

MODUL PEMBELAJARAN

Organisasi dan Arsitektur Komputer



Dosen Pengampu :

Bambang Kurniawan, A.Md, S.Kom, M.Kom

**FAKULTAS ILMU KOMPUTER
UNIVERSITAS HANG TUAH PEKANBARU**

DAFTAR ISI

PRAKATA.....	ii
DAFTAR ISI.....	iii
BAB 1. PENDAHULUAN ORGANISASI DAN ARSITEKTUR KOMPUTER	1
A. Pengertian Organisasi dan Arsitektur Komputer	1
B. Pengertian Struktur dan Fungsi Komputer	2
C. Ringkasan Eksekutif	6
BAB 2. EVOLUSI DAN PERFORMA (KINERJA) KOMPUTER	8
A. Komputer Generasi Pertama	8
B. Komputer Generasi Kedua	11
C. Komputer Generasi Ketiga.....	11
D. Komputer Generasi Selanjutnya.....	13
E. Desain Kinerja	14
F. Ringkasan Eksekutif	15
G. Soal-soal	15
BAB 3. REPRESENTASI DATA KOMPUTER	17
A. Representasi Data Komputer.....	17
B. Sistem Bilangan & Konversi Bilangan Biner, Desimal, Heksadesimal	17
C. Ringkasan Eksekutif	23
BAB 4. FUNGSI DAN INTERKONEKSI KOMPUTER SECARA GENERAL.....	27
A. Komponen Utama Central Processing Unit (CPU)	27
B. Fungsi Komponen-komponen utama CPU.....	29
C. Instruksi dan Kode Operasi.....	31
BAB 5. INTERUPSI.....	39
A. Pengertian dan Fungsi Interupsi	39
B. Proses Interupsi.....	39
C. Interupsi ganda (<i>multiple interrupt</i>)	40
D. Soal dan Pembahasan.....	42
BAB 6. SET INSTRUKSI	43
A. Karakteristik Instruksi Mesin	43

B. Tipe Instruksi dan Jumlah Alamat.....	44
C. Desain set instruksi	46
D. Format dan Mode Pengalamatan.....	47
BAB 8. MEMORI.....	49
A. Pengertian Memori.....	49
B. Karakter Sistem Memori	50
C. Hirarki Memori	52
D. Memori internal	54
E. Cache Memori.....	56
F. Memori Eksternal.....	60
BAB 9. Unit Masukan dan Keluaran (I/O Device).....	62
A. Pengertian dan Fungsi Modul I/O	62
B. Struktur dan Teknik Modul I/O.....	63
DAFTAR PPUSTAKA.....	67

BAB 1. PENDAHULUAN ORGANISASI DAN ARSITEKTUR KOMPUTER

Tujuan Pembelajaran

1. Mahasiswa mampu menjelaskan tentang organisasi komputer
2. Mahasiswa mampu menjelaskan perbedaan utama organisasi dan arsitektur komputer
3. Mahasiswa mampu menjelaskan struktur dan fungsi utama komputer
4. Mahasiswa mampu menjelaskan konsep dasar operasi komputer

A. Pengertian Organisasi dan Arsitektur Komputer

Arsitektur komputer merujuk pada atribut sistem yang visible untuk programmer atau, dengan kata lain, atribut-atribut yang memiliki dampak langsung pada eksekusi logis suatu program. Contoh dari atribut meliputi set instruksi, jumlah bit yang digunakan untuk mewakili berbagai jenis data (misal : angka, karakter), mekanisme I/O(Input/output), dan teknik untuk pengalamatan memori.

Organisasi Komputer merujuk pada unit-unit operasional beserta interkoneksi antar komponen penyusun komputer yang menyatakan spesifikasi arsitektur tertentu. Atribut organisasi meliputi detail hardware yang transparan terhadap programmer, seperti sinyal kontrol, *interface* (antarmuka) serta teknologi memori yang digunakan

Perbedaan pengertian antara arsitektur dan organisasi menjadi penting karena banyak dari pabrikan komputer menawarkan family (keluarga) dari model komputer. Sebagai contoh semua keluarga Intel x86 memiliki arsitektur dasar yang sama dan keluarga IBM System/370 memiliki arsitektur dasar yang sama karena hal tersebut berkaitan dengan kompatibilitas kode yang dibuat. Kode-kode komputer dari keluarga yang sama memiliki fitur "*backwards compatibility mode*" yaitu kemampuan kode untuk mengadaptasi sistem sebelumnya. Meskipun demikian, family computer yang memiliki persamaan arsitektur tidak berarti memiliki persamaan organisasi. Komputer yang diproduksi dengan versi yang berbeda memiliki organisasi yang tidak sama antara satu dengan lainnya. Perkembangan teknologi tidak hanya mempengaruhi organisasi sebuah komputer tetapi juga menghasilkan pengenalan arsitektur yang lebih *powerful* dan kompleks.

B. Pengertian Struktur dan Fungsi Komputer

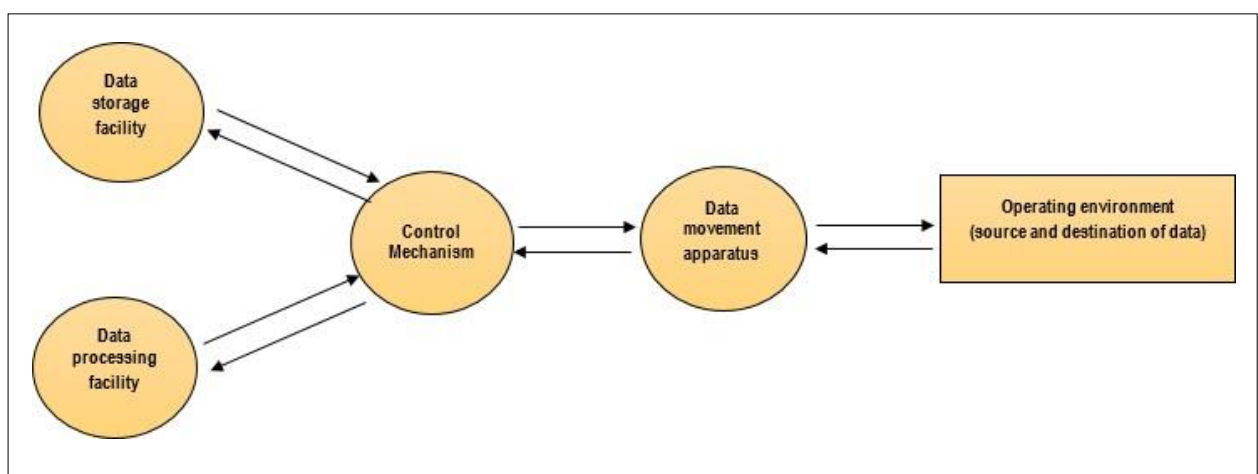
Komputer merupakan sebuah sistem yang kompleks dan terdiri atas jutaan komponen dasar elektronik. Oleh karena itu untuk memudahkan dalam pendeskripsian perlu dikenali sifat hirarkis dari sistem tersebut. Sebuah sistem hirarkis adalah seperangkat subsistem yang saling terkait, yang masing-masing akan berakhir, pada gilirannya, hirarki dalam struktur akan mencapai beberapa tingkat terendah dari subsistem dasar. Perilaku pada setiap level hierarki hanya bergantung pada sistem di tingkat yang lebih rendah berikutnya dan telah disederhanakan. Pada setiap tingkatan tersebut, desainer menghubungkan antara struktur dan fungsi sebuah komputer. Struktur adalah cara di mana komponen saling berhubungan satu sama lain dan fungsi merupakan operasi dari masing-masing komponen sebagai bagian dari struktur.

1. Fungsi

Semua fungsi komputer meliputi 4 hal yaitu :

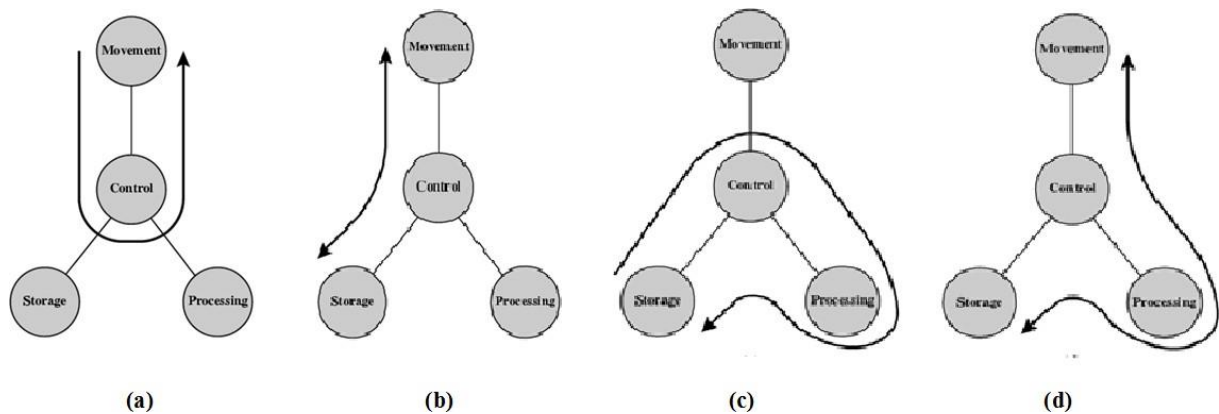
- Data processing (Pengolahan data)
- Data storage (Penyimpanan data)
- Data movement (Pergerakan data)
- Control (Kontrol)

Dalam mendeskripsikan sebuah fungsi komputer ada dua pendekatan yaitu dari level paling rendah ke level tertinggi atau sebaiknya dari gambaran paling umum kemudian mengerucut ke bagian-bagian sub sistem. Pada pembelajaran ini akan digunakan pendekatan yang kedua. Gambaran fungsional dari sebuah komputer tampak pada gambar 1 berikut.



Gambar 1.1 Gambaran fungsional dari sebuah komputer

Jika gambar 1 tersebut diatas dipilah sesuai dengan masing-masing fungsi komputer, maka akan tampak sebagai berikut.



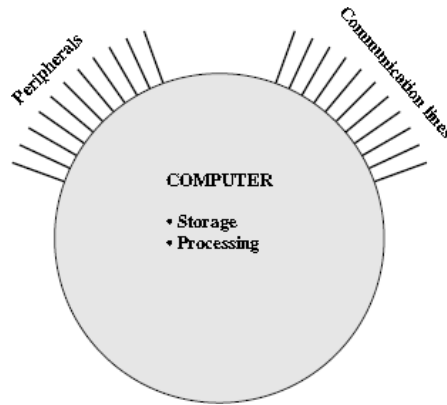
Gambar 1.2 Fungsi Operasi Komputer

(a) Perpindahan Data (b) Penyimpanan Data (c) Pemrosesan dari dan menuju storage (media penyimpanan) (d) Pemrosesan dari storage ke perangkat I/O

Sebuah komputer harus mampu melakukan pemrosesan data dalam format yang beragam dan cakupan pemrosesan yang sangat luas. Lokasi data yang diproses oleh unit pemrosesan tidak selalu sama oleh karena itu salah satu fungsi dasar dari komputer adalah kemampuannya untuk memindahkan data dari satu tempat ke tempat lain yang dibutuhkan sebagai contoh perpindahan katakter dari keyboard ke screen. Meskipun pemrosesan data dilakukan dengan sangat cepat namun kenyataanya butuh satu tempat penyimpanan sementara selama proses berlangsung. Data-data komputer yang dibutuhkan dan data hasil pemrosesan disimpan dalam storage atau media penyimpanan untuk selanjutnya dilakukan pemrosesan ulang dari storage ke perangkat I/O atau sebaliknya sebagai contoh adalah perpindahan data pada saat men-download ataupun meng-upload dari/ke internet. Ketiga operasi komputer tersebut yakni pemrosesan, penyimpanan dan perpindahan data membutuhkan kendali yang dapat mengatur kinerja sistem sehingga bisa harmonis, disinilah operasi kontrol bekerja.

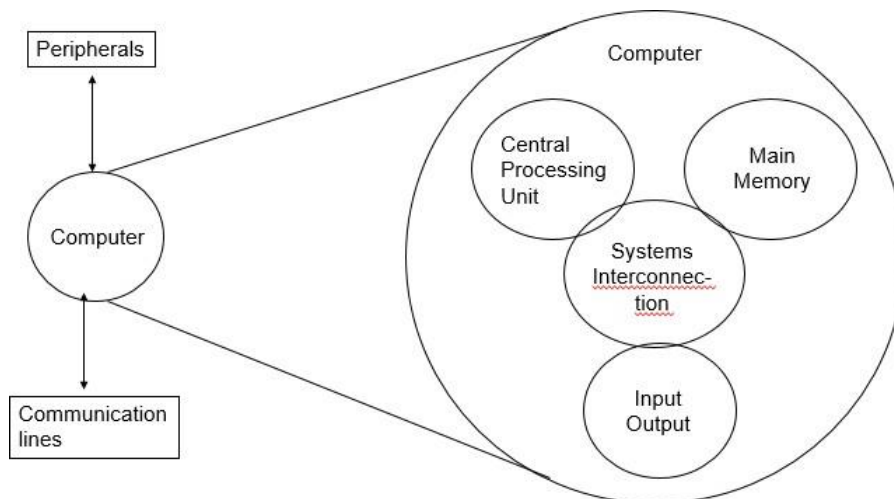
2. Struktur

Dalam pemanfaatannya di kehidupan sehari-hari, komputer digunakan secara bersama-sama dengan berbagai perangkat dari lingkungan eksternal. Secara general perangkat lingkungan terbagi menjadi dua bagian yaitu perangkat peripheral dan jalur komunikasi. Sebagaimana tampak pada gambar 1.3 berikut



Gambar 1.3 Interaksi computer dengan lingkungan eksternal

Struktur internal computer terdiri empat bagian utama yaitu unit pemrosesan terpusat (central processing unit - CPU), memory utama, I/O dan system interkoneksi. Gambar 1.4 menunjukkan struktur top level dari sebuah komputer.

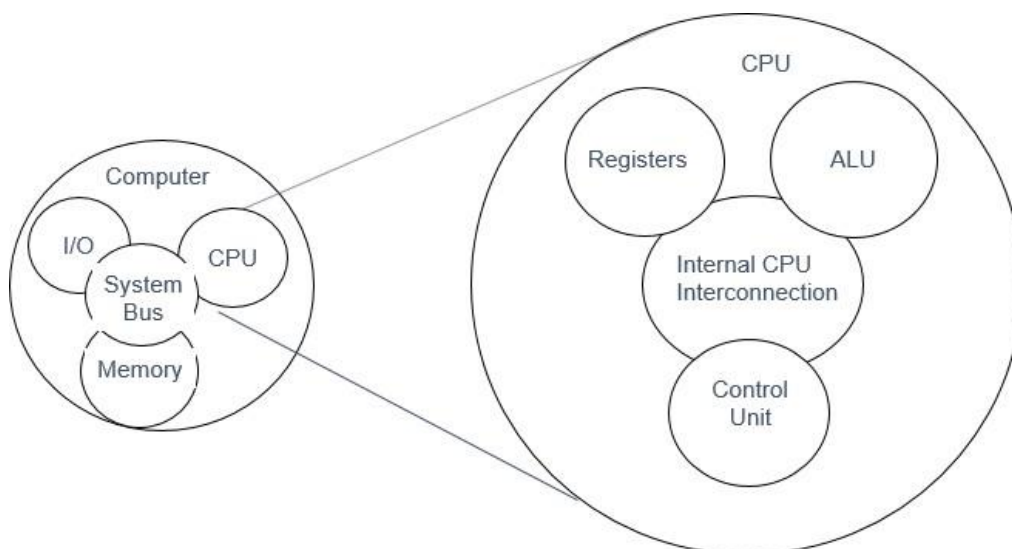


Gambar 1.4 Struktur - Top Level Komputer

CPU bertugas dalam mengontrol operasi komputer dan melakukan fungsi pemrosesan data sehingga biasa disebut processor. Main memory digunakan untuk menyimpan data dan I/O berfungsi sebagai media penghubung antara bagian dalam computer (internal system) dengan lingkungan eksternal. Melalui I/O data bisa dipindahkan dari storage di dalam

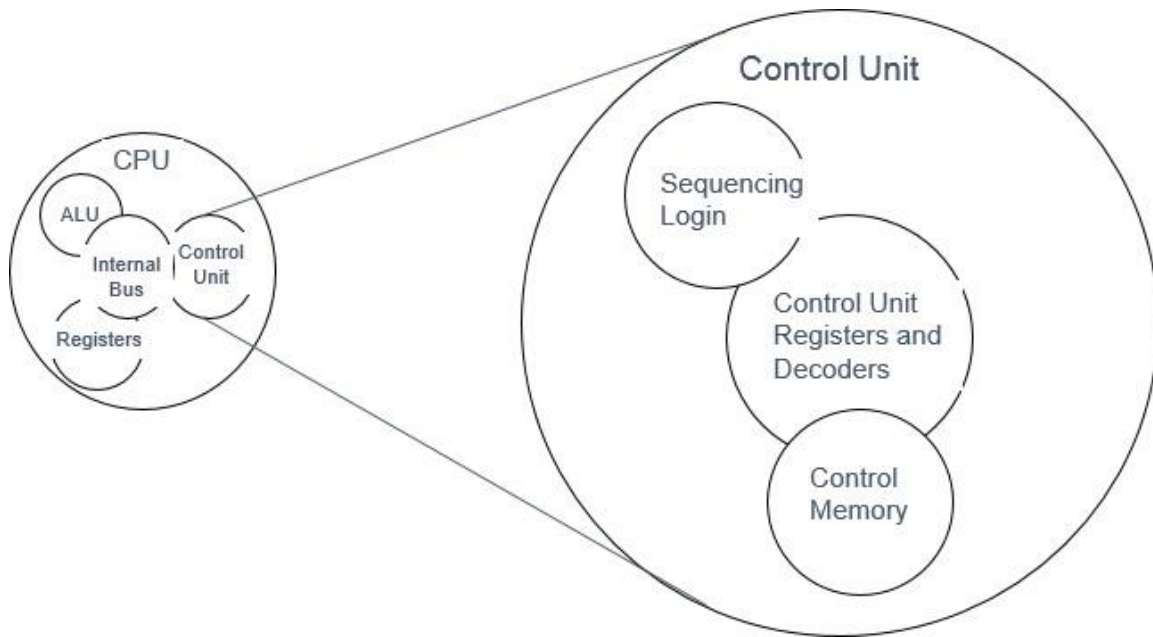
komputer ke media penyimpanan lain di luar computer. Sistem interkoneksi berfungsi untuk menghubungkan semua bagian yang ada pada komputer. Sistem bus ini pula yang menyediakan komunikasi antara CPU, main memory dan I/O. Salah satu sistem interkoneksi yang sangat terkenal adalah sistem bus, yang terdiri dari sejumlah kabel yang terhubung ke komponen-komponen computer.

Maing-masing bagian dari komputer masih terbagi menjadi sub sistem yang lebih kecil. Sebagai contoh pada gambar 1.5 dibawah adalah komponen penyusun CPU. Struktur dari CPU terdiri dari banyak *register*, ALU (*arithmetic and logic unit*), *control unit*, serta internal CPU *interconnection*. *Register* berfungsi untuk menyediakan storage internal pada CPU, ALU melaksanakan fungsi pemrosesan data, *control unit* bertugas dalam mengontrol operasi CPU termasuk operasi computer sedangkan CPU *interconnection* menyediakan meekanisme komunikasi antara *register*, ALU (*arithmetic and logic unit*) dan *control unit*.



Gambar 1.5 Struktur – CPU

Setiap bagian sub sistem dari CPU memiliki hieraki yang lebih rendah lagi. Misalnya control unit merupakan integrase dari sequencing logic, control unit registers and decoders serta control memory. Demikian seterusnya hingga sistem komputer yang kompleks mencapai hierarki pada level yang paling rendah dan sederhana.



Gambar 1.6 Struktur - Control Unit

C. Ringkasan Eksekutif

- Komputer adalah sebuah mesin elektronik yang secara cepat menerima informasi masukan digital dan mengolah informasi tersebut menurut seperangkat instruksi yang tersimpan dalam komputer dan menghasilkan keluaran informasi yang dihasilkan setelah diolah.
- Organisasi Komputer adalah bagian yang terkait erat dengan unit–unit operasional dan interkoneksi antar komponen penyusun sistem komputer dalam merealisasikan aspek arsitekturalnya.
- Arsitektur Komputer lebih cenderung pada kajian atribut–atribut sistem komputer yang terkait dengan seorang programmer.
- Fungsi dasar sistem komputer adalah Fungsi Operasi Pengolahan Data, Penyimpanan Data, Fungsi Operasi Pemindahan Data, Fungsi Operasi Kontrol.
- Struktur internal komputer meliputi: Central Processing Unit(CPU), Memori Utama, I/O, Sistem Interkoneksi.
- Struktur internal CPU meliputi: Control Unit, Aritmetic And Logic Unit(ALU), Register, CPU Interkoneksi.

D. SOAL-SOAL

1. Isikan daftar istilah berikut kedalam kolom yang tersedia pada tabel

- | | | | |
|---------------|---------------|-------------|-----------|
| a. Arsitektur | b. Organisasi | c. Struktur | d. Fungsi |
|---------------|---------------|-------------|-----------|

“Atribut yang tampak oleh programmer” adalah pengertian dari	
“Bagaimana suatu fitur diimplementasikan” adalah pengertian dari	
“Cara komponen terkoneksi satu dengan yang lainnya” adalah	
“Operasi komponen secara individual” adalah pengertian dari	

2. Sebutkan 4 fungsi dasar komputer

-
-
-
-

3. Berikut yang merupakan struktur top level dari komputer adalah

- | | |
|--------------------------|-----------------------------------|
| a. <i>Control Memory</i> | e. <i>ALU</i> |
| b. <i>Main Memory</i> | f. <i>Peripheral</i> |
| c. <i>Register</i> | g. <i>I/O module</i> |
| d. <i>Control Unit</i> | h. <i>Central Processing Unit</i> |

4. Berikut yang merupakan struktur top level dari Central Processing Unit adalah

- | | |
|--------------------------|-----------------------------------|
| a. <i>Control Memory</i> | e. <i>ALU</i> |
| b. <i>Main Memory</i> | f. <i>Peripheral</i> |
| c. <i>Register</i> | g. <i>I/O module</i> |
| d. <i>Control Unit</i> | h. <i>Central Processing Unit</i> |

BAB 2. EVOLUSI DAN PERFORMA (KINERJA) KOMPUTER

Tujuan Pembelajaran

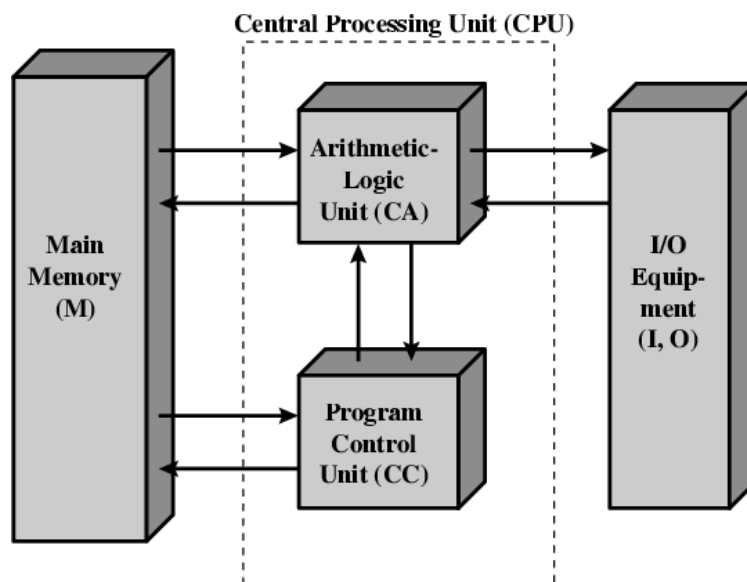
1. Mahasiswa mampu menjelaskan tentang sejarah teknologi komputer
2. Mahasiswa mampu menjelaskan trend teknologi yang telah membuat unjuk kerja yang menjadi fokus rancangan sistem komputer
3. Mahasiswa mampu meninjau bermacam-macam teknik dan strategi yang digunakan untuk mencapai unjuk kerja yang seimbang dan efisien
4. Mahasiswa mampu menjelaskan perkembangan pentium dan Power PC

A. Komputer Generasi Pertama

Perkembangan komputer dari masa kemasa terus mengalami peningkatan yang sangat pesat. Penciptaan perangkat elektronik ini dimulai pada tahun 1940an dengan memanfaatkan tabung hampa udara. Salah satu hasil nyata computer generasi pertama ini adalah ENIAC (*Electronic Numerical Integrator And Computer*). Pada tahun 1943 John Mauchly, seorang Profesor teknik elektro dari University of Pennsylvania beserta mahasiswanya, John Eckert mengusulkan untuk membangun satu komputer general-purpose menggunakan vacuum tubes (tabung hampa udara). Ide tersebut selanjutnya diterima oleh angkatan darat pemerintah setempat. Mesin yang dihasilkan memiliki bobot 30 ton dengan luas 1500 meter persegi dan terdiri atas lebih dari 18.000 vacuum tubes. Komputer ini direncanakan sebagai alat pembuatan jarak dan tabel lintasan peluru kendali senjata baru. Proyek ENIAC berhasil diselesaikan pada tahun 1946 sehingga merupakan satu usaha yang terlambat untuk dimanfaatkan pada perang dunia ke dua, namun mesin ini masih digunakan sampai tahun 1955. Komputer ENIAC bekerja menggunakan sistem bilangan desimal, bukan biner. Mesin ini memiliki memori yang terdiri atas 20 akumulator dan setiap akumulator menampung 10 digit desimal. ENIAC diprogram secara manual dengan switch, membutuhkan konsumsi daya sebesar 140Kw serta mampu melakukan 5000 operasi penjumlahan per detik.

