



Selamat Datang

*Semoga Tuhan memberi
berkah pada kelas ini.*



Perencanaan pembelajaran

- TUJUAN

Memberikan pemahaman kepada mahasiswa tentang bagaimana komputer dibentuk oleh komponen-komponen pendukungnya, agar mahasiswa memahami relasi inti perangkat keras dan perangkat lunaknya.



Arsitektur Komputer

Pertemuan ke-	Pokok Bahasan	Keterangan
1	Evolusi Komputer	
2	Bus-bus Sistem	
3	Internal Memori	
4	External Memori	
5	Input / Output	
6	Input / Output	
7	Review Materi / Kuis (Soal-Soal Essay)	
8	Ujian Tengah Semester (UTS)	
9	Central Processing Unit Dan Unit Control	
10	Set Instruksi Mode Dan Format	
11	Struktur dan Fungsi CPU	
12	Reduced Instruction Set Computers (RISC)	
13	Reduced Instruction Set Architecture (RISA)	
14	Quiz and Review	
15	Ujian Akhir Semester (UAS)	



DAFTAR PUSTAKA

1. Stallings, William.1998.Organisasi dan Arsitektur Komputer. Jilid 1. PT Prenhalindo.Jakarta.
2. Stallings, William.1998.Organisasi dan Arsitektur Komputer. Jilid 2. PT Prenhalindo.Jakarta.
3. Tanenbaum, Andrew S. 2001. Organisasi Komputer Terstruktur. Jilid 1. Salemba Teknika. Jakarta.
4. Tanenbaum, Andrew S. 2001. Organisasi Komputer Terstruktur. Jilid 2. Salemba Teknika. Jakarta
5. Soepono, S.1995.Organisasi Komputer. STMIK Guna Darma. Jakarta.
6. Stalling, W. 2002. Organisasi dan Arsitektur Komputer, Edisi 6. Jakarta: PT. Indeks, Kelompok Gramedia.



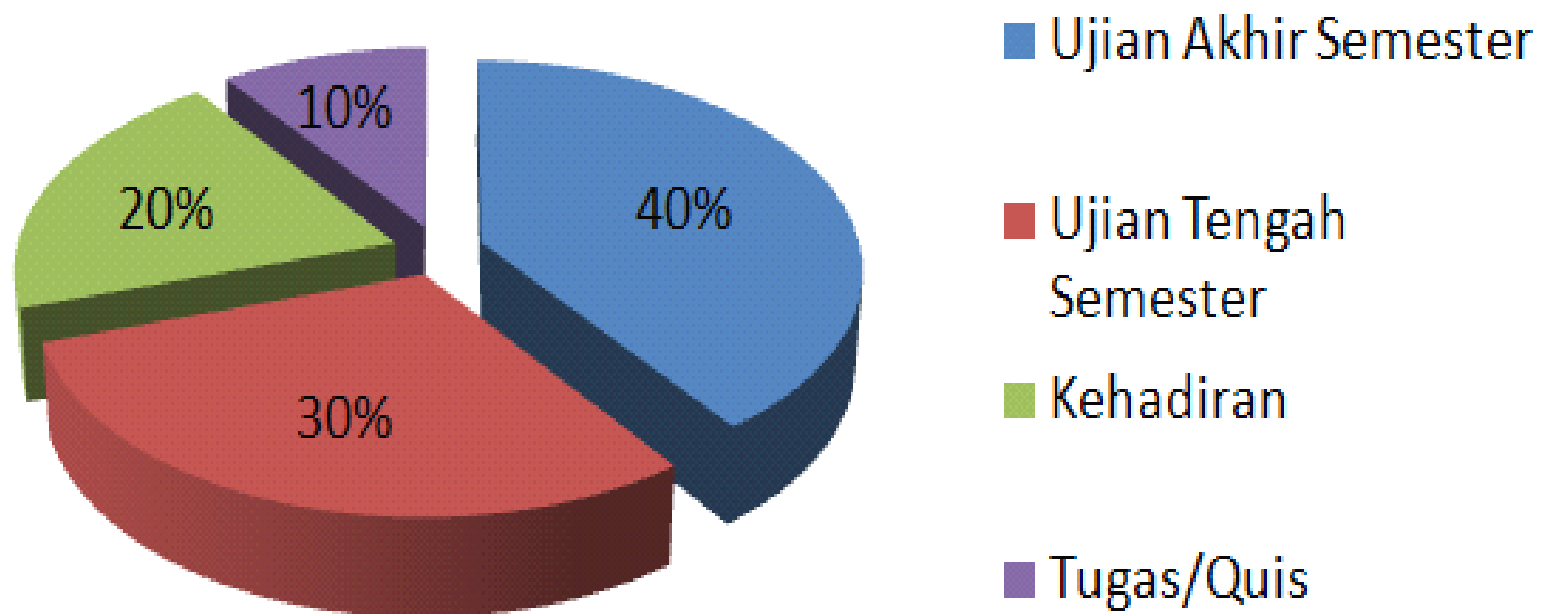
RENCANA PEMBELAJARAN

- Pertemuan 1 s.d 14 disampaikan dengan Metode Ceramah, Metode Diskusi dan Latihan Soal.
- Pada akhir pertemuan akan dilakukan latihan soal (quiz) yang nilainya akan dijadikan “Nilai Tugas”.



Sistem Penilaian

Prosentase





DESKRIPSI SINGKAT TUGAS

Mahasiswa harus menjawab beberapa pertanyaan dalam soal yang diberikan oleh dosen yang bersangkutan.

Soal diambil dari pertemuan yang telah diajarkan oleh dosen.



PERTEMUAN

1

EVOLUSI KOMPUTER



Organisasi Komputer

Organisasi Komputer berkaitan dengan unit-unit operasional dan interkoneksinya yang merealisasikan spesifikasi arsitektural.

Contoh atribut organisasional :
signal-signal kontrol, interface komputer dan pheriperal dan teknologi memori yang digunakan.



Arsitektur Komputer

Arsitektur berkaitan dengan atribut-atribut sebuah sistem yang tampak (visible) bagi seorang pemrogram, yaitu atribut-atribut yang memiliki dampak langsung pada eksekusi logis sebuah program.

Contoh atribut arsitektural :

Set instruksi, jumlah bit, mekanisme I/O, teknik pengalamatan memori



Struktur Dan Fungsi Komputer

- Struktur adalah cara komponen-komponen dalam sebuah komputer saling terkait
- Fungsi adalah operasi pada masing-masing komponen sebagai bagian dari struktur



Fungsi Komputer

Sebuah komputer dapat melakukan fungsi pengolahan data, penyimpanan data, pemindahan data dan kontrol



Struktur Komputer

- **CPU (Central Processing Unit)**
Mengontrol operasi komputer dan membentuk fungsi-fungsi pengolahan datanya
- **Main Memory**
Tempat menyimpan data
- **Input Output (I/O)**
Memindahkan data antara komputer dengan lingkungan luarnya.
- **System Interconnections**
Beberapa mekanisme komunikasi antara CPU, main memory dan I/O



Komponen-komponen CPU

- **Control Unit (CU)**
Mengontrol operasi CPU dan pada gilirannya mengontrol komputer
- **Arithmetic and Logic Unit (ALU)**
Membentuk fungsi-fungsi pengolahan data komputer
- **Register**
Penyimpan internal bagi CPU
- **CPU Interconnections**
Mekanisme komunikasi antara CU, ALU dan register-register



