

Diktat Kuliah MIKROPROSESOR

Dosen : Ir.AMIR HAMZAH,MT

KATA PENGANTAR

Puji syukur penulis ucapkan ke hadapan Allah SWT karena berkat rahmat, hidayah, dan karuniaNya penulis dapat menyelesaikan Diktat untuk Mata Kuliah Mikroprosesor ini. Tujuan penulisan diktat kuliah ini adalah untuk memberikan pemahaman, pengetahuan, serta contoh-contoh yang praktis dan sederhana tentang Mikroprosesor. Contoh-contoh disesuaikan dengan lingkungan dan kehidupan sehari-hari para pembaca, khususnya mahasiswa Jurusan Teknik Elektro Oleh karena itu, diharapkan setelah membaca, mempelajari, dan membuat beberapa latihan serta tugas, mahasiswa mampu membuat perencanaan dan pelaksanaan Mata Kuliah Mikroprosesor. Ucapan terima kasih penulis sampaikan kepada semua pihak yang telah membantu penulis dalam menyelesaikan penulisan diktat ini. Akhirnya sumbang saran sangat penulis harapkan demi kesempurnaan diktat ini. Semoga diktat ini bermanfaat bagi seluruh pembaca, khususnya mahasiswa Teknik Elektro.

Bogor, Agustus 2019

Penulis

Daftar Isi

| | |
|--|-----------|
| Bab I Pendahuluan Mikroprosesor..... | 3 |
| Mikrokontroler..... | 7 |
| Survey Mikrokontroler..... | 9 |
| Mikrokontroler 8 Bit..... | 9 |
| Mikrokontroler 32 Bit..... | 10 |
| Datasheet Mikrokontroler..... | 10 |
| Bab II Metode Pengalamatan..... | 11 |
| Mode Pengalamatan Direct (Langsung) | 11 |
| Register SFR dan Pengalamatan-nya..... | 13 |
| Mode Pengalamatan register Indirect..... | 15 |
| Bab III. Memory | 22 |

Bab I. Pendahuluan Mikroprosesor

Pada tahun 1969 tim insinyur Jepang dari sebuah perusahaan BUSICOM datang ke Amerika Serikat memesan beberapa buah IC untuk membuat kalkulator. Mereka datang ke Perusahaan INTEL dan Marcian Hoff adalah orang yang dapat melayani permintaan itu. Sebab ia adalah orang yang berpengalaman bekerja di bidang komputer. Marcian Hoff memberi saran agar digunakannya IC yang bekerja berdasarkan program sehingga menjadi lebih sederhana.

Gagasan Marcian Hoff ini berhasil dan mikroprosesor pertama kali lahir. Untuk mewujudkan gagasan ini Marcian Hoff dibantu oleh Frederico Faggin. Dalam waktu sembilan bulan mereka sukses dan INTEL memperoleh hak atas penjualan temuan IC itu. Pada tahun 1971 keluar mikroprosesor seri 4004 dengan data bus 4 bit dengan kecepatan 6000 operasi per detik. Tidak lama kemudian Perusahaan Amerika CTC meminta INTEL dan Texas Instrumen untuk membuat mikroprosesor 8 bit. Akhirnya pada tahun 1972 INTEL dan Texas Instrumen berhasil menciptakan mikroprosesor 8008 dengan memori 16 Kbyte, 45 instruksi, dan kecepatan 300000 operasi per detik. Mikroprosesor ini menjadi pendahulu bagi semua mikroprosesor masa kini. INTEL terus melakukan penelitiannya sehingga pada bulan April 1974 menghasilkan mikroprosesor 8080 dengan kemampuan memori 64 Kbyte dan 75 instruksi. Keberhasilan INTEL diikuti oleh MOTOROLA dengan ciptaannya mikroprosesor 8 bit seri 6800, 6820, dan 6850.

Frederico Faggin meninggalkan INTEL membuat perusahaan sendiri diberi nama ZILOG dan pada tahun 1976 mengumumkan temuannya sebuah mikroprosesor seri Z-80. Mikroprosesor Z-80 dikembangkan dari rancangan mikroprosesor 8080. Mikroprosesor Z-80 kompatibel dan mampu menjalankan semua perintah yang ada pada 8080 sehingga Z-80 menjadi mikroprosesor yang tangguh pada waktu itu.

MikroprosesordanSistimMikroprosesor

2

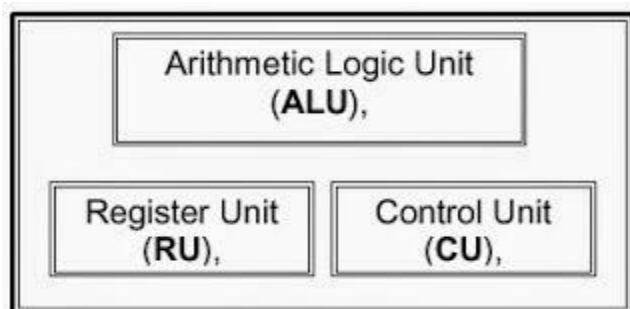


Mikroprosesor dalam perkembangan dunia komputer digital disebut sebagai Central Processing Unit (CPU). Mikroprosesor diberi sebutan CPU karena fungsinya sebagai unit pemroses pusat. CPU bekerja sebagai pusat pemroses dan pengendali bekerjanya sistim komputer. Sebagai salah satu jenis chip dari berbagai jenis chip yang telah diproduksi, mikroprosesor sering juga diberi sebutan Microprocessor Unit (MPU). CPU atau MPU merupakan komponen utama dari sebuah komputer.

Sebuah mikroprosesor secara internal dikonstruksi dari tiga bagian penting yaitu :

- Arithmetic Logic Unit (ALU),
- Register Unit (RU), dan
- Control Unit (CU)

Secara ilustratif konstruksi internal sebuah IC mikroprosesor dapat digambarkan seperti Gambar 1.



