

2. TEORI DASAR

2.1. Mikrokontroler AT89S51

Sebelum pembahasan lebih lanjut ada baiknya kita mengetahui perbedaan mendasar antara mikroprosesor dan mikrokontroler. Suatu mikroprosesor adalah bagian dari CPU sebuah *computer*, tanpa *memory*, I/O, dan peripheral yang dibutuhkan oleh suatu sistem yang lengkap. Dan untuk dapat bekerja, mikroprosesor membutuhkan perangkat pendukung yang dapat berupa RAM, ROM dan I/O. Sedangkan mikrokontroler adalah kombinasi dari sebuah mikroprosesor dengan I/O, RAM dan ROM dalam suatu *chip* tunggal, sehingga dalam penggunaannya mikrokontroler mempunyai suatu kelebihan integrasi dibandingkan dengan mikroprosesor.

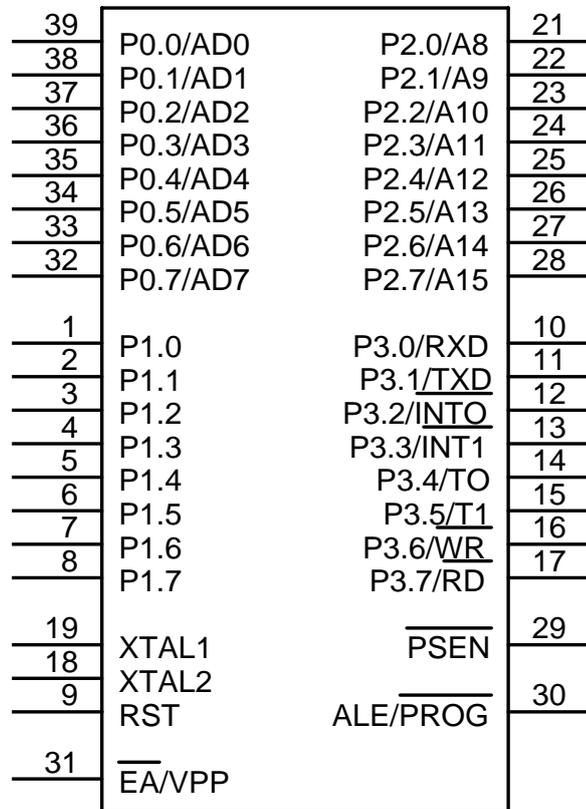
Mikrokontroler adalah mikroprosesor yang dilengkapi dengan fasilitas I/O dan memori (RAM/ROM) dan dikemas dalam suatu *chip* tunggal. Mikrokontroler AT89S51 merupakan bagian dari keluarga mikrokontroler MCS-51 yaitu termasuk jenis 8051. Keluarga ini diawali oleh Intel yang mengenalkan mikrokontroler tipe 8051 pada awal tahun 1980-an.

Spesifikasi secara umum mikrokontroler AT89S51 adalah sebagai berikut :

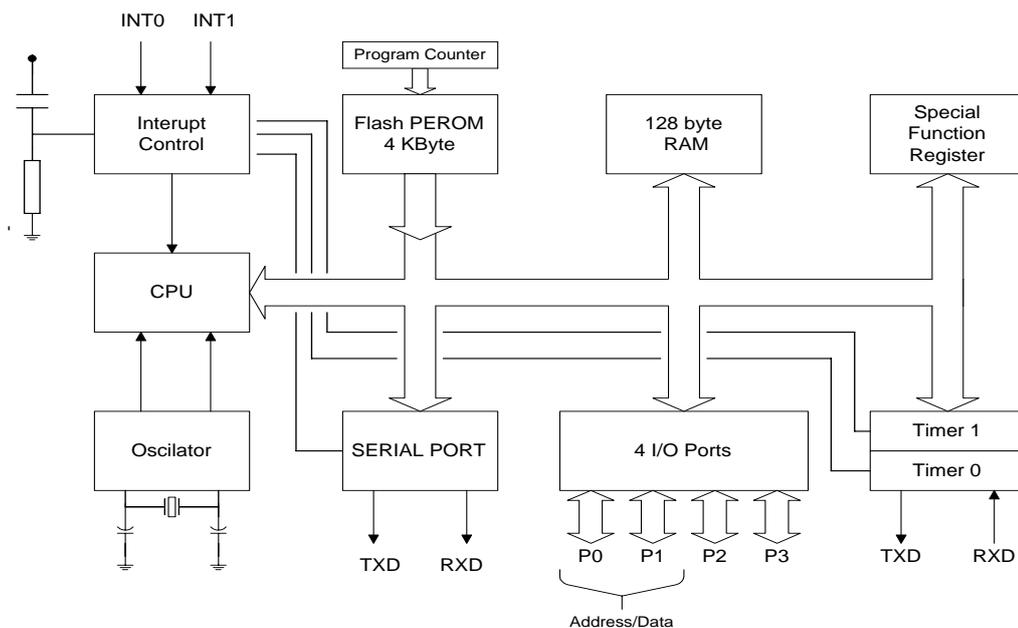
- 4K bytes *Flash Programmable and Erasable Read Only Memory (Flash PEROM)* yang dapat dihapus dan ditulis sampai 1000 kali.
- Kompatibel dengan standar industri MCS-51.
- 1 buah jalur serial *input/output*.
- 2 buah *counter/timer* 16-bit.
- 128 × 8-bit internal RAM.
- 32 jalur *input/output*.
- 256 set instruksi.
- 5 sumber pembangkit interupsi.

2.1.1. Konfigurasi dan Deskripsi AT89S51

Konfigurasi pin mikrokontroler AT89S51 secara umum dapat dilihat pada gambar 2.1 berikut ini :



Gambar 2.1. Konfigurasi pin AT89S51



Gambar 2.2. Konstruksi dasar AT89S51

Berikut ini adalah penjelasan mengenai fungsi dari masing-masing pin mikrokontroler AT89S51 :

- Vcc : tegangan *supply*.
- Gnd : *ground*.
- Port 0 : merupakan *port* I/O bertipe *open drain bidirectional*. Sebagai *port* keluaran, masing-masing kaki dapat mengaktifkan delapan masukan TTL. Pada saat *port* 0 dalam keadaan *high*, dapat digunakan sebagai *input* berimpedansi tinggi. Port 0 juga dapat dikonfigurasi sebagai jalur alamat (A0-A7) atau jalur data (D0-D7) selama pengaksesan memori data atau memori program eksternal.
- Port 1 : merupakan *port* I/O dua arah yang dilengkapi dengan *pullup* internal. *Output buffer* dari port 1 dapat mengaktifkan 4 masukan TTL. Pada saat *port* 1 dalam keadaan *high*, dapat digunakan sebagai *input*.
- Port 2 : merupakan *port* I/O dua arah yang dilengkapi dengan *pullup* internal. *Output buffer* dari port 2 dapat mengaktifkan 4 masukan TTL. Pada saat *port* 2 dalam keadaan *high*, dapat digunakan sebagai *input*. Port 2 juga dapat digunakan sebagai jalur alamat (A8-A15) selama pengambilan instruksi dari memori program eksternal atau selama pengaksesan memori data eksternal yang menggunakan perintah dengan alamat 16-bit.
- Port 3 : merupakan *port* I/O dua arah yang dilengkapi dengan *pullup* internal. *Output buffer* dari port 3 dapat mengaktifkan 4 masukan TTL. Pada saat *port* 3 dalam keadaan *high*, dapat digunakan sebagai *input*. Port 3 juga menyediakan berbagai fungsi khusus, sebagaimana diberikan dalam tabel 2.1.

Tabel 2.1. Fungsi Khusus Masing-Masing Kaki Port 3

Kaki Port	Fungsi Khusus
P3.0	RXD (<i>port</i> masukan serial)
P3.1	TXD (<i>port</i> keluaran serial)
P3.2	$\overline{\text{INT0}}$ (interupsi eksternal 0)
P3.3	$\overline{\text{INT1}}$ (interupsi eksternal 1)

P3.4	T0 (masukan eksternal pewaktu/pencacah 0)
P3.5	T1 (masukan eksternal pewaktu/pencacah 1)
P3.6	$\overline{\text{WR}}$ (sinyal tanda tulis memori data eksternal)
P3.7	$\overline{\text{RD}}$ (sinyal tanda baca memori data eksternal)

- RST : *reset*. Kondisi high selama 2 siklus mesin akan mereset mikrokontroler.
- $\text{ALE}/\overline{\text{PROG}}$: keluaran ALE (*Address Latch Enable*) akan menghasilkan pulsa-pulsa untuk mengunci alamat *byte* rendah (A0-A7) selama mengakses memori eksternal. Pada saat pemrograman *flash* berfungsi sebagai *pulse input*. Pada operasi normal, ALE akan mengeluarkan pulsa sebesar 1/6 frekuensi *oscillator*.
- $\overline{\text{PSEN}}$ (*Program Strobe Enable*) : merupakan sinyal baca untuk memori program eksternal. Sinyal $\overline{\text{PSEN}}$ tidak aktif untuk pengambilan program pada memori program internal.
- $\overline{\text{EA}}/\text{VPP}$: jika mikrokontroler akan mengeksekusi program dari memori eksternal lokasi 0000H hingga FFFFH maka $\overline{\text{EA}}$ (*External Access Enable*) harus dihubungkan ke *ground*. Apabila pin $\overline{\text{EA}}$ dihubungkan dengan V_{CC} , maka mikrokontroler akan mengeksekusi program pada alamat 0000H sampai 0FFFH pada memori program internal dan alamat 1000H sampai FFFFH pada memori program eksternal. Kaki ini juga berfungsi menerima tegangan 12 Volt selama pemrograman flash.
- XTAL1 : merupakan *input* untuk *inverting oscillator amplifier* dan sebagai *input* pada *internal clock operationing circuit*.
- XTAL2 : merupakan *output* dari *inverting oscillator amplifier*.