Evolusi dan Kinerja Komputer



Generasi Pertama

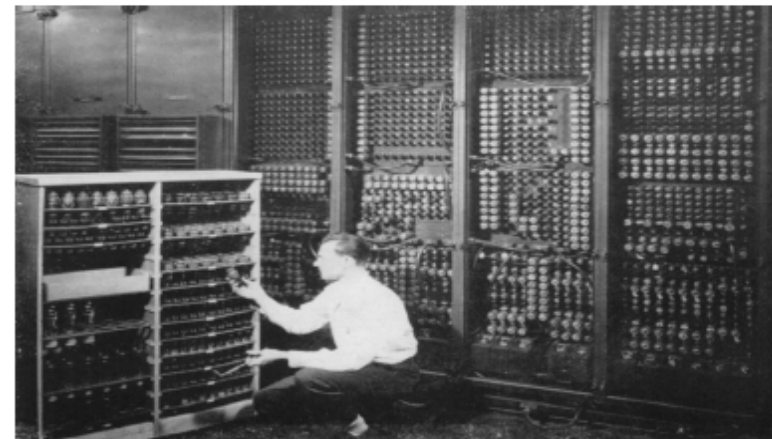
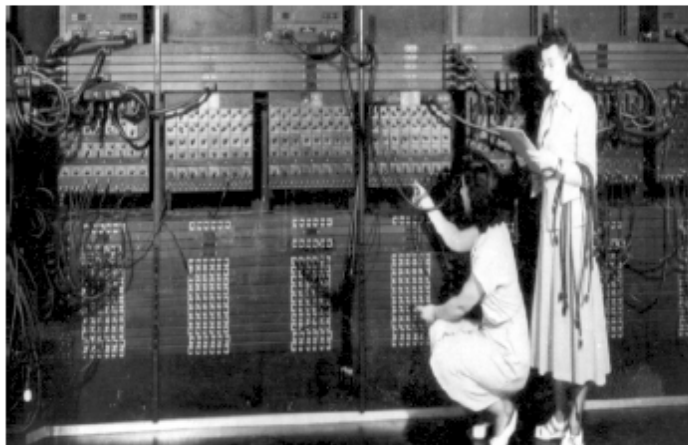
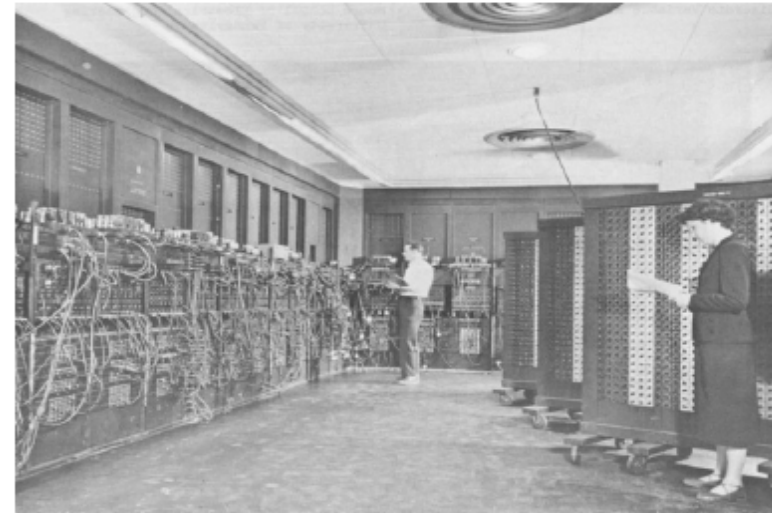
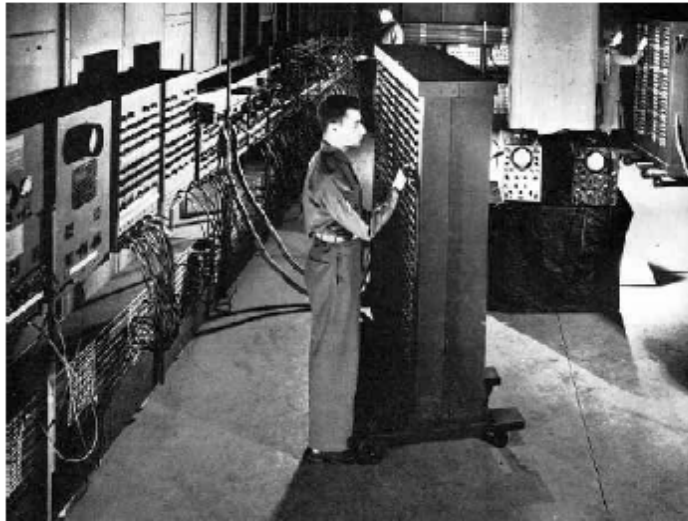
Ciri umum komputer generasi pertama:

- Menggunakan tabung hampa (Vaccum Tube)
- Program dibuat dalam bahasa mesin
- Memori utama dibuat dengan teknologi magnetic core storage
- Ukuran fisik komputer besar
- Membutuhkan daya listrik besar



ENIAC **(Electronic Numerical Integrator And Computer)**

- Dirancang oleh John Mauchly dan John Presper Enkert di Universitas Pennsylvania
- Berat 30 ton, volume 15000 kaki persegi, berisi lebih dari 18000 tabung vakum, membutuhkan daya 140 KW, melakukan operasi penambahan 5000 operasi/detik
- Memori terdiri dari 20 accumulator, masing-masing accumulator menampung 10 digit

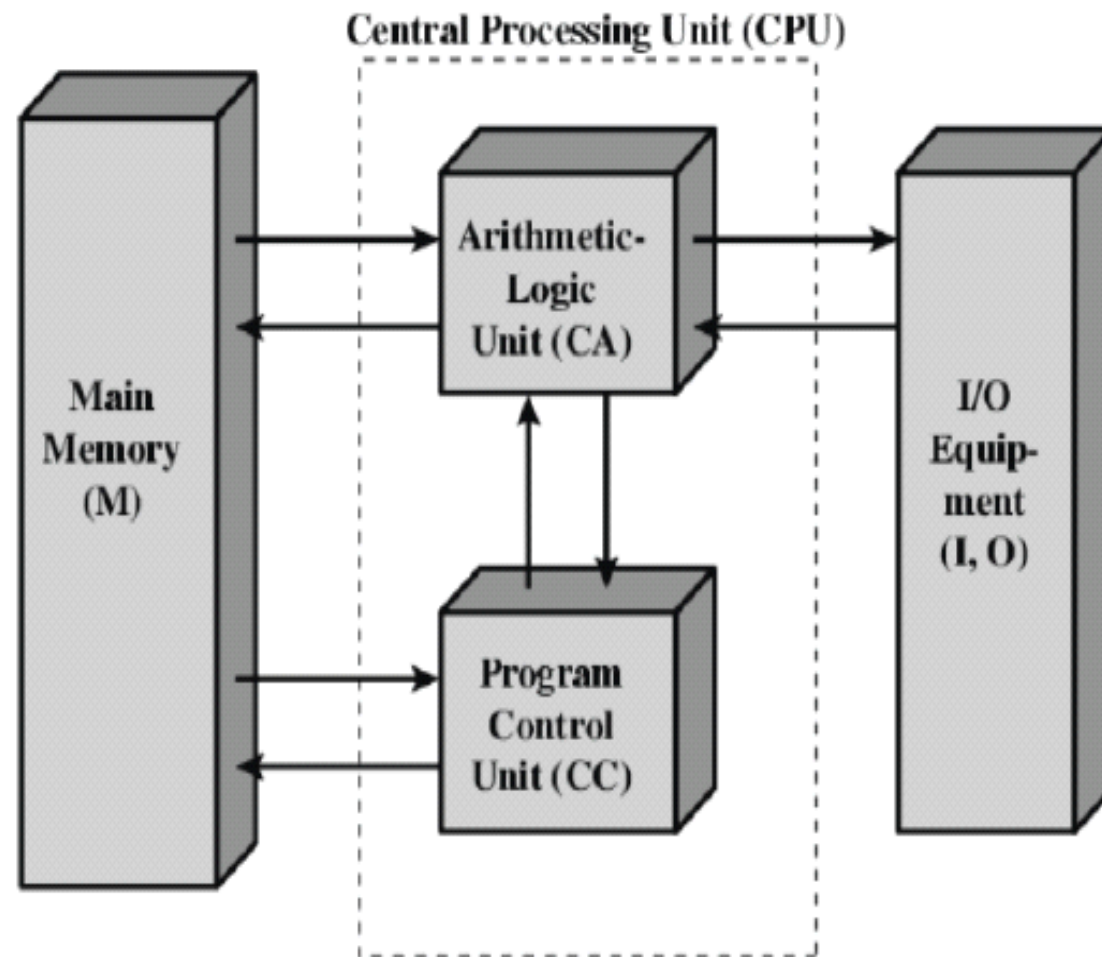


ENIAC



Mesin Von Neumann (Komputer IAS)

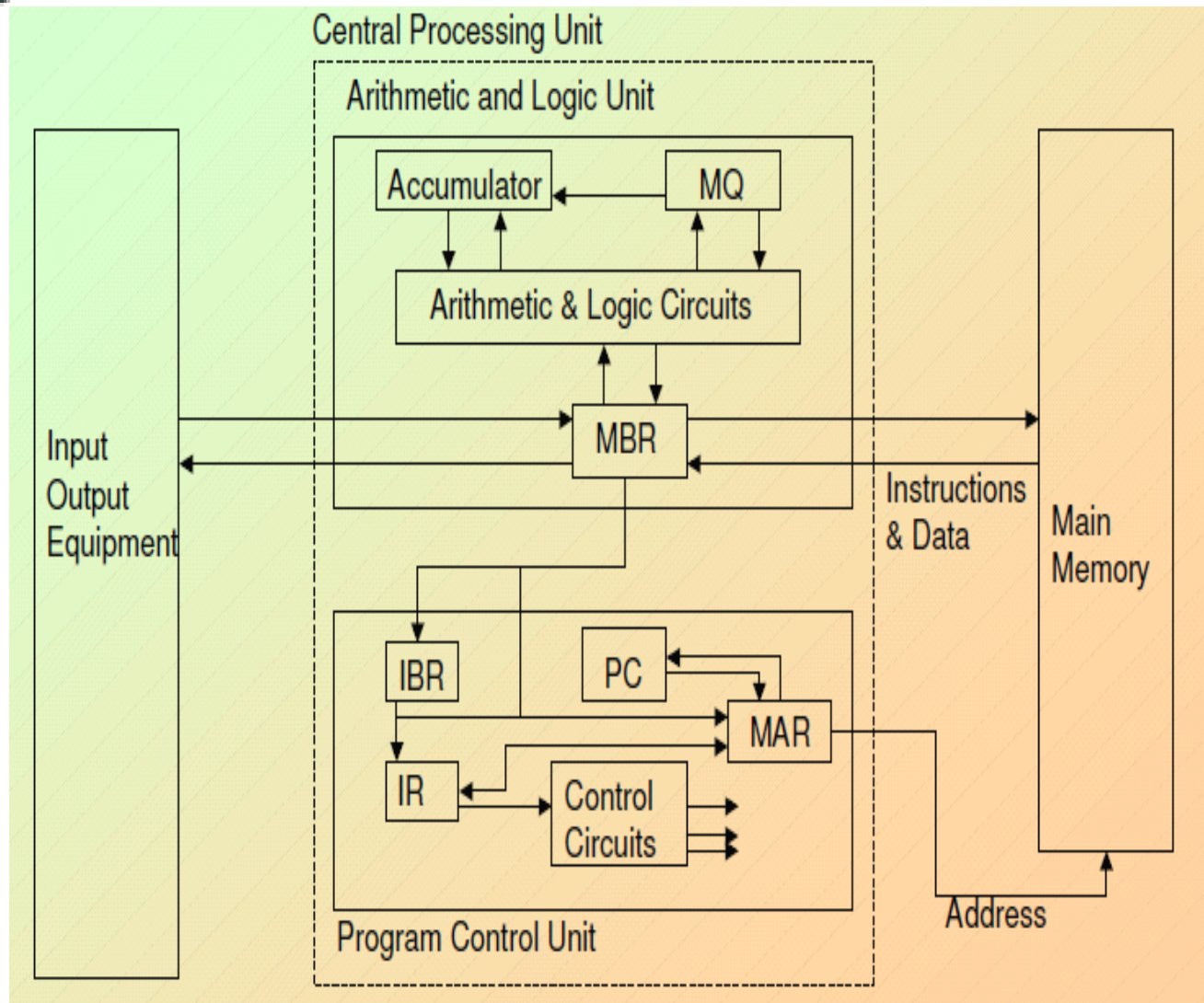
- Memori komputer IAS (Computer of Institute for Advanced Studies) terdiri dari 1000 lokasi penyimpanan yang di sebut word
- Satu word terdiri dari 40 binary digit
- Satu word dapat juga terdiri dari 20 bit instruksi, masing-masing instruksi terdiri dari 8 bit kode operasi dan 12 bit alamat
- Memiliki 21 buah instruksi
- Instruksi-instruksi dibagi menjadi 5 kelompok, yaitu : Data transfer, Unconditional branch, Conditional branch, Arithmetic dan address modify