Gambar1. Blok diagram Internal sebuah Mikroprosesor

ALU adalah bagian yang bekerja melaksanakan operasi aritmetika dan operasi logika. Operasi aritmetika meliputi operasi penjumlahan (ADD atau ADD with Carry), pengurangan (SUB atau SUB

with Borrow), perkalian (MUL), dan pembagian (DIV). Sedangkan operasi logika meliputi operasi

logika AND, OR, XOR, COMPLEMEN, NEGATE. Untuk

mendukung pelaksanaan operasi pada ALU mikroprosesor

membutuhkan sejumlah register. Register adalah sebuah

memori tempat menyimpan data dan tempat menyimpan

hasil operasi. Register khusus yang bekerja sebagai tempat

penampung hasil operasi pengolahan pada ALU disebut

Akumulator. CU mengendalikan aliran data pada bus data

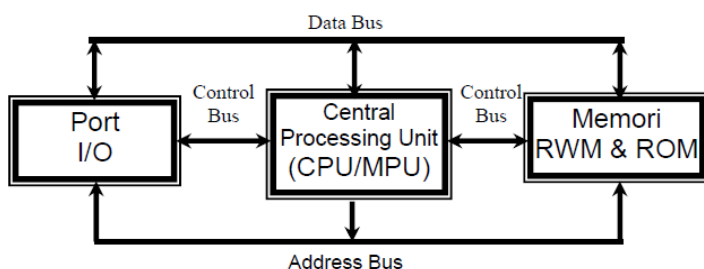
dan bus alamat, kemudian menafsirkan dan mengatur

sinyal yang terdapat pada bus pengendali.

Mikroprosesor sebagai sebuah chip tidak bisa bekerja sendiri. Mikroprosesor memerlukan unit

lain yaitu unit memori dan unit I/O. Dengan menggabungkan

CPU, Memory Unit, dan I/O unit terbangun sebuah sistim yang disebut dengan sistim mikroprosesor.



Gambar 2. Blok Diagram Sistim Mikroprosesor

CPU bekerja mengatur pengendalian dan proses alih data yang terjadi dalam sistim mikroprosesor.

Alih data berlangsung melalui saluran yang disebut dengan data bus. Alih data bisa terjadi dari memori

ke CPU atau dari I/O ke CPU atau sebaliknya dari CPU ke memori atau dari CPU ke I/O. Alih data

dari memori atau dari I/O ke CPU dikenal sebagai proses baca (READ). Alih data dari CPU ke memori atau alih data dari CPU ke I/O dikenal sebagai proses tulis (WRITE). Proses Read atau proses

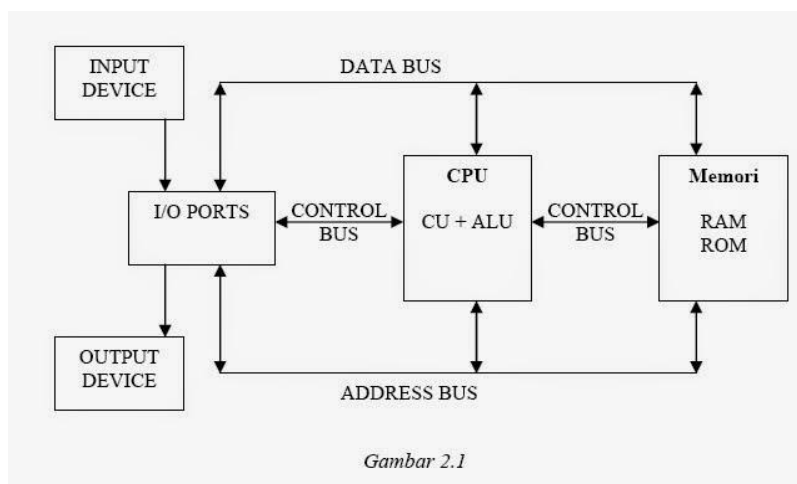
Write dikendalikan melalui saluran yang disebut dengan Control bus. Bus alamat bekerja mengatur

lokasi alamat memori atau I/O dari mana atau kemana data diambil atau dikirim.

Komputer mikro adalah salah satu contoh jenis sistim mikroprosesor. Untuk membangun fungsi sebagai komputer mikro, sebuah mikroprosesor juga harus dilengkapi dengan memori, biasanya memori program yang hanya bisa dibaca (Read Only Memory = ROM) dan memori yang bisa dibaca

dan ditulisi (Read Write Memory = RWM), dekoder memori, osilator, dan sejumlah peralatan input

output seperti port data seri dan paralel. Sebuah komputer mikro dapat digambarkan seperti gambar3. Jadi komputer mikro adalah sebuah sistim mikroprosesor.

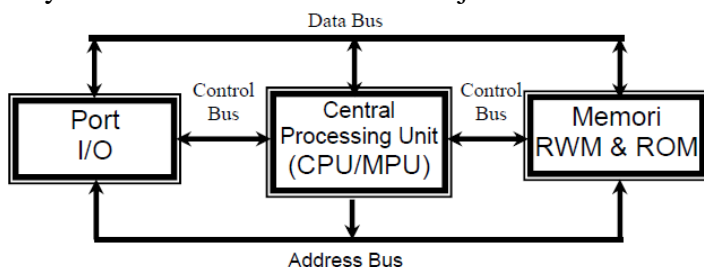


Gambar 3. Blok diagram komputer Mikro

Gambar 3 menunjukkan CPU bekerja bersama unit memori, unit I/O, peralatan input, dan peralatan output. Pokok dari penggunaan mikroprosesor adalah untuk melaksanakan program, mengambil data, membentuk kalkulasi, perhitungan atau manipulasi data, dan menyimpan hasil perhitungan pada peralatan penyimpan atau menampilkan hasil pada sebuah monitor atau cetak keras. Pada komputer mikro Mikroprosesor berkomunikasi dengan memori dan port I/O juga menggunakan saluran yang disebut dengan bus. Bus ada tiga jenis yaitu bus data, bus alamat, dan bus kendali seperti terlihat pada gambar3.