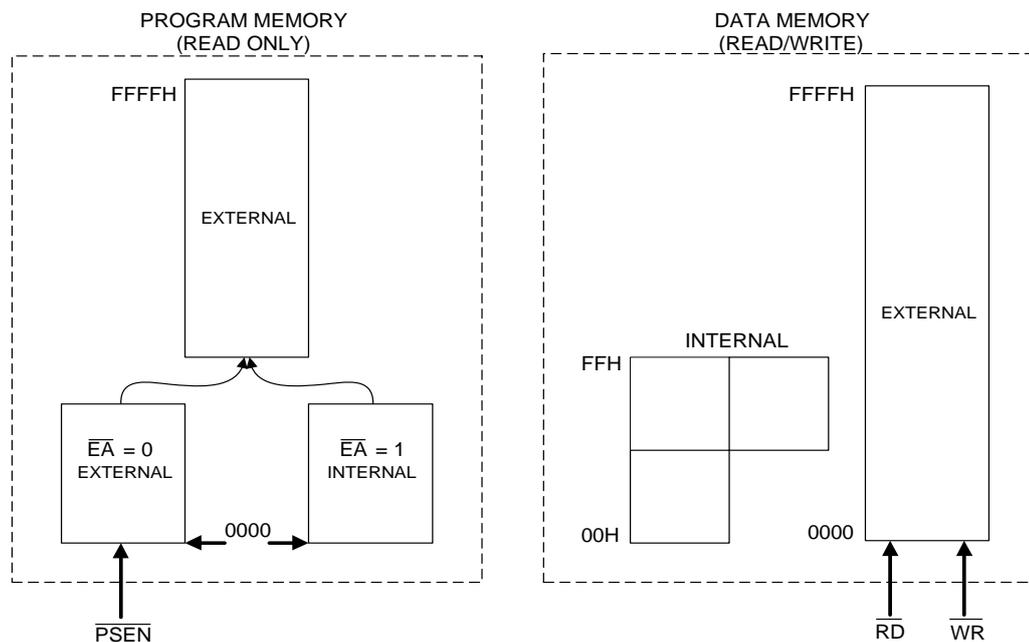
2.1.2. Memori

Mikrokontroler AT89S51 memiliki pembagian ruang alamat (*address space*) untuk memori program dan memori data. Pemisahan memori program dan memori data mengakibatkan memori data dapat diakses dengan alamat 8 bit.

Meskipun demikian, alamat memori data 16 bit dapat diakses melalui suatu register DPTR (*Data Pointer Register*).

Memori program disimpan dalam PEROM dan memori data disimpan dalam RAM. Lebar alamat memori program selalu 16 bit meskipun alamat yang digunakan lebih kecil dari 64K *byte*.

Gambar 2.3 menunjukkan struktur perbedaan antara memori data dan memori program. Pada memori program, ruang memori dapat diperluas sampai 64K *byte*. Untuk memilih penggunaan memori program internal atau memori program eksternal digunakan pin \overline{EA} (*External Access enable*). Setiap eksekusi memori program eksternal dipakai sinyal baca \overline{PSEN} (*Program Strobe Enable*). Memori data internal terdapat dalam *chip* berkapasitas 256 *byte*. Jika diperlukan dapat dilakukan penambahan memori data eksternal, dengan kapasitas maksimum sebesar 64K *byte*. Untuk mengakses memori data eksternal digunakan sinyal baca (RD) dan sinyal tulis (WR).



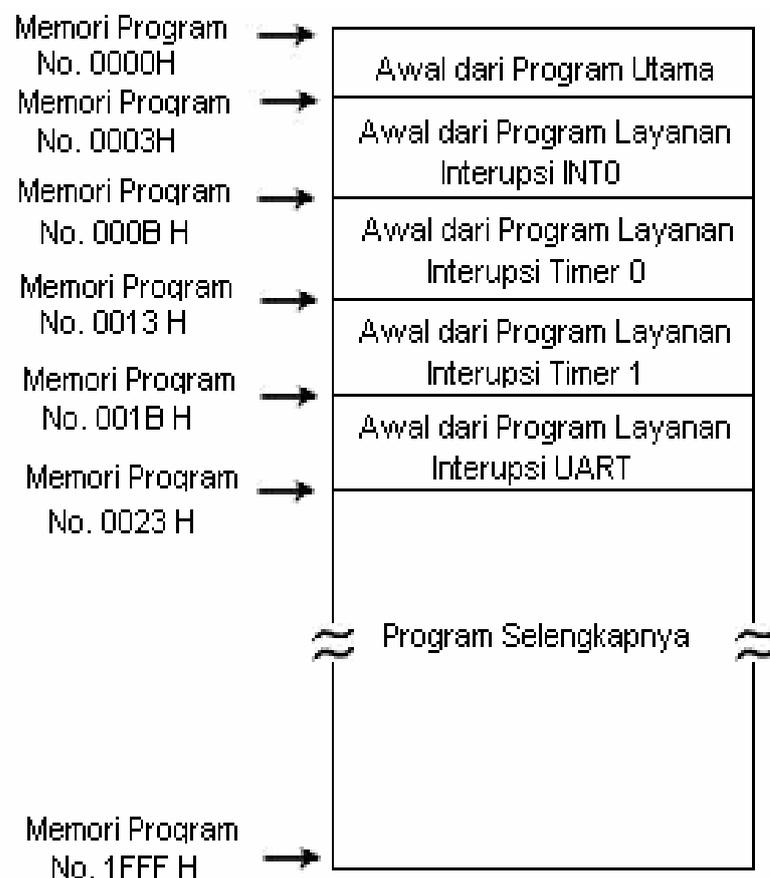
Gambar 2.3. Struktur Memori Mikrokontroler AT89S51

2.1.2.1. Memori Program

Program untuk mengendalikan kerja dari mikrokontroler disimpan di dalam memori program. Program pengendali tersebut merupakan kumpulan dari instruksi kerja mikrokontroler, satu instruksi MCS51 merupakan kode yang

panjangnya bisa 1-4 *byte*. Pada saat mikrokontroler melaksanakan instruksi program, instruksi tersebut *byte* demi *byte* diambil ke CPU dan selanjutnya dipakai untuk mengatur kerja mikrokontroler. Proses pengambilan instruksi dari memori program dikatakan sebagai '*fetch cycles*' dan saat CPU melaksanakan instruksi disebut sebagai '*execute cycles*'.

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



Gambar 2.4. Peta Memori Program.

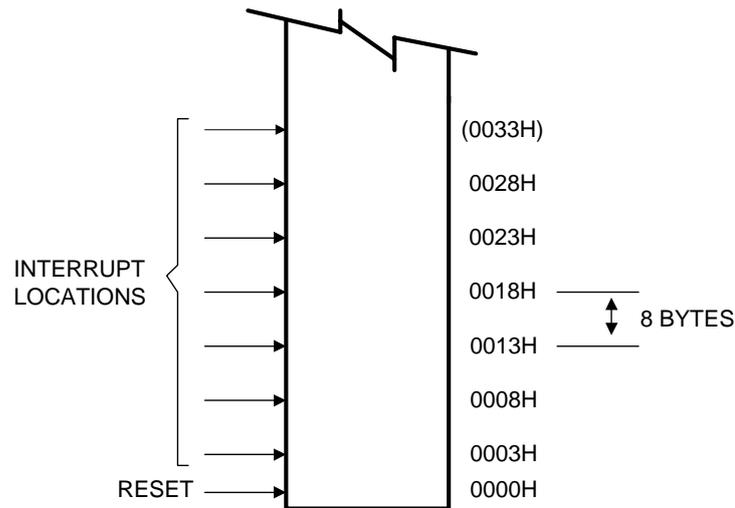
Pada saat MCS51 direset maka isi dari *Program Counter* menjadi 0000. Artinya sesaat setelah reset isi dari memori program nomor 0 dan seterusnya akan diambil ke CPU dan diperlakukan sebagai instruksi yang akan mengatur kerja

mikrokontroler. Dengan demikian, awal dari program pengendali MCS51 harus ditempatkan di memori nomor 0, setelah reset MCS51 menjalankan program mulai dari memori-program nomor 0000, dengan melakukan proses '*fetch cycles*' dan '*execute cycles*' terus menerus tanpa henti.