Komputer generasi pertama selanjutnya adalah *Von Neumann Machine / Turing*. Tahun 1945 Von Neumann yang merupakan seorang ahli matematika sekaligus konsultan proyek ENIAC memperbaiki kelemahan pada ENIAC dengan menciptakan EDVAC (*Electronic Discrete Variable Computer*). EDVAC merupakan komputer pertama menggunakan *stored-program concept*. Selanjutnya pada tahun 1946 Von Neumann beserta koleganya mulai mendesain satu komputer *stored-program concept* yang disebut sebagai komputer IAS di Princeton Institute for Advanced Studies. IAS (Computer of Institute for Advanced Studies)

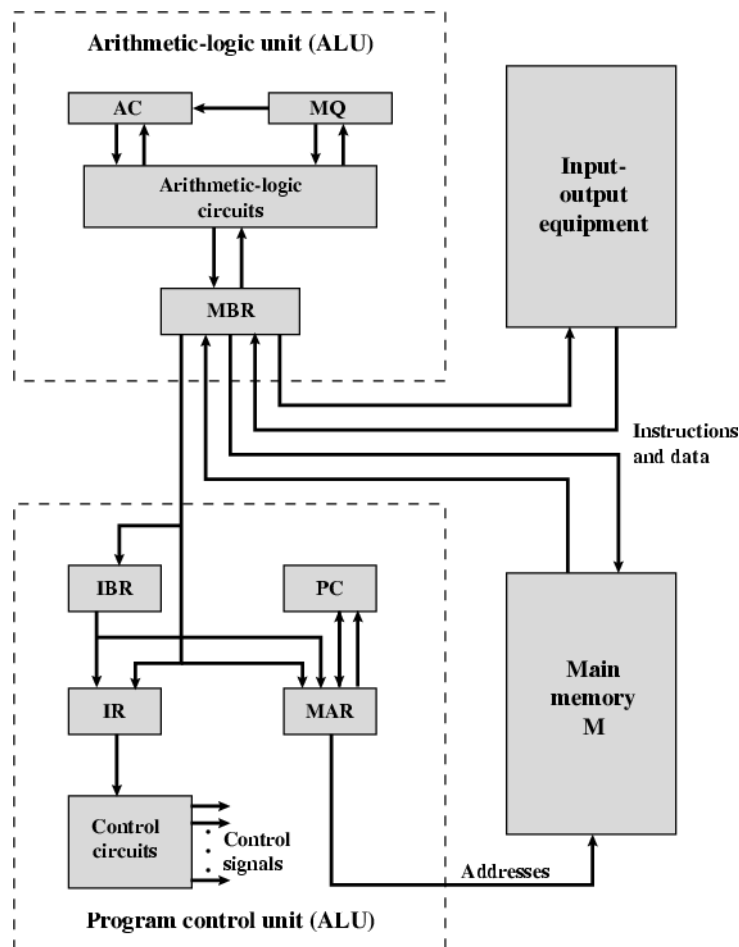
selesai dibangun pada tahun 1952. Struktur dari von Neumann machine tampak pada gambar 2.1 di bawah. Komputer dengan konsep *stored-program* memiliki memori utama untuk menyimpan data maupun instruksi, Arithmetic Logic Unit (ALU), untuk mengolah data biner, Control Unit untuk melakukan kontrol terhadap instruksi–instruksi di dalam memori serta I/O untuk berinteraksi dengan lingkungan luar.



Gambar 2.1 *Structure of von Neumann machine / Structure of IAS Computer*

Spesifikasi detail dari komputer IAS yaitu memiliki 1000 lokasi penyimpanan dengan kapasitas masing-masing 40 bit words untuk menyimpan data dan intruksi. Data dan instruksi yang ada direpresentasikan dalam bentuk biner dengan jumlah instruksi sebesar 2×20 bit. Selanjutnya set register (storage dalam CPU) pada komputer ini terdiri atas *Memory Buffer Register*, *Memory Address Register*, *Instruction Register*, *Instruction Buffer Register*, *Program Counter*, *Accumulator*, dan *Multiplier Quotient*. Gambar 2.2 menunjukkan struktur komputer IAS secara detail. Memory Buffer Register (MBR), berisi sebuah word yang akan disimpan di dalam memori atau digunakan untuk menerima word dari memori. Memory Address Register (MAR), untuk menentukan alamat word di memori untuk dituliskan dari MBR atau dibaca oleh MBR. Instruction Register (IR), berisi instruksi 8 bit kode operasi yang akan dieksekusi. Instruction Buffer Register (IBR), digunakan untuk penyimpanan sementara instruksi sebelah kanan word di dalam memori. Program Counter (PC), berisi alamat pasangan instruksi berikutnya yang akan diambil dari memori. Accumulator (AC) dan Multiplier Quotient (MQ), digunakan untuk penyimpanan sementara operand dan hasil ALU. Misalnya, hasil perkalian 2

buah bilangan 40 bit adalah sebuah bilangan 80 bit; 40 bit yang paling berarti (most significant bit) disimpan dalam AC dan 40 bit lainnya (least significant bit) disimpan dalam MQ.



Gambar 2.2. Structure of IAS – detail

IAS beroperasi secara berulang membentuk siklus instruksi. Komputer IAS memiliki 21 instruksi, yang dapat dikelompokkan menjadi 5 seperti berikut ini : *Data tranfer*, memindahkan data di antara memori dengan register – register ALU atau antara dua register ALU sendiri. *Unconditional branch*, perintah – perintah eksekusi percabangan tanpa syarat tertentu. *Conditional branch*, perintah – perintah eksekusi percabangan yang memerlukan syarat tertentu agar dihasilkan suatu nilai dari percabangan tersebut. *Arithmetic*, kumpulan operasi – operasi yang dibentuk oleh ALU. *Address Modify*, instruksi – instruksi yang memungkinkan perubahan alamat saat di komputasi sehingga memungkinkan fleksibilitas alamat yang tinggi pada program.

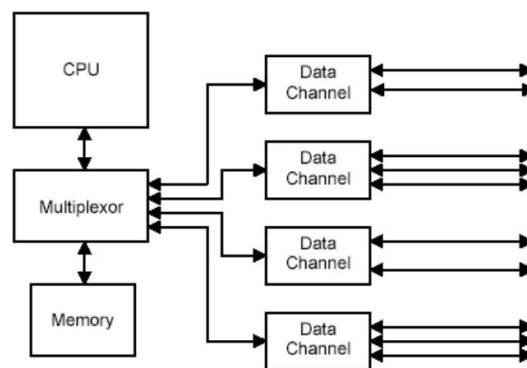
Pada tahun 1947 Eckert-Mauchly mendirikan Eckert-Mauchly Computer Corporation. Perusahaan ini sudah mulai melakukan komersialisasi komputer. Komputer generasi pertama yang dikomersialkan adalah UNIVAC I (Universal Automatic Computer). UNIVAC I menjadi

tulang punggung perhitungan sensus di USA. Pada tahun tersebut merupakan tahun kelahiran industri komputer dengan munculnya 2 buah perusahaan yang saat itu mendominasi pasar, yaitu Sperry dan IBM. Tahun 1950 diluncurkan UNIVAC II, dengan karakteristik komputasi yang lebih cepat serta kapasitas memory yang lebih besar. Pada tahun 1953 IBM memproduksi *stored program computer* pertamanya yang diberi nama IBM seri 701. Komputer ini memiliki kemampuan untuk melakukan kalkulasi *scientific*. IBM seri 702 dipasarkan pada tahun 1955, dengan kemampuan untuk membantu aplikasi bisnis. Pada tahun selanjutnya IBM mengeluarkan seri 700/7000.

B. Komputer Generasi Kedua

Pada komputer generasi kedua terjadi perubahan unit pemrosesan yang sebelumnya memanfaatkan *vacuum tube* (tabung hama udara) berubah menjadi transistor. Kelebihan yang dimiliki oleh transistor adalah secara ukuran lebih kecil, secara bobot lebih ringan, dan memiliki disipasi daya lebih rendah. Transistor berupa perangkat yang berbentuk padat (*Solid State device*) dan terbuat dari pasir silikon. Transistor ditemukan oleh William Shockley dan kawan-kawan pada tahun 1947 di Laboratorium Bell.

Komputer berbasis transistor merupakan mesin generasi kedua. Perusahaan NCR & RCA mulai memproduksi *small transistor machines*, sementara IBM meluncurkan seri 7000 dan DEC membuat PDP-1. Gambar 2.3 berikut merupakan konfigurasi IBM seri 7094



Gambar 2.3 Konfigurasi IBM 7094

C. Komputer Generasi Ketiga

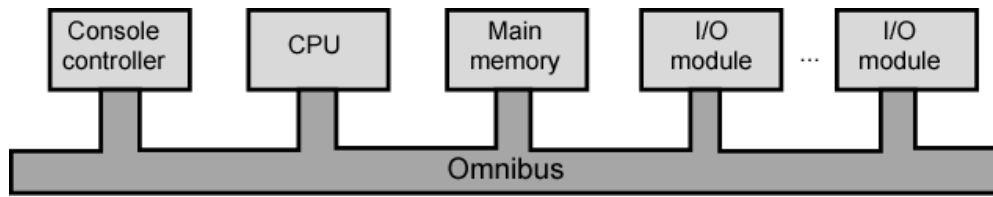
Komputer organisasi ketiga memanfaatkan *Integrated Circuit (IC)* sebagai pengganti transistor. IC merupakan kumpulan dari puluhan transistor yang dipadukan secara bersama dalam satu chip tunggal. IC juga dikenal sebagai perangkat *microelectronics* yang secara literature berarti "*small electronics*". IC terbentuk dari kumpulan gate, kumpulan memori dan

interkoneksinya yang dapat dibuat dengan semikonduktor. Generasi komputer dari tahun ke tahun dapat dirangkum sebagai berikut :

- *Vacuum Tube* - 1946-1957
- Transistor - 1958-1964
- *Small scale integration* - 1965 *used up to 100 devices on a chip*
- *Medium scale integration* - to 1971 *used 100-3,000 devices on a chip*
- *Large scale integration* - 1971-1977 *used 3,000 - 100,000 devices on a chip*
- *Very large scale integration* - 1978 to date, *used 100,000 - 100,000,000 devices on a chip*
- *Ultra large scale integration, used over 100,000,000 devices on a chip*

Contoh dari komputer generasi ketiga yang diluncurkan oleh IBM pada tahun 1964 adalah IBM seri 360. Komputer ini memiliki karakteristik berupa kemiripan pada set instruksi atau identik, dalam kelompok komputer ini berbagai model yang dikeluarkan menggunakan set instruksi yang sama sehingga mendukung kompatibilitas sistem maupun perangkat kerasnya. Selain itu IBM seri 360 juga memiliki sistem operasi mirip atau identik, ini merupakan *feature* yang menguntungkan konsumen sehingga apabila kebutuhan menuntut penggantian komputer tidak kesulitan dalam sistem operasinya karena sama. Pada komputer seri ini kecepatan meningkat, model – model yang ditawarkan mulai dari kecepatan rendah sampai kecepatan tinggi untuk penggunaan yang dapat disesuaikan konsumen sendiri. Seri 360 juga memiliki ukuran memori yang lebih besar, semakin tinggi modelnya akan diperoleh semakin besar memori yang digunakan selain itu juga harga yang meningkat, semakin tinggi modelnya maka harganya semakin mahal.

Contoh lain komputer generasi ketiga adalah DEC PDP-8 yang merupakan minicomputer pertama kali. Komputer ini diciptakan pada 1964 dan mesin yang dibuat sudah tidak memerlukan air conditioned room. DEC PDP-8 merupakan embedded applications dan OEM yang memiliki arsitektur sangat berbeda dengan IBM terutama bagian sistem bus. Pada komputer ini menggunakan omnibus system sebagai mana tampak pada gambar 2.4 dibawah. Sistem ini terdiri atas 96 buah lintasan sinyal yang terpisah, yang digunakan untuk membawa sinyal – sinyal kontrol, alamat maupun data. Arsitektur bus seperti PDP-8 ini nantinya digunakan oleh komputer – komputer modern.



Gambar 2. 4 Struktur Bus DEC - PDP-8

D. Komputer Generasi Selanjutnya

Aplikasi pertama teknologi IC adalah untuk pembuatan processor (*control unit* dan ALU), namun pada saat yang sama IC dimanfaatkan pula untuk pembuatan memori semikonduktor. Pada tahun 1970 Fairchild Semiconductor International, Inc. memproduksi memori semikonduktor pertama dengan kapasitas 256 bits. Chip memory yang dihasilkan memiliki karakteristik *non-destructive read* serta memiliki kecepatan yang lebih tinggi dari *magnetic core memory*. Memory yang dibuat hanya membutuhkan *70 billionths of a second* untuk membaca 1 bit data. Kapasitas memori semikonduktor meningkat rata-rata dua kali setiap tahun.

Pemanfaatan perangkat IC pada processor komputer generasi keempat dan seterusnya merupakan integrasi chip yang luar biasa mulai dari LSI, VLSI bahkan hingga kini ULSI. Pada tahun 1971 Intel memproduksi komputer dengan processor dengan ukuran micro (*microprocessor*) untuk pertama kalinya. Intel seri 4004 telah menggabungkan semua komponen CPU dalam satu chip tunggal serta memiliki 4 jalur bus (4 bit data). Selanjutnya pada tahun 1972 seri 8008 dengan kapasitas pemrosesan 8 bit diluncurkan. Kedua generasi intel tersebut merupakan komputer yang memiliki aplikasi spesifik. Komputer *general purpose microprocessor* pertama diterbitkan intel tahun 1974 dengan seri 8080. Evolusi microprocessor intel dari tahun ke tahun tampak pada gambar 2. 5 dibawah.

(a) 1970s Processors

	4004	8008	8080	8085	8088
Introduced	1971	1972	1974	1978	1979
Clock speeds	108 kHz	108 kHz	2 MHz	5 MHz, 8 MHz, 10 MHz	5 MHz, 8 MHz
Bus width	4 bits	8 bits	8 bits	16 bits	8 bits
Number of transistors	2,300	3,500	6,000	29,000	29,000
Feature size (μm)	10		6		6
Addressable memory	640 Bytes	16 KB	64 KB	1 MB	1 MB

(b) 1980s Processors

	80286	386TM DX	386TM SX	486TM DX CPU
Introduced	1982	1985	1988	1989
Clock speeds	6MHz-12.5MHz	16MHz-33MHz	16MHz-33MHz	25MHz-50MHz
Bus width	16 bits	32 bits	16 bits	32 bits
Number of transistors	134,000	275,000	275,000	1.2 million
Feature size (μm)	1.5	1	1	0.8-1
Addressable memory	16 MB	4 GB	16 MB	4 GB
Virtual memory	1 GB	64 TB	64 TB	64 TB
Cache	-	-	-	8 KB

(c) 1990s Processors

	486TM SX	Pentium	Pentium Pro	Pentium II
Introduced	1991	1995	1995	1997
Clock speeds	16MHz-33MHz	60MHz-165MHz	130MHz-200MHz	200MHz-300MHz
Bus width	32 bits	32 bits	64 bits	64 bits
Number of transistors	1.185 million	3.1 million	5.5 million	7.5 million
Feature size (μm)	1	0.8	0.6	0.35
Addressable memory	4 GB	4 GB	64 GB	64 GB
Virtual memory	64 TB	64 TB	64 TB	64 TB
Cache	8 KB	8 KB	512 KB L1 and 1 MB L2	512 KB L2

(d) Recent Processors

	Pentium III	Pentium 4	Core 2 Duo	Core 2 Quad
Introduced	1999	2000	2005	2008
Clock speeds	430-660MHz	1.3-1.8GHz	1.05-1.2GHz	3GHz
Bus width	64 bits	64 bits	64 bits	64 bits
Number of transistors	9.5 million	42 million	167 million	820 million
Feature size (nm)	250	180	65	45
Addressable memory	64 GB	64 GB	64 GB	64 GB
Virtual memory	64 TB	64 TB	64 TB	64 TB
Cache	512 KB L2	256 KB L2	2 MB L2	6 MB L2

Gambar 2.5 Evolusi microprocessor intel

E. Desain Kinerja

Pertimbangan dalam pemilihan sebuah komputer adalah : kinerja/kecepatan pemrosesan yang tinggi, kapasitas memori yang besar dan biaya yang murah. Ketiga dasar tersebut menjadi tujuan dalam arsitektur komputer. Komputer modern merupakan implementasi teknologi tinggi dan kompleks oleh karena itu dalam perancangannya dibutuhkan pemilihan spesifikasi sesuai kebutuhan seperti teknologi komponen penyusun, kapasitas penyimpanan, kinerja serta biaya perangkat.

Dalam melakukan desain sebuah komputer modern diperlukan beberapa cara untuk meningkatkan kecepatan performa pemrosesan diantaranya adalah sebagai berikut :

- *Pipelining*

- *On board cache*
- *On board L1 & L2 cache*
- *Branch prediction*
- *Data flow analysis*
- *Speculative execution*

Keseimbangan kinerja antara processor dan memori sangat diperlukan dimana ketika kecepatan processor meningkat maka kapasitas memori juga perlu ditingkatkan. Namun demikian perlu diketahui bahwa terdapat celah yaitu kecepatan (bukan kapasaitas) yang ada pada memori sangat jauh lebih rendah dibandingkan dengan kecepatan processor.

F. Ringkasan Eksekutif

- a. Evolusi prsesor komputer dimulai dari vacuum tube, selanjutnya transistor, IC, LSI, VLSI dan kemudian ULSI
- b. Generasi Komputer dari tahun ke tahun:
 - Generasi pertama : *Vacuum tube* - 1946-1957
 - Generasi kedua : *Transistor* - 1958-1964
 - Generasi ketiga : *IC* → *Small scale integration* - 1965 on ; up to 100 devices on a chip & *Medium scale integration* - to 1971 ; 100-3,000 devices on a chip
 - Generasi keempat : *Large scale integration* - 1971-1977 ; used 3,000 - 100,000 devices on a chip
 - Generasi kelima : *Very large scale integration* - 1978 -1991 ;100,000 - 100,000,000 devices on a chip
 - Generasi keenam : *Ultra large scale integration* – 1991 – Now ; Over 100,000,000 devices on a chip

G. Soal-soal

1. Urutan yang benar generasi komputer dari yang pertama sampai generasi ke empat adalah
 - ULSI 1.
 - IC 2.
 - Vacuum Tube 3.
 - Transistor 4.

2. Sebutkan 6 jenis register yang ada dalam CPU komputer IAS

- a.
- b.
- c.
- d.
- e.
- f.

3. Sebutkan 5 kelompok/kategori instruksi yang ada pada operasi berulang (siklus intruksi) komputer IAS

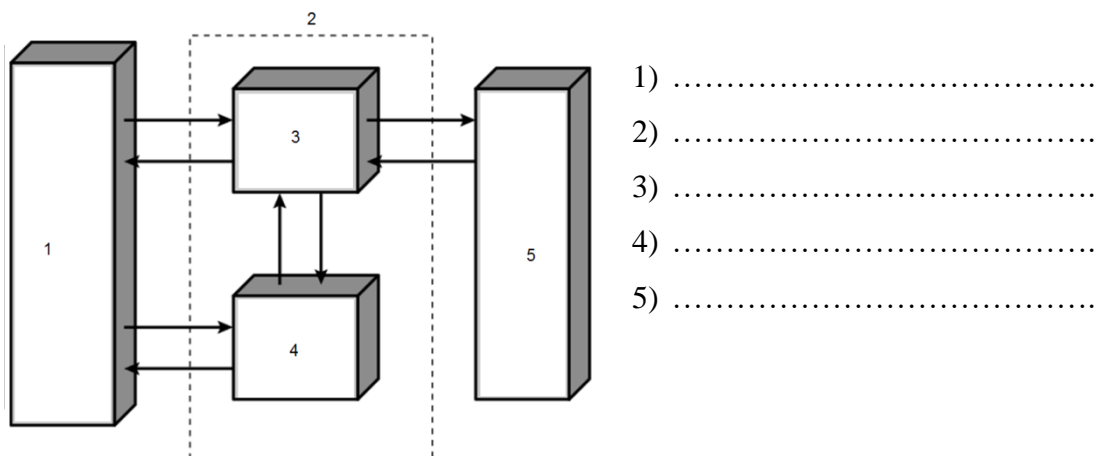
- a.
- b.
- c.
- d.
- e.

