Gambar 2.14 Simbol Dioda

2.8 Transistor

Seperti kita ketahui bahwa transistor bisa berfungsi sebagai saklar. Transistor yang digunakan pada rangkaian ini adalah *type* PNP



Gambar 2.15 Rangkaian skema dasar konfigurasi saklar menggunakan transistor

Sesuai dengan rangkaian pada gambar 2.15, agar ada arus yang mengalir dari Vcc maka pada Y0 harus diberi logika 0 sehingga transistor menjadi On (kondisi jenuh) dan mengalirkan arus dari Vcc ke CA, karena sevensegment yang digunakannya menggunakan konfigurasi sevensegment CA (Common Anode)

2.9 Bahasa Pemrograman

Bahasa assembly adalah suatu bentuk bahasa mesin yang mudah dimengerti oleh manusia, sedangkan bahasa mesin dimengerti oleh mikrokontroller. Assembly adalah program komputer yang mentranslitrasi program dari bahasa assembly ke bahasa mesin, Sedangkan bahasa *assembly* adalah ekuivalensi bahasa mesin dalam bentuk alpanumerik. *Mnemonics* alpanumerik digunakan sebagai alat bantu bagi programmer untuk memprogram mesin komputer daripada menggunakan serangkaian 0 dan 1 (bahasa mesin) yang panjang dan rumit. Bahasa assembly menghasilkan program yang kecil dan cepat.

2.9.1 Instruksi Bahasa Assembly

Ada beberapa perangkat lunak (software) yang dapat digunakan sebagai program bantu untuk membuat dang mensimulasikan program, diantaranya AVSIM51, ALDS dan Pinnacle. Pinnacle memiliki beberapa fitur yang cukup lengkap seperti : Register Data Pointer (DPTR), Ports, Timer, Internal RAM, External RAM dan lain-lain. Program sumber (source code) dibuat dengan pinnacle 52, hasil kerja program yang telah dikompile dalam bahasa assembler ini adalah assembly listing. Instruksi yang digunakan dalam penyusunan program adalah sebagai berikut :

EQU

Digunakan untuk mendefinisikan sebuah lambang assembler secara bebas.

ORG (Origin)

Digunakan untuk mengarahkan lokasi tempat instruksi yang ada dibawahnya

CLR

Memberikan nilai “0” pada bit tertentu

SETB

Memberikan nilai ”1” pada bit tertentu

CALL

Merupakan instruksi untuk melakukan lompatan dengan area sebesar 2 KB

LCALL

Instruksi ini hampir sama dengan instruksi CALL, hanya saja instruksi ini digunakan jika label yang dipanggil letaknya lebih jauh dari 2 KB

MOV

Intruksi ini berfungsi melakukan pemindahan data dari variabel pada kode operasi kedua dan disimpan di variabel pada kode operasi pertama.

MOVC

Intruksi ini digunakan untuk memori program. Tanda ‘@’ digunakan untuk menandai A+DPTR yang berfungsi untuk menyatakan lokasi memori yang isinya disalin ke akumulator A, dalam hal ini nilai tersimpan dalam DPTR (Data Pointer Register) ditambah dengan nilai yang tersimpan dalam akumulator A sama dengan lokasi memori program yang diakses.

JMP

Digunakan untuk melakukan lompatan dan menjalankan program yang berada dialamat yang ditentukan oleh label tertentu.

SJMP

Digunakan untuk melakukan lompatan untuk jarak yang pendek (Short JUMP)

LJMP

Instruksi ini berfungsi untuk melakukan lompatan untuk jarak yang jauh (Long Jump)

JB

Lompat jika Bit tertentu bernilai 1, artinya jika bit yang telah ditentukan bernilai 1, maka lompat ke alamat yang telah ditentukan dan dilanjutkan ke instruksi berikutnya.

JNB

Kebalikan dari JB, JNB merupakan instruksi untuk beralih ke alamat tertentu jika Bit tertentu bernilai 0..

DJNZ

Instruksi ini berfungsi melakukan pengurangan pada Rn (R0…….R7) dengan 1 dan lompat ke alamat yang ditentukan apabila hasilnya bukan 00. apabila hasilnya telah mencapai 00, maka program akan terus menjalankan instruksi dibawahnya.

CJNE

Instruksi ini berfungsi melakukan perbandingan antara data tujuan dan data sumber serta melakukan lompatan ke alamat yang ditentukan jika hasil perbandingan tidak sama.

RET

Digunakan untuk melakukan lompatan ke alamat yang disimpan dalam SP dan SP-1. Instruksi ini biasa digunakan pada saat kembali dari subrutin yang dipanggil dengan instruksi ACALL atau LCALL.

END

Instruksi ini biasanya diletakan diakhir baris dari file program sumber assembler sebagai tanda akhir pernyataan (statement) bagi program assembler.