Jika sarana interupsi diaktifkan, dan tegangan di kaki *INT0 (P3.0)* diubah dari '1' menjadi '0', maka proses menjalankan program akan dihentikan sebentar, mikrokontroler melayani dulu permintaan interupsi, selesai melayani permintaan interupsi CPU akan melanjutkan mengerjakan program utama lagi. Untuk melaksanakan hal tersebut, pertama CPU akan menyimpan nilai *Program Counter* ke *Stack* (*Stack* merupakan satu bagian kecil dari data memori – *RAM*), kemudian mengganti isi *Program Counter* dengan 0003. Artinya MCS51 akan melaksanakan program yang ditempatkan di memori program mulai *byte* ke 3 untuk melayani interupsi yang diterima dari kaki *INT0*. Selesai melayani interupsi, nilai *Program Counter* yang tadi disimpan ke dalam *Stack* akan dikembalikan ke *Program Counter*, dengan demikian CPU bisa melanjutkan pekerjaan di program utama.

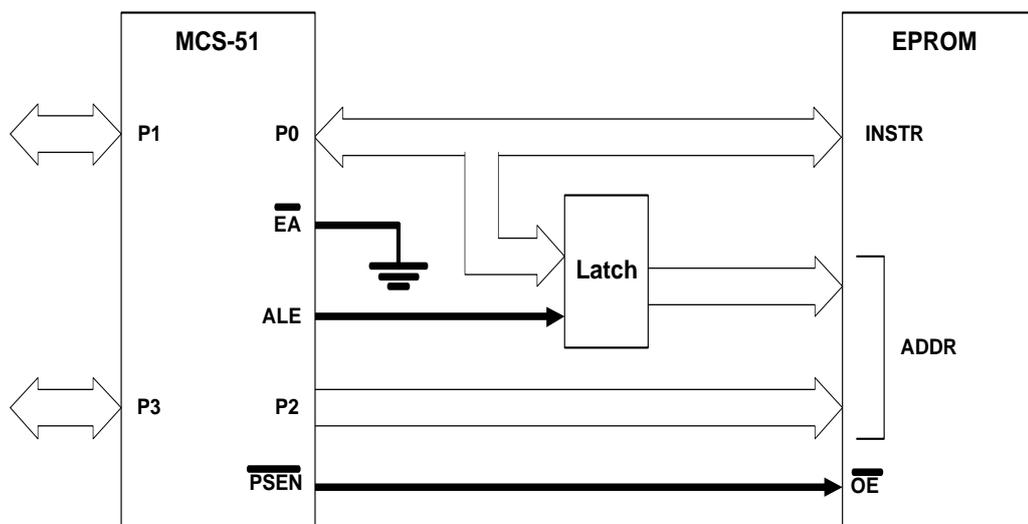
Selain *INT0*, AT89S51 bisa menerima interupsi dari *INT1*, *UART* dan *Timer*. Agar permintaan interupsi itu bisa dilayani dengan program yang berlainan, maka masing-masing sumber interupsi itu mempunyai nomor awal program untuk layanan interupsi yang berlainan.

Setelah melaksanakan rutin *reset*, mikrokontroler memulai mengeksekusi program pada alamat 0000H. Setiap interupsi mempunyai lokasi yang tetap dalam memori program. Interupsi menyebabkan CPU melompat ke lokasi tersebut di mana pada lokasi tersebut terdapat sub-rutin yang harus dilaksanakan.



Gambar 2.5. Alamat Memori Program Bagian Bawah

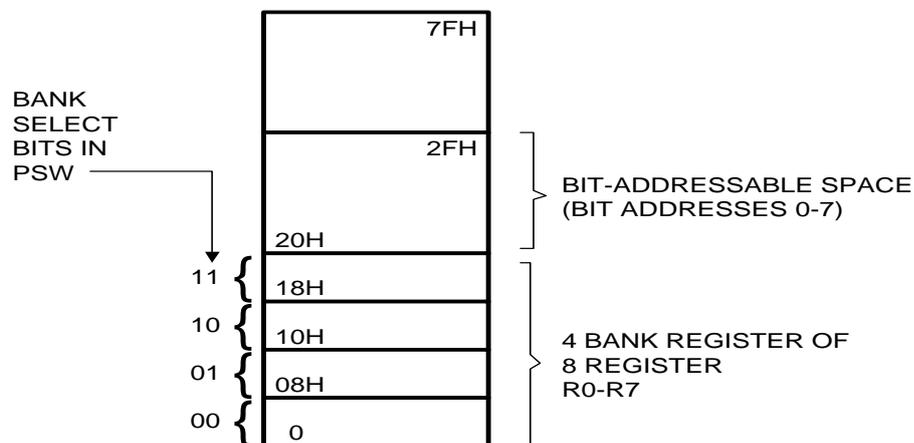
Konfigurasi perangkat keras yang diperlukan untuk mengeksekusi memori program eksternal terlihat pada gambar 2.6. Enam belas saluran I/O (pada *port 0* dan *port 2*) difungsikan sebagai *bus* selama pengambilan instruksi dari memori program eksternal. *Port 0* merupakan *bus* alamat yang dimultipleks dengan *bus* data. Sebagai *bus* alamat, *port 0* mengeluarkan alamat rendah (A0-A7), dan kemudian berubah menjadi kondisi mengambang untuk menerima data dari memori program eksternal. Pada saat *port 0* mengeluarkan alamat rendah, sinyal ALE (*Address Latch Enable*) menahan alamat tersebut pada *latch*. *Port 2* merupakan alamat tinggi (A8-A15) yang bersama dengan alamat rendah (A0-A7) membentuk alamat 16-bit.



Gambar 2.6. Konfigurasi Perangkat Keras Untuk Memori Program Eksternal

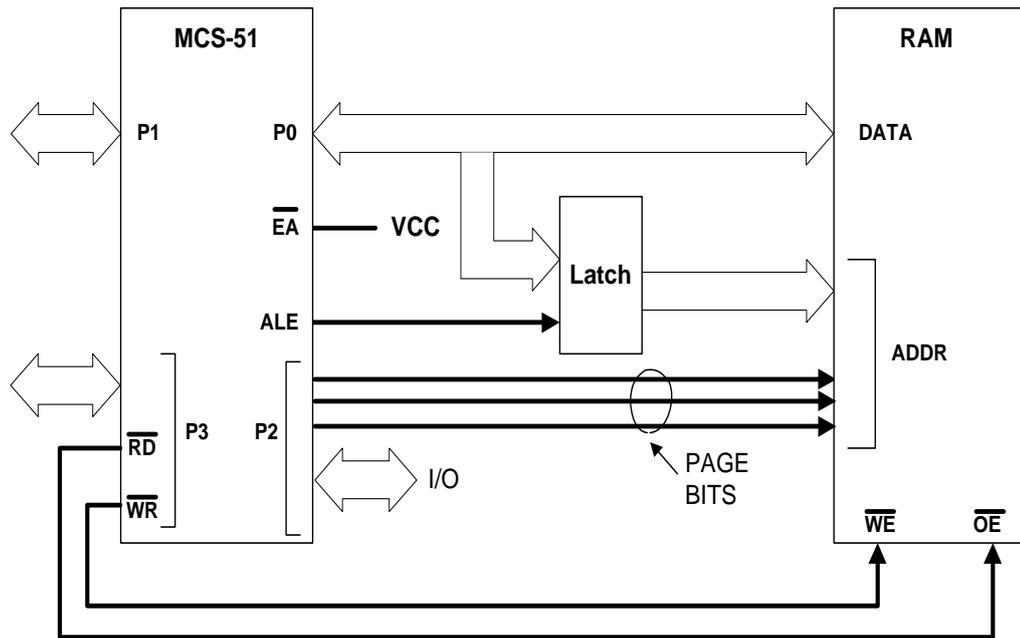
2.1.2.2 Memori Data

Mikrokontroler AT89S51 memiliki memori data yang berupa RAM internal sebesar 128 *byte*. Dari jumlah tersebut, 32 *byte* terbawah dikelompokkan menjadi 4 bagian yang biasa disebut *bank*. Masing-masing *bank* terdiri dari 8 *register* yang dapat diakses program dengan cara pengalamatan *register*. Pemilihan *bank* tersebut dilakukan dengan melalui suatu *register* yang disebut *Program Status Word* (PSW). Sedangkan 16 *byte* berikutnya di atas keempat *bank register* tersebut membentuk satu blok memori yang dapat dialamati per *bit*. Memori data ini dapat diakses dengan pengalamatan langsung atau pengalamatan tak langsung.



Gambar 2.7. Alamat Memori Data Bagian Bawah

Konfigurasi perangkat keras yang diperlukan untuk mengakses memori data eksternal (RAM) ditunjukkan pada gambar 2.8.