Struktur Komputer IAS



Struktur IAS

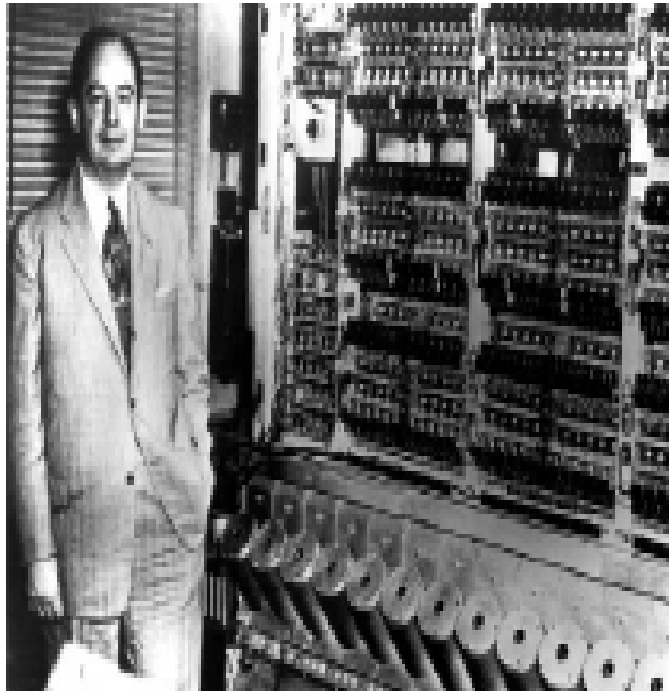




UNIVAC

(Universal Automatic Computer)

- Dirancang oleh Mauchly dan Eckert
- Produk pertama yang dihasilkan oleh The Eckert-Mauchly Computer Company
- Komputer komersial pertama yang dipasarkan



IAS



UNIVAC



Generasi Kedua

Ciri umum komputer generasi kedua :

- Teknologi dasar rangkaiannya berupa transistor
- Menggunakan bahasa pemrograman fortran, cobol, Algol
- Menggunakan memori sekunder berupa magnetic tape
- Ukuran fisik komputer lebih kecil dari komputer generasi pertama
- Membutuhkan daya listrik yang lebih kecil





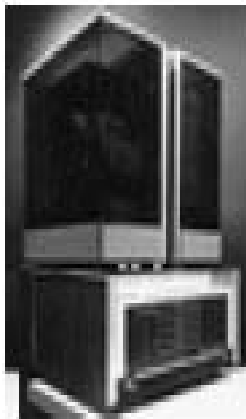
Generasi Ketiga

Ciri umum komputer generasi ketiga :

- Teknologi dasar pembangun rangkaian yang digunakan adalah IC (Integrated Circuit)
- Layar monitor dapat menampilkan gambar dan grafik
- Menggunakan magnetic tape sebagai memori sekunder
- Memiliki fitur jaringan
- Penggunaan daya listrik lebih hemat



Generasi Ketiga



Generasi Keempat





Generasi-generasi Selanjutnya

Setelah generasi ketiga perkembangan komputer didasarkan pada perkembangan IC (integrated Circuit)

- SSI (Small Scale Integration)
Sampai 100 komponen elektronik per chip
- MSI (Medium Scale Integration)
100-3000 komponen elektronik per chip
- LSI (Large Scale Integration)
3000-100000 komponen elektronik per chip
- VLSI (Very Large Scale Integration)
100000-1 juta komponen per chip
- ULSI (Ultra Large Scale Integration)
Lebih dari 1 juta komponen per chip



Perancangan Dan Kinerja Komputer

Kinerja sebuah sistem komputer merupakan hasil proses dari seluruh komponen komputer, yang melibatkan CPU, memori utama, memori sekunder, bus, peripheral.



Berbagai teknik yang dikembangkan untuk meningkatkan kinerja komputer, antara lain :

- **Branch Prediction**, teknik dimana prosesor memungkinkan mengamati terlebih dahulu di dalam software dan melakukan prediksi percabangan atau kelompok instruksi yang akan dieksekusi berikutnya
- **Data Flow Analysis**, prosesor akan menganalisa instruksi-instruksi yang tidak tergantung pada hasil atau data lainnya untuk membuat penjadwalan yang optimum dalam eksekusi
- **Speculative Execution**, dengan modal prediksi cabang dan analisis data, maka prosesor dapat melakukan eksekusi spekulatif terlebih dahulu sebelum waktunya.



Evolusi Pentium

Feature	8008	8080	8086	80386	80486
Tahun	1972	1974	1978	1985	1989
Jml Instruksi	66	111	133	154	235
Lebar bus alamat	8	16	20	32	32
Lebar bus data	8	8	16	32	32
Jumlah Flag	4	5	9	14	14
Jumlah Register	8	8	16	8	8
Kemampuan memori	16KB	64KB	1MB	4GB	4GB
I/O Port	24	256	64K	64K	4GB
Bus bandwidth	-	0.75MB/dt	5MB/dt	32MB/dt	32MB/dt
Waktu menambah register-register	-	1,3 μ dt	0.3 μ det	0.125 μ dt	0.006 μ dt



Karakteristik	286	386	486	Pentium	P6	P7
Mulai	1978	1982	1986	1989	1990	1993
Resmi	1982	1985	1989	1993	1995	1997
Pengiriman	1983	1986	1990	1994	1996	1998
Transistor	130.000	275.000	1,2 juta	3,1 juta	5,5 juta	10+ juta
Laju awal (MIPS)	1	5	20	100	250	500