4. Evolusi komputer ditandai dengan perubahan pada 4 komponen yaitu kecepatan prosesor, ukuran komponen dan memory serta kapasitas dan kecepatan I/O yang mengalami proses peningkatan dan penurunan. Manakah dari keempat komponen tersebut yang mengalami

Peningkatan :

Penurunan :

5. Beri keterangan struktur komputer IAS berikut



BAB 3. REPRESENTASI DATA KOMPUTER

Tujuan Pembelajaran

1. Mahasiswa memahami jenis data pada suatu komputer
2. Mahasiswa mengenal istem bilangan berbasis dua, sepuluh dan enam belas
3. Mahasiswa bisa melakukan perhitungan konversi antara bilangan biner, desimal dan hexadesimal

A. Representasi Data Komputer

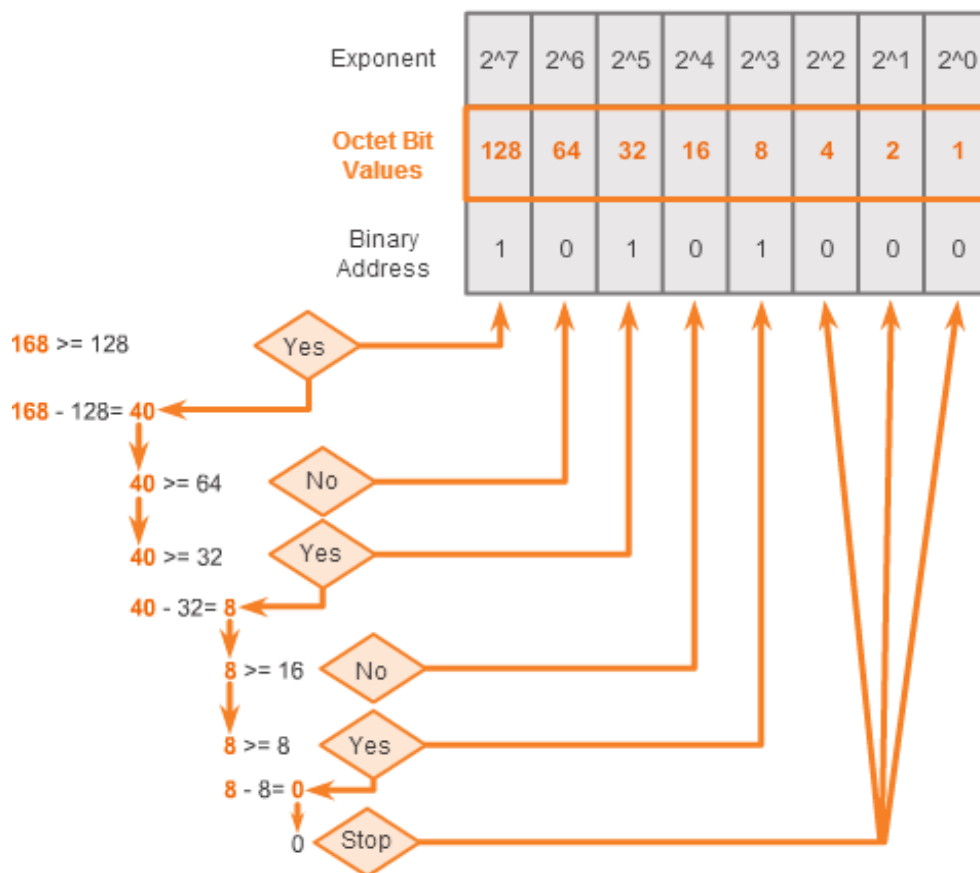
Komputer yang ada pada saat ini merupakan komputer biner. Semua bilangan yang disimpan dalam memori adalah data biner karena arsitektur komputer yang ada menggunakan sistem bilangan biner. Bilangan heksadesimal banyak digunakan dalam pemrograman bahasa assembly ataupun bahasa mesin. Selanjutnya terkait dengan format data pada komputer terbagi menjadi dua yaitu ASCII (American Standard code for information interchange) dimana data dengan format ini mewakili karakter alfanumerik dalam sistem komputer. Format data yang digunakan ASCII adalah 7 bit dengan bit ke-8 sebagai MSB (Most Significant Bit) yang digunakan untuk sebagai parity checker (melakukan pengecekan pada beberapa sistem yang terkait). Sebagai contoh pada sebuah printer pencetakan data alfanumerik terjadi ketika MSB dari format data ASCII bernilai 0 (MSB=0) dan untuk pencetakan grafik MSB bernilai 1. Dalam komputer, kumpulan karakter extended ASCII menggunakan kode 80H – FFH. Karakter-karakter ini menyimpan huruf-huruf asing dan tanda baca, karakter Yunani, karakter matematika, karakter box – drawing, dan karakter khusus lainnya. Format data lainnya selain ASCII adalah BCD (Binary coded decimal). Data BCD disimpan dalam bentuk packed BCD atau unpacked BCD. Data packet BCD data desimal disimpan dalam bentuk 2 digit per byte sedangkan unpacked BCD data desimal disimpan 1 digit per byte. Packed BCD digunakan untuk beberapa instruksi penjumlahan dan pengurangan BCD dalam kumpulan instruksi mikroprosesor. Unpacked BCD digunakan untuk keypad atau keyboard dimana memerlukan satu byte tiap karakternya. Contoh desimal 735 jika dikonversi dalam packet BCD menjadi 0000 0111 - 0011 0101 sedangkan unpacked BCD menjadi 0000 0111 - 0000 0011- 0000 0101.

B. Sistem Bilangan & Konversi Bilangan Biner, Desimal, Heksadesimal

Sistem bilangan biner merupakan sistem bilangan berbasis dua terdiri atas 0 dan 1. Angka tersebut dikenal dengan istilah logika digital (0 &1). Setiap digit dalam sistem bilangan biner disebut bit (binary digit). Delapan buah bit disebut Byte yang merupakan unit

penyimpanan terkecil pada sebuah komputer. Tipe penyimpanan yang lebih besar adalah word (16 bit = 2byte) dan doubleword (32 bit = 8byte). Sistem bilangan Heksadesimal merupakan sistem bilangan berbasis enambilas dengan angka 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E, dan F. Bilangan heksadesimal banyak dimanfaatkan pada pemrograman assembly (pokok bahasan pemrograman assembly akan dibahas pada pertemuan selanjutnya).

Cara konversi Bilangan dari biner ke desimal dapat dilakukan sebagaimana langkah-langkah berikut. Miisalkan ada pertanyaan berapa nilai biner dari bilangan 168 ? Perhatikan gambar 3.1 berikut, dimana pada baris terdapat beberapa kolom yang mewakili nilai eksponensial bilangan basis dua. Dimulai dari dua pangkat nol yang sama dengan satu, kemudian berlanjut pada dua pangkat satu, dua pangkat dua, dua pangkat tiga demikian seterusnya hingga mendekati nilai yang dibutuhkan. Dalam satu octet atau 8 bit yang dimulai dari dua pangkat nol maka diperoleh nilai tertinggi dua pangkat tujuh atau sama dengan serratus dua puluh delapan. Nilai hasil pemangkatan (eksponensial) dari bari pertama diletakkan pada baris kedua (warna oranye). Kemudian pada baris ketiga kita letakkan angka 0 atau 1 sesuai dengan nilai yang ditanyakan.



Gambar 3.1 Konversi bilangan biner ke desimal

Pada contoh ini konversi 168 menghasilkan bilangan biner 10101000. Nilai 10101000 diperoleh dengan meletakkan angka 1 pada kolom yang jika dijumlahkan totalnya akan membentuk 128, atau jika dalam matematika sederhana maka 128 didapatkan dengan menjumlahkan bilangan $128 + 32 + 8$. Pada kolom yang angkanya tidak dibutuhkan dibubuhkan nilai 0 sehingga pada kolom dibawah bilangan 64, 16, 4, 2 dan 1 di beri nilai 0. Atau untuk memudahkan maka kita bisa mengalikan antara kolom yang ditandai dengan nilai nol maupun satu pada bariis dua dan tiga yaitu : $128(1) + 64(0) + 32(1) + 16(0) + 8(1) + 4(0) + 2(0) + 1(0) = 128$. Selanjutnya untuk membentuk biner maka nilai satu pada bagian pengali dikumpulkan menjadi 10101000. Selanjutnya perhatikan tabel berikut untuk lebih memudahkan konversi yang diinginkan

Tabel 3.1 Contoh konversi bilangan 176 ke dalam bentuk biner

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
1	0	1	1	0	0	0	0

Tabel 3.2 Contoh konversi bilangan 255 ke dalam bentuk biner

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
1	1	1	1	1	1	1	1

Tabel 3.3 Contoh konversi bilangan desimal dalam bentuk biner

Value	Bit Value							
	128	64	32	16	8	4	2	1
255	1	1	1	1	1	1	1	1
254	1	1	1	1	1	1	1	0
252	1	1	1	1	1	1	0	0
248	1	1	1	1	1	0	0	0
240	1	1	1	1	0	0	0	0
224	1	1	1	0	0	0	0	0
192	1	1	0	0	0	0	0	0
128	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0

Jenis bilangan biner terbagi menjadi dua yaitu Sign and Unsign Integer serta Floating Point. Sign integer adalah bilangan bulat bertanda yaitu tanda positif ataupun negatif,

sedangkan unsign integer adalah bilangan bulat tidak bertanda. Unsign Integer (integer tidak bertanda) berarti semua bilangannya adalah positif. Selanjutnya bilangan floating point merupakan bilangan pecahan atau bilangan real. Floating point ranges untuk bilangan 32 bit adalah eksponensial dari 8 bit atau $\pm 2^{256} \approx 1.5 \times 10^{77}$. Akurasi pada bilangan floating point dipengaruhi oleh perubahan LSB mantissa, dimana 23 bit mantissa $2^{-23} \approx 1.2 \times 10^{-7}$ atau sekitar 6 decimal dibelakang koma.

Konversi bilangan desimal ke dalam bentuk biner ataupun sebaliknya biner ke desimal yang tidak bertanda lebih mudah karena semua nilainya positif. Bilangan biner hanya ada angka 0 dan 1 untuk merepresentasikan segalanya. Contoh bilangan positif yang terdapat dalam sistem biner 41 adalah 00101001. Hal yang perlu diketahui bahwa dalam sistem bilangan biner dengan format integer tidak ada tanda minus serta tidak ada tanda koma. Cara untuk mengubah bilangan biner integer menjadi bentuk negative adalah *Sign-Magnitude* dan *Two's compliment*.

Pada *Sign-Magnitude*, bit paling kiri merupakan bit tanda (*Left most bit is sign bit*). Bit 0 berarti positif dan bit 1 berarti negatif. Sebagai contoh +18 (positif delapan belas) = 00010010 sedangkan -18 (negatif delapan belas) = 10010010. Permasalahan yang muncul ketika menggunakan sistem *Sign-Magnitude* adalah terdapat dua representasi nilai nol yaitu (+0 dan -0) serta sangat perlu mempertimbangkan tanda dan magnitude dalam operasi aritmatika. Dalam penulisannya untuk membedakan bilangan biner bertanda dengan tidak bertanda adalah dengan menuliskan "*Sign-Magnitude*" pada nilai dibelakang bilangan biner. Contoh lain konversi bilangan desimal ke bilangan binernya dengan "sign maginitude" adalah sebagai berikut

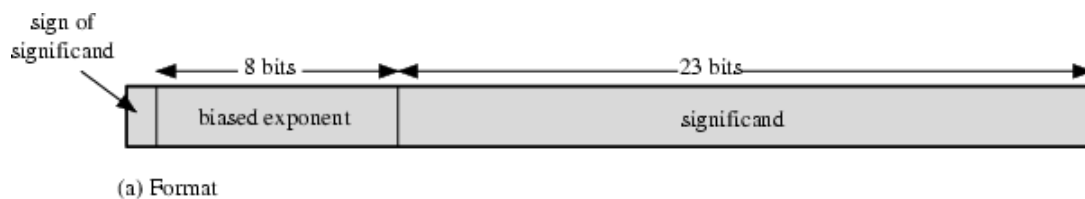
- +15 = 00001111 (... sign maginitude ...)
- -15 = 10001111 (... sign maginitude ...)
- +215 = 0000 0000 11010111 (... sign maginitude ...)
- -215 = 1000 0000 11010111 (... sign maginitude ...)

Cara konversi pada bentuk sign bit selanjutnya adalah dengan metode Two's Compliment. Pada metode ini terdapat dua langkah yang harus dilakukan setelah bilangan desimal dikonversi ke dalam bentuk unsigned bit. Langkah pertama atau first complement adalah dengan mengubah biner 0 menjadi 1 dan sebaliknya 1 menjadi nol. Selanjutnya pada langkah kedua atau second complement adalah menambahkan biner 1 pada bit paling kanan. Perhatikan contoh berikut untuk mendapatkan gambaran yang lebih jelas. Perintah yang dibuat

adalah konversikan bilangan desimal -15 dan -215 ke dalam bentuk bilangan biner bertanda dengan “Two’s Complement”.

- a. Jawab +15 : 00001111
 1’st Complement : 11110000
 2’s Complement : _____ 1+
 Sehingga -15 : 11110001 (Two’s Complement)
- b. Jawab +215 : 00000000 11010111
 1’st Complement : 11111111 00101000
 2’s Complement _____ 1+
 Sehingga -215 : 11111111 00101001 (Two’s Complement)

Konversi selanjutnya adalah bilangan biner dalam bentuk floating point. Floating point merupakan sistem bilangan pecahan.



$$\begin{aligned}
 1.1010001 \times 2^{10100} &= 0\ 10010011\ 101000100000000000000000 = 1.638125 \times 2^{20} \\
 -1.1010001 \times 2^{10100} &= 1\ 10010011\ 101000100000000000000000 = -1.638125 \times 2^{20} \\
 1.1010001 \times 2^{-10100} &= 0\ 01101011\ 101000100000000000000000 = 1.638125 \times 2^{-20} \\
 -1.1010001 \times 2^{-10100} &= 1\ 01101011\ 101000100000000000000000 = -1.638125 \times 2^{-20}
 \end{aligned}$$

(b) Examples

Gambar 3.2. Cara konversi biner *floating point*

Sebagai contoh konversi bilangan-bilangan desimal berikut ke bilangan biner (floating point)

- 0,5 = 2^{-1} → 0,1
- 0,25 = 2^{-2} → 0,01
- 0,125 = 2^{-3} → 0,001
- 0,0625 = 2^{-4} → 0,0001
- 190,125 = 1011 1110,001

Sedangkan contoh konversi bilangan-bilangan biner ke bilangan desimal adalah sebagai berikut

- 0000011,1001 = 3,5625
 → $2^1 2^0 2^{-1} 2^{-4}$
 → $2^1 = 2 \mid 2^0 = 1 \mid 2^{-1} = 0,5 \mid 2^{-4} = 0,0625$
 → $2 + 1 + 0,5 + 0,0625 = 3,5625$
- 0011001,011 = 24,375
 → $2^4 2^3 2^0 2^{-2} 2^{-3}$
 → $2^4 = 16 \mid 2^3 = 8 \mid 2^0 = 1 \mid 2^{-2} = 0,25 \mid 2^{-3} = 0,125$
 → $16 + 8 + 0,25 + 0,125 = 24,375$

Sistem Bilangan Heksadesimal merupakan bilangan berbasis enam belas. Penomoran bilangan basis 16 menggunakan angka 0 sampai 9 dan huruf A sampai F. Perhatikan pola bilangan biner yang sesuai dengan nilai pada desimal dan heksadesimal pada tabel berikut

Tabel 3.4 Contoh konversi bilangan desimal, biner dan heksadesimal

Hexadecimal	Decimal	Binary	Hexadecimal	Decimal	Binary
0	0	0000	00	0	0000 0000
1	1	0001	01	1	0000 0001
2	2	0010	02	2	0000 0010
3	3	0011	03	3	0000 0011
4	4	0100	04	4	0000 0100
5	5	0101	05	5	0000 0101
6	6	0110	06	6	0000 0110
7	7	0111	07	7	0000 0111
8	8	1000	08	8	0000 1000
9	9	1001	0A	10	0000 1010
A	10	1010	0F	15	0000 1111
B	11	1011	10	16	0001 0000
C	12	1100	20	32	0010 0000
D	13	1101	40	64	0100 0000
E	14	1110	80	128	1000 0000
F	15	1111	C0	192	1100 0000
			CA	202	1100 1010
			F0	240	1111 0000
			FF	255	1111 1111

C. Ringkasan Eksekutif

- Konversikan bilangan-bilangan biner berikut ke bilangan desimalnya

$$00001111 = 17$$

- Konversikan bilangan-bilangan biner berikut ke bilangan desimalnya

$$0000011,1001 = 3,5625$$

$$2^1 2^0 2^{-1} 2^{-4} \rightarrow 2^1 = 2 \mid 2^0 = 1 \mid 2^{-1} = 0,5 \mid 2^{-4} = 0,0625 \rightarrow 2 + 1 + 0,5 + 0,0625 = 3,5625$$

$$0011001,011 = 3,5625$$

$$2^4 2^3 2^0 2^{-2} 2^{-3} \rightarrow 2^4 = 16 \mid 2^3 = 8 \mid 2^0 = 1 \mid 2^{-2} = 0,25 \mid 2^{-3} = 0,125 \rightarrow 16 + 8 + 0,25 + 0,125$$

- Konversikan bilangan-bilangan desimal berikut ke bilangan binernya

$$84 = 1010100$$

- Konversikan bilangan-bilangan desimal berikut ke bilangan binernya

$$190,125 = 1011 1110,001$$

$$190 = 1011 1110$$

$$0,5 = 2^{-1} \rightarrow 0,1$$

$$0,25 = 2^{-2} \rightarrow 0,01$$

$$0,125 = 2^{-3} \rightarrow 0,001$$

$$0,0625 = 2^{-4} \rightarrow 0,0001$$

- Konversikan bilangan desimal berikut ke bilangan binernya dengan “sign maginitude”

$$+15 = \quad \quad \quad 00001111 \quad (\dots \text{sign maginitude } \dots)$$

$$-15 = \quad \quad \quad 10001111 \quad (\dots \text{sign maginitude } \dots)$$

$$+215 = \quad 0000 0000 \quad 11010111 \quad (\dots \text{sign maginitude } \dots)$$

$$-215 = \quad 1000 0000 \quad 11010111 \quad (\dots \text{sign maginitude } \dots)$$

- Konversikan bilangan desimal berikut ke bilangan binernya dengan “Two’s Complement”

$$-15 = \quad \quad \quad 11110001 \quad (\dots \text{Two’s Complement } \dots)$$

$$\text{Jawab } +15 : \quad \quad \quad 00001111$$

$$1\text{'st Complement} : \quad \quad \quad 11110000$$

$$2\text{'s Complement} : \quad \quad \quad \underline{\hspace{10em}} 1+$$

$$-15 \quad \quad \quad 11110001 \quad (\dots \text{Two’s Complement } \dots)$$

$$\begin{aligned}
-215 &= 11111111\ 00101001 && (\dots \text{Two's Complement } \dots) \\
\text{Jawab } +215 &: 00000000\ 11010111 \\
\text{1'st Complement} &: 11111111\ 00101000 \\
\text{2's Complement} &: \underline{\hspace{10em}} 1+ \\
-215 &= 11111111\ 00101001 && (\dots \text{Two's Complement } \dots)
\end{aligned}$$

✚ Hitunglah aritmatika bilangan biner berikut

$$1101 \times 101 = 0100\ 0001$$

D. Soal-Soal

1. Konversikan bilangan-bilangan biner berikut ke bilangan desimalnya

- a. 00001100 =
- b. 00000011 =
- c. 00011100 =
- d. 00111100 =
- e. 00101010 =

2. Konversikan bilangan-bilangan biner berikut ke bilangan desimalnya

- a. 00011100,011 =
- b. 00110011,10011 =
- c. 1010101010,1 =
- d. 110011,101 =
- e. 1110101,11 =

3. Konversikan bilangan-bilangan desimal berikut ke bilangan binernya

- a. 64 =
- b. 111 =
- c. 223 =
- d. 345 =
- e. 455 =

4. Konversikan bilangan-bilangan desimal berikut ke bilangan binernya

- a. 25,25 =
- b. 34,75 =
- c. 114,625 =
- d. 215,500 =
- e. 27,1875 =

5. Konversikan bilangan desimal berikut ke bilangan binernya dengan “sign maginitude”

- a. -5 =(.....)
- b. -81 =(.....)
- c. -165 =(.....)
- d. -204 =(.....)
- e. -267 =(.....)

6. Konversikan bilangan desimal berikut ke bilangan binernya dengan “Two’s Complement”

- a. -5 =(.....)
- Jawab +5 :
- 1’st Complement :
- 2’s Complement+
- 5(.....)

- b. -81 =(.....)
- Jawab +81 :
- 1’st Complement :
- 2’s Complement+
- 81(.....)

- c. -165 =(.....)
- Jawab +165 :
- 1’st Complement :
- 2’s Complement+
- 165(.....)

d. -214 =(.....)

Jawab +5 :

1'st Complement :

2's Complement+

-214(.....)

e. -267 =(.....)

Jawab +267 :

1'st Complement :

2's Complement+

-267(.....)

7. Hitunglah aritmatika bilangan biner berikut

a. 1101 x 101 =

b. 1011 x 110 =

c. 111101 x 1011 =

d. 1010 x 111 =

e. 10111 x 1 1001 =

8. Hitunglah aritmatika bilangan biner berikut

a. 10010110 : 0011 =

b. 11110000 : 1100 =

c. 10100010 : 1001 =

d. 11101110 : 10001 =

e. 11110000 : 10000 =

5. Konversi nilai berikut kedalam bentuk yang diinginkan

• Biner → Hexadecimal 1011 1110 1100 1010 → _____

• Hexadecimal → Biner A 8 5 D → _____

BAB 4. FUNGSI DAN INTERKONEKSI KOMPUTER SECARA GENERAL

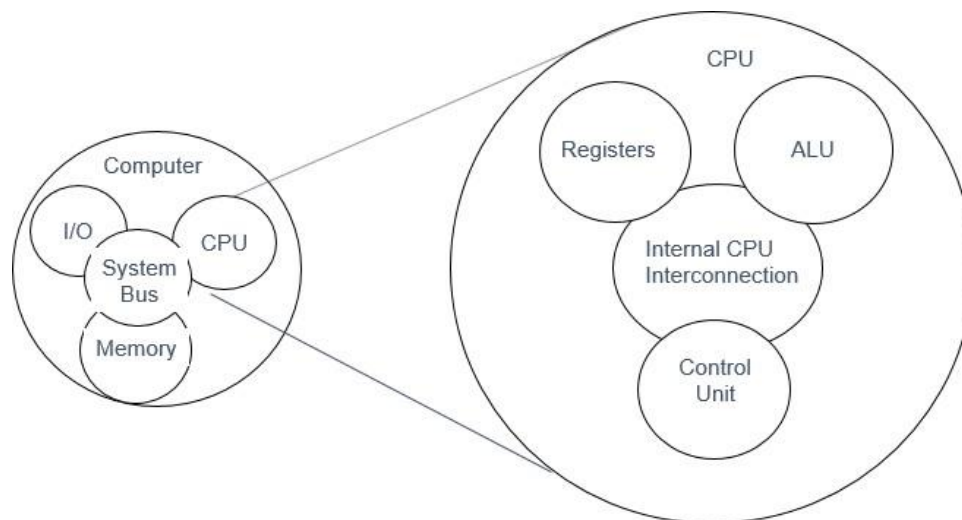
Tujuan Pembelajaran

1. Mahasiswa mengerti komponen utama CPU dan Fungsi CPU
2. Mahasiswa memahami struktur dan fungsi internal prosesor, organisasi ALU, *control unit* dan *register*
3. Mahasiswa mampu menjelaskan fungsi prosesor dalam menjalankan instruksi mesin

A. Komponen Utama Central Processing Unit (CPU)

Central Processing Unit merupakan komponen terpenting dari sistem komputer. CPU merupakan komponen pengolah data berdasarkan instruksi yang diberikan kepadanya. Dalam mewujudkan fungsi dan tugasnya, CPU tersusun atas beberapa komponen. Komponen utama Central Processing Unit terdiri atas :

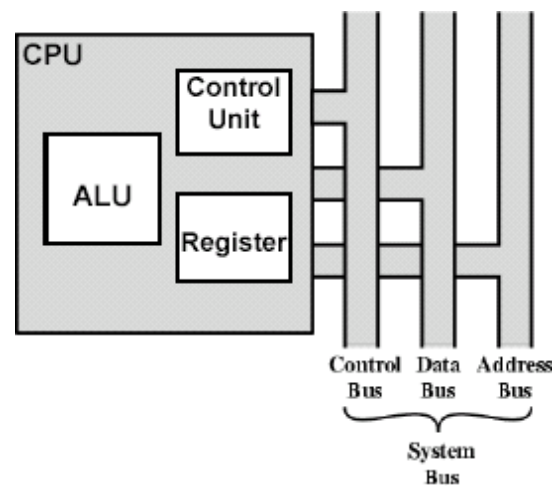
- Arithmetic and Logic Unit (ALU)
- Control Unit
- Registers
- CPU Interconnections