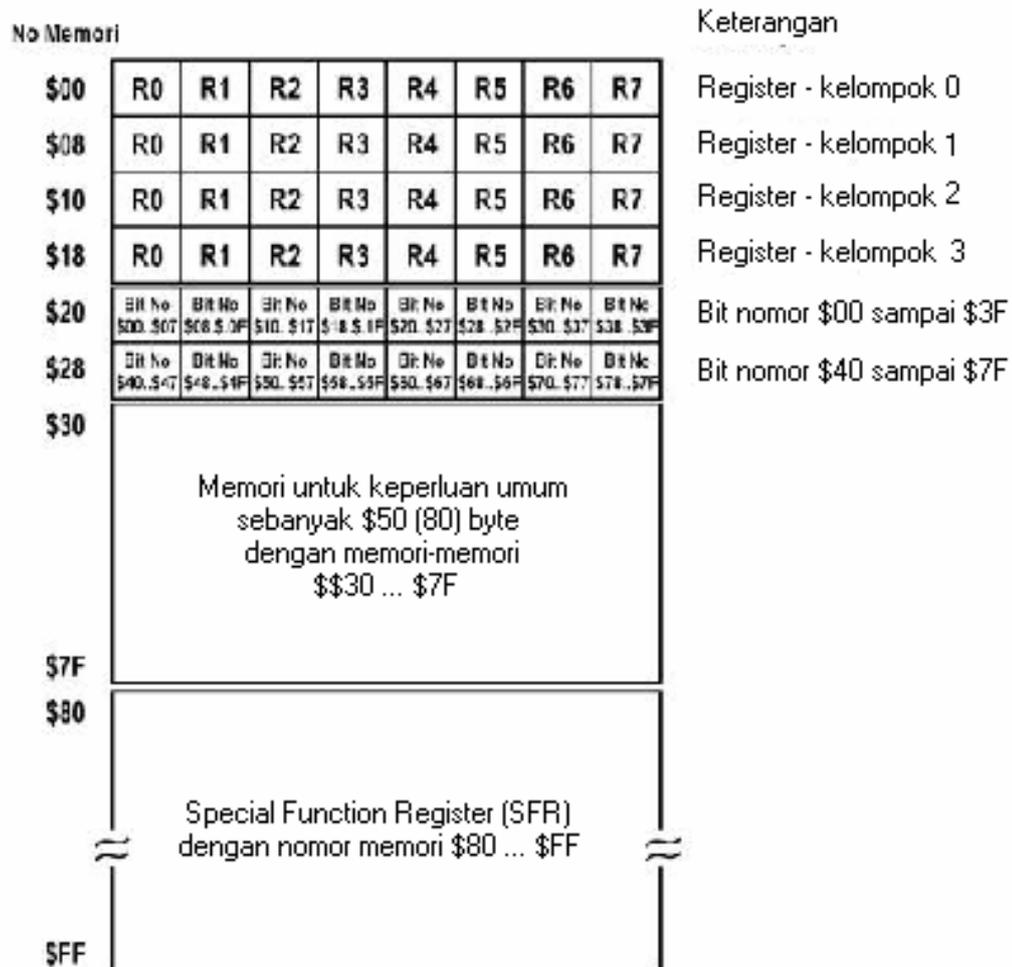
Gambar 2.8 Konfigurasi Perangkat Keras untuk Memori Data Eksternal

Memori data dalam mikrokontroler AT89S51 mempunyai penomoran yang terpisah yaitu untuk memori program \$0000 - \$07FF, sedangkan memori data \$00 - \$FF. Memori data ini dibagi dalam dua bagian, nomor \$00 sampai \$7F merupakan memori seperti *RAM* yang berupa memori data biasa, yang dibagi sebagai berikut :

- Nomor \$00 sampai \$18 dipakai sebagai Register Serba Guna (*General Purpose Register*).
- Nomor \$20 sampai \$2F dipakai untuk menyimpan informasi dalam level bit.
- Nomor \$30 sampai \$7F dipakai untuk menyimpan data maupun dipakai sebagai *stack*.
- Nomor \$80 - \$FF digunakan sebagai *Special Function Register*.

Register Serba Guna (*General Purpose Register*) sebesar 32 *byte* yang menempati memori nomor \$00 sampai \$18 dikelompokkan menjadi 4 kelompok register (*register bank*), 8 *byte* dari memori ini dikenal sebagai *Register 0... Register 7 (R0, R1, R2, R3, R4, R5, R6, R7)*. Selain itu *Register 0* dan *Register 1 (R0 dan R1)* bisa dipakai sebagai register penampung alamat yang dipakai dalam penyebutan memori secara tidak langsung (*indirect memory addressing*). Empat kelompok register serba guna ini tidak bisa dipakai secara bersamaan, saat setelah

reset yang aktif adalah kelompok *register 0*. Kalau yang diaktifkan adalah kelompok *register 1*, maka yang dianggap sebagai *R0* bukan lagi memori data nomor 0 melainkan memori data nomor 8, demikian pula kalau yang diaktifkan kelompok *register 3* maka memori data nomor \$18 menjadi *R0*.



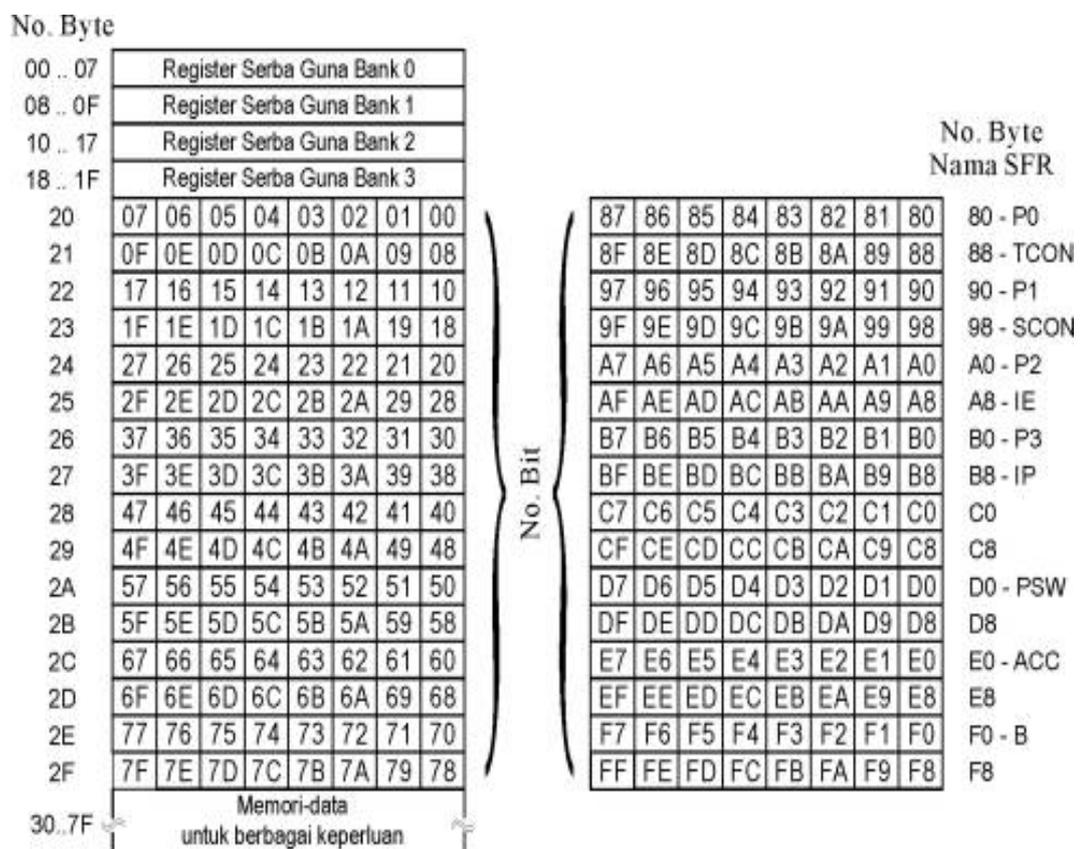
Gambar 2.9. Peta Memori Data

2.1.3. Memori Level Bit

Pada umumnya mikrokontroler mengolah data 8 bit sekaligus, misalnya mengisi akumulator dengan data 8 bit sekaligus, isi akumulator yang 8 bit dijumlahkan dengan isi memori yang 8 bit dan lain sebagainya. Memori data nomor \$20 sampai \$2F bisa digunakan untuk menampung informasi dalam level bit. Setiap byte memori di daerah ini bisa dipakai menampung 8 bit informasi yang masing-masing diberi nomor tersendiri, misalkan bit 0 dari memori data

nomor \$20 bisa disebut sebagai bit nomor 0, bit 1 memori data nomor \$20 disebut sebagai bit nomor 1. Bit 0 dari memori data nomor \$21 disebut sebagai bit nomor 8, bit 7 memori data nomor \$21 disebut sebagai bit nomor 15 dan seterusnya.

Dengan demikian dari 16 byte memori yang ada bisa dipakai untuk menyimpan 128 bit (16 x 8 bit) yang diberi nomor \$00 sampai \$7F. Disamping itu operasi bit bisa berlaku di memori data nomor \$80 sampai dengan \$FF yang biasa disebut *Special Function Register (SFR)*. Hanya *SFR* dengan nomor memori data yang diakhiri dengan angka heksa desimal 0 atau 8 yang bisa dipakai untuk operasi bit, bit pada memori data daerah ini sebanyak 128 bit, mendapat nomor \$80 sampai dengan \$FF. Secara keseluruhan operasi bit bisa diperlakukan pada 256 lokasi bit seperti terlihat dalam gambar 2.10.