THE END



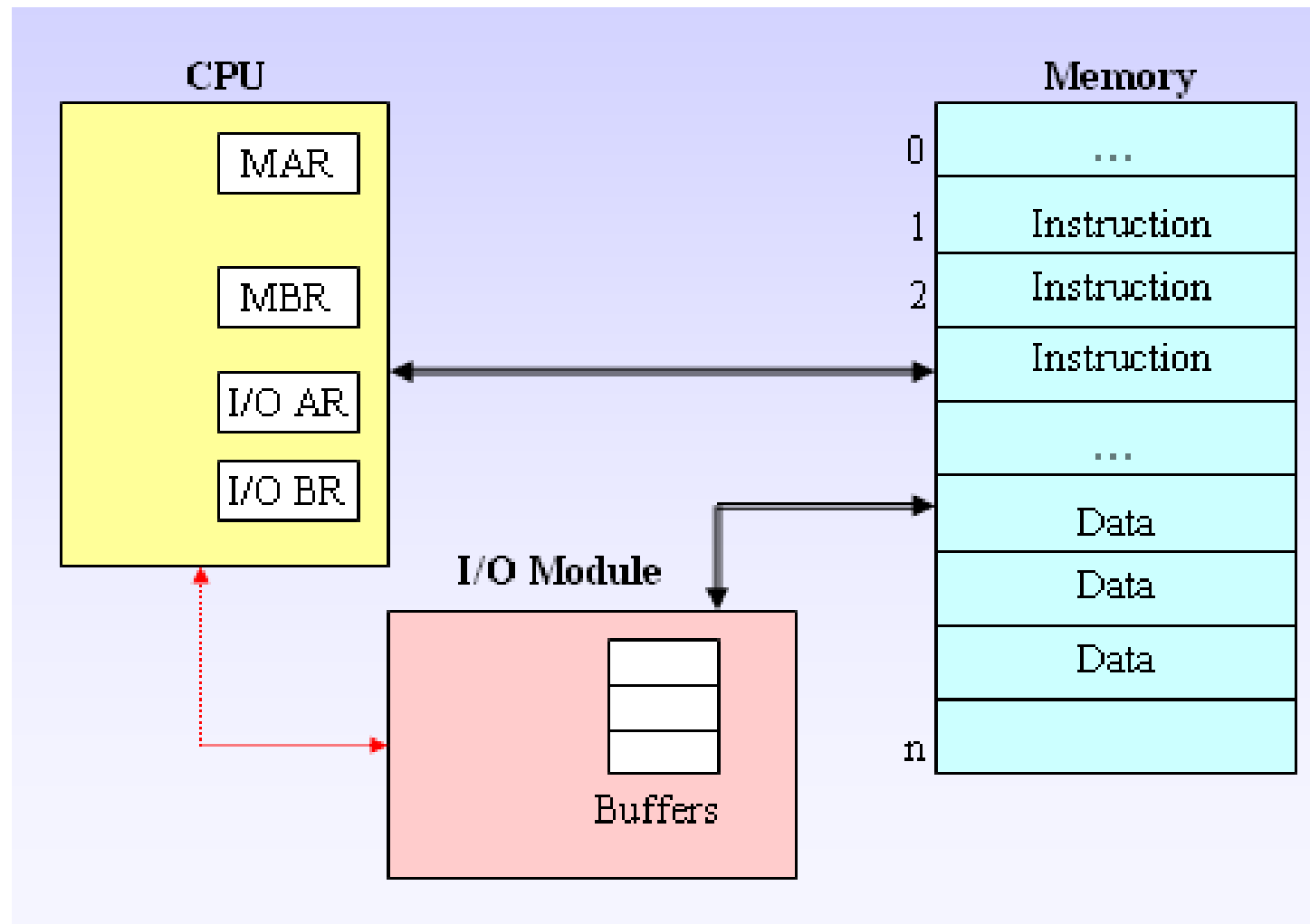
PERTEMUAN

2

BUS-BUS SISTEM



Komponen-komponen Komputer





Komponen CPU

Register

Register yang terdapat dalam CPU, yaitu :

MAR (Memory Address Register)

Menentukan alamat di dalam memori yang akan diakses untuk operasi Read/Write

MBR (Memory Buffer Register)

Berisi data yang akan di tuliskan ke dalam memori atau menerima data yang di baca dari memori



I/O AR (I/O Address Register)

Menspesifikasikan perangkat I/O yang akan diakses

I/O BR (I/O Buffer Register)

Menyimpan data yang akan dituliskan ke port atau data yang akan disalin dari port.

Alamat port ditunjuk oleh I/O AR

PC (Program Counter)

Mencatat alamat memori dimana instruksi di dalamnya akan dieksekusi



IR (Instruction Register)

Menampung instruksi yang akan dilaksanakan

AC (Accumulator)

Menyimpan data sementara baik data yang sedang diproses atau data yang dihasilkan



Modul I/O

- ✚ Memindahkan data dari perangkat eksternal ke CPU dan sebaliknya
- ✚ Modul ini berisi buffer internal untuk menampung data ini sementara sampai data itu di kirimkan.



Fungsi Komputer

- Fungsi dasar komputer adalah eksekusi program
- Program yang akan di eksekusi oleh CPU ada dalam memori

Ada 2 langkah pengambilan instruksi :

1. CPU membaca instruksi yang ada di memori (fetch)
2. CPU mengeksekusi setiap instruksi (execute)



Siklus Fetch

- ✚ Pada CPU yang umum, suatu register Program Counter (PC) di pakai untuk mengawasi instruksi yang akan di baca selanjutnya.
- ✚ Instruksi yang di baca akan di muatkan kedalam sebuah register (IR) Instructions Register



Aksi-aksi yang dilakukan oleh CPU ketika menginterpretasikan instruksi di bagi menjadi 4 kategori :

1. CPU --- Memori

Data di pindahkan dari CPU ke memori atau sebaliknya

2. CPU --- I/O

Data dapat di pindahkan ke atau dari dunia luar dengan pemindahan antara CPU dan modul I/O.



3. Pengolahan data

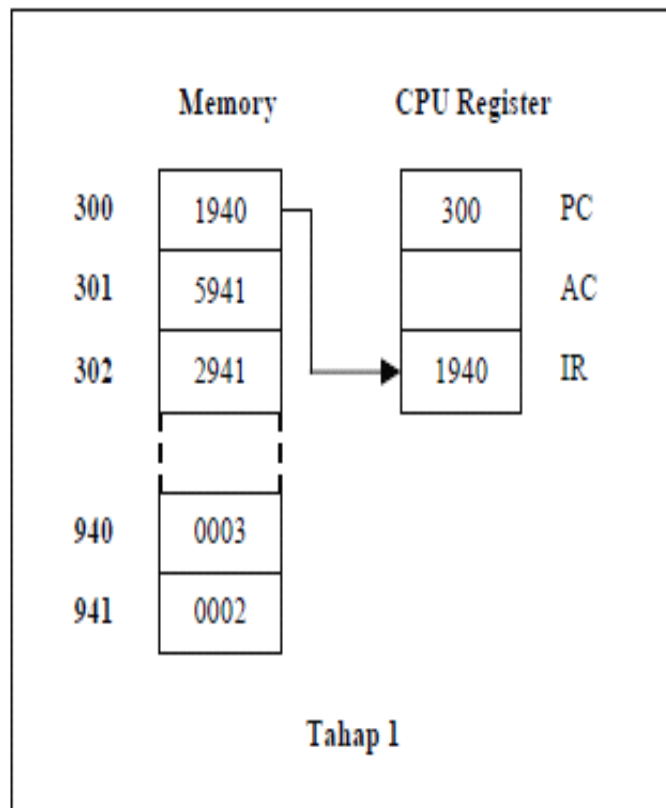
CPU dapat membentuk sejumlah operasi aritmatik /logik.