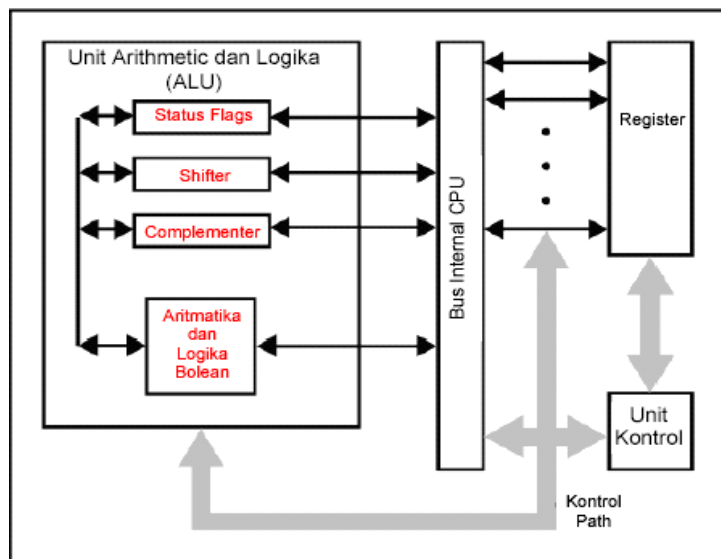
Gambar 4.1. Komponen Utama Central Processing Unit

ALU merupakan unit komputer yang melakukan pengolahan data baik data aritmetika maupun logika. ALU terdiri dari dua bagian, yaitu unit aritmetika dan unit logika boolean, yang masing – masing memiliki spesifikasi tugas tersendiri. Control Unit Bertugas mengontrol operasi CPU dan secara keseluruhan mengontrol komputer sehingga terjadi sinkronisasi kerja

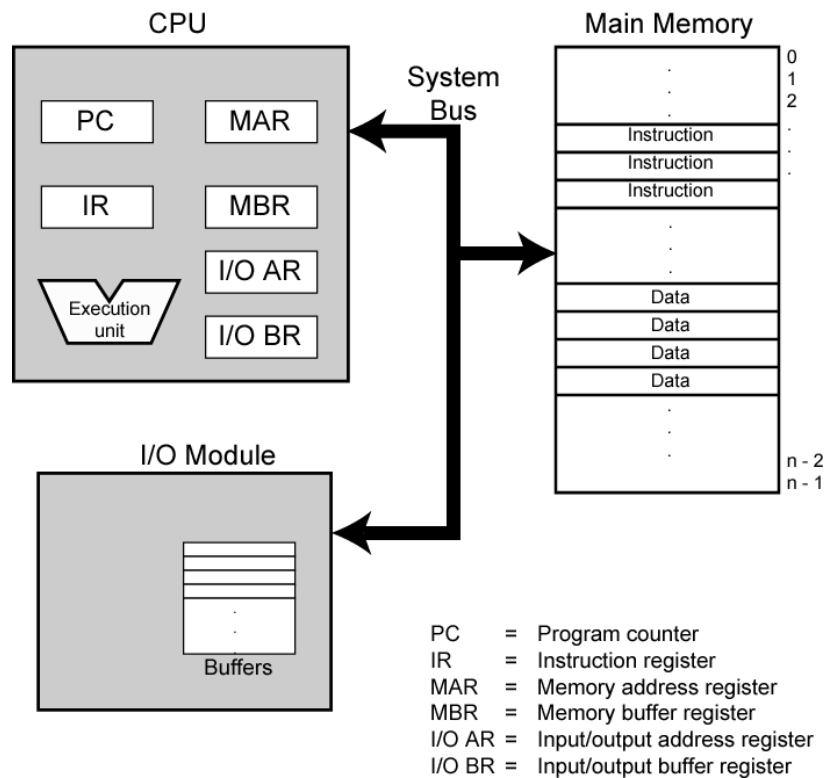
antar komponen dalam menjalankan fungsi – fungsi operasinya. Termasuk dalam tanggung jawab unit control adalah mengambil instruksi – instruksi dari memori utama dan menentukan jenis instruksi tersebut. Registers merupakan Media penyimpanan internal CPU yang digunakan saat proses pengolahan data. Memori ini bersifat sementara, biasanya digunakan untuk menyimpan data saat diolah ataupun data untuk pengolahan selanjutnya. CPU Interconnections atau Sistem koneksi dan bus yang menghubungkan komponen internal dan bus – bus eksternal CPU. Komponen internal CPU yaitu ALU, unit kontrol, dan register. Komponen eksternal CPU adalah sistem lainnya, seperti memori utama, piranti masukan/keluaran (I/O).



Gambar 4.2. Komponen internal Central Processing Unit



Gambar 4.3. Struktur detail internal CPU



Gambar 4.4. Komponen Komputer : Top Level View

B. Fungsi Komponen-komponen utama CPU

Fungsi dasar dari Central Processing Unit adalah melakukan eksekusi program. Pengertian dari program adalah set instruksi yang tersimpan dalam memori. Oleh karena itu, fungsi CPU adalah menjalankan program – program yang disimpan dalam memori utama dengan cara mengambil instruksi – instruksi, menguji instruksi tersebut dan mengeksekusinya satu persatu sesuai alur perintah. Pandangan paling sederhana dari proses eksekusi program adalah dengan mengambil pengolahan instruksi yang terdiri dari dua langkah, yaitu operasi pembacaan instruksi (fetch) dan operasi pelaksanaan instruksi (execute)

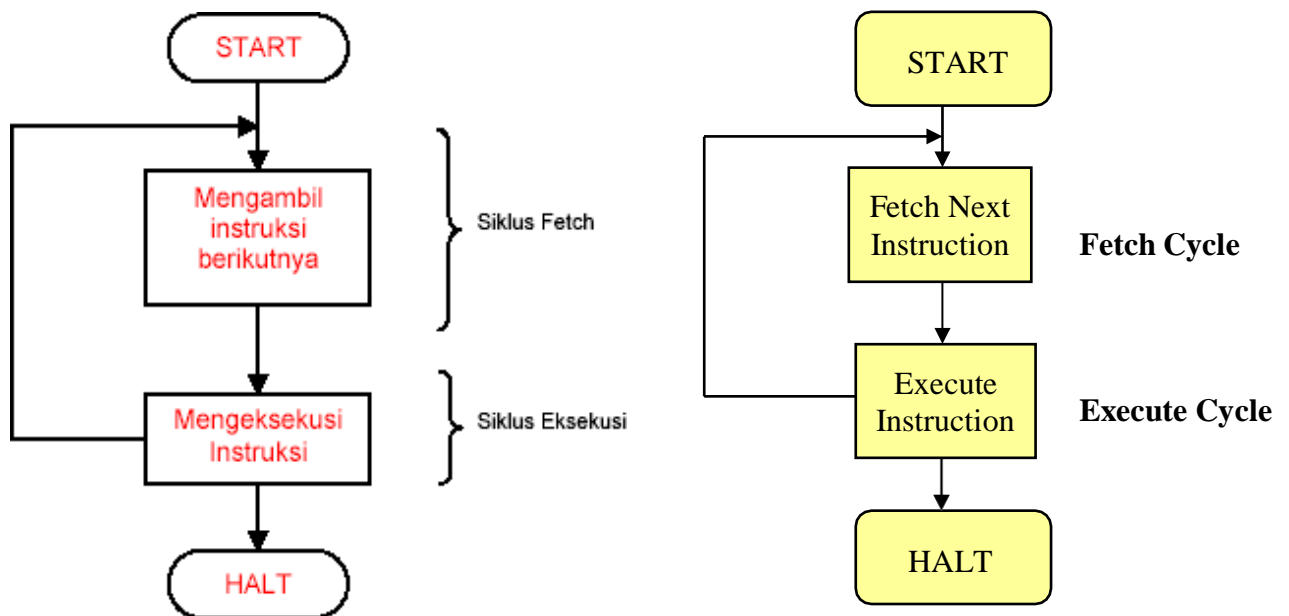
Siklus fetch terdiri atas serangkaian proses sebagai berikut.

1. Pada setiap siklus instruksi, CPU awalnya akan membaca instruksi dari memori
2. Terdapat register dalam CPU yang berfungsi mengawasi dan menghitung instruksi selanjutnya, yang disebut Program Counter (PC)
3. PC akan menambah satu hitungannya setiap kali CPU membaca instruksi
4. Instruksi – instruksi yang dibaca akan dimasukkan dalam register instruksi (IR).
5. Instruksi – instruksi ini dalam bentuk kode – kode biner yang dapat diinterpretasikan oleh CPU kemudian dilakukan aksi yg diperlukan sebagaimana penjelasan berikutnya.

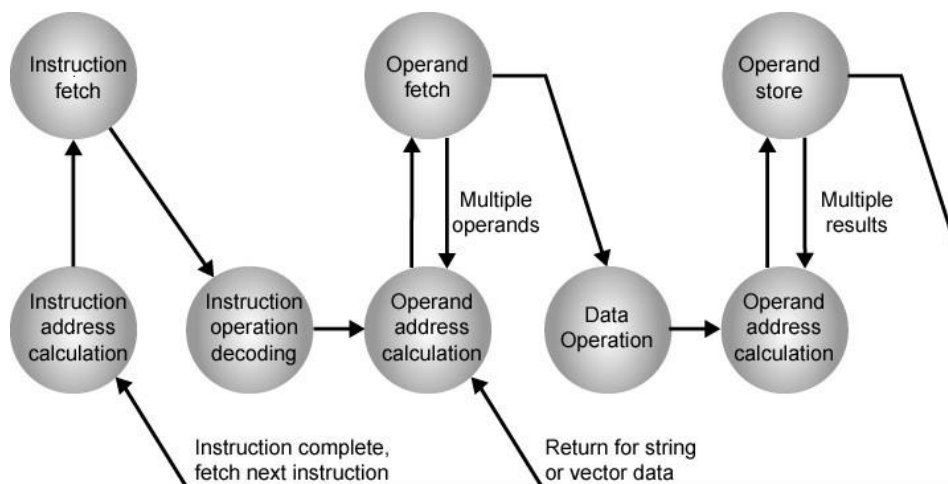
Kegiatan atau aksi CPU terbagi menjadi 4 kategori yaitu :

1. CPU – Memori, perpindahan data dari CPU ke memori dan sebaliknya
2. CPU –I/O, perpindahan data dari CPU ke modul I/O dan sebaliknya.
3. Pengolahan Data, CPU melakukan sejumlah operasi aritmatika dan logika terhadap data.
4. Kontrol, merupakan instruksi untuk pengontrolan fungsi atau kerja. Misalnya instruksi perubahan urutan eksekusi.

Siklus eksekusi sebuah instruksi terdiri atas beberapa proses meliputi *Instruction Address Calculation (IAC)*, yaitu mengkalkulasi atau menentukan alamat instruksi berikutnya yang akan dieksekusi. Biasanya melibatkan penambahan bilangan tetap ke alamat instruksi sebelumnya. Misalnya, bila panjang setiap instruksi 16 bit padahal memori memiliki panjang 8 bit, maka tambahkan 2 ke alamat sebelumnya. *Instruction Fetch (IF)*, yaitu membaca atau mengambil instruksi dari lokasi memorinya ke CPU. *Instruction Operation Decoding (IOD)*, yaitu menganalisa instruksi untuk menentukan jenis operasi yang akan dibentuk dan operand yang akan digunakan. *Operand Address Calculation (OAC)*, yaitu menentukan alamat operand, hal ini dilakukan apabila melibatkan referensi operand pada memori. *Operand Fetch (OF)*, adalah mengambil operand dari memori atau dari modul I/O. *Data Operation (DO)*, yaitu membentuk operasi yang diperintahkan dalam instruksi. *Operand store (OS)*, yaitu menyimpan hasil eksekusi ke dalam memori.



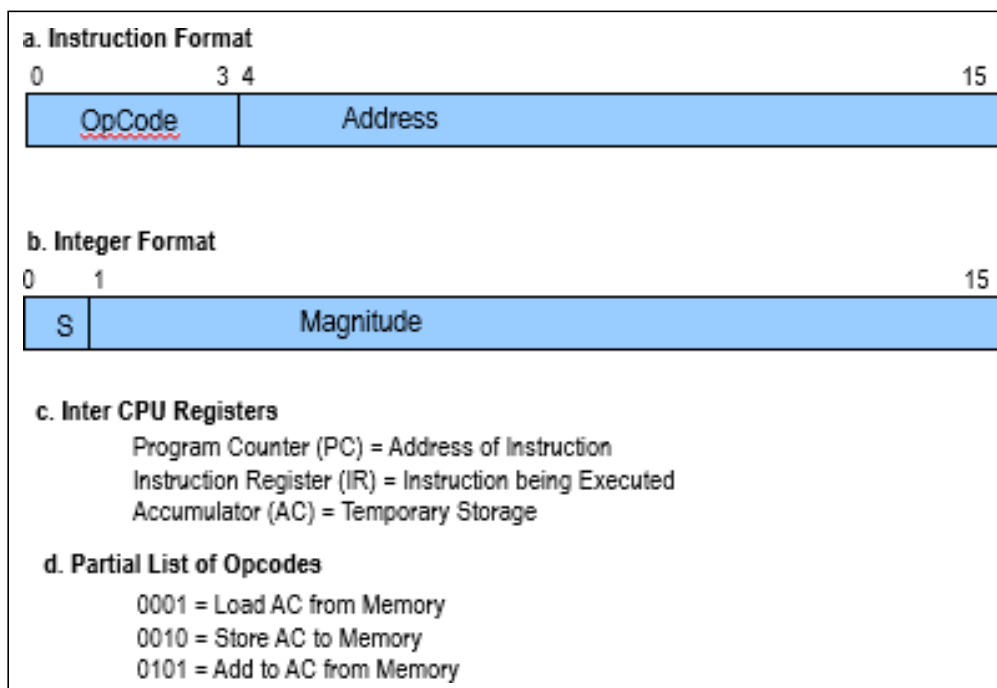
Gambar 4.5. Diagram alir siklus instruksi



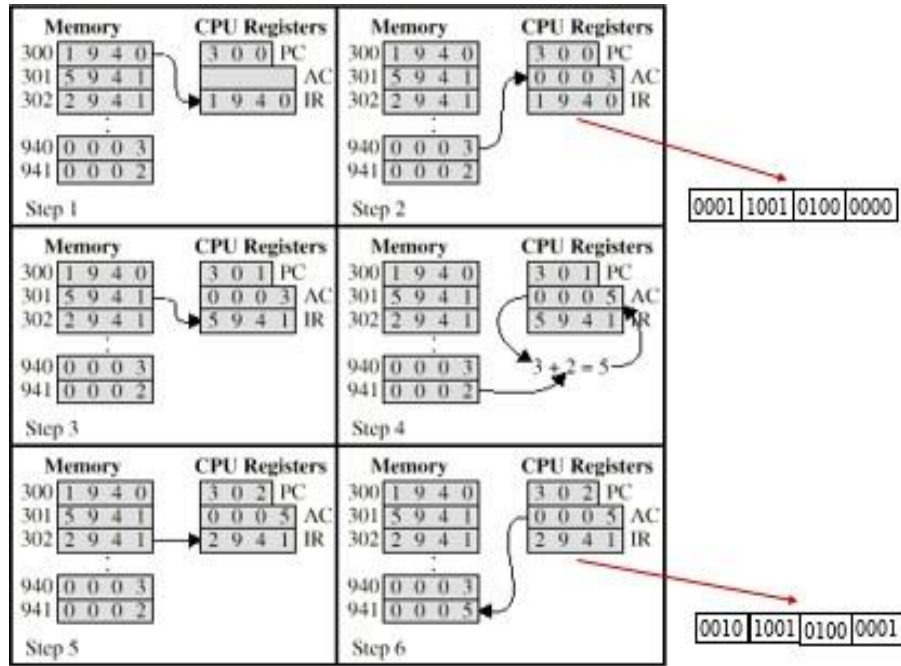
Gambar 4.6. Diagram siklus instruksi

C. Instruksi dan Kode Operasi

Format instruksi terdiri atas dua bagian yaitu *opcode* (*operation code*) dan *address operation*. Panjangnya instruksi (jumlah bit instruksi) ditentukan oleh *word length* masing-masing sistem. Gambar 4.7 dibawah menunjukkan contoh format instruksi sepanjang 16 bit, terbagi atas 4 bit opcode dan 12 bit alamat data. Kode operasi yang tampak pada gambar dibawah dimulai dari 0000 sampai dengan 1111 setiap kombinasi empat bit biner memiliki makna yang berbeda sebagaimana tampak pada bagian “d” pada gambar.



Gambar 4.7. Format Instruksi dan Kode



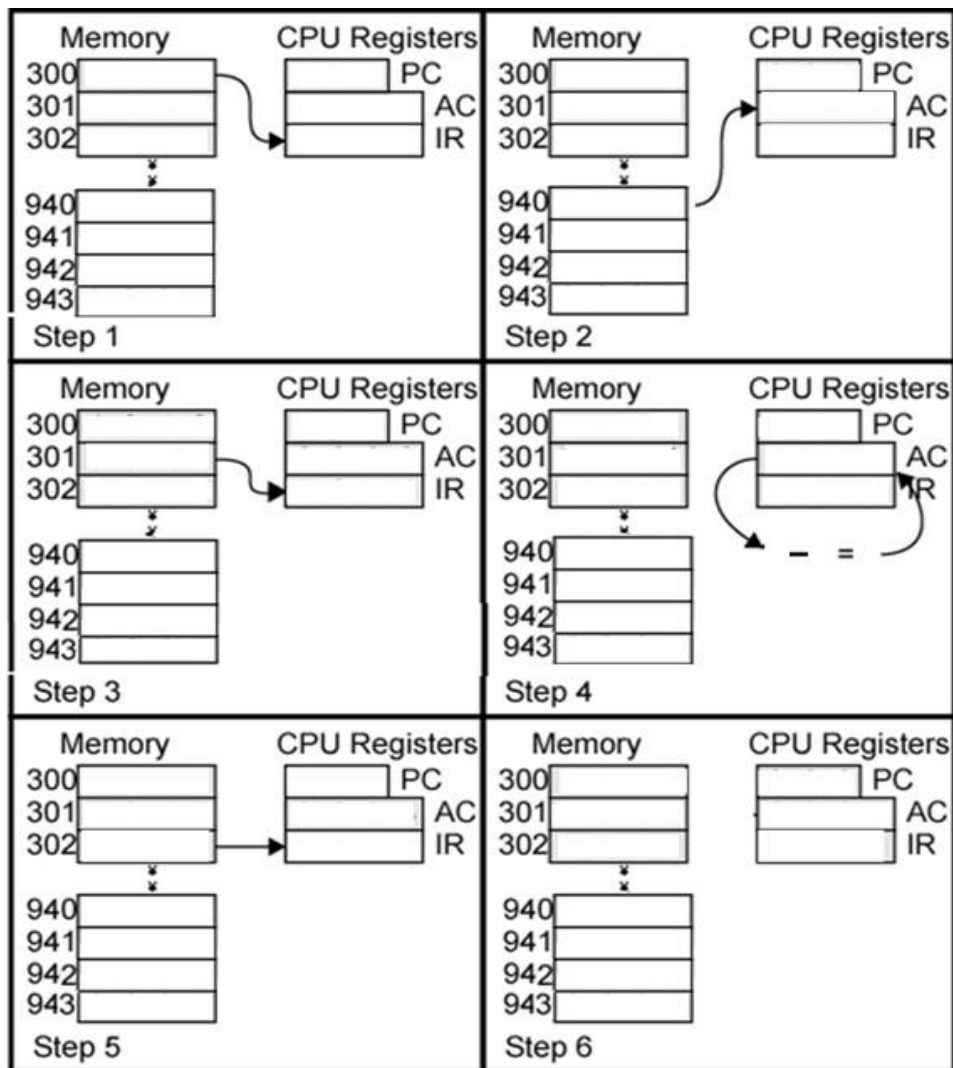
Gambar 4. 8 Contoh *execution in hypothetical machine*

Langkah-langkah yang menunjukkan eksekusi program pada mesin sebuah komputer tampak pada gambar 4.8 di atas,. Dari gambar dapat disampaikan informasi sebagai berikut. Jika diketahui bahwa panjang word = 16 bit, terbagi menjadi 4 bit op-code (terdapat 2^4 jenis kode operasi yang berbeda) dan 12 bit alamat (2^{12} lokasi memory). Selanjutnya jika diasumsikan bahwa Program Counter (PC) berada bernilai 300 yang berarti 300 adalah alamat instruksi pertama. Selanjutnya instruksi yang ada pada lokasi 300 (alamat memory dari instruksi : 300) di load ke IR. Pertama PC ke MAR, BACA, kemudian ditransfer ke MBR lalu dari MBR ke IR. Empat bit pertama (op-code) dalam IR mengindikasikan bahwa AC di load. Sementara 12 bit sisanya (angka 940) menunjukkan alamat operan, nilai 1940 adalah bilangan hexadesimal dimana angka 1 mewakili opcode dan 940 mewakili alamat memori. Instruksi selanjutnya (5941) dipanggil dari lokasi 301 dan *program counter* meningkat (misal 1 kali). Kode 5941 memiliki maksud bahwa 5 artinya “ADD” sedangkan 941 adalah alamat data yang akan dieksekusi instruksi. Isi dari accumulator (memori) yang lama serta konten yang berlokasi di alamat 941 ditambahkan, kemudian hasilnya disimpan dalam *accumulator* (memori). Instruksi selanjutnya (2941) dipanggil dari lokasi 302 dan *program counter increment* (meningkat) yang artinya simpan konten dari *accumulator* pada lokasi 941.

D. Soal-Soal

1. Soal Latihan Pengurangan

TABEL EKSEKUSI PROGRAM



Jika diketahui PC : program counter; IR : Instruction Register ; AC : Accumulator, temporary storage. Perintah terdiri dari 16 bit (4 Hexadecimal); 4 bit pertama adalah “instruksi” dan 12 bit sisanya adalah alamat memori dengan daftar opcode dan informasi sebagai berikut

<u>Daftar Opcode + lokasi alamatnya</u>	<u>Alamat + isi data Memori</u>
0001 : LOAD AC from memory; ada di alamat 300	940 adalah 0002
0110 : SUB to AC from memory; ada di alamat 301	941 adalah 0003
0010 : STORE AC to memory; ada di alamat 302	942 adalah 0004

Isikan informasi yang diketahui berikut pada tabel eksekusi program yang disediakan. Pada step pertama : program counter bernilai 300; (alamat instruksi pertama 300) sistem memanggil intruksi pertama dengan nilai 1942 dan diisikan (di-load) ke dalam IR. [1 artinya load –lihat kode biner di tabel ; 942 artinya alamat data ada di lokasi 942 → dimana data pada alamat memory 942 berisi 004]. Pada step 2 : Sistem mengeksekusi intruksi dengan kode 1942 artinya load (memproses) data di lokasi 942 ; data bernilai 004 ; ke dalam AC. Pada step 3 : Intruksi selanjutnya diambil dari lokasi 301 ; kode instruksi “SUBkonten AC dengan konten dilokasi 941”. Pada step 4 : Konten dari AC dan konten dilokasi941 di eksekusi dengan instruksi “SUB” lalu hasilnya disimpan dalam AC (*accumulator*). Pada step 5: Intruksi STORE diambil dari lokasi 302, yang artinya “STORE hasil perhitungan aritmetika ke memory 943”. Pada step 6 : Simpan hasil perhitungan dari AC ke lokasi sesuai instruksi pada step 5 (alamat 943).

2. Soal Latihan Penjumlahan

Jika diketahui PC : program counter; IR : Instruction Register ; AC : Accumulator, temporary storage. Perintah terdiri dari 16 bit (4 Hexadecimal); 4 bit pertama adalah “instruksi” dan 12 bit sisanya adalah alamat memory. dengan daftar opcode dan informasi sebagai berikut.