Mikrokontroler

Mikrokontroler adalah komputer mikro dalam satu chip tunggal. Mikrokontroler memadukan CPU, ROM, RWM, I/O paralel, I/O seri, counter-timer, dan rangkaian clock dalam satu chip tunggal. Mikrokontroler berbeda dengan mikroprosesor dalam beberapa hal. Mikrokontroler memadukan memori untuk menyimpan program atau data periperal I/O untuk berkomunikasi dengan alat luar. Pemanfaatan mikrokontroler saat sangat populer di bidang kendali dan instrumentasi elektronik. Hal ini terjadi karena mikrokontroler memiliki keunggulan dan kemudahan dalam penggunaannya. Disamping harganya yang sangat murah dibandingkan sistim lainnya. Gambar 4 menunjukkan blok diagram sebuah mikrokontroler.



Gambar 4 Diagram Mikrokontroler

Jika dibandingkan dengan mikroprosesor pada setiap mikrokontroler terdapat unit-unit CPU/MPU,

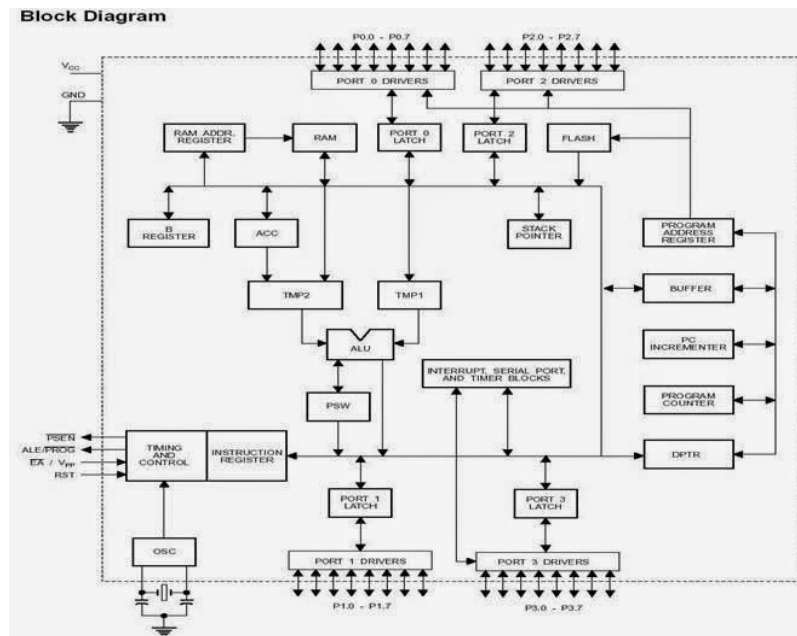
Memory Unit, I/O Unit yang dihubungkan oleh tiga bus yaitu bus data bus alamat dan bus kendali.

Dengan demikian maka sebuah mikrokontroler adalah sebuah sistem mikroprosesor dalam sebuah

chip tunggal. Dengan menambahkan peralatan input dan peralatan output sebuah mikrokontroler dapat berfungsi sebagai komputer mikro dalam satu chip tunggal.

Mikrokontroler sebagai sebuah chip telah mengalami perkembangan baik dari sisi arsitektur, teknologi dan kemampuannya. Struktur internal sebuah mikrokontroler AT89C51 bisa dilihat pada

gambar 5 sebagai contoh kasus.



Gambar 5. Blok Diagram IC Mikrokontroler AT89C51

Sama halnya dengan mikroprosesor, mikrokontroler adalah piranti yang dirancang untuk kebutuhan umum. Penggunaan pokok dari mikrokontroler adalah untuk mengontrol kerja mesin atau sistem menggunakan program yang disimpan pada sebuah ROM. Untuk melihat perbedaan konsep diantara mikroprosesor dan mikrokontroler di bawah ini ditunjukkan tabel perbandingan konfigurasi, arsitektur, dan set instruksi diantara mikroprosesor Z-80CPU dengan mikrokontroler AT89C51.

Tabel 1. Perbandingan Konfigurasi Z-80 CPU dan AT89C51

| No | Konfigurasi Pin | Mikroprosesor Z-80 CPU | Mikrokontroler 8051 |
|----|----------------------|---------------------------|------------------------|
| 1. | Jumlah pin | 40 | 40 |
| 2. | Jumlah pin alamat | 16 | 16 |
| 3. | Jumlah pin data | 8 | 8 |
| 4. | Jumlah pin interrupt | 2 | 2 |
| 5. | Pin I/O | - | 32 |
| 6. | Register 8 bit | 20 | 34 |
| 7. | Register 16 bit | 4 | 2 |
| 8. | Ukuran Stack | 64 K | 128 |
| 9. | ROM Internal | - | 4 Kbyte |

Sebagai catatan dari tabel 1 ini, jika dilakukan perbandingan tidaklah menunjukkan bahwa satu lebih baik dari lainnya. Kedua rancangan memiliki penekanan tujuan yang berbeda. Mikrokontroler dilengkapi dengan port I/O, memori, timer, dan counter.

Survey Mikrokontroler

Seperti halnya sebuah mikroprosesor, mikrokontroler juga berkembang sesuai rancangan dan model model aplikasinya. Mikrokontroler berdasarkan jumlah bit data yang dapat diolah dapat dibedakan dalam :

- Mikrokontroler 4 Bit
- Mikrokontroler 8 Bit
- Mikrokontroler 16 Bit
- Mikrokontroler 32 Bit

Mikrokontroler 4 Bit

Mikrokontroler 4 bit merupakan mikrokontroler dengan jumlah bit data terkecil. Mikrokontroler jenis ini diproduksi untuk meminimalkan jumlah pin dan ukuran kemasan.
Tabel 2 Contoh Mikrokontroler 4 bit

Mikrokontroler 8 Bit

Mikrokontroler 8 bit merupakan mikrokontroler yang paling banyak digunakan untuk pekerjaan pekerjaan perhitungan skala kecil. Dalam komunikasi data, Data ASCII serial juga disimpan dalam ukuran 8 bit. Kebanyakan IC memori dan fungsi logika dibangun menggunakan data 8 bit

sehingga interfacebus data menjadi sangat mudah dibangun. Penggunaan mikrokontroler 8 bit jauh lebih banyak dibandingkan dengan mikrokontroler 4 bit. Aplikasinya juga sangat pariatif mulai dari aplikasi kendali sederhana sampai kendali mesin berkecepatan tinggi.