4. Control

Sebuah instruksi yang dapat mengubah urutan eksekusi



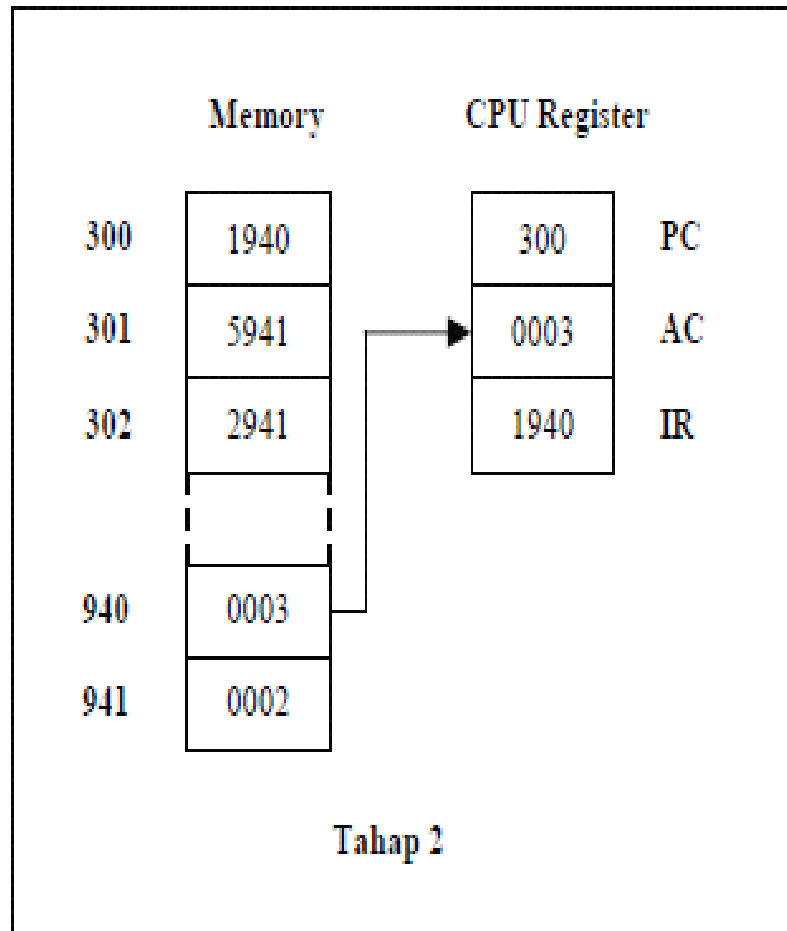
Contoh Eksekusi Program Dalam CPU



Tahap 1

Program Counter (PC) berisi 300 alamat instruksi pertama

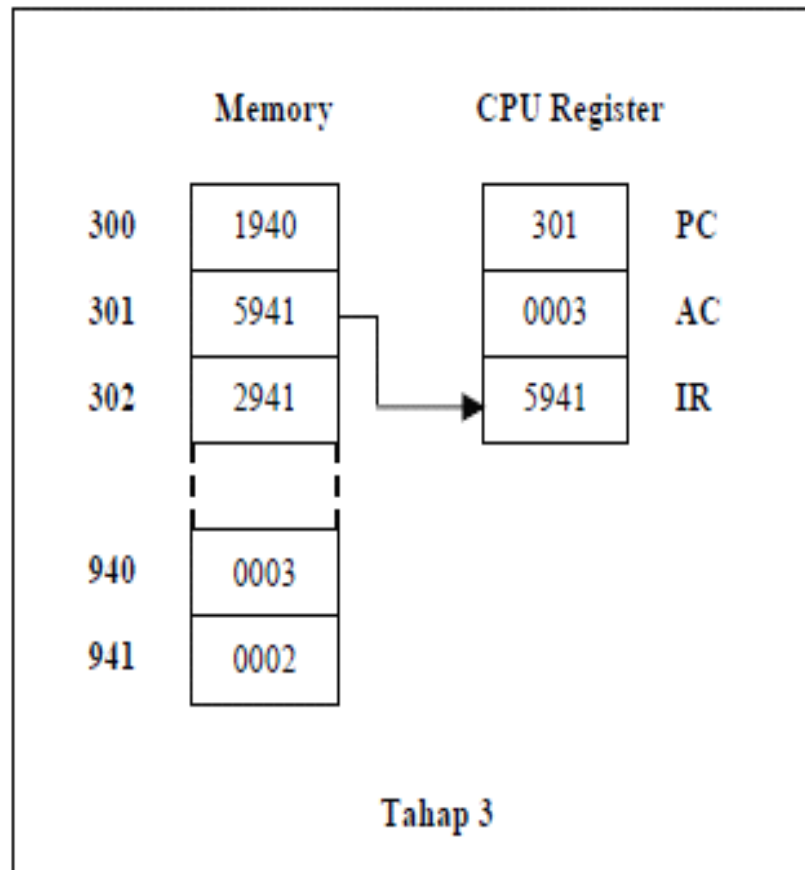
Alamat ini dimuatkan ke dalam Instruction Register (IR)



Tahap 2

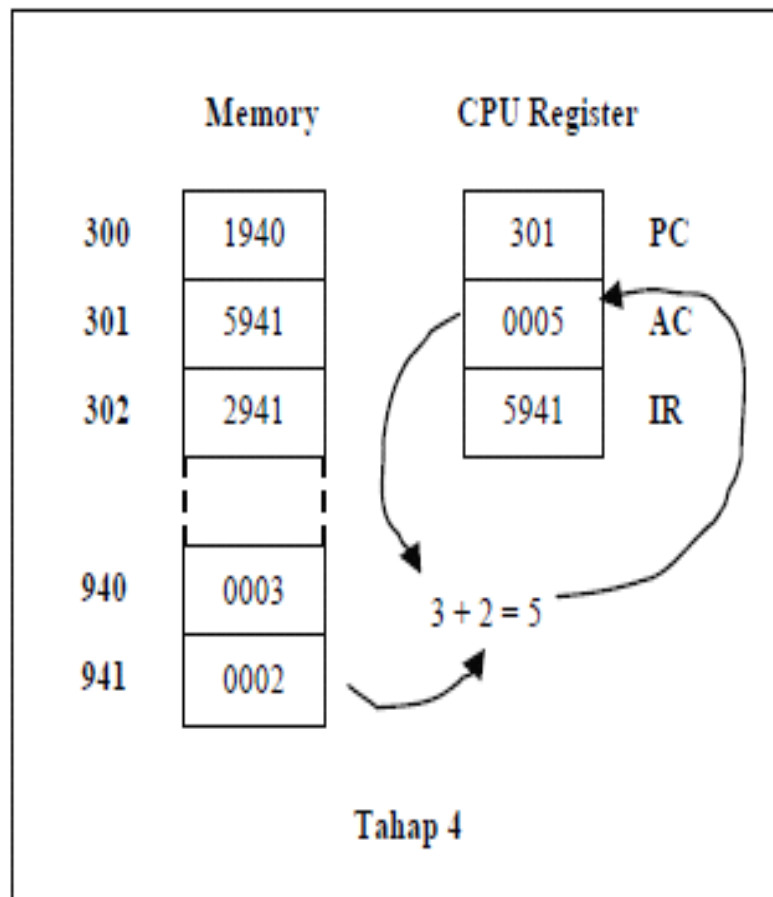
4 bit pertama di dalam IR mengindikasikan bahwa akumulator (AC) akan dimuatkan

12 bit sisanya menentukan alamat, yaitu 940



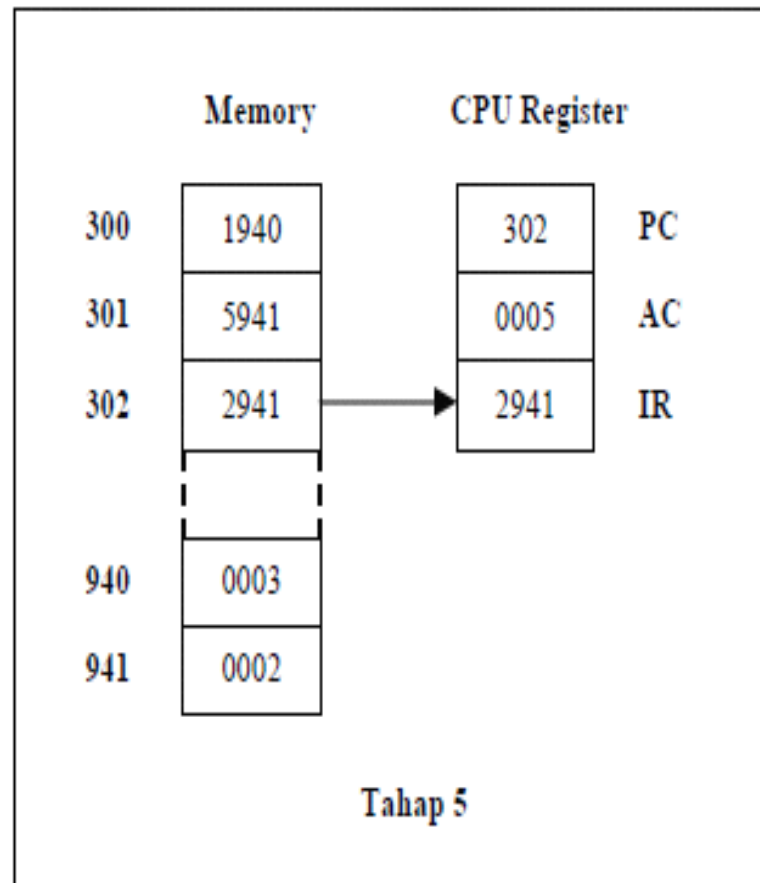
Tahap 3

PC dinaikkan nilainya
dan instruksi berikutnya
akan diambil



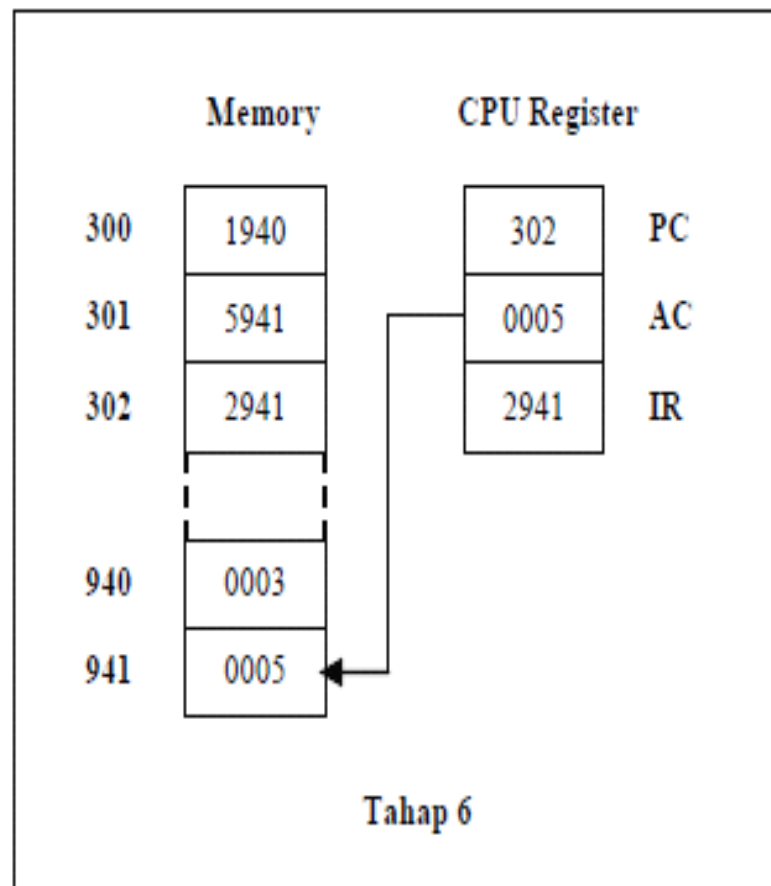
Tahap 4

Isi AC yang lama dan isi lokasi 941 ditambahkan dan hasilnya disimpan di dalam AC



Tahap 5

PC dinaikkan nilainya dan instruksi berikutnya akan diambil



Tahap 6

Isi AC akan disimpan pada lokasi 941



Interrupts

Kelas-kelas Interrupt :

1. Program

dibangkitkan dengan beberapa kondisi yang terjadi sebagai hasil dari suatu eksekusi instruksi

2. Timer

dibangkitkan oleh timer di dalam processor



3. I/O

di bangkitkan oleh I/O kontroller untuk memberi signal penyelesaian normal atau memberikan signal bergagai kondisi error

4. H/W Failure

di bangkitkan oleh kegagalan seperti kegagalan daya atau memori parity error



Interrupt disediakan terutama sebagai cara untuk meningkatkan efisiensi pengolahan, karena sebagian besar perangkat eksternal jauh lebih lambat di bandingkan prosessor



Interrupt & Siklus Instruksi

- Dengan memakai interrupt, processor dapat dipakai dalam mengeksekusi instruksi-instruksi lain operasi I/O yang sedang di laksanakan
- Ketika perangkat eksternal telah siap untuk dilayani, maka modul I/O untuk perangkat eksternal itu mengirimkan signal interrupt request ke prosessor.