Gambar 2.10. Denah Memori Bit

2.1.4. Register Dasar MCS51

Setiap mikroprosesor/mikrokontroler selalu dilengkapi dengan register dasar untuk keperluan penulisan program. Ada register yang merupakan register baku yang dijumpai di semua jenis mikroprosesor/mikrokontroler, ada pula register yang spesifik. Register baku antara lain *Program Counter*, Akumulator, *Stack Pointer Register*, *Program Status Word*. Sebagai register yang khas *MCS51* antara lain adalah *Register B*, *Data Pointer High byte* dan *Data Pointer Low Byte*. Register ini menempati kedudukan antara lain :

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

Program Counter (PC) dalam mikrokontroler AT89S51 merupakan register dengan kapasitas 16 bit. Didalam *PC* dicatat nomor memori program yang menyimpan instruksi berikutnya yang akan diambil (*fetch cycles*) sebagai instruksi untuk dikerjakan (*execute*). Saat setelah reset *PC* bernilai 0000H, berarti *MCS51* akan segera mengambil isi memori program nomor 0 sebagai instruksi. Nilai *PC* otomatis bertambah 1 setelah prosesor mengambil instruksi 1 *byte*. Ada instruksi yang hanya 1 *byte*, ada instruksi yang sampai 4 *byte*, dengan demikian penambahan nilai *PC* setelah menjalankan instruksi, tergantung pada jumlah *byte* instruksi bersangkutan.

Akumulator adalah sebuah register yang berfungsi untuk menampung (*Accumulate*) hasil-hasil pengolahan data dari banyak instruksi *MCS51*. Akumulator bisa menampung data 8 bit dan merupakan register yang paling banyak digunakan, lebih dari setengah instruksi *MCS51* melibatkan akumulator.

Salah satu bagian dari memori data dipakai sebagai *stack*, yaitu tempat yang dipakai untuk menyimpan sementara nilai *PC* sebelum prosesor menjalankan sub-rutin, nilai tersebut akan diambil kembali dari *stack* dan dikembalikan ke *PC* saat prosesor selesai menjalankan sub-rutin. *Stack Pointer Register* adalah register yang berfungsi untuk mengatur kerja *stack*, dalam *Stack Pointer Register* disimpan nomor memori data yang dipakai untuk operasi *stack* berikutnya.

Program Status Word (PSW) berfungsi mencatat kondisi prosesor setelah melaksanakan instruksi. Sedangkan *Register B* merupakan register dengan kapasitas 8 bit, merupakan register pembantu akumulator saat menjalankan instruksi untuk perkalian dan pembagian. *Data Pointer High Byte (DPH)* dan *Data Pointer Low Byte (DPL)* masing-masing merupakan register dengan kapasitas 8 bit. Tapi dalam pemakaiannya kedua register ini digabungkan menjadi satu register 16 bit yang dinamakan sebagai *Data Pointer Register (DPTR)*. Sesuai dengan namanya, register ini dipakai untuk mengamati data dalam jangkauan yang luas.

2.1.5. SFR (*Special Function Register*)

SFR adalah register-register yang digunakan untuk pelayanan *latch port*, *timer*, *program status word*, kontrol peripheral dan lain sebagainya. SFR terletak pada alamat 80H sampai FFH pada memori data internal.

Tabel 2.2. Nama dan Alamat Register pada SFR

Simbol	Nama Register	Alamat	Nilai Setelah Reset
ACC *)	<i>Accumulator</i>	E0H	00H
B *)	<i>B Register</i>	F0H	00H
PSW *)	<i>Program Status Word</i>	D0H	00H
SP	<i>Stack Pointer</i>	81H	07H
DPTR	<i>Data Pointer</i>		
DPL	<i>Low Bytes</i>	82H	00H
DPH	<i>High Bytes</i>	83H	00H
P0 *)	<i>Port 0</i>	80H	FFH
P1 *)	<i>Port 1</i>	90H	FFH
P2 *)	<i>Port 2</i>	A0H	FFH
P3 *)	<i>Port 3</i>	B0H	FFH
SBUF	<i>Serial Data Buffer</i>	99H	Independen
TH0	<i>Timer/counter 0 high byte</i>	8CH	00H
TL0	<i>Timer/counter 0 low byte</i>	8AH	00H
TH1	<i>Timer/counter 1 high byte</i>	8DH	00H

TL1	<i>Timer/counter 1 low byte</i>	8BH	00H
IP *)	<i>Interrupt Priority Control</i>	B8H	XXX00000B
IE *)	<i>Interrupt Enable Control</i>	A8H	0XX00000B
TMOD	<i>Timer/counter Mode Control</i>	89H	00H
TCON *)	<i>Timer/counter Control</i>	88H	00H
SCON *)	<i>Serial Control</i>	98H	00H
PCON	<i>Power Control</i>	87H	0XXXXXXXXB

*) dapat dialamati per *bit*.

2.1.6. Interupsi

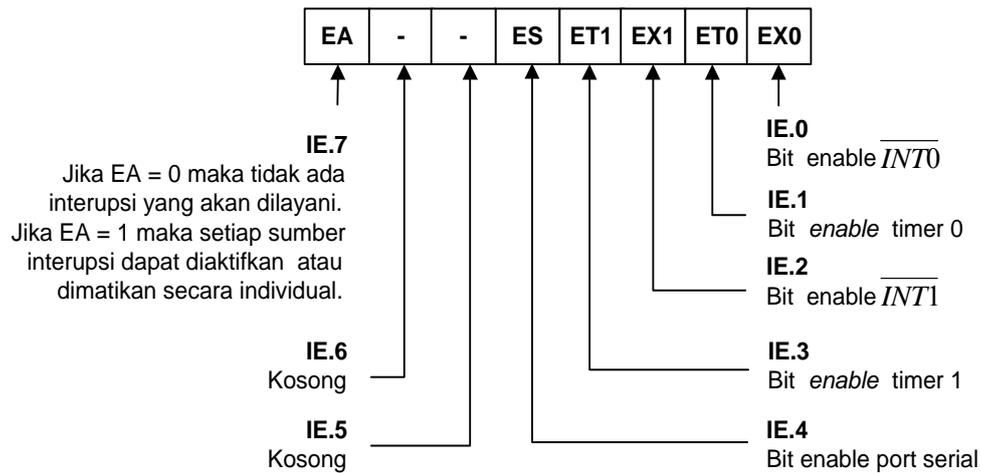
Apabila CPU dalam mikrokontroler AT89S51 sedang melaksanakan suatu program, kita dapat menghentikan pelaksanaan program tersebut secara sementara dengan meminta interupsi. Pada mikrokontroler AT89S51 terdapat beberapa saluran interupsi. Interupsi pada AT89S51 dibedakan dalam 2 jenis, yaitu :

1. Interupsi yang tak dapat dihalangi oleh perangkat lunak (*non maskable interrupt*), misalnya *reset*.
2. Interupsi yang dapat dihalangi oleh perangkat lunak (*maskable interrupt*).

Contoh interupsi jenis ini adalah $\overline{\text{INT0}}$ dan $\overline{\text{INT1}}$ (eksternal) serta *Timer/counter 0*, *Timer/counter 1*, dan interupsi dari *port* serial (internal).

Mikrokontroler AT89S51 menyediakan 5 sumber interupsi, yaitu 2 interupsi eksternal, 2 interupsi *timer*, dan 1 interupsi *port* serial. Alamat awal layanan rutin interupsi dari setiap sumber interupsi dapat dilihat pada tabel 2.3.

Masing-masing sumber interupsi dapat di *enable/disable* secara perangkat lunak yaitu dengan mengatur register IE (*Interrupt Enable*). Sebagai contoh, jika ingin mengaktifkan *timer 0*, maka nilai yang harus diberikan ke IE adalah 82H (memberikan logika 1 ke EA dan ET0).



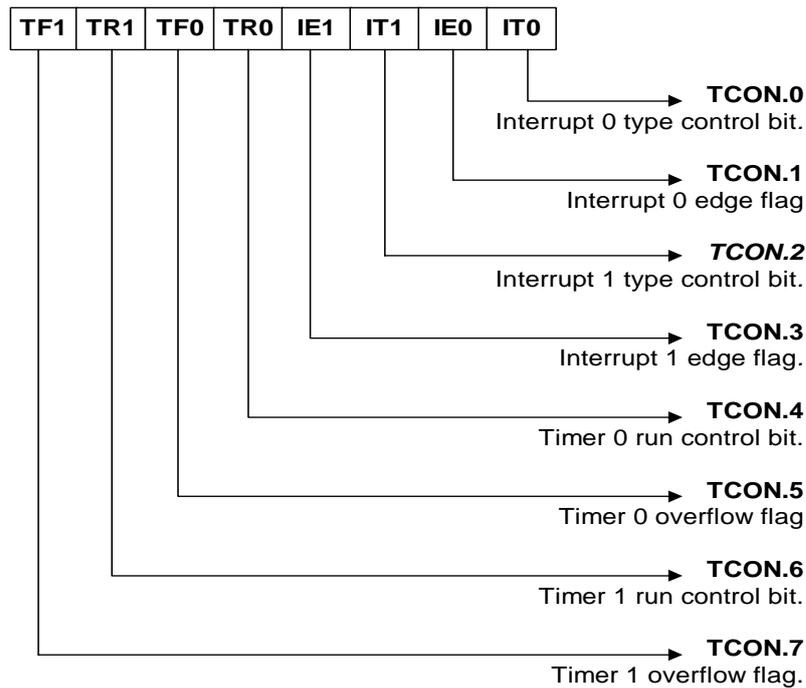
Gambar 2.11. Susunan bit Register IE.