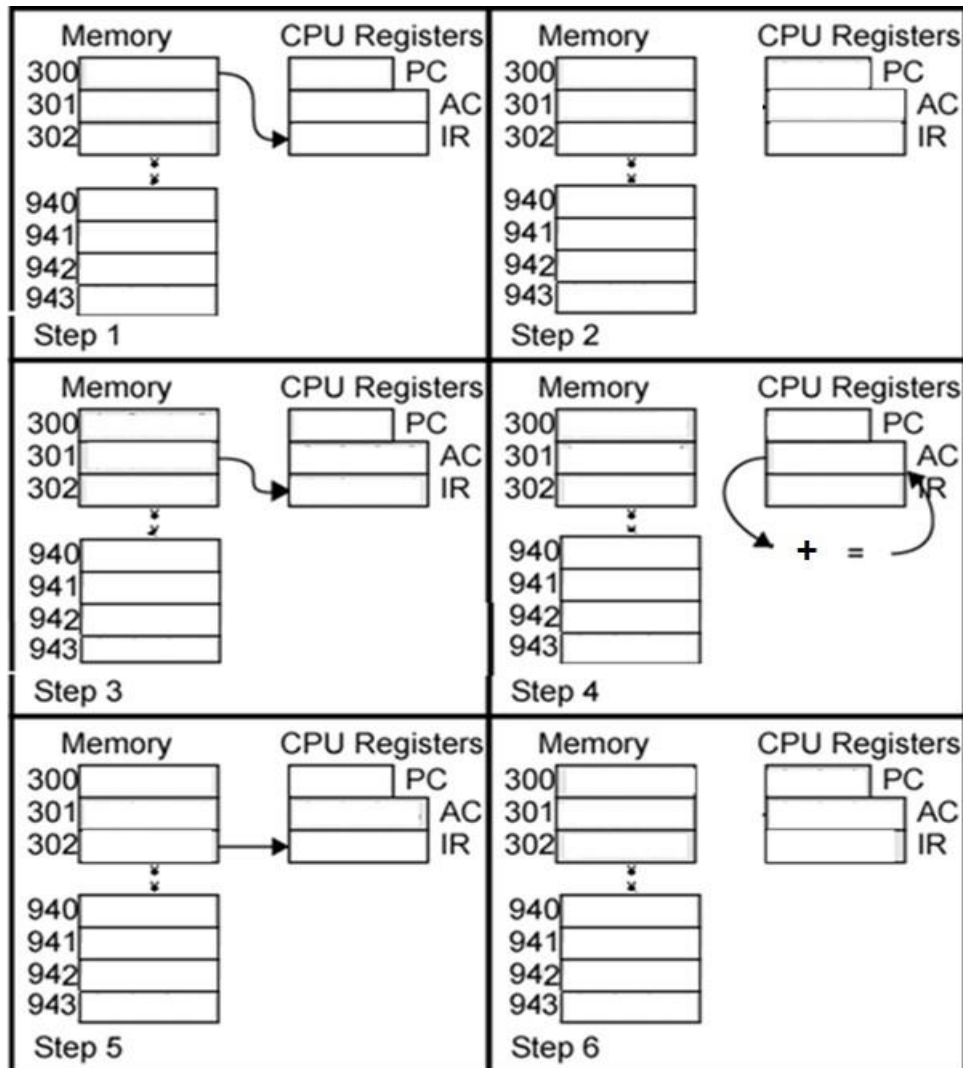
Daftar Opcode + alamat nya	Alamat + isi data Memori
0001 : LOAD AC from memory; ada di alamat 300	940 adalah 0002
0101 : ADD to AC from memory; ada di alamat 301	941 adalah 0003
0010 : STORE AC to memory; ada di alamat 302	942 adalah 0004

Isikan informasi yang diketahui pada tabel eksekusi program yang disediakan

- Pada step 1: program counter bernilai 300; (alamat instruksi pertama 300) sistem memanggil intruksi pertama dengan nilai 1940 dan diisikan (di-load) ke dalam IR.
- Pada step 2 : Sistem mengeksekusi intruksi dengan kode 1940 ke dalam AC
- Pada step 3 : Intruksi selanjutnya diambil dari lokasi 301 ; kode instruksi “ADD konten AC dengan konten dilokasi 942”
- Pada step 4 : Konten dari AC dan konten dilokasi 942 di eksekusi dengan instruksi “ADD” lalu hasilnya disimpan dalam AC
- Pada step 5: Intruksi STORE diambil dari lokasi 302, yang artinya “STORE hasil perhitungan aritmetika ke memory 943”

- Pada step 6 : Simpan hasil perhitungan dari AC ke lokasi sesuai instruksi pada step 5 (alamat 943)

TABEL EKSEKUSI PROGRAM



3. Soal Latihan Perkalian

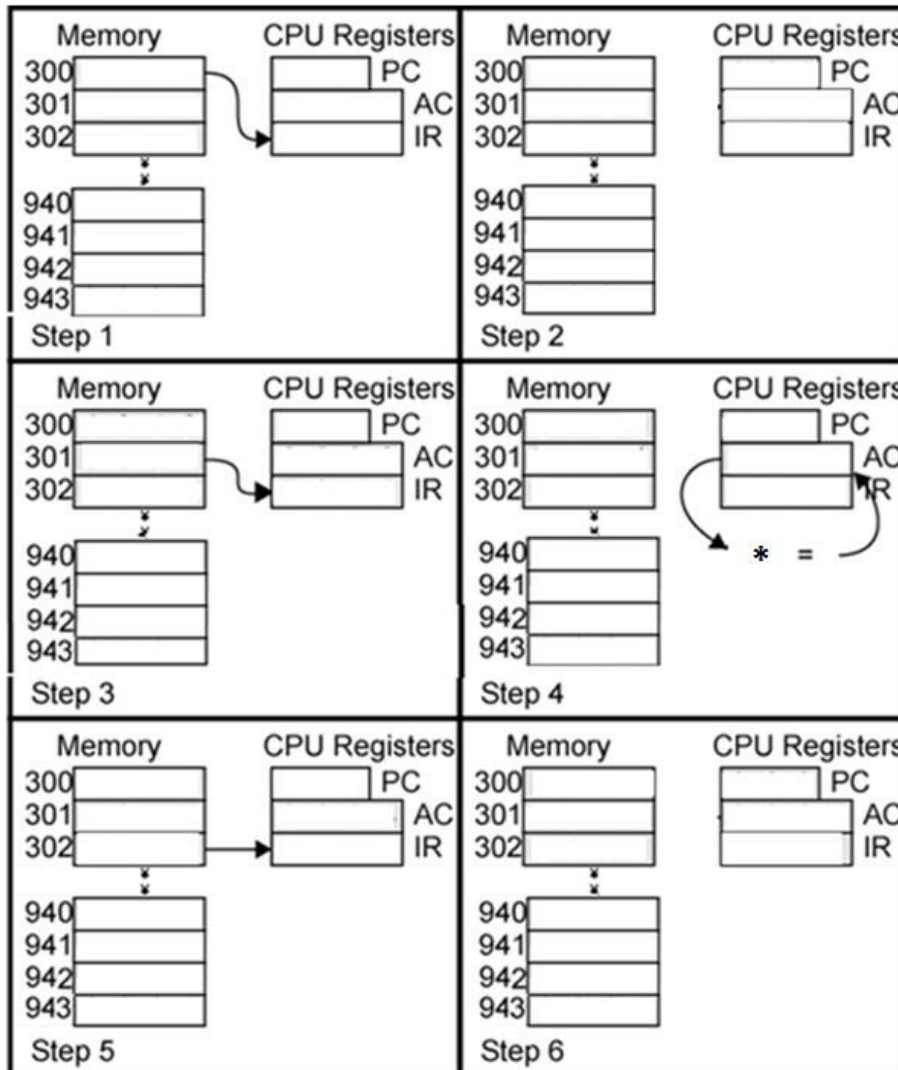
Jika diketahui PC : program counter; IR : Instruction Register ; AC : Accumulator, temporary storage. Perintah terdiri dari 16 bit (4 Hexadecimal); 4 bit pertama adalah “instruksi” dan 12 bit sisanya adalah alamat memori dengan daftar opcode dan informasi sebagai berikut.

Daftar Opcode + alamat nya	Alamat + isi data Memori
0001 : LOAD AC from memory; ada di alamat 300	940 adalah 0002
1011 : MUL to AC from memory; ada di alamat 301	941 adalah 0003
0010 : STORE AC to memory; ada di alamat 302	942 adalah 0004
	943 adalah 0005

Isikan informasi yang diketahui pada tabel eksekusi program yang disediakan

- Pada step 1: program counter bernilai 300; (alamat instruksi pertama 300) sistem memanggil intruksi pertama dengan nilai 1940 dan diisikan (di-load) ke dalam IR.
- Pada step 2 : Sistem mengeksekusi intruksi dengan kode 1940 ke dalam AC
- Pada step 3 : Intruksi selanjutnya diambil dari lokasi 301 ; kode instruksi “MUL konten AC dengan konten dilokasi 941”
- Pada step 4 : Konten dari AC dan konten dilokasi 941 di eksekusi dengan instruksi “MUL” lalu hasilnya disimpan dalam AC
- Pada step 5: Intruksi STORE diambil dari lokasi 302, yang artinya “STORE hasil perhitungan aritmetika ke memory 942”
- Pada step 6 : Simpan hasil perhitungan dari AC ke lokasi sesuai instruksi pada step 5 (alamat 942)

TABEL EKSEKUSI PROGRAM



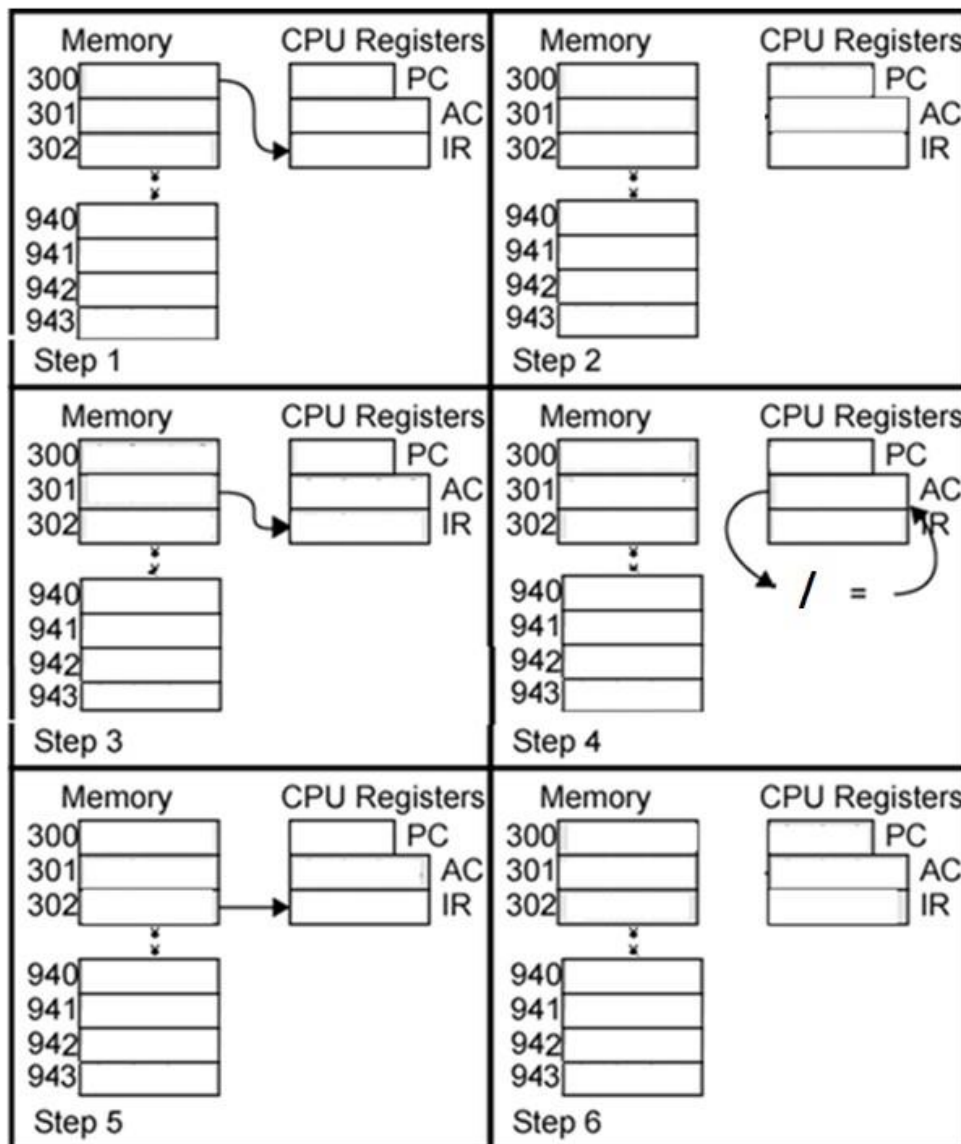
4. Soal Latihan 4 Pembagian

Jika diketahui PC : program counter; IR : Instruction Register ; AC : Accumulator, temporary storage. Perintah terdiri dari 16 bit (4 Hexadecimal); 4 bit pertama adalah “instruksi” dan 12 bit sisanya adalah alamat memory. dengan daftar opcode dan informasi sebagai berikut.

Daftar Opcode + alamat nya	Alamat + isi data Memori
0001 : LOAD AC from memory; ada di alamat 300	940 adalah 0002
1100 : DIV to AC from memory; ada di alamat 301	941 adalah 0003
0010 : STORE AC to memory; ada di alamat 302	942 adalah 0004
	943 adalah 0005

Isikan informasi yang diketahui pada tabel eksekusi program yang disediakan

- Pada step 1: program counter bernilai 300; (alamat instruksi pertama 300) sistem memanggil intruksi pertama dengan nilai 1942 dan diisikan (di-load) ke dalam IR.
- Pada step 2 : Sistem mengeksekusi intruksi dengan kode 1942 ke dalam AC
- Pada step 3 : Intruksi selanjutnya diambil dari lokasi 301 ; kode instruksi “DIV konten AC dengan konten dilokasi 940”
- Pada step 4 : Konten dari AC dan konten dilokasi 940 di eksekusi dengan instruksi “DIV” lalu hasilnya disimpan dalam AC
- Pada step 5: Intruksi STORE diambil dari lokasi 302, yang artinya “STORE hasil perhitungan aritmetika ke memory 943”
- Pada step 6 : Simpan hasil perhitungan dari AC ke lokasi sesuai instruksi pada step 5 (alamat 943)



BAB 5. INTERUPSI

A. Pengertian dan Fungsi Interupsi

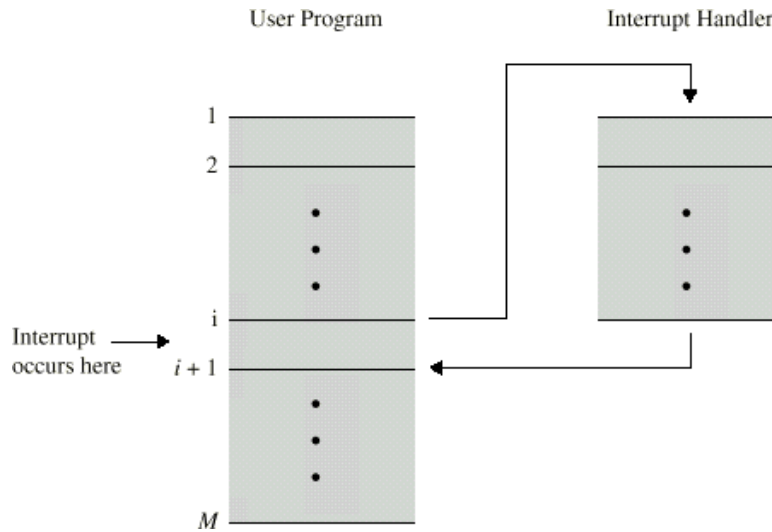
Interupsi merupakan suatu mekanisme dimana komponen dari sistem komputer / modul komputer seperti Perangkat I/O, Memory, dan lain lain menginterupsi urutan normal sebuah proses dari CPU karena terjadi kondisi yang sangat penting. Interupsi merupakan mekanisme penghentian atau pengalihan pengolahan instruksi dalam CPU kepada interupsi rutin. Hampir semua modul (memori dan I/O) memiliki mekanisme yang dapat menginterupsi kerja CPU.

Secara umum fungsi interupsi adalah untuk manajemen pengeksekusian instruksi rutin agar efektif dan efisien antara CPU, modul – modul I/O serta memori. Setiap komponen komputer dapat menjalankan tugasnya secara bersamaan, meskipun kecepatan eksekusi setiap modul berbeda, akan tetapi kendali tetap terletak pada CPU. Selanjutnya pada interupsi terdapat pembagian kelas sinyal sebagai berikut

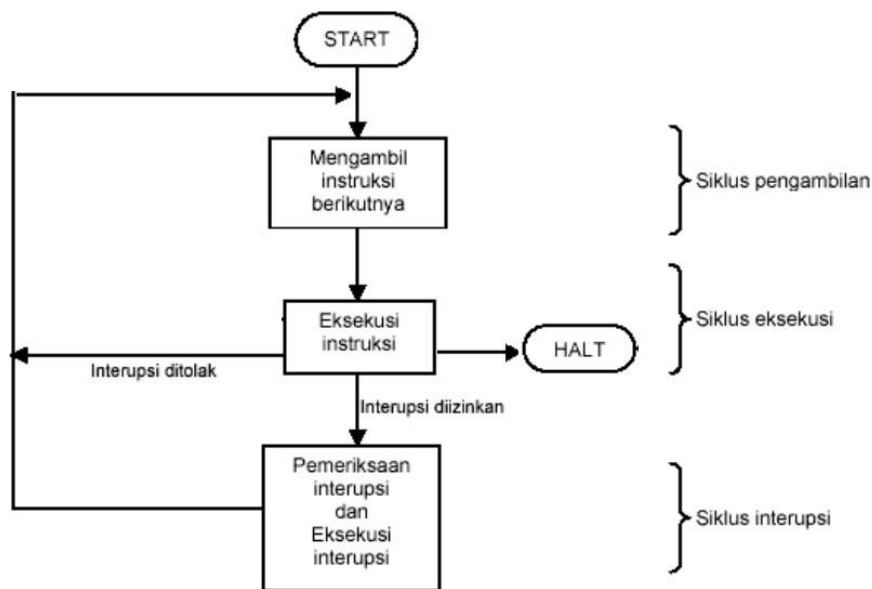
- *Program*, yaitu interupsi yang dibangkitkan dengan beberapa kondisi yang terjadi pada hasil eksekusi program. Contohnya: arimatika overflow, pembagian nol, operasi ilegal.
- *Timer*, adalah interupsi yang dibangkitkan karena pewaktuan internal (pewaktuan dalam prosesor). Sinyal ini memungkinkan sistem operasi menjalankan fungsi tertentu secara reguler.
- *I/O*, sinyal interupsi yang dibangkitkan oleh modul I/O sehubungan pemberitahuan kondisi error dan penyelesaian suatu operasi.
- *Hardware failure*, adalah interupsi yang dibangkitkan oleh kegagalan daya atau kesalahan paritas memori.

B. Proses Interupsi

Adanya mekanisme interupsi, prosesor dapat digunakan untuk mengeksekusi instruksi–instruksi lain. Saat suatu modul telah selesai menjalankan tugasnya dan siap menerima tugas berikutnya maka modul ini akan mengirimkan permintaan interupsi ke prosesor. Kemudian prosesor akan menghentikan eksekusi yang dijalankannya untuk menhandel interupsi rutin. Setelah program interupsi selesai maka prosesor akan melanjutkan eksekusi programnya kembali. Saat sinyal interupsi diterima prosesor ada dua kemungkinan tindakan, yaitu interupsi diterima/ditanggihkan dan interupsi ditolak



Gambar 5.1 Proses Interupsi *transfer of command*



Gambar 5.2. diagram alir siklus eksekusi oleh prosesor dengan adanya fungsi interupsi

C. Interupsi ganda (*multiple interrupt*)

Interupsi ganda sering terjadi pada sistem operasi kompleks, misalnya suatu komputer akan menerima permintaan interupsi saat proses pencetakan dengan printer selesai, disamping itu dimungkinkan dari saluran komunikasi akan mengirimkan permintaan interupsi setiap kali data tiba. Jika pada saat bersamaan terdapat banyak interupsi pada program yang dijalankan oleh CPU, maka reaksi prosesor atas sinyal interupsi yang diterimanya ada dua kemungkinan

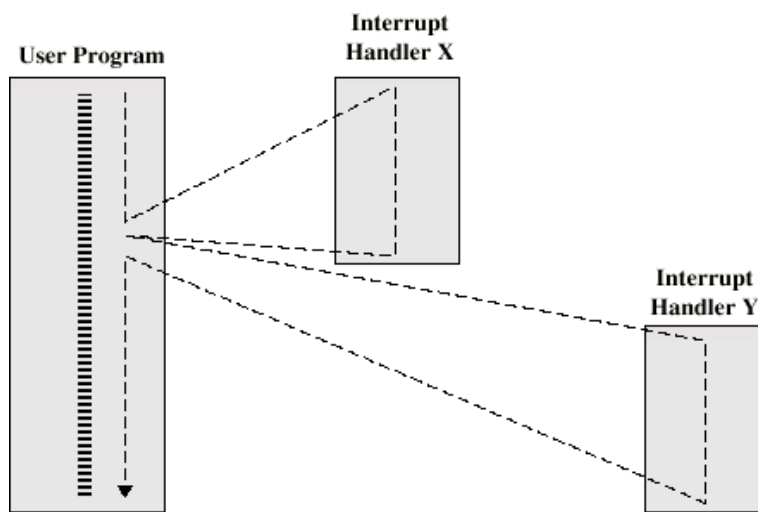
tindakan, yaitu interupsi diterima/ditanggguhkan dan interupsi ditolak. Terdapat dua pendekatan Interupsi ganda yaitu

1. Pendekatan pengolahan interupsi berurutan / Sekuensial

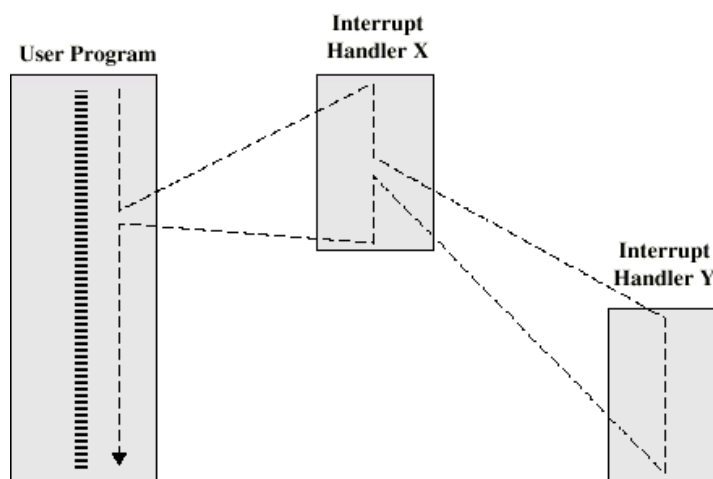
Pendekatan interupsi ini dilakukan denolak atau tidak mengizinkan interupsi lain saat suatu interupsi ditangani prosesor. Setelah prosesor selesai menangani suatu interupsi maka interupsi lain baru di tangani.