Mikrokontroler16Bit

Keterbatasan-keterbatasan yang ada pada mikrokontroler 8 bit berkaitan dengan semakin kompleksnya pengolahan data dan pengendalian serta kecepatan tanggap/respon disempurnakan dengan menggunakan mikrokontroler 16 bit. Salah satu solusinya adalah dengan menaikkan kecepatan clock, dan ukuran data. Mikrokontroler 16 bit digunakan untuk mengatur tangan robot, dan aplikasi Digital Signal Processing (DSP).

Mikrokontroler32Bit

Mikrokontroler 32 bit ditargetkan untuk aplikasi Robot, Instrumen cerdas, Avionics, Image Processing, Telekomunikasi, Automobil, dan sebagainya.Program-program aplikasinya bekerja dengan sistim operasi.

Data Sheet Mikrokontroler

Data sheet sebuah mikrokontroler memuat informasi tentang Feature, deskripsi, konfigurasi pin, blok diagram, deskripsi fungsi masing-masing pin, organisasi memori termasuk register, pemrograman memori, karakteristik AC dan karakteristik DC. Gambar 6 menunjukkan salah contoh bentuk bagian depan data sheet mikrokontroler AT89S51. Data sheet dapat diakses dari berbagai sumber seperti bukuteks, kumpulan buku data sheet, berbagai situs di internet.

Banyaknya bentuk pengalamatan pada cpu sehingga membuat kita mempunyai pilihan untuk melakukan peng-akses-an RAM yang paling efisien dalam desain program kita.

Terdapat 5 mode pengalamatan pada mikon. Seperti berikut:

(1) Immediate (segera)

- (2) Register
- (3) Direct (langsung)
- (4) Register Indirect (tidak langsung)
- (5) Indexed

BAB II. Mode Pengalamatan

Mode pengalamatan Immediate

Adalah sebuah bentuk pengalamatan paling sederhana. Tidak ada referensi kemana-mana. Operand (data yang akan diolah) benar-benar ada dalam instruksinya. Keuntungan dari mode pengalamatan ini adalah Menghemat siklus instruksi sehingga proses keseluruhan menjadi cepat (karena tidak memerlukan referensi memori).

. Contohnya berikut ini

```
MOV A,#25h    ;isi 25h pada A
MOV R4,#62    ;isi bilangan desimal 62 pada R4
MOV B,#40h    ;isi B dengan 40h
MOV DPTR,#4521h ;DPTR = 4521h
```

Karena register DPTR adalah 16-bit, dia dapat diakses pula sebagai dua register 8-bit, yaitu DPH (DPTR High byte) dan DPL (DPTR Low byte). Lihat contoh di bawah ini.

```
MOV DPTR,#2550h
```

Mode Pengalamatan Register

Mode ini melibatkan dan menggunakan register sebagai tempat untuk menyimpan atau tempat data yang dimanipulasi. Contoh dari mode pengalamatan register adalah sebagai berikut ini.

```
MOV A,R0    ;salin isi R0 pada A
```

```

MOV R2,A ;salin isi A pada R2
MOV A,R5 ;salin isi R5 pada A
MOV A,R7 ;salin isi R7 pada A
MOV R5,B ;salin isi B pada R5

```

Mode Pengalamatan Direct (Langsung)

Adalah sebuah mode pengalamatan sederhana karena hanya memerlukan satu referensi memory sebagai operannya. Artinya operand (data yang akan diolah) diletakkan di memory sehingga lebih besar dari operand yang "hanya" tertulis di instruksinya.

Contoh mode pengalamatan langsung (direct addressing) adalah sebagai berikut:

```

MOV R0,40h ;simpan isi 40h pada R0 (immediate)
MOV 56h,A ;simpan isi A pada lokasi 56h (langsung)
MOV R4,7Fh ;salin isi lokasi 74h ke R4 (Regsiter)

```

Register SFR dan Pengalamatan-nya

Dari penjelasan tentang register , kita tahu bahwa R0 s/d R7 adalah bagian dari memory RAM. Lalu dimana tempatnya untuk register register semacam A, B, PSW, dan DPTR , mereka tersimpan pada suatu register kusus yang disebut sebagai SFR (Special Function Registers). Umumnya register-register tersebut dibuat bukan untuk tempat kita menyimpan data, namun register-register tersebut memiliki fungsi terutama untuk mengendalikan peralatan-peralatan yang terdapat pada chip, misalnya Timer, Serial, Kontrol Power, Port dan lain-lain. Dan semua register-register tersebut juga memiliki nama tertentu. Misalnya register alamat E0h, disebut juga register A, dan register B memiliki alamat F0h. Dalam Tabel 5-1 kita akan melihat secara lengkap register-register yang termasuk dalam golongan SFR ini.

```

MOV A,#55h ;isi A dengan 55h
MOV 0E0h,#55h ;sama artinya dgn di atas

MOV B,#25h ;isi A dengan 25h

```

MOV 0F0h,#25h ;sama artinya dgn di atas

MOV A,R2 ;Salin R2 pada A

MOV 0E0h,R2 ;sama artinya dgn di atas

MOV B,R0 ;Salin R0 pada B

MOV 0F0h,R0 ;sama artinya dgn di atas

Tabel 5-1 adalah daftar dari Special Function Register 8051 dan alamat-alamatnya. Hal berikut ini yang harus diperhatikan untuk mengalamat register SFR ini.