Tabel 2.3. Alamat Layanan Rutin Interupsi

Nama	Alamat Interupsi
Reset	00H
$\overline{INT0}$	03H
Timer 0	0BH
$\overline{INT1}$	13H
Timer 1	1BH
Sint	23H

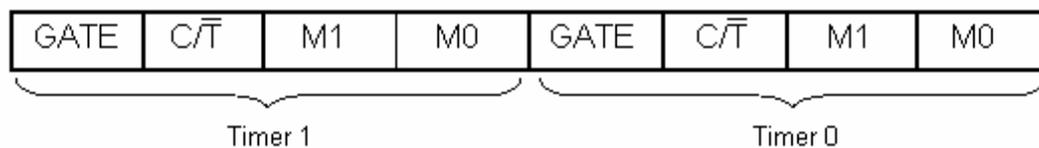
2.1.7. Timer dan Counter

Pada mikrokontroler AT89S51 terdapat dua buah *timer/counter* 16 bit yang dapat diatur melalui perangkat lunak, yaitu *timer/counter* 0 dan *timer/counter* 1. Pengontrol kerja *timer/counter* adalah register *Timer/counter Control* (TCON) dimana susunan bit-bitnya ditunjukkan pada gambar 2.12.



Gambar 2.12. Susunan bit Register TCON.

Mode penggunaan *timer 0* dan *timer 1* ada empat macam, yaitu mode 0 (*timer 13 bit*), Mode 1 (*timer 16 bit*), mode 2 (*timer 8 bit auto-reload*), dan mode 3 (*timer 8 bit yang terpisah*). Register pengontrol pemilihan mode pada operasi *timer/counter* adalah register *Timer Mode* (TMOD) dimana susunan bit-bitnya ditunjukkan pada gambar 2.13.



Gambar 2.13. Susunan bit Register TMOD.

- **GATE** : jika *gate* dalam keadaan *high*, *timer/counter* x akan aktif jika pin $\overline{\text{INTx}}$ dan TRx (TCON) dalam keadaan *high*. jika *gate* dalam keadaan *low*, *timer* x akan aktif jika TRx dalam keadaan *high*.
- $\overline{\text{C/T}}$: *timer* atau *counter*, *clear* (0) untuk operasi *timer* dengan masukan dari

clock internal, *set* (1) untuk operasi *counter* dengan masukan dari *pin* T0 atau *pin* T1.

- M : mode kerja dari *timer/counter*.

2.1.8. Rangkaian reset

Rangkaian reset digunakan untuk mereset mikrokontroler. Rangkaian *power on reset* diperlukan agar AT89S51 dapat direset secara otomatis pada saat pertama kali dihidupkan. Keadaan reset diperoleh bila *pin* 9 yaitu RST diberi logika “1” dalam beberapa mili detik setelah catu daya diberikan.

Rumus waktu untuk reset :

$$\tau = R \times C \dots\dots\dots(2)$$

dimana : τ = waktu reset mikrokontroler

R = nilai dari resistor

C = nilai dari kapasitor

Untuk rangkaian reset ini digunakan resistor sebesar 5600 Ω dan nilai kapasitor sebesar 10 μF , jadi waktu reset dari mikrokontroler adalah sebesar 56 ms.

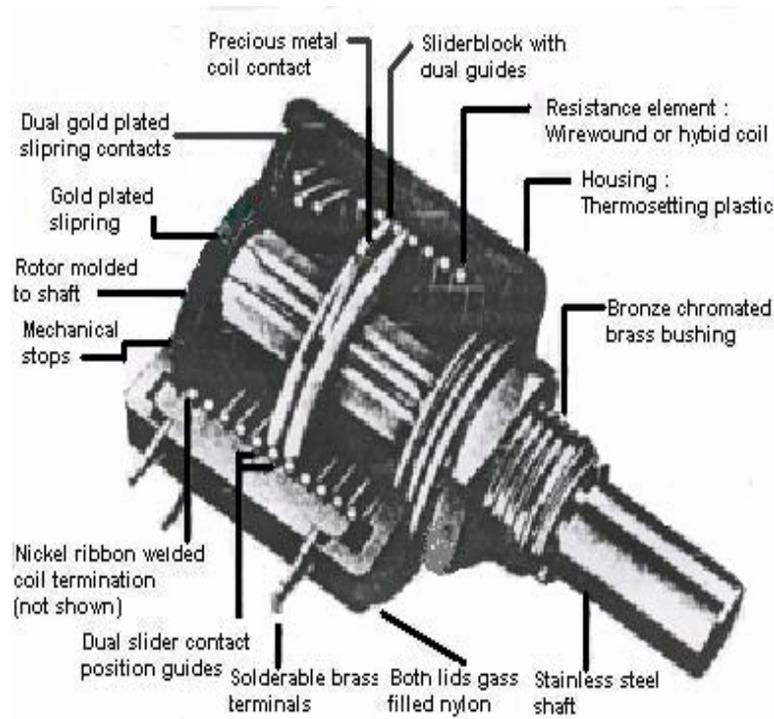
2.1.9. Rangkaian Clock

Pada mikrokontroler AT89S51 sudah terdapat rangkaian *oscillator* yang digunakan sebagai sumber *clock* bagi mikrokontroler itu sendiri. Untuk dapat menjalankan osilator tersebut maka kita harus menghubungkan sebuah kristal dengan *pin* 18 (XTAL2) dan *pin* 19 (XTAL1) serta dua buah kapasitor dihubungkan dengan *ground*.

Clock generator internal dari mikrokontroler AT89S51 menentukan siklus mikrokontroler itu sendiri. Satu siklus mesin (*machine cycle*) terdiri dari 6 keadaan yaitu S1 sampai S6 dan setiap keadaan tersebut sama dengan dua kali periode osilator. Jadi setiap satu siklus mesin adalah sepanjang 12 periode osilator. Pemilihan frekuensi kristal yang digunakan adalah 11,0592 MHz dan nilai kapasitansi yang digunakan adalah 30 pF.

2.2. Potensiometer

Potensiometer adalah *transduser* elektromekanik yang mengubah perubahan mekanik menjadi besaran listrik. Perubahan mekanik disini berkaitan dengan pergeseran atau perputaran (*linear/rotational displacement*). Potensiometer putar dapat ditemukan secara komersil dalam bentuk satu putaran atau banyak putaran, dengan gerak putaran yang terbatas ataupun tidak terbatas. Potensiometer disebut juga *variable resistor* atau disingkat *VR* (*variable resistor*). Pada tugas akhir ini, potensiometer digunakan sebagai sensor posisi.



Gambar 2.14. Potensiometer Linier

Berdasarkan perubahan nilai hambatannya, potensiometer dibedakan menjadi dua yaitu potensiometer log dan potensiometer linier. Potensiometer log adalah potensiometer yang perubahan nilai hambatannya berdasarkan skala log (bersifat logaritmik). Potensiometer log biasanya digunakan pada pengatur *volume, bass, treble*. Sedangkan potensiometer linier adalah potensiometer yang

perubahan nilai hambatannya beraturan sesuai dengan sudut putar atau jarak gesernya.

2.3. ADC0804

ADC (*Analog to Digital Converter*) adalah pengubah sinyal analog menjadi data digital yang proporsional dengan besar tegangan input. Pada tugas akhir ini, ADC0804 berfungsi sebagai pengubah tegangan keluaran dari potensiometer menjadi data digital yang dapat diolah oleh mikrokontroler. ADC0804 termasuk konverter A/D jenis *Successive Approximation Register* (SAR), dengan waktu konversi 100 μs , mempunyai input analog tegangan differential ($V_{\text{in}+}$ dan $V_{\text{in}-}$), *on-chip clock generator* (clock internal), memiliki jumlah data output sebanyak 8 bit sehingga resolusinya untuk tegangan referensi 5V adalah sekitar 20 μV .

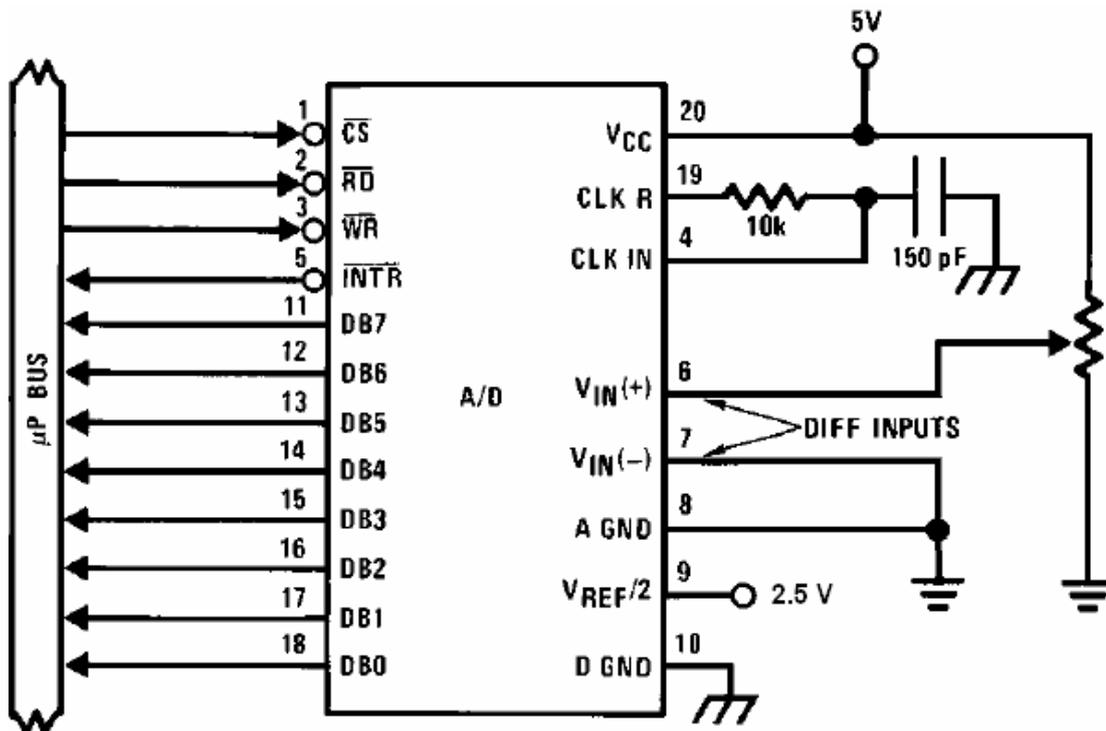
Rangkaian resistor dan kapasitor pada pin CLK IN dan CLK R digunakan sebagai *self clocking* pada A/D, dimana :