2. Pengolahan interupsi bersarang yaitu mendefinisikan prioritas bagi interupsi

Interrupt handler mengizinkan interupsi berprioritas lebih tinggi ditangani terlebih dahulu



Gambar 5.3. *Multiple Interrupts – Sequential*



Gambar 5.4. *Multiple Interrupts – Nested*

Pada kasus *multiple interrupt* dengan pendekatan pengolahan interupsi berurutan/ Sequential, maka ketika interupsi interupsi ditolak Apa yang dilakukan prosessor? Interupsi ditolak atau di-*disable* terjadi ketika processor mengabaikan interupsi berikutnya karena processor sedang memproses satu interupsi. Interupsi berikutnya ditunda (pending) dan dilakukan pengecekan setelah interupsi pertama diproses. Pada kondisi ini (*Disable interrupts*) Interupsi ditangani sesuai urutan kejadian. Sedangkan pada kasus *multiple interrupt* dengan pendekatan bersarang atau *nested*, maka ketika knterupsi interupsi diterima/ditangguhkan Apa yang dilakukan Prosessor? Prosesor menangguhkan eksekusi program yang dijalankan dan menyimpan konteksnya. Tindakan yang dilakukan processor adalah menyimpan alamat instruksi dan data lain yang relevan yang akan dieksekusi berikutnya. Prosesor menyetel program counter (PC) ke alamat awal *routine interrupt handler*. Pada kondisi ini interupsi dengan prioritas rendah dapat diinterupsi dengan interupsi yang memiliki prioritas lebih tinggi. Setelah interupsi dengan prioritas lebih tinggi selesai diproses maka program akan kembali menjalankan interupsi prioritas rendah.

D. Soal dan Pembahasan

Suatu sistem memiliki tiga perangkat I/O: printer, disk, dan saluran komunikasi, masing – masing prioritasnya 2, 4 dan 5. Bagaimana proses interupsinya?

- Pada awal sistem melakukan pencetakan dengan printer, saat itu terdapat pengiriman data pada saluran komunikasi sehingga modul komunikasi meminta interupsi.
- Proses selanjutnya adalah pengalihan eksekusi interupsi modul komunikasi, sedangkan interupsi printer ditangguhkan.
- Saat pengekseskusion modul komunikasi terjadi interupsi disk, namun karena prioritasnya lebih rendah maka interupsi disk ditangguhkan.
- Setelah interupsi modul komunikasi selesai akan dilanjutkan interupsi yang memiliki prioritas lebih tinggi, yaitu disk.
- Bila interupsi disk selesai dilanjutkan eksekusi interupsi printer. Selanjutnya dilanjutkan eksekusi program utama

BAB 6. SET INSTRUKSI

A. Karakteristik Instruksi Mesin

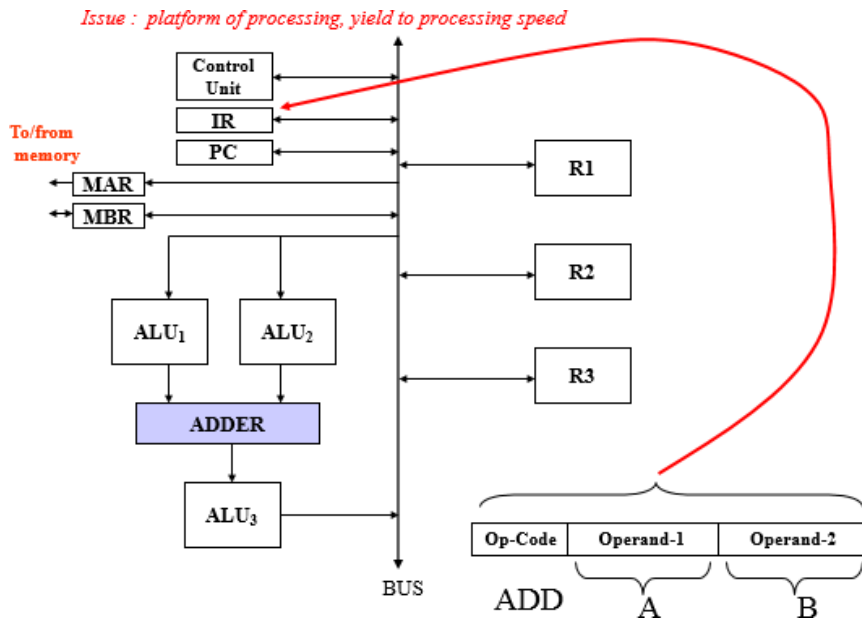
Instruction set atau set instruksi merupakan kumpulan berbagai jenis instruksi dari CPU tertentu. Sedangkan elemen instruksi mesin merupakan setiap instruksi harus berisi informasi yang dibutuhkan oleh CPU untuk eksekusi. Elemen instruksi biasanya terdiri atas :

- Kode operasi berfungsi sebagai penentu operasi
- Sumber operan sebagai referensi data
- Hasil operand sebagai referensi data
- Instruksi selanjutnya yang akan dieksekusi (bisa diberikan dalam instruksi saat ini atau secara implisit diberikan - instruksi selanjutnya setelah instruksi saat ini).

Operan sumber dan operan hasil dapat berada di tiga bagian yaitu memori utama atau memori virtual yang merupakan alamat yang harus diberikan ketika sebuah instruksi dibuat. Selanjutnya yang kedua adalah register CPU baik register eksplisit maupun implisit. Termasuk jenis register eksplisit adalah pada penggunaan dua atau tiga register, sedangkan instruksi dengan satu register (satu instruksi operan), kemungkinan besar yang dilibatkan adalah register implisit (biasanya untuk operasi aritmatika) akumulator secara implisit dihubungkan sebagai operan hasil. Jenis operan sumber dan hasil yang ketika adalah perangkat I/O yang digunakan untuk instruksi I/O. Instruksi harus menentukan perangkat I/O atau modul yang digunakan, atau pada kasus memory-mapped I/O maka operan tujuan adalah lokasi memori lain.

Sementara itu, seperti yang dipelajari pada mesin IAS, instruksi biasanya diwakili dengan pernyataan-pernyataan yang sifatnya mudah dipahami atau mudah diingat oleh manusia (mnemonik). Beberapa contoh instruksi adalah sebagai berikut :

ADD	artinya “tambahkan”
SUB	artinya “kurangkan”
DIV	artinya “bagi”
MUL	artinya “kalikan”
ADD R, Y,	artinya tambahkan data R (operan pertama) dan Y (operan kedua)



Gambar 6.1 Processor Organization

B. Tipe Instruksi dan Jumlah Alamat

Tipe-tipe instruksi pada mesin komputer terbagi atas empat jenis yang meliputi instruksi untuk pengolahan data aritmatika dan instruksi logika, penyimpanan data yang merupakan instruksi memori, pergerakan Data dari dan menuju I/O atau biasa disebut sebagai instruksi I/O, Kontrol atau pengendalian yang merupakan instruksi engujian dan instruksi cabang serta aritmatika yang mampu memberikan kemampuan komputasi. Jumlah alamat pada instruksi menunjukkan jumlah operan yang dapat dilibatkan dalam operasi tertentu: bisa jadi satu, dua atau tiga, atau bahkan nol (*stack machine*).

Contoh *Instruction Set* pada mesin DLX (Hennessy & Patterson) adalah ADD R1, R2, R3 (instruksi RR) atau ADD R1, R2, A (instruksi RX). Kemudian, contoh lain untuk instruksi yang melibatkan tiga alamat operan adalah sebagai berikut. Buatlah set instruksi atau program untuk mengeksekusi persamaan $Y = (A-B) / (C + D * E)$ dengan menggunakan tiga alamat instruksi:

Instruksi		Komentar
SUB	Y,A,B	$Y \leftarrow A - B$
MPY	T,D,E	$T \leftarrow D \times E$
ADD	T,T,C	$T \leftarrow T + C$
DIV	Y,Y,T	$Y \leftarrow Y / T$

Program aritmatika diatas hanya terdiri atas empat set instruksi, dalam program ini terlihat bahwa word length untuk alamat operan lebih panjang ketika menggunakan tiga alamat yang terlibat sekaligus dalam setiap instruksi mesin. Sedangkan instruksi yang melibatkan dua buah alamat untuk permasalahan yang sama yaitu $Y = (A - B) / (C + D * E)$ dapat dilihat pada contoh di bawah ini.

<u>Instruksi</u>		<u>Komentar</u>
MOVE	Y,A	$Y \leftarrow A$
SUB	Y,B	$Y \leftarrow Y - B$
MOVE	T,D	$T \leftarrow D$
MPY	T,E	$T \leftarrow T * E$
ADD	T,C	$T \leftarrow T + C$
DIV	Y,T	$Y \leftarrow Y / T$

Dalam program dengan dua alamat tersebut diatas tampak bahwa program terdiri atas enam instruksi, dimana pada saat jumlah alamat yang terlibat berkurang (dua alamat, dibandingkan contoh sebelumnya tiga alamat) maka instruksi semakin panjang.

Penyelesaian persamaan $Y = (A - B) / (C + D * E)$ dengan menggunakan satu alamat instruksi dapat dilihat pada contoh program di bawah ini

<u>Instruksi</u>		<u>Komentar</u>
LOAD	D	$AC \leftarrow D$
MPY	E	$AC \leftarrow AC * E$
ADD	C	$AC \leftarrow AC + C$
STOR	Y	$Y \leftarrow AC$
LOAD	A	$AC \leftarrow A$
SUB	B	$AC \leftarrow AC - B$
DIV	Y	$AC \leftarrow AC / Y$
STOR	Y	$Y \leftarrow AC$

Semakin sedikit alamat instruksi maka program semakin panjang, tampak pada contoh di atas dibutuhkan delapan baris set instruksi untuk menyelesaikan satu persoalan yang sama dengan

dua contoh sebelumnya. Selanjutnya, persamaan $Y = (A-B)/(C+D*E)$ dapat diselesaikan dengan menggunakan instruksi zero address seperti berikut.

<u>Instruksi</u>	<u>Komentar</u>
PUSH A	
PUSH B	
SUB	(stack yang bawah dikurangi dengan data pada stack atas)
PUSH D	
PUSH E	
MUL	
PUSH C	
ADD	
DIV	(stack yang bawah dibagi dengan data pada stack atas)
POP Y	(data disimpan atau dikeluarkan pada operan Y)

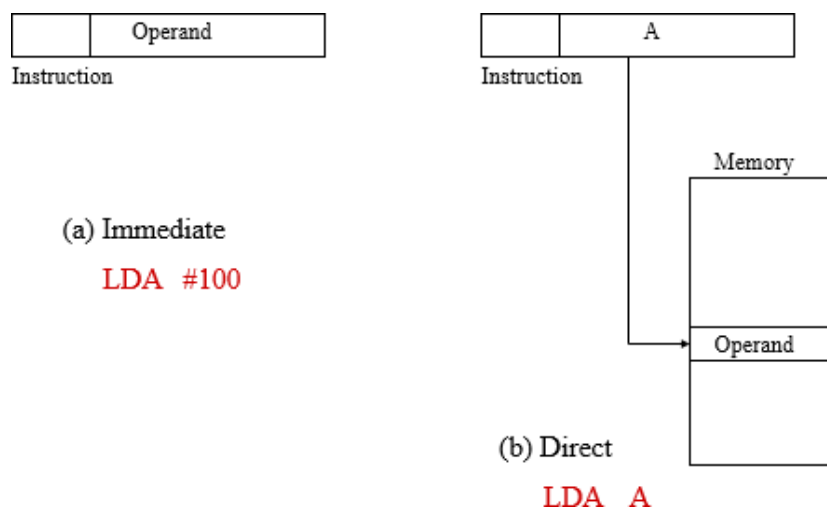
Jumlah set instruksi dengan zero address paling banyak dibandingkan dengan satu, dua atau tiga alamat. Hal yang perlu diperhatikan ketika membuat program dengan *zero address instruction* adalah bahwa data yang pertama di input masuk ke stack terlebih dahulu diikuti dengan instruksi-instruksi selanjutnya, sehingga ketika peletakan instruksi tidak pas akan mengakibatkan program tidak berjalan sebagaimana kebutuhan.

C. Desain set instruksi

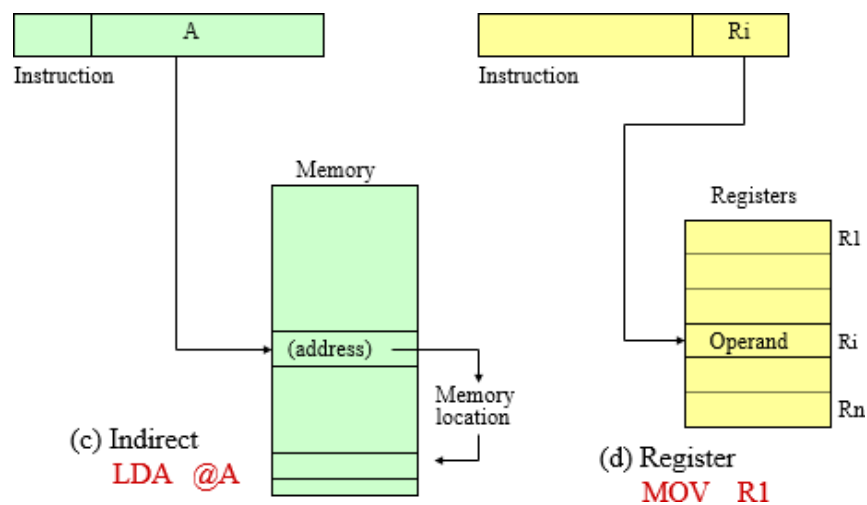
Fakta tentang desain set instruksi ternyata masih dalam perselisihan. Masalah desain yang mendasari perselisihan tersebut meliputi *operation repertoire* yang menyangkut pembahasan tentang sejauh mana tingkat kerumitan sebuah instruksi, tipe data yang menyangkut pertanyaan apakah orang membutuhkan semua jenis data, format instruksi yang berkaitan dengan format jenis tetap atau variabel kemudian ukuran panjang instruksi serta jumlah operan yang terlibat. Dua pembahasan yang menjadi perselisihan selanjutnya berkaitan dengan register dan pengalamatan. Pada register ukuran serta tujuan pemanfaatannya masih diperdebatkan termasuk pada pengalamatan tentang berapa banyak pengalamatan yang dibutuhkan beserta mode pengalamatannya. Sementara itu hal lain yang perlu diperhatikan dalam desain instruksi adalah tipe-tipe operan yang terlibat pada sebuah set instruksi. Instruksi mesin bekerja pada data dengan kategori data yang paling penting adalah alamat, angka, karakter serta data logika. Angka yang ada pada set instruksi mesin bisa dalam bentuk integer, fixed point atau floating point maupun desimal. Sedangkan karakter yang biasa digunakan dalam pengolahan data bisa dalam bentuk karakter ASCII atau EBCDIC tergantung mesinnya

D. Format dan Mode Pengalamatan

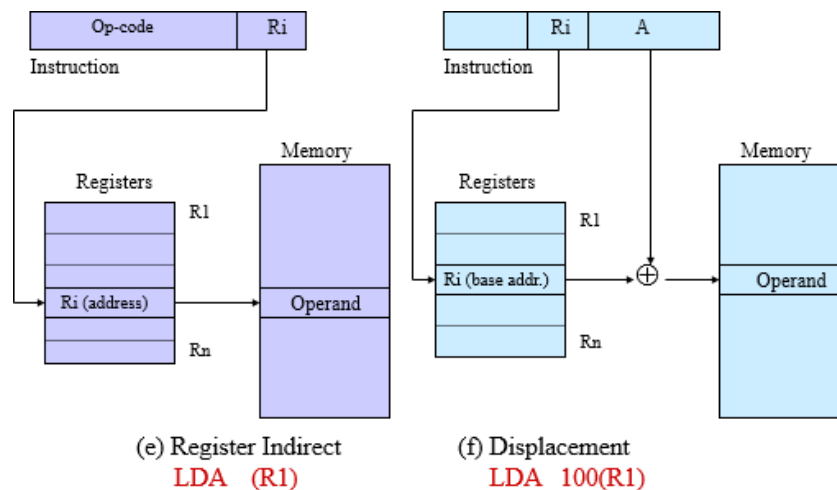
Instruksi dirancang untuk dapat merujuk ke berbagai lokasi di memori utama. Berbagai teknik pengalamatan telah digunakan untuk mencapai akses tersebut. Pertukaran antara rentang alamat dan / atau kompleksitas perhitungan alamat sangat perlu diperhatikan. Jenis-jenis teknik pengalamatan instruksi meliputi pengalamatan segera, langsung, tidak langsung, daftar, daftar tidak langsung, perpindahan, tumpukan (stack). Sementara permasalahan yang ada terdiri atas flexibility lokasi operan, trade off antara fleksibilitas lokasi dan kecepatan pemrosesan. Diagram untuk berbagai mode pengalamatan tampak pada gambar berikut.



Gambar 6.2. Metode akses memori secara immediate dan direct



Gambar 6.3. Metode akses memori secara indirect dan register direct



Gambar 6.4. Metode akses register secara indirect dan displacement

Contoh penulisan dari mode pengalamatan atau addressing mode adalah sebagai berikut.

- *Immediate* : ADD #100 ; add 100 to Accum.
- *Direct*: ADD A ; add content of addr. A to accumulator
- *Indirect* : ADD @A ; content of location A is address of operand. add content of loc. with that address to accum.
- *Register Direct*: ADD R1 ; add content of R1 to accumulator
- *Reg. Indirect* : ADD (R1) ; add content of mem. Location whose addr. is in R1 to acc.

Pengaksesan alamat dengan displacement merupakan metode yang "langka", namun masih mungkin terjadi sebagaimana contoh ini. ADD 1050 (R1) dimana alamat dasar (asal) adalah berada di R1 dan *displacement*-nya adalah 1050. Dimana dalam instruksi dua alamat (two address instruction), akan terlihat seperti ADD 1050 (R1), R2 yang artinya tambahkan data di R2 dengan data yang alamat dasarnya ada di R1 dan *displacement*-nya 1050. Tiga displacement addressing yang umum meliputi pengalamatan relative (*relative addressing*), pengalamatan register dasar (*base register addressing*) serta pengindeksan (*indexing*). Pada pengalamatan relative, alamat operan merupakan jumlah alamat dalam instruksi ditambah isi dari program counter (PC). Pada pengalamatan register dasar, isi base-register ditambahkan ke alamat operan dalam instruksi. Sedangkan pada skema pengindeksan, bidang alamat instruksi ditambahkan ke isi register indeks untuk mendapatkan alamat memori yang tepat (dari operan). Tiga skema alamat tersebut membuat register untuk menunjuk ke beberapa segmen memori tertentu dan menggunakan alamat operan dalam instruksi sebagai *displacement*.

BAB 8. MEMORI

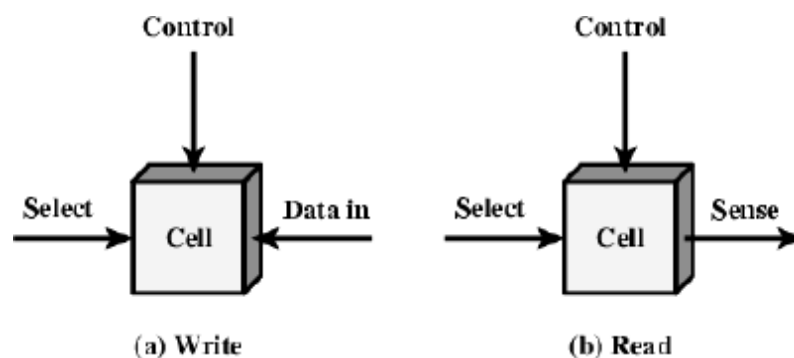
Tujuan Pembelajaran

1. Menjelaskan tentang memori utama komputer
2. Menjelaskan tipe dari memori, waktu dan pengontrolan
3. Menjelaskan pembetulan kesalahan
4. Menjelaskan cache memori termasuk didalamnya adalah fungsi pemetaan

A. Pengertian Memori

Memori adalah bagian dari komputer tempat program-program dan data-data disimpan. Memori juga diartikan sebagai tempat informasi, dibaca dan ditulis, dimana terdapat berbagai jenis, teknologi, organisasi, unjuk kerja dan harga memori. Secara garis besar, berdasarkan lokasinya memori terbagi dua yaitu memori internal dan eksternal. Memori internal adalah memori yang dapat diakses langsung oleh prosesor. Contoh memori internal adalah register yang terdapat di dalam prosesor, cache memori dan memori utama berada di luar prosesor. Sedangkan memori eksternal adalah memori yang diakses prosesor melalui piranti I/O, contohnya disket dan hardisk.

Elemen dasar memori disebut sebagai sel memori. Sifat sel memori yaitu memiliki dua keadaan stabil (atau semi-stabil), yang dapat digunakan untuk merepresentasikan bilangan biner 1 atau 0. Sifat lainnya adalah bahwa sel memori mempunyai kemampuan untuk ditulisi (sedikitnya satu kali) dan kemampuan untuk dibaca.



Gambar 7.1 Terminal fungsi sel memori

B. Karakter Sistem Memori

Karakteristik memori terbagi atas beberapa bagian yaitu lokasi, kapasitas, satuan transfer, metode akses, kinerja, tipe fisik beserta karakter fisik. Adaaun detail klasifikasi karakteristik sismtem meori tersebut dapat dilihat pada tabbel berikut.

Tabel 7.1 karakteristik sistem memori

Karakteristik	Macam/ Keterangan
Lokasi	1. CPU 2. Internal (main) 3. External (secondary)
Kapasitas	1. Ukuran word 2. Jumlah word
Satuan transfer	1. Word 2. Block
Metode akses	1. Sequential access 2. Direct access 3. Random access 4. Associative access
Kinerja	1. Access time 2. Cycle time 3. Transfer rate
Tipe fisik	1. Semikonduktor 2. Magnetik
Karakteristik fisik	1. Volatile/nonvolatile 2. Erasable/nonerasable

Berdasarkan tabel karakteristik tersebut diatas maka berdasarkan lokasi nya memori terbagi atas CPU, internal dan eksternal. Memori yang berada di dalam chip prosesor (CPU) adalah register. Register diakses langsung oleh prosesor dalam menjalankan operasinya. Register digunakan sebagai memori sementara dalam perhitungan maupun pengolahan data dalam prosesor. Sementara memori internal secara lokasi berada diluar chip prosesor, dimana memori ini juga diakses langsung oleh prosesor. Memori internal terbagi atas memori utama dan cache memori. Memori eksternal merupakan jenis memori yang lokasinya diluar komputer dan diakses oleh prosesor melalui piranti I/O. Contoh memori eksternal ini dapat berupa disk maupun pita.

Kapasitas Memori memori internal maupun eksternal biasanya dinyatakan dalam mentuk byte (1 byte = 8 bit) atau word. Panjang word umumnya 8, 16, 32 bit. Sementara memori eksternal biasanya lebih besar kapasitasnya daripada memori internal, hal ini disebabkan karena teknologi dan sifat penggunaannya yang berbeda. Satuan Transfer memori

internal sama dengan jumlah saluran data yang masuk ke dan keluar dari modul memori. Jumlah saluran ini sering kali sama dengan panjang word, tapi dimungkinkan juga tidak sama. Konsep Satuan Transfer terbagi menjadi beberapa diantaranya yaitu word, addressable units serta unit of transfer. Word merupakan satuan “alami” organisasi memori. Ukuran word biasanya sama dengan jumlah bit yang digunakan untuk representasi bilangan dan panjang instruksi. Addressable units, pada sejumlah sistem, addressable units adalah word. Namun terdapat sistem dengan pengalamatan pada tingkatan byte. Pada semua kasus hubungan antara panjang A suatu alamat dan jumlah N addressable unit adalah $2^A = N$. Unit of transfer, adalah jumlah bit yang dibaca atau dituliskan ke dalam memori pada suatu saat. Pada memori eksternal, transfer data biasanya lebih besar dari suatu word, yang disebut dengan block.

Metode Akses memori terbagi menjadi empat yaitu Sequential access, Direct access, Random access serta Associative access. Metode sequential access bekerja dengan cara memori diorganisasi menjadi unit – unit data yang disebut record. Akses harus dibuat dalam bentuk urutan linier yang spesifik. Informasi pengalamatan yang disimpan dipakai untuk memisahkan record – record dan untuk membantu proses pencarian. Terdapat shared read/write mechanism untuk penulisan/pembacaan memorinya. Pita magnetik merupakan memori yang menggunakan metode sequential access. Sedangkan metode direct access sama seperti sequential access dimana terdapat shared read/write mechanism. Setiap blok dan record memiliki alamat unik berdasarkan lokasi fisiknya. Akses dilakukan langsung pada alamat memori dan disk adalah memori direct access. Metode yang selanjutnya yaitu random access, pada metode ini setiap lokasi memori dipilih secara random dan diakses serta dialamati secara langsung. Contohnya adalah memori utama. Metode yang terakhir yaitu associative access yang merupakan jenis memori random akses yang memungkinkan perbandingan lokasi bit yang diinginkan untuk pencocokan. Data dicari berdasarkan isinya bukan alamatnya dalam memori. Contoh memori ini adalah cache memori.