1. SFR memiliki alamat 80h s/d FFh. Kesemuanya hanya bisa diakses dengan cara mode pengalamat langsung (Direct). Beberapa diantara juga bisa dialamati secara bit. Sama persis dengan semua lokasi RAM yaitu 00 s/d 7Fh yang juga bisa dilamati dengan mode pengalamatan langsung.
2. Tidak semua lokasi dalam SFR digunakan, karena tidak ada peralatan yang dihubungkan untuk lokasi tersebut. Lokasi yang tidak digunakan pada lokasi SFR 80h s/d FFh tersebut dibiarkan kosong, dan kita diminta untuk tidak memodifikasi (menulis) nya. Karena mungkin pada produk yang lebih baru, lokasi-lokasi tersebut digunakan untuk peralatan yang baru dengan fungsi-fungsi tertentu.

Tabel 5-1 Alamat-alamat dari SFR (Special Function Register)

| Simbol | Nama | Alamat |
|--------|---------------------|--------|
| ACC * | Accumulator | 0E0h |
| B * | Register B | 0F0h |
| PSW * | Program Status Word | 0D0h |
| SP | Stack Pointer | 81h |

| | | |
|--------|-------------------------------|------|
| DPTR | Data Pointer 2-bytes | |
| DPL | Low byte | 82h |
| DPH | High byte | 83h |
| P0 * | Port 0 | 80h |
| P1 * | Port 1 | 90h |
| P2 * | Port 2 | 0A0h |
| P3 * | Port 3 | 0B0h |
| IP * | Kontrol Prioritas Interupsi | 0B8h |
| IE * | Kontrol Enable Interupsi | 0A8h |
| TMOD | Kontrol Mode Timer/.Counter | 89h |
| TCON | Kontrol Timer/Counter | 88h |
| T2CON | Kontrol Timer/Counter 2 | 0C8h |
| T2MOD | Kontrol Mode Timer/.Counter 2 | 0C9h |
| TH0 | Timer/Counter 0 high byte | 8Ch |
| TL0 | Timer/Counter 0 Low byte | 8Ah |
| TH1 | Timer/Counter 1 high byte | 8Dh |
| TL1 | Timer/Counter 1 Low byte | 8Bh |
| TH2 | Timer/Counter 2 high byte | 0CDh |
| TL2 | Timer/Counter 2 Low byte | 0CCh |
| RCAP2H | T/C 2 Capture high byte | 0CBh |
| RCAP2L | T/C 2 Capture low byte | 0Cah |
| SCON * | Serial Control | 98h |
| SBUF | Serial data buffer | 99h |
| PCON | Power Control | 87h |

* = Bit Addressable (dibahas pada bab 8)

Pada mode pengalamatan langsung (direct), kita harus perhatikan bahwa alamat data yang bisa ditangani dalam mode ini adalah dalam ukuran byte. Yaitu dengan alamat 00 s/d FFh. Sehingga mode pengalamatan ini hanya mampu untuk mengamati lokasi-lokasi tersebut. Sebagian untuk RAM dan sebagian lagi untuk SFR.

Mode Pengalamatan register Indirect

Adalah sebuah mode pengalamatan yang memerlukan lebih dari satu referensi (baik memory atau register) untuk mengambil operand-nya. dalam mode ini, register digunakan untuk menunjuk lokasi dari register yang lain. Kita dapat mengakses seluruh lokasi RAM yang lokasinya ditunjukkan oleh isi register.

Contoh mode pengalamatan tidak langsung adalah:

MOV A,R0 ;pindahkan isi register R0 pada A

MOV A,@R0 ;pindahkan isi lokasi yg ditunjuuk R0 pada A

MOV R1,B ;pindahkan isi B pada R1

MOV @R1,B ;pindahkan isi B pd lokasi yg ditunjuuk R1

Kelebihan Mode Pengalamatan InDirect

Salah satu kelebihan dari pengalamatan register InDirect ini, kita dapat mengakses data secara dinamis jauh lebih baik dari mode Pengalamatan Direct karnaakan menjadi jauh lebih efisien dan hanya dimungkinkan dengan menggunakan mode pengalamatan tidak langsung.karena dapat menggunakan looping yang tidak dapat digunakan dengan mode pengalamatan langsung. Nah inilah perbedaan penting dari dua mode ini.

Mode Pengalamatan Ter-Index dan mengakses On-Chip ROM

Mode pengalamatan ter-index digunakan secara luar untuk mengakses element data (of look-up table entries) dalam lokasi ROM program dalam 8051. Instruksi yang digunakan untuk hal itu adalah “MOVC A,@A+DPTR”. Register 16-bit pada DPTR dan register A digunakan sebagai pembentuk alamat dari element data yang tersimpan dalam ROM program. Karena data yang hendak diakses adalah data kode yang tersimpan dalam ROM Program, maka simbol MOVC digunakan untuk membedakan dengan MOV. “C” yang berarti adalah Code. Instruksi ini adalah jumlah dari isi register A dan isi DPTR kemudian menjadi penunjuk (pointer) 16-bit yang dapat mengakses seluruh jangkauan data 16-bit dalam CPU.

Banyaknya bentuk pengalamatan pada cpu sehingga membuat kita mempunyai pilihan untuk melakukan peng-akses-an RAM yang paling efisien dalam desain program kita.

Terdapat 5 mode pengalamatan pada mikon. Seperti berikut:

- (1) Immediate (segera)
- (2) Register
- (3) Direct (langsung)
- (4) Register Indirect (tidak langsung)
- (5) Indexed

Mode pengalamatan Immediate

Adalah sebuah bentuk pengalamatan paling sederhana. Tidak ada referensi kemana-mana. Operand (data yang akan diolah) benar-benar ada dalam instruksinya. Keuntungan dari mode pengalamatan ini adalah Menghemat siklus instruksi sehingga proses keseluruhan menjadi cepat (karena tidak memerlukan referensi memori).