$$f_{\text{CLK}} \cong \frac{1}{1.1RC} \text{ dengan nilai } R \cong 10 \text{ k}\Omega$$

Sehingga jika harga $R = 10 \text{ k}\Omega$ dan $C = 150 \text{ pF}$, maka :

$$f_{\text{CLK}} \cong \frac{1}{1.1 \times 10000 \times 150 \times 10^{-12}} \approx 600 \text{ kHz}$$

Karena tegangan referensi yang digunakan sebesar 5 Volt, maka besar tegangan yang diberikan pada pin $V_{\text{REF}/2}$ (kaki 9) adalah sebesar 2.5 Volt.



Gambar 2.15. Rangkaian ADC0804

Adapun beberapa fungsi dari pin-pin ADC 0804 yang lainnya adalah sebagai berikut:

- **CS**, pin ini merupakan pin input aktif *low*. Jika pin ini diberi logika “0” maka akan menyebabkan ADC0804 aktif.
- **WR**, pin ini merupakan pin input aktif *low*. Pada saat pin WR diberi logika “0” dan pin CS diberi logika “0”, maka data analog mulai dikonversikan ke dalam besaran digital.
- **INTR**, pin ini merupakan pin output aktif *low*. Pin ini digunakan oleh ADC0804 untuk memberi sinyal kepada peralatan luar sebagai tanda proses konversi data telah selesai.
- **RD**, pin ini merupakan pin input aktif *low*. Pada saat pin RD diberi logika “0” dan pin CS diberi logika “0”, maka ADC0804 akan mengeluarkan data digital.
- **D0–D7**, merupakan output data digital, dimana D0 adalah *List Significant Bit* (LSB) dan D7 adalah *Most Significant Bit* (MSB).

2.4. Sensor Ketepatan Letak

Guna membatasi gerakan – gerakan dari suatu mekanik dapat menggunakan sensor pembatas gerakan dalam hal ini digunakan *limit switch*. *Limit switch* yang digunakan mempunyai 2 kondisi awal yaitu kondisi *Normally Open* (NO) dan kondisi *Normally Close* (NC) atau yang biasa disebut *change over*, kedua kondisi tersebut dapat dilihat pada gambar symbol dioda pada gambar 2.16. Kondisi awal NO akan tertutup dan NC akan terbuka apabila pada tuas dari *limit switch* ditekan.



Gambar 2.16. Symbol Limit Switch



Gambar 2.17. Bentuk fisik limit switch

2.5. Relay

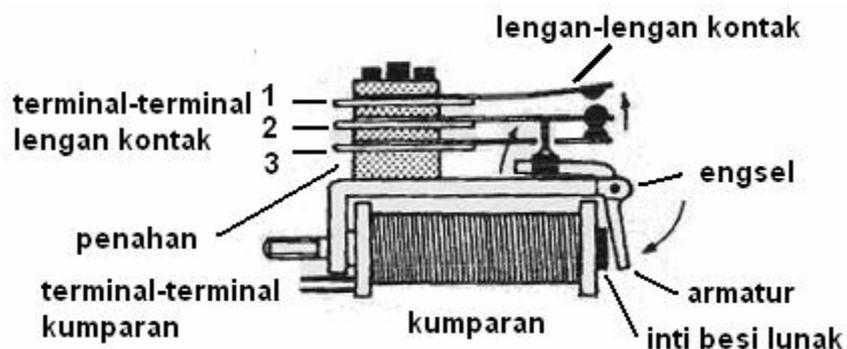
Relay adalah sebuah alat elektromagnetik yang dapat mengubah kontak-kontak saklar sewaktu alat ini menerima sinyal listrik. Salah satu bentuk relay adalah relay 4 kontak, seperti ditunjukkan pada gambar 2.18. Relay pada rangkaian skripsi ini digunakan sebagai pengontrol arus yang lebih besar yang

akan dicatukan pada motor DC. Fungsi lain dari relay pada skripsi ini adalah sebagai pembalik polaritas catu tegangan pada motor DC.



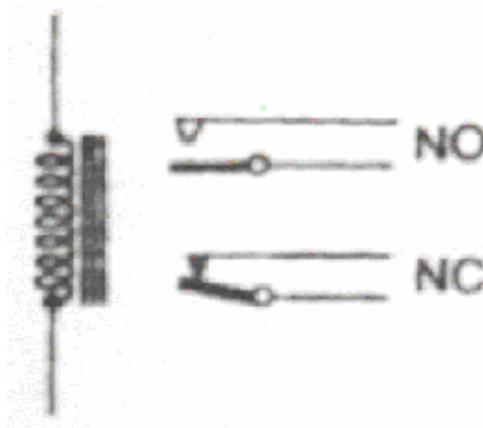
Gambar 2.18. Bentuk fisik relay 4 kontak

Relay tersusun atas sebuah kumparan kawat beserta sebuah inti besi lunak, seperti gambar 2.19. Ketika arus kontrol kecil melewati kumparan, inti besi lunak akan dimagnetisasi. Armatur ditarik oleh inti yang di magnetisasi. Gerakan armatur ini akan menutup kontak 1 dan 2 dan akan membuka kontak 2 dan 3. Dengan kata lain gerakan armatur tadi telah mengubah kontak 1 dan 3. Kontak-kontak ini dapat dipakai untuk mengontrol arus yang lebih besar.



Gambar 2.19. Penampang Relay

Relay adalah saklar dimana fungsi utamanya untuk mengontrol arus yang lebih besar dalam rangkaian dengan arus kecil yang melalui kumparan. Simbol relay dalam rangkaian ditunjukkan pada gambar 2.20.

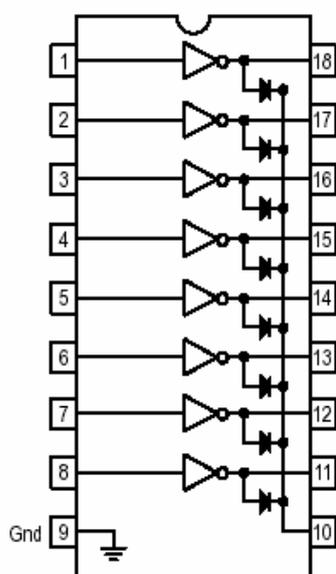


Gambar 2.20. Simbol Relay

Simbol ini terdiri atas sebuah kumparan dan 2 set kotak, satu biasanya terbuka (*normaly open*) atau NO lainnya biasanya tertutup (*normaly close*) NC. Ketika arus melewati kumparan, kontak NO tertutup, sebaliknya kontak NC terbuka

2.6. Penguat ULN2803

ULN2803 merupakan IC yang tersusun dari delapan buah transistor yang dihubungkan secara *darlington* yang berfungsi sebagai antar muka (*interface*) antara rangkaian digital yang mengkonsumsi arus dan tegangan rendah (misalnya TTL, CMOS, PMOS/NMOS) dengan peralatan yang memerlukan arus dan tegangan tinggi (misalnya lampu, *relay*, motor *stepper*). Keistimewaan dari IC ini adalah *outputnya* merupakan *open collector*. Dalam penggunaannya, IC ULN2803 kompatibel dengan rangkaian yang menggunakan IC TTL. Pada perencanaan alat, *output* dari IC ULN2803 langsung dihubungkan dengan motor dimana tegangan *output* dari IC ULN2803 adalah sebesar 12 volt. Sedangkan *input* dari IC ini dihubungkan dengan rangkaian TTL.



Gambar 2.21. Konfigurasi pin IC ULN2803

2.7 Sensor

Sensor merupakan suatu pengubah besaran non elektrik menjadi besaran elektrik. Besaran fisis tersebut dapat berupa suhu, kelembaban, tekanan, cahaya, dan lain sebagainya. Besaran elektrik yang dihasilkan dapat berupa tegangan, arus, hambatan, kapasitansi, dan lain sebagainya.