- Prosesor menanggapi dengan menahan operasi program yang sedang di lakukannya, mencabangkannya ke suatu program untuk melayani perangkat I/O itu, yang di kenal dengan Interrupt Handler, dan kembali melaksanakan eksekusi mula-mula, setelah perangkat itu di layani.



Multiple Interrupt

1. Dengan tidak mengijinkan terjadinya interrupt lain pada saat suatu interrupt sedang di proses. (interrupt lain di tangguhkan)

Keuntungannya :

Pendekatan tersebut cukup baik dan sederhana karena interrupt di tangani dalam urutan yang cukup ketat.

Kekurangannya :

Pendekatan ini tidak memperhitungkan prioritas relatif atau kebutuhan waktu kritis



2. Dengan mendefinisikan prioritas bagi interrupt dan mengijinkan interrupt berprioritas tinggi menyebabkan interrupt handler yang berprioritas lebih rendah untuk menginterupsi diri sendiri.



Fungsi I/O

Sebuah I/O dapat bertukar data secara langsung dengan CPU.

Sebuah I/O juga dapat bertukar data langsung dengan memori.



Pertukaran Data Antara I/O dan Memori

- Dalam kasus ini CPU memberikan otoritas kepada modul I/O untuk membaca dari atau menulis ke memori, sehingga perpindahan data terjadi tanpa terpaut dengan CPU
- Selama perpindahan seperti itu, modul I/O mengeluarkan perintah baca/ tulis ke memori, yang membebaskan CPU dari tanggung jawab pertukaran data
- Operasi seperti ini di kenal dengan DMA (Dirrect Memory Access)



Struktur Interkoneksi

- + Komputer terdiri dari CPU – Memori – I/O
- + Komponen bus/ lintasan yang menghubungkan berbagai modul di sebut dengan struktur interkoneksi



Struktur interkoneksi harus mendukung jenis perpindahan berikut ini:

1. Memori ke CPU

CPU membaca sebuah instruksi atau satuan data dari memori

2. CPU ke Memori

CPU menuliskan sebuah satuan data ke memori



3. I/O ke CPU

CPU membaca data dari perangkat I/O melalui sebuah modul I/O

4. CPU ke I/O

CPU mengirimkan data ke perangkat I/O

5. I/O ke memori atau memori ke I/O

Pada kedua kasus ini sebuah modul I/O di izinkan untuk dapat bertukar data secara langsung tanpa melalui CPU dengan menggunakan DMA



Interkoneksi Bus

Bus adalah media transmisi yang dapat di gunakan bersama

Bila dua buah perangkat melakukan transmisi dalam waktu yang bersamaan, maka signal-signalnya akan bertumpang tindih dan menjadi rusak. Berarti harus hanya satu buah perangkat saja yang akan berhasil melakukan transmisi pada suatu saat tertentu.



Struktur Bus

Bus-bus yang menghubungkan komponen utama sistem (CPU, Memori, I/O) disebut dengan BUS SISTEM

Biasanya bus sistem terdiri dari 50 – 100 saluran yang terpisah



Klasifikasi Bus

Saluran Data :

memberikan lintasan bagi perpindahan data antara 2 modul sistem.

Saluran ini secara kolektif disebut BUS DATA

Saluran Alamat :

digunakan untuk menandakan sumber atau tujuan data pada bus data.

Saluran Kontrol :

di gunakan untuk mengontrol akses ke saluran alamat dan penggunaan data dan saluran alamat.



Elemen-elemen Rancangan BUS

A. Jenis

1. Dedicated

Suatu saluran bus di dedicated secara permanen diberi sebuah fungsi atau subset fisik komponen-komponen komputer

2. Multiplexed

Metode penggunaan saluran yang sama untuk berbagai keperluan



Keuntungan :

- 1. Dedicated :**
Throughput yang tinggi, karena terjadi kemacetan yang kecil
- 2. Multiplexed :**
Memerlukan saluran yang sedikit menghemat ruang dan biaya



Kerugian :

1. Dedicated :
Meningkatnya ukuran dan biaya sistem
2. Multiplexed :
Rangkaian lebih kompleks, terjadi penurunan kinerja, kerana event-event tertentu yang menggunakan saluran bersama-sama tidak dapat berfungsi secara paralel



B. Metode Arbitrasi

- 1. Tersentralisasi :**
sebuah perangkat H/W (pengontrol bus arbiter) bertanggung jawab atas alokasi waktu pada bus
- 2. Terdistribusi :**
tidak terdapat pengontrol sentral, tapi setiap modul terdiri dari access control logic dan modul-modul berkerja sama untuk memakai bus bersama-sama



C. Timing

1. Synchronous

terjadinya event pada bus ditentukan oleh clock

2. Asynchronous

terjadinya event pada bus ditentukan oleh event sebelumnya



D. Lebar Bus

- Ⓐ Semakin lebar bus data, semakin besar bit yang dapat di transfer pada suatu saat
- Ⓐ Semakin lebar bus alamat, semakin besar range lokasi yang dapat di referensi



Jenis Transfer Data

- + **Read**
- + **Write**
- + **Read modify write**
- + **Read after write**
- + **Blok**



Bus PCI (Peripheral Component Interconnect) (1990)

- **Merupakan bus yang tidak tergantung processor dan bandwidth tinggi yang dapat berfungsi sebagai bus mezzanine/bus berkecepatan tinggi**
- **Mezzanine adalah bus berkecepatan tinggi yang sangat terintegrasi dengan sistem**



Future Bus+

Future Bus+ adalah standard bus asinkron yang berkinerja tinggi



Syarat-syarat Future Bus

- Tidak tergantung pada arsitektur, processor dan teknologi tertentu
- Memiliki protokol transfer asinkron dasar
- mengizinkan protokol tersinkronisasi pada sumber untuk kebutuhan optional
- tidak berdasarkan pada teknologi terancang



- terdiri dari protokol-protokol paralel terdistribusi penuh dan arbitrase yang mendukung baik protokol circuit switched maupun protokol split transactions
- Menyediakan dukungan bagi sistem-sistem yang fault-tolerant dan yang memiliki reliabilitas tinggi
- menawarkan dukungan langsung terhadap memori berbasis cache yang dapat digunakan bersama
- memberikan definisi transportasi pesan yang compatible



- Future bus+ mendukung bus-bus data 32,64, 128,256 bit
- Future bus+ mendukung baik model terdistribusi maupun tersentralisasi
- Future bus+ merupakan salah satu standar bus yang secara teknis paling kompleks
- Future bus+ merupakan spesifikasi bus yang dapat di gunakan untuk bus prosessor –memori atau yang dapat di gunakan dengan PCI untuk mendukung peripheral-peripheral berkecepatan tinggi.



Perbedaan PCI dan Future bus+

- ✗ PCI di tujukan bagi implementasi murah yang membutuhkan bidang fisik secara minimal
- ✗ Future bus+ dimaksudkan untuk memberikan fleksibilitas yang tinggi dan fungsionalitas yang luas untuk memenuhi kebutuhan berbagai sistem yang berkinerja tinggi terutama sistem-sistem yang mahal.



The End



PERTEMUAN

3

INTERNAL MEMORI



Karakteristik-karakteristik penting sistem memori

A. Lokasi

CPU
Internal
External

B. Kapasitas

Ukuran Word
Ukuran Block

C. Satuan Transfer

Word
Block

D. Metode Akses

Sequential Acces
Direct Access
Random Access
Associative Access



E. Kinerja

Access time

Cycle time

Transfer rate

F. Tipe Fisik

Semi Konduktor

Permukaan Magnetik

G. Karakteristik Fisik

Volatile/Non Volatile

Erasable/Non Erasable



Kapasitas

- ❖ Karakteristik memori yang jelas adalah kapasitasnya
- ❖ Kapasitas ini dinyatakan dalam byte (1 byte = 8 bit) atau word.
- ❖ Panjang word yang umum adalah 8, 16 dan 32 bit
- ❖ Kapasitas eksternal memory biasanya dinyatakan dalam byte



Metode Akses

1. Sequential access

Memori diorganisasikan menjadi unit-unit data yang disebut record. Waktu untuk mengakses record sangat bervariasi

Contoh : Pita magnetik

2. Direct access

Setiap block dan record memiliki alamat-alamat yang unik berdasarkan lokasi fisik

Contoh : Disk



3. Random access

Setiap addressable locations di dalam memori memiliki mekanisme yang unik dan pengalamatan yang secara fisik wired-in. Waktu untuk mengakses lokasi tertentu tidak tergantung pada urutan akses sebelumnya dan bersifat konstan.