. Contohnya berikut ini

MOV A,#25h ;isi 25h pada A

MOV R4,#62 ;isi bilangan desimal 62 pada R4

MOV B,#40h ;isi B dengan 40h

MOV DPTR,#4521h ;DPTR = 4521h

Karena register DPTR adalah 16-bit, dia dapat diakses pula sebagai dua register 8-bit, yaitu DPH (DPTR High byte) dan DPL (DPTR Low byte). Lihat contoh di bawah ini.

```
MOV DPTR,#2550h
```

Mode Pengalamatan Register

Mode ini melibatkan dan menggunakan register sebagai tempat untuk menyimpan atau tempat data yang dimanipulasi. Contoh dari mode pengalamatan register adalah sebagai berikut ini.

```
MOV A,R0 ;salin isi R0 pada A
```

```
MOV R2,A ;salin isi A pada R2
```

```
MOV A,R5 ;salin isi R5 pada A
```

```
MOV A,R7 ;salin isi R7 pada A
```

```
MOV R5,B ;salin isi B pada R5
```

Mode Pengalamatan Direct (Langsung)

Adalah sebuah mode pengalamatan sederhana karena hanya memerlukan satu referensi memory sebagai operannya. Artinya operand (data yang akan diolah) diletakkan di memory sehingga lebih besar dari operand yang "hanya" tertulis di instruksinya.

Contoh mode pengalamatan langsung (direct addressing) adalah sebagai berikut:

```
MOV R0,40h ;simpan isi 40h pada R0 (immediate)
```

```
MOV 56h,A ;simpan isi A pada lokasi 56h (langsung)
```

```
MOV R4,7Fh ;salin isi lokasi 74h ke R4 (Regsiter)
```

Register SFR dan Pengalamatan-nya

Dari penjelasan tentang register , kita tahu bahwa R0 s/d R7 adalah bagian dari memory RAM. Lalu dimana tempatnya untuk register register semacam A, B, PSW, dan DPTR , mereka tersimpan pada suatu register kusus yang disebut sebagai SFR (Special Function Registers). Umumnya register-register tersebut dibuat bukan untuk tempat kita menyimpan data, namun register-register tersebut memiliki fungsi terutama untuk mengendalikan peralatan-peralatan yang terdapat pada chip, misalnya Timer, Serial, Kontrol Power, Port dan lain-lain. Dan semua register-register tersebut juga memiliki nama tertentu. Misalnya register alamat E0h, disebut juga register A, dan register B memiliki alamat F0h. Dalam Tabel 5-1 kita akan melihat secara lengkap register-register yang termasuk dalam golongan SFR ini.

```
MOV A,#55h    ;isi A dengan 55h
MOV 0E0h,#55h ;sama artinya dgn di atas

MOV B,#25h    ;isi A dengan 25h
MOV 0F0h,#25h ;sama artinya dgn di atas

MOV A,R2      ;Salin R2 pada A
MOV 0E0h,R2   ;sama artinya dgn di atas

MOV B,R0      ;Salin R0 pada B
MOV 0F0h,R0   ;sama artinya dgn di atas
```

Tabel 5-1 adalah daftar dari Special Function Register 8051 dan alamat-alamatnya. Hal berikut ini yang harus diperhatikan untuk mengalamat register SFR ini.

1. SFR memiliki alamat 80h s/d FFh. Kesemuanya hanya bisa diakses dengan cara mode pengalamat langsung (Direct). Beberapa diantara juga bisa dialamati secara bit. Sama persis dengan semua lokasi RAM yaitu 00 s/d 7Fh yang juga bisa dilamati dengan mode pengalamatan langsung.

2. Tidak semua lokasi dalam SFR digunakan, karena tidak ada peralatan yang dihubungkan untuk lokasi tersebut. Lokasi yang tidak digunakan pada lokasi SFR 80h s/d FFh tersebut dibiarkan kosong, dan kita diminta untuk tidak memodifikasi (menulis) nya. Karena mungkin pada produk yang lebih baru, lokasi-lokasi tersebut digunakan untuk peralatan yang baru dengan fungsi-fungsi tertentu.

Tabel 5-1 Alamat-alamat dari SFR (Special Function Register)

| Simbol | Nama | Alamat |
|--------|-------------------------------|--------|
| ACC * | Accumulator | 0E0h |
| B * | Register B | 0F0h |
| PSW * | Program Status Word | 0D0h |
| SP | Stack Pointer | 81h |
| DPTR | Data Pointer 2-bytes | |
| DPL | Low byte | 82h |
| DPH | High byte | 83h |
| P0 * | Port 0 | 80h |
| P1 * | Port 1 | 90h |
| P2 * | Port 2 | 0A0h |
| P3 * | Port 3 | 0B0h |
| IP * | Kontrol Prioritas Interupsi | 0B8h |
| IE * | Kontrol Enable Interupsi | 0A8h |
| TMOD | Kontrol Mode Timer/.Counter | 89h |
| TCON | Kontrol Timer/Counter | 88h |
| T2CON | Kontrol Timer/Counter 2 | 0C8h |
| T2MOD | Kontrol Mode Timer/.Counter 2 | 0C9h |

| | | |
|--------|---------------------------|------|
| TH0 | Timer/Counter 0 high byte | 8Ch |
| TL0 | Timer/Counter 0 Low byte | 8Ah |
| TH1 | Timer/Counter 1 high byte | 8Dh |
| TL1 | Timer/Counter 1 Low byte | 8Bh |
| TH2 | Timer/Counter 2 high byte | 0CDh |
| TL2 | Timer/Counter 2 Low byte | 0CCh |
| RCAP2H | T/C 2 Capture high byte | 0CBh |
| RCAP2L | T/C 2 Capture low byte | 0Cah |
| SCON * | Serial Control | 98h |
| SBUF | Serial data buffer | 99h |
| PCON | Power Control | 87h |

* = Bit Addressable (dibahas pada bab 8)

Pada mode pengalamatan langsung (direct), kita harus perhatikan bahwa alamat data yang bisa ditangani dalam mode ini adalah dalam ukuran byte. Yaitu dengan alamat 00 s/d FFh. Sehingga mode pengalamatan ini hanya mampu untuk mengalami lokasi-lokasi tersebut. Sebagian untuk RAM dan sebagian lagi untuk SFR.