Parameter utama unjuk kerja memori meliputi access time, memory cycle time serta transfer rate. Pada random access memory, waktu akses adalah waktu yang dibutuhkan untuk melakukan operasi baca atau tulis. Memori non-random akses merupakan waktu yang dibutuhkan dalam melakukan mekanisme baca atau tulis pada lokasi tertentu. Selanjutnya parameter memory cycle time digunakan pada random access memory. Terdiri dari access time ditambah dengan waktu yang diperlukan transient agar hilang pada saluran sinyal. Pada transfer rate atau kecepatan data transfer ke unit memori atau dari unit memori. Random access memory sama dengan $1/(\text{cycle time})$. Sedangkan non-random access memory dengan perumusan :

$$TN = TA + (N/R)$$

TN = waktu rata – rata untuk membaca atau menulis N bit

TA = waktu akses rata – rata

N = jumlah bit

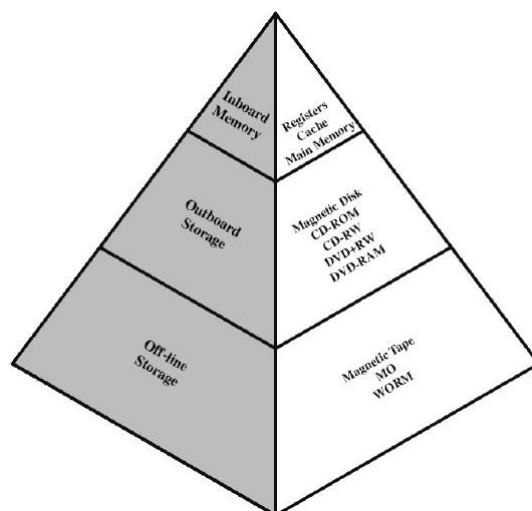
R = kecepatan transfer dalam bit per detik (bps)

Berdasarkan bentuk fisiknya memori dibedakan menjadi volatile dan non-volatile serta erasable dan nonerasable. Pada media penyimpanan volatile (volatile memory) informasi akan hilang apabila daya listriknya dimatikan, sedangkan pada non-volatile memory data yang tersimpan tidak hilang walau daya listriknya hilang. Memori permukaan magnetik adalah contoh non-volatile memory, sedangkan memori semikonduktor ada yang volatile dan nonvolatile. Selanjutnya pada pembagian media erasable dan nonerasable, terdapat beberapa jenis memori semikonduktor yang tidak bisa dihapus kecuali dengan menghancurkan unit storage-nya, memori ini dikenal dengan ROM (Read Only Memory).

Keandalan memori dihitung berdasarkan ukuran kapasitas atau berapa banyak data yang data disimpan. Kebutuhan ukuran ini merupakan sesuatu yang sulit dijawab, karena berapapun kapasitas memori tentu ukuran aplikasi yang akan menggunakannya juga menjadi factor penentu. Keandalan selanjutnya dihitung berdasarkan kecepatan akses memori dimana memori harus mampu mengikuti kecepatan CPU sehingga terjadi sinkronisasi kerja antar CPU dan memori tanpa adanya waktu tunggu karena komponen lain belum selesai prosesnya. Faktor penentu keandalan yang lain meliputi harga dari sebuah memori yang bisa dikatakan bahwa nilainya relatif. Bagi produsen selalu mencari harga produksi paling murah tanpa mengorbankan kualitasnya untuk memiliki daya saing di pasaran. Hubungan ketiganya yakni harga, kapasitas dan waktu akses adalah bahwa semakin kecil waktu akses maka semakin besar harga per bitnya, semakin besar kapasitas maka semakin kecil harga per bitnya serta semakin besar kapasitas maka semakin besar waktu aksesnya.

C. Hirarki Memori

Menurunnya hirarki mengakibatkan beberapa hal terjadi diantaranya meliputi penurunan harga/bit, peningkatan kapasitas, peningkatan waktu akses serta penurunan frekuensi akses memori oleh CPU. Kunci keberhasilan hirarki ini pada penurunan frekuensi aksesnya. Semakin lambat memori maka keperluan CPU untuk mengaksesnya semakin sedikit. Secara keseluruhan sistem komputer akan tetap cepat namun kebutuhan kapasitas memori besar terpenuhi.



Gambar 7.2 Hirarki Memori

Dari gambar hirarki di atas maka memori dapat diklasifikasikan berdasarkan tipe, teknologi, ukuran atau kapasitas serta waktu akses seperti pada table berikut.

Tabel 7.2 Spesifikasi memori

Tipe memori	Teknologi	Ukuran	Waktu akses
Cache Memory	semikonduktor RAM	128 – 512 KB	10 ns
Memori Utama	semikonduktor RAM	4 – 128 MB	50 ns
Disk magnetik	Hard Disk	Gigabyte	10 ms, 10MB/det
Disk Optik	CD-ROM	Gigabyte	300ms, 600KB/det
Pita magnetik	Tape	100 MB	Det -mnt, 10MB/mnt

Satuan pokok memori adalah digit biner atau disebut bit. Bit dapat berisi sebuah angka 0 atau 1. Memori juga dinyatakan dalam byte, 1 byte = 8 bit. Kumpulan byte dinyatakan dalam word. Panjang word yang umum adalah 8, 16, dan 32 bit. Tingkatan satuan memori dari yang ukuran kecil sampai besar tampak pada table di bawah ini

Tabel 7.3. Tingkatan Satuan Memori

Symbol		Number of bytes	
Kilobytes	Kb	2e10	1024
Megabyte	Mb	2e20	1,048,576
Gigabyte	Gb	2e30	1,073,741,824
Terabyte	Tb	2e40	1,099,511,627,776

D. Memori internal

Memori semikonduktor terbagi atas beberapa tipe diantaranya adalah random akses (RAM), ROM, PROM, EPROM, Flas Memory serta EEPROM. Klasifikasi detail untuk setiap jenis memori tersebut tampak pada table 8.3 di bawah. Pada komputer lama, bentuk umum random access memory untuk memori utama adalah sebuah piringan ferromagnetik berlubang yang dikenal sebagai core, istilah yang tetap dipertahankan hingga saat ini.

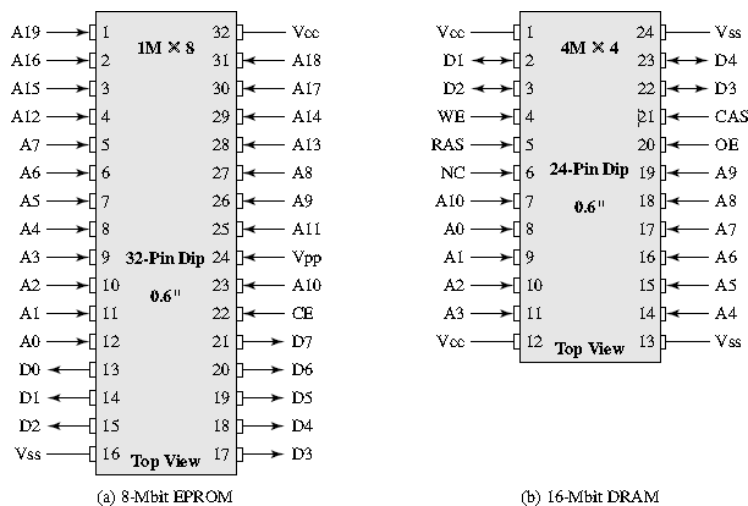
Tabel 7.3 Tipe-tipe memori semikonduktor

Tipe Memori	Kategori	<i>Erasure</i>	Mekanisme Penulisan	<i>Volatility</i>
<i>Random-access memory (RAM)</i>	<i>Read-write memory</i>	<i>Electrically, byte-level</i>	<i>Electrically</i>	<i>Volatile</i>
<i>Read-only memory (ROM)</i>	<i>Read-only memory</i>	<i>Not possible</i>	<i>Masks</i>	<i>Nonvolatile</i>
<i>Programmable ROM (PROM)</i>				
<i>Erasable PROM (EPROM)</i>	<i>Read-mostly memory</i>	<i>UV light, chip-level</i>	<i>Electrically</i>	
<i>Electrically Erasable PROM (EEPROM)</i>		<i>Electrically, byte-level</i>		
<i>Flash memory</i>		<i>Electrically, block-level</i>		

Pada memori random akses data diakses secara langsung melalui logik pengalamanan wired-in. Pada memori jenis ini dimungkinkan adanya pembacaan dan penulisan data ke memori secara cepat dan mudah. RAM bersifat volatile dimana data tersimpan selama ada daya (listrik) dan akan hilang jika tidak ada sumber listrik. Random akses memori terbagi atas RAM dinamik dan RAM static. RAM dinamik disusun oleh sel – sel yang menyimpan data sebagai muatan listrik pada kapasitor. Kapasitor memiliki kecenderungan alami untuk mengosongkan muatan, maka RAM dinamik memerlukan pengisian muatan listrik secara periodik untuk memelihara penyimpanan data. Biasanya dimanfaatkan untuk operasi data besar. Sedangkan RAM static, nilai biner disimpan dengan menggunakan konfigurasi gate logika flipflop

tradisional. RAM jenis ini memiliki kemampuan menyimpan data selama ada daya listriknya dan lebih cepat dibanding RAM dinamik.

ROM (Read Only Memory) merupakan jenis memori yang sangat berbeda dengan RAM. Pada ROM data bersifat permanen, tidak bisa diubah dimana ini menguntungkan untuk penyimpanan data yang permanen. Namun terdapat kerugian yaitu jika terdapat kesalahan data atau adanya perubahan data sehingga perlu penyesuaian –penyesuaian. PROM (Programmable ROM) merupakan jenis memori non-volatile yang terbagi atas tiga macam yaitu EPROM, EEPROM dan flash memory. EEPROM electrically erasable programmable read only memory. Jenis memori yang dapat ditulisi kapan saja tanpa menghapus isi sebelumnya. EEPROM menggabungkan kelebihan non-volatile dengan fleksibilitas dapat di-update.



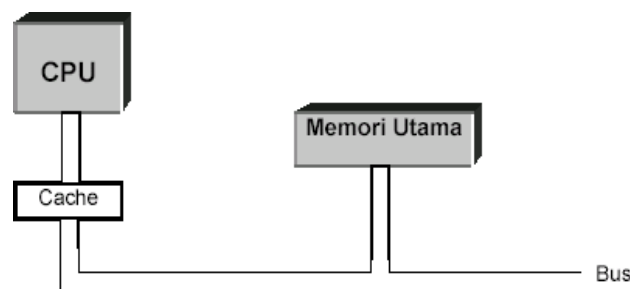
Gambar 7.3 Typical Memory Package Pins and Signals

Pengemasan memori sebagaimana tampak pada Gambar 7.3 di atas (a) EPROM yang merupakan keping 8 Mbit yang diorganisasi sebagai 1Mx8. Organisasi dianggap sebagai kemasan satu *word* per keping. Kemasan terdiri dari 32 pin, yang merupakan salah satu ukuran kemasan keping standar. Sedangkan gambar (b) keping 16 Mbit yang diorganisasikan sebagai 4M x 4. Terdapat sejumlah perbedaan dengan keping ROM, karena ada operasi tulis maka pin – pin data merupakan input/output yang dikendalikan oleh WE (*write enable*) dan OE (*output enable*). Pada pengemasan DRAM ini data yang akan dibaca terdiri dari 8 saluran (D0 –D7). Catu daya keping adalah Vcc dengan pin grounding Vss serta pin *chip enable* (CE). Karena mungkin terdapat lebih dari satu keping memori yang terhubung pada bus yang sama maka pin CE digunakan untuk mengindikasikan valid atau tidaknya pin ini. Pin CE diaktifkan oleh logik yang terhubung dengan bit berorde tinggi bus alamat (di atas A19). Tegangan program adalah Vpp.

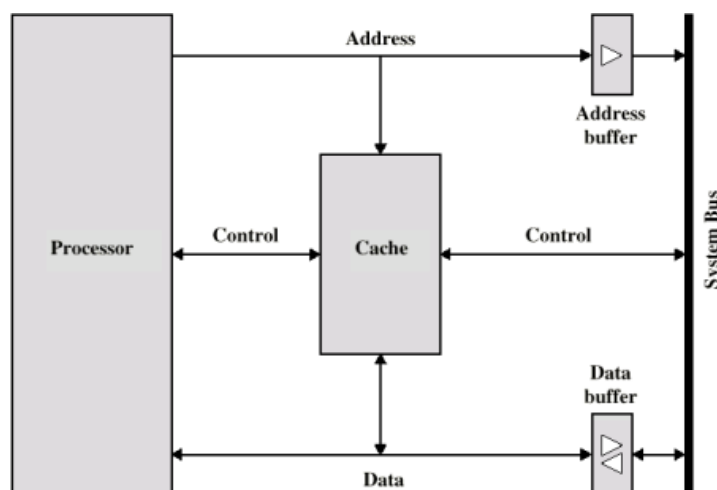
Dalam melaksanakan fungsi penyimpanan, memori semikonduktor dimungkinkan mengalami kesalahan. Kesalahan berat yang biasanya merupakan kerusakan fisik memori. Kesalahan ringan yang berhubungan data yang disimpan. Kesalahan ringan dapat dikoreksi kembali. Koreksi kesalahan data yang disimpan diperlukan dua mekanisme yaitu mekanisme pendeteksian kesalahan serta perbaikan kesalahan.

E. Cache Memori

Salah satu jenis memori adalah cache memori yang memiliki kemampuan untuk mempercepat kerja memori utama sehingga mendekati kecepatan prosesor. Memori utama lebih besar kapasitasnya namun lambat operasinya, sedangkan cache memori berukuran kecil namun lebih cepat. Cache memori berisi salinan memori utama. Ukuran cache memori adalah kecil, semakin besar kapasitasnya maka akan memperlambat proses operasi cache memori itu sendiri, disamping harga cache memori yang sangat mahal.



Gambar 7. 4 Ilustrasi Cache memori



Gambar 7.5. Organisasi Cache Memori

Cache Addresses	Write Policy
Logical	Write through
Physical	Write back
Cache Size	Write once
Mapping Function	Line Size
Direct	Number of caches
Associative	Single or two level
Set Associative	Unified or split
Replacement Algorithm	
Least recently used (LRU)	
First in first out (FIFO)	
Least frequently used (LFU)	
Random	

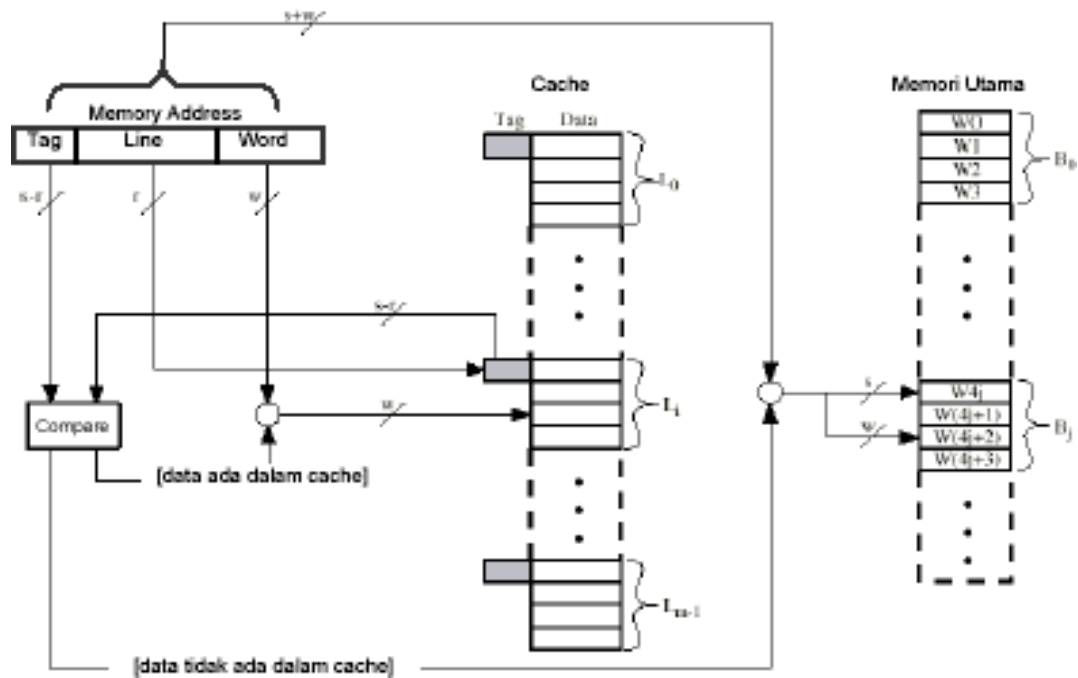
Gambar 7.6. Elemen Cache Memori

Secara bentuk fisik cache memori terbagi atas logical dan fisik, sedangkan untuk ukuran bervariasi menyesuaikan kebutuhan. Sebagai contoh AMD mengeluarkan prosesor K5 dan K6 dengan cache yang besar (1MB), kinerjanya tidak bagus. Intel mengeluarkan prosesor tanpa cache untuk alasan harga yang murah, yaitu seri Intel Celeron pada tahun 1998-an, kinerjanya sangat buruk terutama untuk operasi data besar, floating point. Sejumlah penelitian telah menganjurkan bahwa ukuran cache antara 1KB dan 512KB akan lebih optimum [STA96].

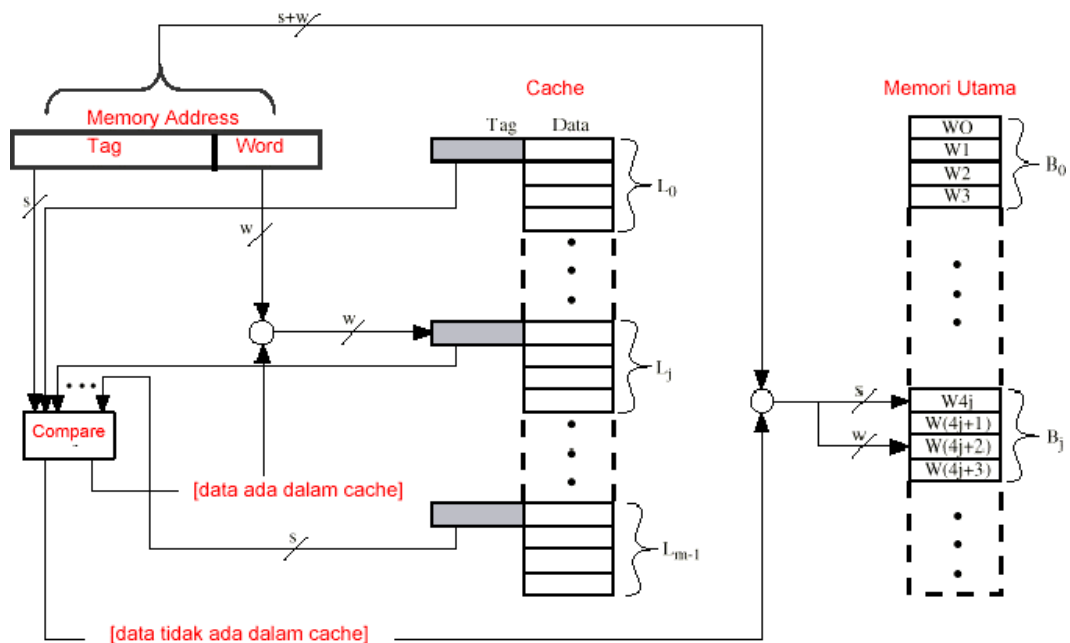
Hubungan antara ukuran blok dan hit ratio sangat rumit untuk dirumuskan, tergantung pada karakteristik lokalitas programnya dan tidak terdapat nilai optimum yang pasti telah ditemukan. Ukuran antara 4 hingga 8 satuan yang dapat dialamati (word atau byte) cukup beralasan untuk mendekati nilai optimum [STA96].

Pemetaan Cache mempunyai kapasitas yang kecil dibandingkan memori utama. Aturan blok – blok mana yang diletakkan dalam cache. Terdapat tiga metode, yaitu pemetaan langsung, pemetaan asosiatif, dan pemetaan asosiatif set. Pemetaan langsung merupakan teknik paling sederhana, yaitu teknik ini memetakan blok memori utama hanya ke sebuah saluran cache saja. Pemetaan Asosiatif mengatasi kekurangan pemetaan langsung. Tiap blok memori utama dapat dimuat ke sembarang saluran cache. Alamat memori utama diinterpretasikan dalam field tag dan field word oleh kontrol logika cache. Tag secara unik mengidentifikasi sebuah blok memori utama. Mekanisme untuk mengetahui suatu blok dalam cache dengan memeriksa setiap tag saluran cache oleh kontrol logika cache. Fleksibilitas dalam penggantian blok baru yang ditempatkan dalam cache. Kelebihan dari pemetaan ini adalah bahwa algoritma penggantian dirancang untuk memaksimalkan hit ratio, yang pada pemetaan langsung terdapat kelemahan. Sedangkan kekurangannya yaitu kompleksitas rangkaian sehingga mahal secara ekonomi. Pemetaan Asosiatif Set menggabungkan kelebihan yang ada pada pemetaan

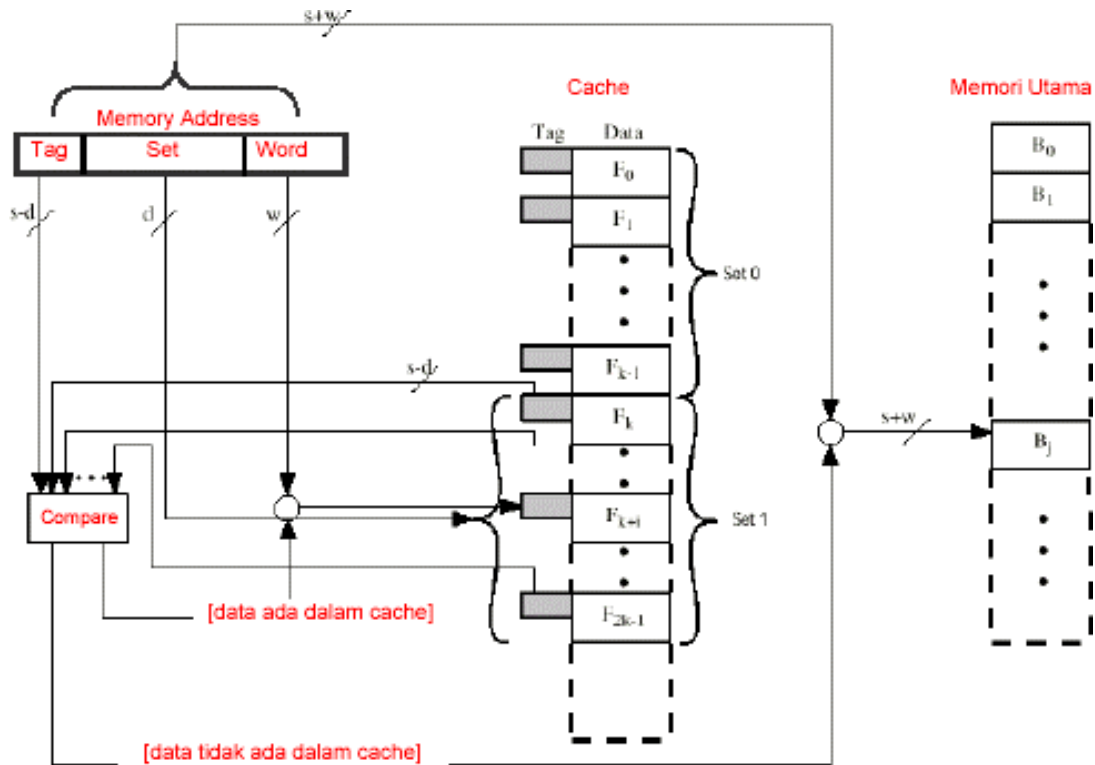
langsung dan pemetaan asosiatif. Memori cache dibagi dalam bentuk set-set. Alamat memori utama diinterpretasikan dalam tiga field, yaitu: field tag, field set, field word. Setiap blok memori utama dapat dimuat dalam sembarang saluran cache.