Contoh : main memori

4. Associative access

Sebuah word dicari berdasarkan pada isinya bukan berdasarkan pada alamatnya

Contoh : Cache memory



Kinerja

❖ Access Time

waktu yang di butuhkan untuk melakukan operasi baca atau tulis

❖ Cycle Time

access time ditambah dengan waktu tambahan yang di perlukan agar transient hilang dari signal

❖ Transfer Rate

kecepatan data agar dapat di transfer ke unit memori atau di transfer dari unit memori

$$T_n = T_a + N/R$$

T_n = waktu rata-rata untuk W/R N bit

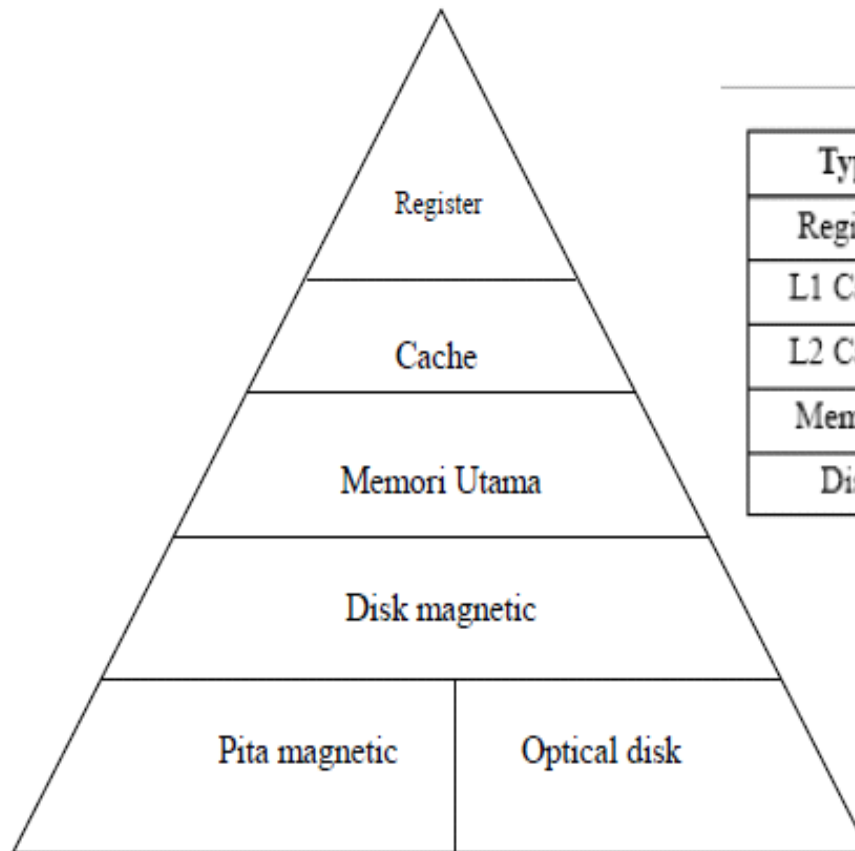
T_a = waktu akses rata-rata

N = jumlah bit

R = kecepatan transfer, dalam bit/detik (bps)



HIRARKI MEMORI



Memory Hierarchy

Type	Size	Speed	Bandwidth
Register	< 1 KB	1 ns	9600 MB/s
L1 Cache	< 256 KB	10 ns	3200 MB/s
L2 Cache	< 8 MB	30 ns	800 MB/s
Memory	< 4 GB	100 ns	133 MB/s
Disk	> 1 GB	20 ms	4 MB/s



MEMORI UTAMA SEMI KONDUKTOR



Jenis memori semi konduktor Random Access

RAM Terbagi 2 :

1. RAM Statis

2. RAM Dinamis



1.RAM Statis

- + Nilai nilai biner di simpan dengan menggunakan konfigurasi gate logic flip-flop tradisional.
- + Ram statis akan menampung data sepanjang daya listrik di sediakan untuknya



2. RAM Dinamis

- ✚ Disusun oleh sel-sel yang menyimpan data sebagai muatan listrik pada kapasitor
- ✚ Keberadaan dan ke tidak beradaan pada kapasitor di interpretasikan sebagai bilangan 1 atau 0, kerana kapasitor memiliki kecenderungan alami untuk mengosongkan muatan, maka RAM dinamis memerlukan pengisian muatan



SRAM	DRAM
Lebih cepat	Lebih lambat
Lebih mahal	Lebih murah
6 transistor/bit sehingga densitas kecil	1 transistor dan 1 kapasitor untuk 1 bit sehingga densitas sangat tinggi (banyak bit per chip)
Lebih sulit dibuat	Lebih mudah dibuat
Digunakan sebagai L1,L2 cache	Digunakan sebagai RAM
Tidak membutuhkan refresh karena isi tidak hilang selama daya/listrik tetap hidup	Membutuhkan refresh 300 ms
Dibuat menggunakan rangkaian yang sama dengan flipflop dasar	Tidak menggunakan flipflop, terdiri dari serangkaian sel-sel, masing-masing sel berisi 1 transistor dan 1 kapasitor berukuran kecil. Kapasitor dapat diisi atau dikosongkan



ROM

Berisi pola data permanen yang tidak dapat di ubah

Jenis-jenis ROM

- Programmable ROM (PROM)
- Erasable Programmable ROM (EPROM)
- Electrically Erasable Programmable ROM (EEPROM)



Cache Memory

- ❖ Cache memory di tujukan untuk memperoleh kecepatan memori yang mendekati kecepatan memori tercepat yang bisa di peroleh, sekaligus memberikan ukuran memori yang besar dengan harga yang lebih murah
- ❖ Cache adalah memori kecil yang berkecepatan tinggi
- ❖ Cache dapat kita lihat sebagai suatu memori buffer bagi memori utama



- ❖ Cache berisi salinan sebagian memori utama
- ❖ Pada saat sebuah cpu speed membaca word memori maka dilakukan pemeriksaan untuk mengetahui apakah word itu terdapat dalam chace.
- ❖ Bila sudah ada, maka word akan dikirimkan ke CPU, sedang bila tidak ada blok main memori yang terdiri dari sejumlah word yang tetap akan di baca cache dan kemudian di kirim ke CPU



- ❖ Meskipun cache menggunakan informasi yang tersimpan dalam memori utama, tetapi ia tidak berhadapan secara langsung dengan memori utama.
- ❖ Apa yang tersimpan dalam cache dan juga ditempatkan ke dalam memori utama dikerjakan sesuai perintah CPU.
- ❖ Agar suatu cache dapat efektif maka ia harus lebih cepat dari memori utama.



- ❖ Pada umumnya, waktu akses memori cache berkisar antara 50-100 nanodetik (lebih cepat 5-10 kali lebih cepat dari memori utama)
- ❖ Cache dapat menampung 64 KB



Operasi Pembacaan Cache

- A. Direct**
- B. Assosiative**
- C. Set assosiative**



A. Pemetaan Langsung

- Alamat memori utama dibagi menjadi 2 field yaitu tag dan indeks.
- Jumlah bit dalam indeks berhubungan dengan ukuran cache
- Ketika suatu word direferensikan, bit indeks pada alamat digunakan untuk pengaksesan cache



- Jika field tag cocok dengan bit tag pada alamat terjadi suatu hit, jump jika sebaliknya akan terjadi suatu miss
- Organisasi cache ini mempunyai waktu akses yang lebih cepat dari pada RAM karena mempunyai field alamat yang lebih kecil
- Kelemahan pada cara adalah tidak dapat menyimpan secara bersamaan apabila dua buah word mempunyai indeks yang sama dan tag yang berbeda.



B. Pemetaan Asosiatif

- @ Ketika suatu memori asosiatif digunakan untuk sebuah cache maka diperlukan suatu pemetaan yang berbeda
- @ Dengan pemetaan asosiatif alamat word memori utama dan isinya (data) tersimpan di dalam cache.



- ② Untuk mereferensikan suatu word tertentu, alamat disimpan ke bagian yang berhubungan pada register argumen (A) dan register kunci (K) di setup untuk membandingkan hanya field alamatnya.
- ② Pemetaan asosiatif memungkinkan adanya penyimpanan semua word yang mempunyai indeks yang sama dan tag yang berbeda kedalam cache ini.



C. Pemetaan kelompok assosiatif

- Merupakan kombinasi dari kedua organisasi sebelumnya
- Dalam jenis organisasi ini, dua word dengan indeks yang sama dan tag yang berbeda dapat disimpan didalam kelompok yang sama
- Karena setiap word dalam kelompok memori assosiatif hanya menyertakan bit tag tambahan, bukan alamat lengkap, maka waktu aksesnya lebih cepat dan harganya lebih murah



Organisasi DRAM

A. Enhanced DRAM (EDRAM)

- Dengan mengintegrasikan cache SRAM yang kecil pada keping DRAM generik
- EDRAM mencakup beberapa feature lainnya yang dapat meningkatkan kinerja
- Operasi refresh dapat dilakukan secara paralel dengan operasi pembacaan cache



B. Cache DRAM (CDRAM)

Mencakup cache SRAM- cache SRAM yang lebih besar dari EDRAM

C. Synchronous DRAM (SDRAM)

SDRAM saling bertukar data dengan processor yang di sinkronkan dengan signal pewaktu eksternal dan bekerja dengan kecepatan penuh bus prosessor atau memori tanpa mengenal keadaan wait



D. Rambus DRAM (RDRAM)

- Menggunakan pendekatan terhadap masalah memori bandwidth yang lebih revolusioner
- RDRAM memiliki kelajuan sekitar 500 Mbps (Dram = 33 Mbps).