Mode Pengalamatan register Indirect

Adalah sebuah mode pengalamatan yang memerlukan lebih dari satu referensi (baik memory atau register) untuk mengambil operand-nya. dalam mode ini, register digunakan untuk menunjuk lokasi dari register yang lain. Kita dapat mengakses seluruh lokasi RAM yang lokasinya ditunjukkan oleh isi register.

Contoh mode pengalamatan tidak langsung adalah:

MOV A,R0 ;pindahkan isi register R0 pada A

MOV A,@R0 ;pindahkan isi lokasi yg ditunjuuk R0 pada A

MOV R1,B ;pindahkan isi B pada R1

MOV @R1,B ;pindahkan isi B pd lokasi yg ditunjuuk R1

Kelebihan Mode Pengalamatan InDirect

Salah satu kelebihan dari pengalamatan register InDirect ini, kita dapat mengakses data secara dinamis jauh lebih baik dari mode Pengalamatan Direct karnaakan menjadi jauh lebih efisien dan hanya dimungkinkan dengan menggunakan mode pengalamatan tidak langsung.karena dapat menggunakan looping yang tidak dapat digunakan dengan mode pengalamatan langsung. Nah inilah perbedaan penting dari dua mode ini.

Mode Pengalamatan Ter-Index dan mengakses On-Chip ROM

Mode pengalamatan ter-index digunakan secara luar untuk mengakses element data (of look-up table entries) dalam lokasi ROM program dalam 8051. Intruksi yang digunakan untuk hal itu adalah “MOVC A,@A+DPTR”. Register 16-bit pada DPTR dan register A digunakan sebagai pembentuk alamat dari element data yang tersimpan dalam ROM program. Karena data yang hendak diakses adalah data kode yang tersimpan dalam ROM Program, maka simbol MOVC digunakan untuk membedakan dengan MOV. “C” yang berarti adalah Code. Instruksi ini adalah jumlah dari isi register A dan isi DPTR kemudian menjadi penunjuk (pointer) 16-bit yang dapat mengakses seluruh jangkauan data 16-bit dalam CPU.

BAB III

MEMORI

3.1 Pengertian Memori

Memori merupakan sumberdaya yang bersangkutan denganruang atau letak selain sebagai pengingat, memori juga bertindakselaku penyimpanan (storage). Ada isi memori yang sudah bersifattetap, sehingga tidak dapat diubah lagi, ada juga memori yangtidak bersifat tetap. Keluasan ruang memori menyebabkanpencarian bagian atau letak memori tertentu menjaditidak mudah.Berdasarkan atas keperluan penyimpanan informasi,dikenal adanya alamat memori mutlak atau alamat memori fisik.Berdasarkan atas kemudahan tataolah, dikenal adanya alamatmemori relatif atau alamat memori logika. Pada memori kerja,

alamat mutlak adalah alamat fisik pada memori kerja, sedangkan alamat relatif adalah alamat memori yang secara tidak langsung menunjuk ke salah satu sel pada memori kerja.

3.2 Pemuatan Informasi ke Memori

Memori yang mempunyai fungsi utama sebagai penyimpan informasi atau data, maka sudah barang tentu perlu diketahui teknik atau cara pemuatan informasi ke ruang memori yang digunakan. Beberapa cara pemuatan data ke memori adalah Pemuatan Mutlak. Pemuatan informasi ke memori-kerja, alamat yang tercantum di dalam tata olah sama dengan alamat yang ditempatinya di dalam memori-kerja.

Pemuatan Relokasi

Kondisi dimana pemuatan informasi ke memori-kerja, alamat yang tercantum di dalam tata olah tidak mesti sama dengan alamat yang ditempatinya di dalam memori-kerja. Pemuatan Sambung (linker) Menyambungkan suatu informasi ke informasi lain di dalam memori-kerja. Pemuatan sambung sering digunakan pada tata olah atau penggalan tata olah yang tersimpan di dalam pustaka (library). Pemuatan dinamik (pemuatan tumpang atau overlay) Jika ukuran tata olah itu melampaui ukuran ruang memori kerja, tata olah dapat dipenggal ke dalam sejumlah segmen. Segmen itulah yang kemudian dimuat ke dalam memori-kerja. Pelaksanaan pekerjaan berlangsung segmen demi segmen.

3.3 Jenis-Jenis memori

Setiap program dan data yang sedang diproses oleh prosesor akan disimpan di dalam memori fisik. Data yang disimpan dalam memori fisik bersifat sementara, karena data yang disimpan di dalamnya akan tersimpan selama masih dialiri daya listrik. Ketika sumber daya listrik dimatikan atau direset, data yang disimpan dalam memori fisik akan hilang. Memori biasa dibedakan menjadi dua macam: ROM dan RAM. Selain itu, terdapat pula memori yang disebut Cache Memory.

3.3.1 ROM (Read Only Memory)

ROM adalah kependekan dari Read Only Memory, yaitu perangkat keras berupa chip memori semikonduktor yang isinya

hanya dapat dibaca. Jenis memori ini datanya hanya bisa dibaca dan tidak bisa ditulis secara berulang-ulang. Memori ini berjenis non-volatile, artinya data yang disimpan tidak mudah menguap (hilang) walaupun catu dayanya dimatikan. Karena itu memori ini biasa digunakan untuk menyimpan program utama dari suatu sistem, sehingga lebih aman karena tidak takut datanya hilang ditengah jalan.

3.3.1.1 Jenis-jenis ROM

Sampai sekarang dikenal beberapa jenis ROM yang pernah beredar dan terpasang pada komputer, antara lain PROM,

EPROM dan EEPROM.

a. PROM (Programmable Read-Only-Memory)

Jika isi ROM ditentukan oleh vendor, PROM dijual dalam keadaan kosong dan kemudian dapat diisi dengan program oleh pemakai. Setelah diisi dengan program, isi PROM tak bisa dihapus.

b. EPROM (Erasable Programmable Read-Only-Memory)

Berbeda dengan PROM, isi EPROM dapat dihapus diprogram. Penghapusan dilakukan dengan menggunakan sinar ultraviolet.

c. EEPROM (Electrically Erasable Programmable Read-Only Memory)

EEPROM dapat menyimpan data secara permanen, tetapi isinya masih bisa dihapus secara elektrik melalui program. Salah satu jenis EEPROM adalah Flash Memory. Flash Memory biasa digunakan pada kamera digital, konsol video game, dan chip BIOS.