Gambar 7.7 Pemetaan Langsung



Gambar 7.8 Pemetaan Asosiatif



Gambar 7.9. Pemetaan Asosiatif Set

Algoritma penggantian merupakan suatu mekanisme penggantian blok-blok dalam memori cache yang lama dengan data baru. Pemetaan langsung tidak memerlukan algoritma ini. Pemetaan asosiatif dan asosiatif set, berperan penting meningkatkan kinerja cache memori. Algoritma *Least Recently Used* (LRU), yaitu mengganti blok data yang terlama berada dalam cache dan tidak memiliki referensi. Algoritma *First In First Out* (FIFO), yaitu mengganti blok data yang awal masuk. Algoritma *Least Frequently Used* (LFU) adalah mengganti blok data yang mempunyai referensi paling sedikit. Algoritma Random, yaitu penggantian tidak berdasarkan pemakaian datanya, melainkan berdasar slot dari beberapa slot kandidat secara acak.

Apabila suatu data telah diletakkan pada cache maka sebelum ada penggantian data baru maka harus dicek apakah data tersebut telah mengalami perubahan atau tidak. Apabila data telah berubah maka data pada memori utama harus di-update. Masalah penulisan ini sangat kompleks, apalagi memori utama dapat diakses langsung oleh modul I/O, yang memungkinkan data pada memori utama berubah. Perubahan data pada memori utama ini tentunya akan mempengaruhi data yang telah dikirim pada cache. Perubahan ini menyebabkan data yang ada pada cache menjadi tidak valid, sehingga perlu dilakukan pembenaran berupa update atau

perubahan data. Write Policy –"write through" merupakan operasi penulisan melibatkan data pada memori utama dan sekaligus pada cache memori sehingga data selalu valid. Kekurangan teknik ini adalah lalu lintas data ke memori utama dan cache sangat tinggi. Write Policy –"write back " merupakan teknik meminimasi penulisan dengan cara penulisan pada cache saja. Pada saat akan terjadi penggantian blok data cache maka baru diadakan penulisan pada memori utama. Masalah muncul jika data di memori utama belum di-update telah diakses modul I/O sehingga data di memori utama tidak valid. Write Policy-Multi cache merupakan multi cache untuk multi prosesor yang memiliki masalah yang lebih kompleks. Masalah validasi data tidak hanya antara cache dan memori utama namun validasi antar cache juga harus diperhatikan.

Berdasarkan jumlahnya cache memori terbagi menjadi dua yaitu *single or two level* serta *unified or split*. Pada jenis *single or two level cache*, *layer 1* atau L1 merupakan cache internal dimana lokasi memori berada di dalam chip yang tidak membutuhkan bus eksternal dan waktu aksesnya sangat cepat. Sementara *layer 2* atau cache tingkat 2 (L2) merupakan cache eksternal yang lokasinya ada diluar chip. Selanjutnya jenis yang kedua dari sistem cache terdapat cache data dan cache instruksi yang disebut *unified cache*. Keuntungan *unified cache* yaitu *hit rate* yang tinggi karena telah dibedakan antara informasi data dan informasi instruksi dan hanya sebuah cache saja yang perlu dirancang dan diimplementasikan. Pemanfaatan *split cache* dapat dilihat pada mesin–mesin *superscalar* seperti Pentium dan PowerPC dimana sistem lebih menekankan pada paralel proses dan perkiraan –perkiraan eksekusi yang akan terjadi. Kelebihan utama *split cache* adalah kemampuannya dalam mengurangi persaingan antara prosesor instruksi dan unit eksekusi untuk mendapatkan cache, hal ini sangat utama bagi perancangan prosesor–prosesor *pipelining*.

F. Memori Eksternal

Memori eksternal dibutuhkan karena kapasitas yang terdapat pada memori utama tidak mencukupi sehingga diperlukan peralatan tambahan untuk menyimpan data yang lebih besar dan dapat dibawa kemana-mana. Semakin besar peralatan penyimpanan maka akan mempengaruhi waktu pemrosesan data. Contoh-contoh memori eksternal meliputi Magnetik Disk (Floppy Disk, IDE Disk, SCSI Disk), RAID, Optical Disk (CDROM, CD-R, CD-RW, DVD) serta Pita Magnetik.

Magnetik Disk adalah piringan bundar yang terbuat dari bahan tertentu (logam atau plastik) dengan permukaan dilapisi bahan yang dapat di magnetisasi. Mekanisme baca/tulis menggunakan kepala baca atau tulis yang disebut *head*, merupakan komparan pengkonduksi (*conducting coil*). Desain fisiknya, *head* bersifat stasioner sedangkan piringan disk berputar

sesuai kontrolnya. Dua metode layout data pada disk, yaitu constant angular velocity dan multiple zoned recording. Disk diorganisasi dalam bentuk cincin – cincin konsentris yang disebut track. Tiap track pada disk dipisahkan oleh gap (gap: mencegah atau mengurangi kesalahan pembacaan maupun penulisan yang disebabkan melesetnya head atau karena interferensi medan magnet). Sejumlah bit yang sama akan menempati track – track yang tersedia. Semakin ke dalam disk maka kerapatan (density) disk akan bertambah besar. Data dikirim ke memori ini dalam bentuk blok, umumnya blok lebih kecil kapasitasnya daripada track. Blok – blok data disimpan dalam disk yang berukuran blok, yang disebut sector. Track biasanya terisi beberapa sector, umumnya 10 hingga 100 sector tiap tracknya.

RAID (Redundancy Array of Independent Disk) merupakan organisasi disk memori yang mampu menangani beberapa disk dengan sistem akses parallel dan redundansi ditambahkan untuk meningkatkan reliabilitas. Kerja parallel menghasilkan resultan kecepatan disk yang lebih cepat. Teknologi database sangat penting dalam model disk ini karena pengontrol disk harus mendistribusikan data pada sejumlah disk dan juga membaca kembali. Karakteristik umum disk RAID yaitu bahwa RAID adalah sekumpulan disk drive yang dianggap sebagai sistem tunggal disk. Data RAID didistribusikan ke drive fisik array dimana kapasitas redundant disk digunakan untuk menyimpan informasi paritas, yang menjamin recoveribility data ketika terjadi masalah atau kegagalan disk. RAID merupakan salah satu jawaban masalah kesenjangan kecepatan disk memori dengan CPU dengan cara menggantikan disk berkapasitas besar dengan sejumlah disk–disk berkapasitas kecil dan mendistribusikan data pada disk–disk tersebut sedemikian rupa sehingga nantinya dapat dibaca kembali.

Sistem pita magnetik menggunakan teknik pembacaan dan penulisan yang identik dengan sistem disk magnetic. Medium pita magnetik berbentuk track – track paralel, sistem pita lama berjumlah 9 buah track sehingga memungkinkan penyimpanan satu byte sekali simpan dengan satu bit paritas pada track sisanya. Sistem pita baru menggunakan 18 atau 36 track sebagai penyesuaian terhadap lebar word dalam format digital. Seperti pada disk, pita magnetik dibaca dan ditulisi dalam bentuk blok – blok yang bersambungan (kontinyu) yang disebut physical record. Blok – blok tersebut dipisahkan oleh gap yang disebut inter-record gap. Kecepatan putaran pita magnetik adalah rendah sehingga transfer data menjadi lambat. Pita magnetik mulai ditinggalkan digantikan oleh jenis – jenis produk CD.

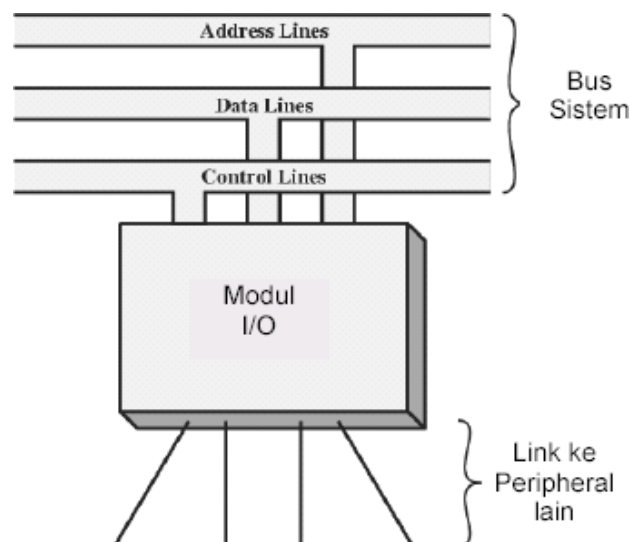
BAB 9. Unit Masukan dan Keluaran (I/O Device)

Tujuan Pembelajaran

1. Mahasiswa mampu menjelaskan tentang system komputer unit masukan/keluaran
2. Mahasiswa mampu menjelaskan prinsip dan teknik unit masukan/keluaran
3. Mahasiswa mampu menjelaskan peralatan luar (External device)

A. Pengertian dan Fungsi Modul I/O

Komponen utama sistem komputer meliputi CPU, memori (primer dan sekunder), modul masukan/keluaran (I/O devices), serta sistem interkoneksi. Modul I/O atau modul masukan/keluaran merupakan peralatan antarmuka (interface) bagi sistem bus atau switch sentral dan mengontrol satu atau lebih perangkat peripheral. Tidak hanya sekedar modul penghubung, tetapi sebuah piranti yang berisi logika dalam melakukan fungsi komunikasi antara peripheral dan bus komputer. Piranti atau perangkat eksternal tidak langsung dihubungkan dengan bus sistem komputer disebabkan karena bervariasinya metode operasi piranti peripheral, sehingga tidak praktis apabila sistem komputer harus menangani berbagai macam sistem operasi piranti peripheral tersebut. Kecepatan transfer data piranti peripheral umumnya lebih lambat dari pada laju transfer data pada CPU maupun memori. Format data dan panjang data pada piranti peripheral seringkali berbeda dengan CPU, sehingga perlu modul untuk menselaraskannya.

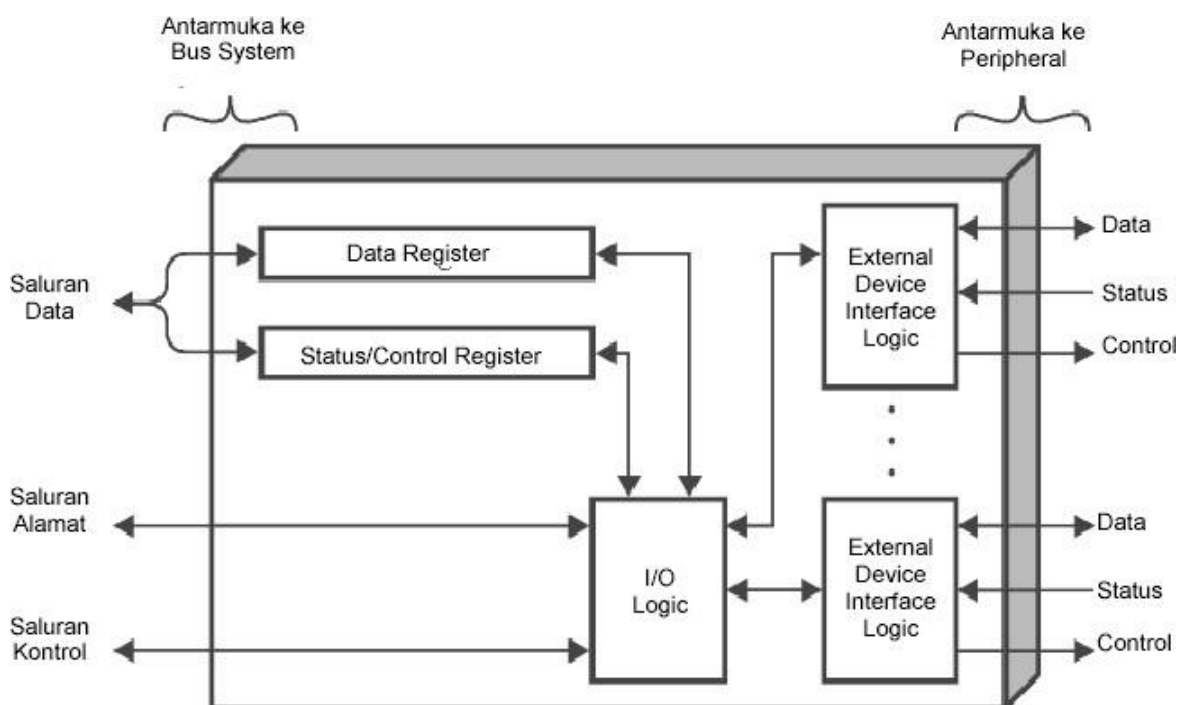


Gambar 8.1. Model generik dari suatu modul I/O

Fungsi modul I/O yaitu sebagai piranti antarmuka ke CPU dan memori melalui bus sistem. Fungsi utama lainnya dari modul ini adalah sebagai piranti antarmuka dengan peralatan peripheral lainnya dengan menggunakan link data tertentu. Modul I/O adalah suatu komponen dalam sistem komputer yang bertanggung jawab atas pengontrolan perangkat luar serta pertukaran data antara perangkat luar tersebut dengan memori utama ataupun dengan register-register CPU atau disebut juga antarmuka dengan perangkat eksternalnya untuk menjalankan fungsi-fungsi pengontrolan. Adapun fungsi modul I/O lainnya meliputi kontrol dan pewaktuan, komunikasi CPU, komunikasi perangkat eksternal, pem-buffer-an data serta deteksi kesalahan

B. Struktur dan Teknik Modul I/O

Modul I/O mengalami berbagai macam perkembangan seiring dengan perkembangan berbagai jenis komputer. Namun, bagaimanapun kompleksitas suatu modul I/O struktur dari sistem ini masih terdapat kemiripan. Antarmuka modul I/O ke CPU melalui bus sistem komputer terdapat tiga saluran yaitu saluran data, saluran alamat serta saluran kontrol. Bagian terpenting dari modul ini adalah blok logika I/O yang berhubungan dengan semua peralatan antarmuka peripheral, terdapat fungsi pengaturan dan switching pada blok.



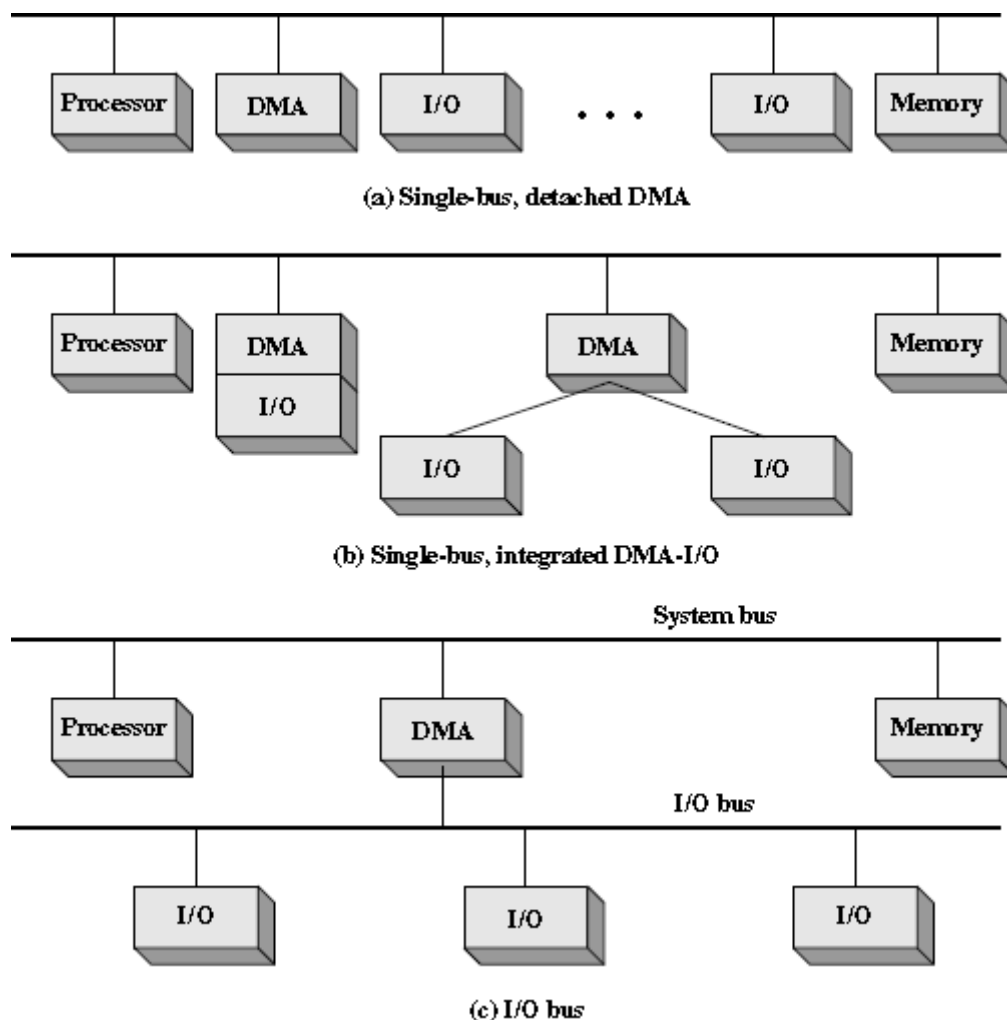
Gambar 8.2. Blok diagram struktur modul I/O

Metode akses perangkat eksternal terhadap modul Input Output terbagi menjadi tiga yaitu *programmed*, *interrupt driven* serta *Direct Memory Access (DMA)*. Pada *programmed* atau I/O Terprogram data saling dipertukarkan antara CPU dan modul I/O. CPU mengeksekusi program yang memberikan operasi I/O kepada CPU secara langsung yang meliputi pemindahan data, pengiriman perintah baca maupun tulis dan monitoring perangkat. Kelemahan dari program I/O adalah bahwa CPU akan menunggu sampai operasi I/O selesai dilakukan modul I/O sehingga akan membuang waktu, CPU lebih cepat proses operasinya. Dalam teknik ini, modul I/O tidak dapat melakukan interupsi kepada CPU terhadap proses – proses yang diinteruksikan padanya. Seluruh proses merupakan tanggung jawab CPU sampai operasi lengkap dilaksanakan. Pada *programmed I/O* implementasi perintah dalam instruksi I/O terbagi menjadi dua yaitu *memory-mapped I/O* dan *isolated I/O*.

Memory-mapped I/O terdapat ruang tunggal untuk lokasi memori dan perangkat I/O. CPU memperlakukan register status dan register data modul I/O sebagai lokasi memori dan menggunakan instruksi mesin yang sama untuk mengakses baik memori maupun perangkat I/O. Kongsuensinya adalah diperlukan saluran tunggal untuk pembacaan dan saluran tunggal untuk penulisan. Keuntungan *memory-mapped I/O* adalah efisien dalam pemrograman, namun memakan banyak ruang memori alamat. Pada implemtasi *Isolated I/O* dilakukan pemisahan ruang pengalamatan bagi memori dan ruang pengalamatan bagi I/O. Dengan teknik ini diperlukan bus yang dilengkapi dengan saluran pembacaan dan penulisan memori ditambah saluran perintah output. Keuntungan *isolated I/O* adalah sedikitnya instruksi I/O.

Pada teknik *Interrupt – Driven I/O* proses pengaksesan dilakukan lebih optimal dengan tidak membuang – buang waktu. Prosesnya yang terjadi yaitu CPU mengeluarkan perintah I/O pada modul I/O, bersamaan perintah I/O dijalankan modul I/O maka CPU akan melakukan eksekusi perintah – perintah lainnya. Apabila modul I/O telah selesai menjalankan instruksi yang diberikan padanya akan melakukan interupsi pada CPU bahwa tugasnya telah selesai. Kendali perintah masih menjadi tanggung jawab CPU, baik pengambilan perintah dari memori maupun pelaksanaan isi perintah tersebut. CPU melakukan multitasking beberapa perintah sekaligus dimana tidak ada waktu tunggu bagi CPU yang membuat proses bisa lebih cepat. Cara kerja teknik interupsi di sisi modul I/O yaitu modul I/O menerima perintah, misal read. Modul I/O melaksanakan perintah pembacaan dari peripheral dan meletakkan paket data ke register data modul I/O. Modul mengeluarkan sinyal interupsi ke CPU melalui saluran kontrol. Modul menunggu datanya diminta CPU. Saat permintaan terjadi modul meletakkan data pada bus data dan modul siap menerima perintah selanjutnya.

Direct memory acces atau akses memori langsung mampu mengatasi kelemahan yang terdapat pada interrupt driven dan programmed I / O, dimana kedua teknik sebelumnya memerlukan intervensi CPU aktif dan memiliki transfer rate yang terbatas. Fungsi DMA adalah sebagai modul tambahan (hardware) pada sistem bus dimana kontroler DMA mengambil alih tugas dari CPU untuk penanganan I/O. Cara kerja DMA menyesuaikan instruksi dari CPU pada saat CPU memberitahu DMA controller untuk melakukan tugas seperti membaca atau menulis data, mengetahui alamat perangkat, memulai alamat blok memori data, serta mencari tahu jumlah data yang akan ditransfer. Setelah CPU menginstruksikan perintah pada DMA controller, CPU melakukan pekerjaan lain dan DMA controller menangani pekerjaan yang berkenaan dengan instruksi yang diberikan CPU. Jika pekerjaan sudah selesai maka DMA controller akan mengirimkan interupsi atau pesan pada CPU.



Gambar 8.3. Konfigurasi Teknik *Direct memory acces*

Konfigurasi DMA terbagi menjadi tiga jenis yang pertama yaitu *Single Bus, Detached DMA controller*. Pada konfigurasi ini setiap transfer menggunakan bus dua kali seperti perjalanan data dari I/O ke DMA lalu DMA ke memori sehingga pada saat sistem bus digunakan dua kali maka kerja CPU juga ditunda dua kali. Konfigurasi kedua disebut sebagai *Single Bus, Integrated DMA controller*. Pada konfigurasi ini *controller* dapat mendukung lebih dari satu perangkat sehingga setiap terjadi perpindahan atau transfer data maka sistem menggunakan bus hanya satu kali seperti perpindahan data dari DMA ke memori. Jika sistem bus digunakan satu kali maka kinerja CPU juga ditunda satu kali sehingga bisa dikatakan bahwa konfigurasi jenis ini lebih baik dari yang sebelumnya. Konfigurasi jenis ketiga yaitu *Separate I/O Bus* (system I/O bus yang terpisah). Konfigurasi bus yang dibangun mendukung semua perangkat DMA yang aktif. Dalam kedua kasus ini (Gambar 7.13b dan c), sistem bus digunakan secara bersama oleh modul DMA, prosesor dan memori. Namun pada praktiknya penggunaan jalur bus ini hanyalah untuk pertukaran data antara DMA dengan memori.

DAFTAR PPUSTAKA

1. Computer Organisation and Architecture-8th ed by William Stallings
2. Computer Organization and Design 3rd Edition by David A. Patterson, John L. Hennessy
3. Morgan.Kaufmann.Embedded.Computing.Dec.2004.eBook-LinG
4. Computer Architecture A Quantitative Approach 4th_Edition by David A. Patterson, John L. Hennessy
5. Computer Architecture. A Quantitative Approach. 3rd Edition by David A. Patterson, John L. Hennessy
6. Desain Digital dan Arsitektur Komputer, Syahrul dan Abdullah Basalamah, 2016