Berdasarkan sumber dayanya, sensor dibagi menjadi 2 yaitu sensor aktif dan sensor pasif. Sensor aktif merupakan sensor yang mempunyai 3 port, yaitu port input, port output, dan port untuk eksitasi listrik yaitu umumnya berupa tegangan dan arus. Sedangkan sensor pasif merupakan sensor yang mempunyai daya sendiri dalam menghasilkan besaran elektris. Sensor ini mempunyai 2 port yaitu port input dan port output. Energi elektrikal pada port output berasal dari energi fisik inputnya. Untuk sensor jarak ada dua macam jenis yang bisa dipakai, yaitu sensor *polaroid* dan sensor *ultrasonic*.

2.7.1. Sensor *Polaroid*

Prinsip dasar dari pendeteksi jarak sonar sama dengan sistem – sistem lain yang sering digunakan. Pendeteksian diaktifkan oleh desingan suara sonik pada suatu frekwensi yang spesifik. Biasanya sensor berdesing dengan frekwensi transisi antara + 200v dan - 200v. Transisi ini diberikan kepada transducer pada

frekwensi sekitar 50 kHz. Dengan acuan, telinga manusia dapat mendengar bunyi dengan frekwensi 20 Hz sampai 20 kHz. Gelombang dipancarkan secara radial menjauh dari *transducer* melalui udara dengan kecepatan rambat bunyi kira-kira 343,2 m/s dan frekwensi 0,9 ms/foot. Pada saat gelombang menjangkau suatu obyek, maka gelombang tersebut akan dibalikkan arahnya oleh objek tersebut. Gelombang yang dibalikkan ini merambat kembali menuju ke *transducer* dengan kecepatan yang sama. Saat gelombang balik ini mengenai *transducer*, maka akan tercipta suatu *output* berupa voltase yang akan dikirimkan ke *stepped-gain amplifier*. Di dalam *stepped-gain amplifier* ini data diolah sehingga diperoleh jarak.



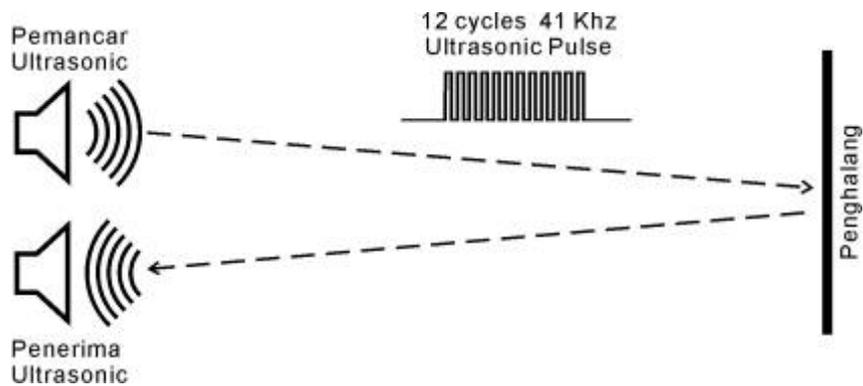
Gambar 2.22. Tampak luar *transducer* sensor *polaroid*



Gambar 2.23. Bagian dalam *transducer*

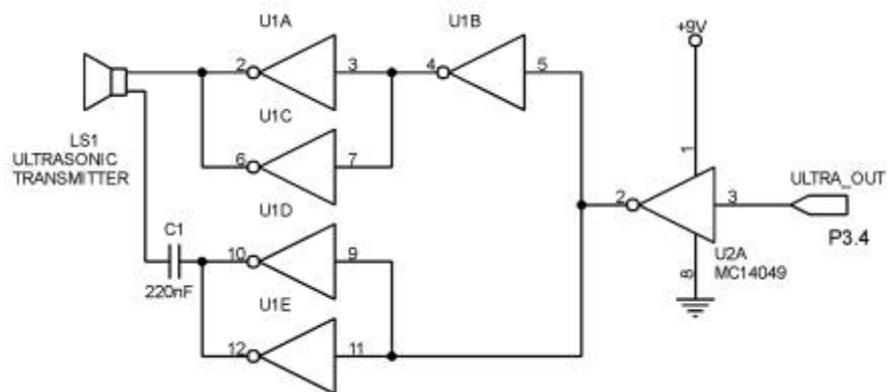
2.7.2. Sensor *Ultrasonic*

Prinsip kerja *echo sounder* untuk pengukuran jarak digambarkan dalam Gambar 2.24. Pulsa *ultrasonic*, yang merupakan sinyal *ultrasonic* dengan frekwensi lebih kurang 41 KHz sebanyak 12 periode, dikirimkan dari pemancar *ultrasonic*. Ketika pulsa mengenai benda penghalang, pulsa ini dipantulkan, dan diterima kembali oleh penerima *ultrasonic*. Dengan mengukur selang waktu antara saat pulsa dikirim dan pulsa pantul diterima, jarak antara alat pengukur dan benda penghalang bisa dihitung.



Gambar 2.24. Prinsip *Echo Sounder*

2.7.2.1. Pemancar Pulsa *Ultrasonic*

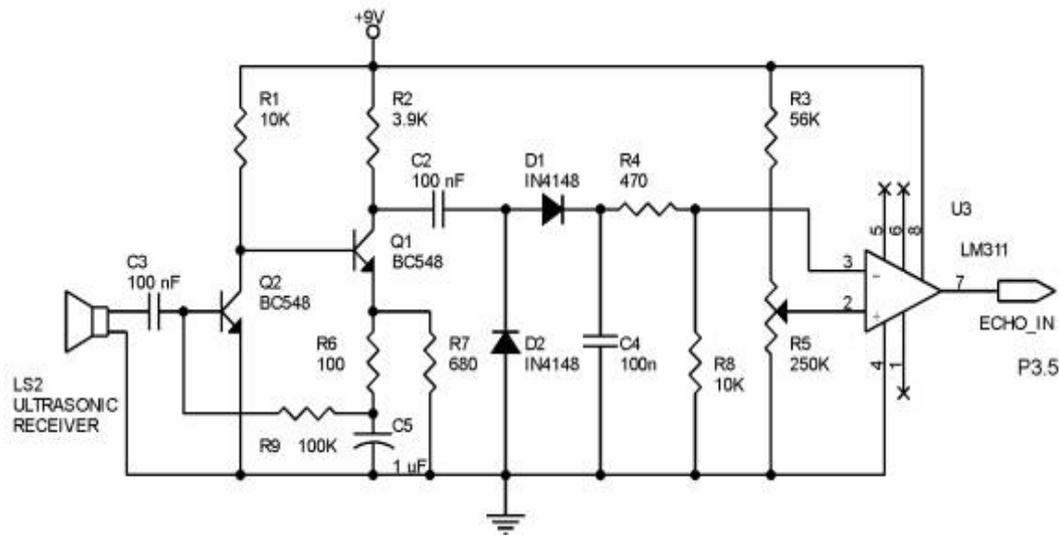


Gambar 2.25. Rangkaian Pemancar Ultra Sonic

Pulsa *ultrasonic* diperkuat dan dipancarkan dengan rangkaian pemancar *ultrasonic* di gambar 2.25. Rangkaian ini dibangun dengan *Inverter CMOS MC14049*, inverter **U1B** dipakai untuk membalik fasa sehingga tegangan di *output* gabungan **U1A & U1C** akan selalu berlawanan dengan tegangan di *output* gabungan **U1D & U1E**, dengan demikian amplitudo *ultrasonic* yang sampai di transduser *ultrasonic* menjadi 2 kali lipat. **C1** dipakai untuk menahan arus DC, sehingga hanya sinyal *ultrasonic* saja yang bisa masuk ke transduser *ultrasonic*.

2.7.2.2. Penerima Pulsa *Ultrasonic*

Rangkaian penerima *ultrasonic* pada Gambar 2.26., merupakan rangkaian yang umum dipakai untuk penerima *ultrasonic*.



Gambar 2.26. Rangkaian Penerima *Ultrasonic*

2.8. Regulator

Di dalam *automatic control*, *regulator* adalah suatu alat yang mempunyai fungsi untuk mengatur besar kecilnya arus dan tegangan yang akan masuk ke dalam rangkaian elektronik. Sebagai salah satu contoh, dalam rangkaian ECU yang dibuat ini dipakai *regulator* yang disebut *voltage regulator*. *Voltage regulator* dapat berupa suatu trafo dengan perbandingan transformasi voltase yang dapat disesuaikan, atau suatu sirkuit elektronik yang bisa menghasilkan suatu voltase yang diinginkan. *Voltage regulator* sendiri merupakan suatu pengatur elektrik yang dirancang secara otomatis untuk memelihara tegangan dalam satu unit rangkaian elektronik secara konstan. Ini bisa dilakukan dengan menggunakan suatu mekanisme *electromechanical*, atau komponen elektronik aktif atau pasif. Tergantung pada disain yang digunakan, mungkin saja digunakan untuk mengatur satu atau lebih arus bolak-balik atau DC *voltages*.

Banyak *regulator power supply* DC sederhana untuk mengatur voltase menggunakan *shunt regulator* seperti dioda zener, *avalanche breakdown diode*, dan *voltage regulator tube*. Masing-masing dari alat ini mulai konduksi pada

voltase tertentu dan akan berkonduksi sebanyak arus yang dibutuhkan untuk menahan voltase terminal hingga mencapai voltase yang telah ditentukan.