3.3.2 RAM (Random Access Memory)

RAM merupakan perangkat keras berupa chip semikonduktor yang sifat memorinya dapat dibaca dan ditulisi. Jenis memori ini data yang disimpan bisa hilang ditengah proses karena sifat memori ini adalah berjenis volatile, artinya data yang disimpan akan hilang (menguap) jika catu dayanya dimatikan.

Berdasarkan cara kerja dari RAM, maka terdapat dua jenis RAM yaitu statik dan dinamik. RAM dinamik tersusun oleh sel-sel yang menyimpan data sebagai muatan listrik pada kapasitor. Untuk Dynamic RAM (DRAM) dapat dikategorikan dalam beberapa tipe, diantaranya adalah

- Fast Page Mode DRAM (FPM DRAM)
- Extended Data Output DRAM (EDO DRAM)
- Synchronous DRAM (SDRAM)
- Rambus DRAM (RDRAM)
- Double Data Rate SDRAM (DDR SDRAM)

Sedangkan RAM yang mempunyai fungsi khusus untuk video diantaranya adalah

- Video RAM (VRAM)
- Windows RAM (WRAM)
- Synchronous Graphic RAM (SGRAM)

Untuk RAM statik menyimpan informasinya dengan cara menyimpan nilai-nilai biner dari suatu informasi dengan menggunakan konfigurasi gerbang logika flip-flop, jenis RAM ini adalah Static RAM (SRAM).

Berdasarkan jenis modulnya dapat dibagi sebagai berikut:

- Single Inline Memory Module (SIMM)
- Double Inline Memory Module (DIMM)
- Rambus Inline Memory Module (RIMM)

Gambar 3.5 Single inline memory module (SIMM)

Berdasarkan jumlah pin nya RAM ada yang mempunyai 30pin, 72 pin, dan 168 pin, sedangkan untuk kecepatannya sudah mencapai nanosecond. Kecepatan memori utama sangat rendah dibandingkan kecepatan prosesor modern. Untuk performa yang baik, prosesor tidak dapat membuang waktunya dengan menunggu untuk mengakses instruksi dan data pada memori utama. Karenanya sangat penting untuk memikirkan suatu skema yang mengurangi waktu dalam mengakses informasi. Karena kecepatan memori utama dibatasi oleh batasan elektronik dan packaging, maka solusinya harus dicari pada sistem arsitektur yang berbeda. Solusi yang efisien adalah menggunakan memori cache cepat yang sebenarnya membuat memori utama tampak lebih cepat bagi prosesor daripada sebenarnya. Jenis memori ini adalah memori berkapasitas terbatas, berkecepatan tinggi yang lebih mahal dibanding memori utama. Cache memori terletak diantara memori utama dan register CPU, dan berfungsi agar CPU tidak langsung mengacu ke memori utama tetapi di cache memori yang kecepatan aksesnya lebih tinggi. Metode ini akan meningkatkan kinerja system. Buffer berkecepatan tinggi yang digunakan untuk menyimpan data yang diakses pada saat itu dan data yang berdekatan dalam memori utama. Waktu akses memori cache lebih cepat 5 – 10 kali dibandingkan memori utama. Dahulu cache disimpan di luar prosesor dan dapat ditambahkan. Untuk meningkatkan kinerja, saat ini cache ditanamkan di prosesor.

3.4 Karakteristik Memori

Sebelum menggunakan memori sebagai penyimpanan informasi, perlu dimengerti tentang beberapa karakteristik dari memori tersebut. Memori yang digunakan untuk mendukung kinerja prosesor memiliki beberapa karakteristik yaitu lokasi, kapasitas, satuan transfer, metode akses, kinerja, tipe fisik, karakteristik fisik.

3.4.1 Lokasi Memori

Berdasarkan lokasinya memori dapat berlokasi di internal dan eksternal. Memori yang berada di dalam dapat diakses oleh prosesor tanpa melalui input output. Sebagai contoh register, cache

memori, dan main memori. Untuk memori eksternal cara mengaksesnya harus melalui peralatan input output. Contohnya adalah hardisk, flasdisk, CDROM.

3.4.2 Kapasitas Memori

Kapasitas memori adalah kemampuan memori untuk menyimpan informasi dalam satuan tertentu. Satuan penyimpanan memori adalah byte atau word. Sebagaimana kita ketahui 1 byte adalah 8 bit, sedangkan untuk word bisa 8, 16, 32bit, tergantung pada pembuat prosesornya.

3.4.3 Satuan Transfer

Satuan transfer untuk memori internal adalah banyaknya bit yang dapat dibaca atau ditulis dari atau ke memori dalam setiap detik. Banyaknya bit tersebut setara dengan banyaknya jalur data yang terhubung ke memori (lebar bus). Biasanya sebanyak satu word tetapi dapat lebih banyak lagi seperti 32, 64 atau 128. Jika memori eksternal digunakan satuan block yang ukurannya lebih dari satu word. Untuk satuan alamat addressable unit adalah ukuran memori terkecil yang dapat diberi alamat sendiri. Besarnya satuan alamat tergantung pembuat prosesor.

Tugas

1. Jelaskan pengertian dari memori ?
2. Sebutkan dan jelaskan cara-cara pemuatan informasi ke dalam memori ?
3. Sebutkan dan jelaskan jenis-jenis dari memori ?
4. Apa yang dapat anda jelaskan tentang volatile dan nonvolatile?
5. Untuk meningkatkan kinerja mengakses informasi dari memori utama ditambahkan cache memori antara memori utama dan register CPU, jelaskan tentang cache memori ?.