E. RAMLINK

- Ralink berkonsentrasi pada interface processor atau memori di bandingkan pada arsitektur internal keping DRAM
- Ralink adalah memori interface yang memiliki koneksi point to point yang di susun dalam bentuk cincin.



The End



PERTEMUAN

4

EXTERNAL MEMORI



- Kebutuhan akan memori utama saja tidak mencukupi maka diperlukan peralatan tambahan untuk menyimpan data yang lebih besar dan dapat dibawa kemana-mana.
- Semakin besarnya peralatan penyimpanan maka akan mempengaruhi waktu pemrosesan data.



Peralatan Penyimpanan Data

- ☐ Magnetic Disk
 - Floppy Disk
 - IDE Disk
 - SCSI Disk
- ☐ RAID
- ☐ Optical Disk
 - CDROM
 - CD-R
 - CD-RW
 - DVD
- ☐ Pita Magnetik



Magnetik Disk

- Disk merupakan sebuah piringan bundar yang terbuat dari logam/ plastik yang di lapiasi dengan bahan yang dapat di magnetisasi
- Data direkam diatasnya dan di baca oleh head (kumparan pengkonduksi/conducting coil)
- Head merupakan perangkat yang relatif berukuran kecil yang dapat membaca atau menulis
- Lebar head = lebar track



- 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 untuk 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



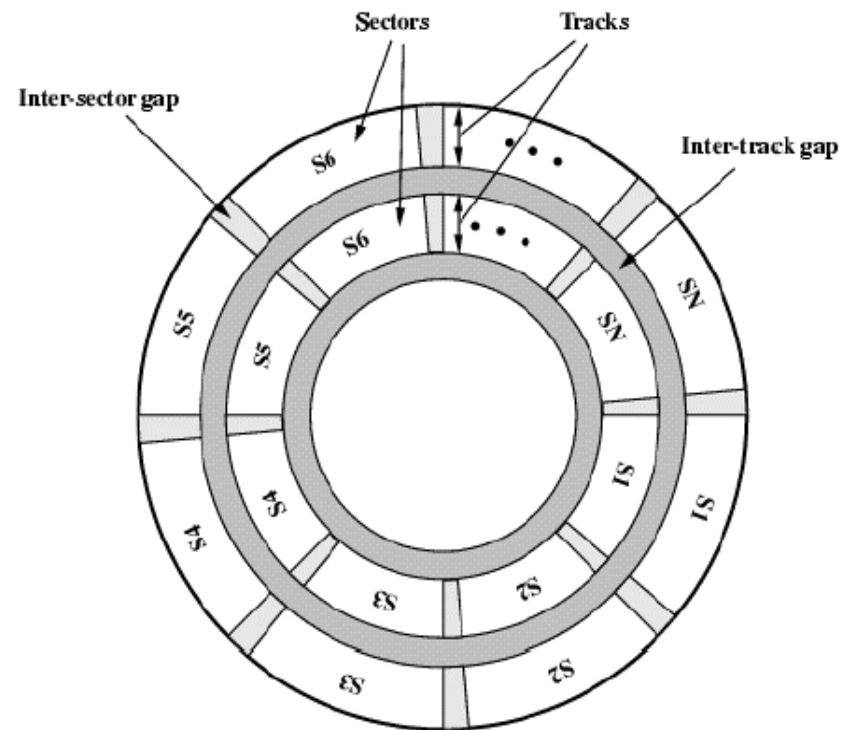
- Semakin ke dalam disk maka kerapatan (density) disk akan bertambah besar
- Data dikirim ke memori ini dalam bentuk blok, umumnya blok lebih kecil kapasitasnya daripada track
- Blok – blok data disimpan dalam disk yang berukuran blok, yang disebut sector
- Track biasanya terisi beberapa sector, umumnya 10 hingga 100 sector tiap tracknya



Layout dan Pembacaan

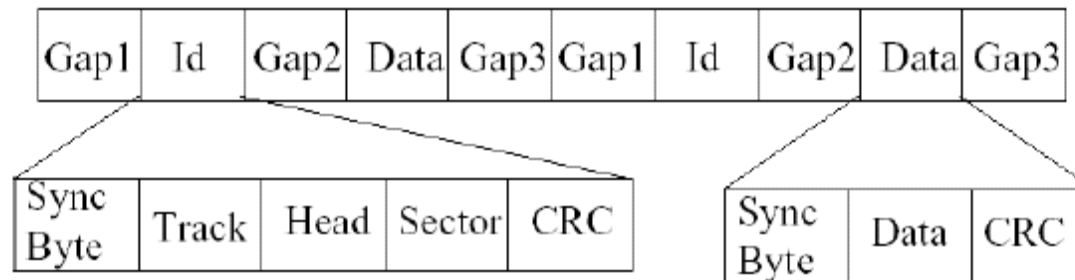
BACA dan TULIS ?

- Head harus bisa mengidentifikasi titik awal atau posisi – posisi sector maupun track
- Data yang disimpan akan diberi header data tambahan yang menginformasikan letak sector dan track suatu data
- Tambahan header data ini hanya digunakan oleh sistem disk drive saja tanpa bisa diakses oleh pengguna





Format data pada track disk



- Field ID merupakan header data yang digunakan disk drive menemukan letak sector dan tracknya.
- Byte SYNCH adalah pola bit yang menandakan awal field data

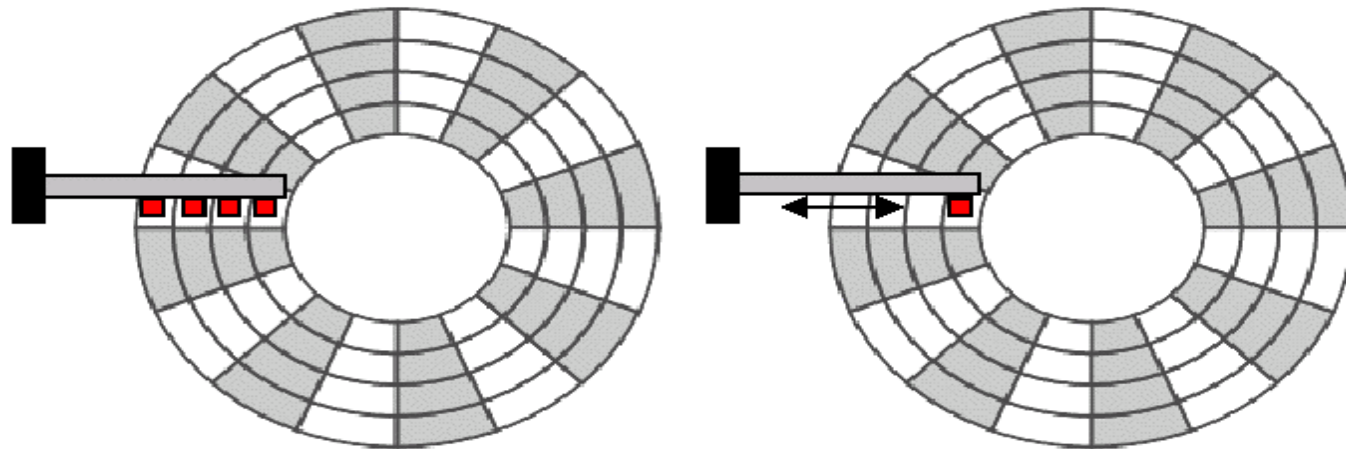


Karakteristik Magnetik Disk

Karakteristik	Macam
Gerakan head	1. Fixed head (satu per track) 2. Movable head (satu per surface)
Portabilitas disk	1. Nonremovable disk 2. Removable disk
Sides	1. Single-sided 2. Double-sided
Platters	1. Single-platter 2. Multiple-platter
Mekanisme head	1. Contact (floppy) 2. Fixed gap 3. Aerodynamic gap (Winchester)



Gerakan Head



(a) Fixed head

(b) Movable head

- Pada head tetap setiap track memiliki kepala head sendiri, sedangkan pada head bergerak, satu kepala head digunakan untuk beberapa track dalam satu muka disk.
- Pada head bergerak adalah lengan head bergerak menuju track yang diinginkan berdasarkan perintah dari disk drive-nya



Portabilitas disk

- Disk yang tetap (non-removable disk)
- Disk yang dapat dipindah (removable disk).

Keuntungan disk yang dapat dipindah atau diganti – ganti adalah tidak terbatas dengan kapasitas disk dan lebih fleksibel



Sides/Sisi dan Platters/Piringan

Sides

- *Satu sisi disk (single sides)*
- *Dua muka disk (double sides)*

Platters

- *Satu piringan (single platter)*
- *Banyak piringan (multiple platter)*



Mekanisme head

- Head yang menyentuh disk (contact) seperti pada floppy disk, head yang mempunyai celah utara tetap maupun yang tidak tetap tergantung medan magnetnya. Celah atau jarak head dgn disk tergantung kepadatan datanya, semakin padat datanya di-butuhkan jarak head yang semakin dekat, namun semakin dekat head maka faktor resikonya semakin besar, yaitu terjadinya kesalahan baca.
- Teknologi Winchester dari IBM mengantisipasi masalah celah head diatas dengan model head aerodinamik. Head berbentuk lembaran timah yang berada dipermukaan disk apabila tidak bergerak, seiring perputaran disk maka disk akan mengangkat headnya.

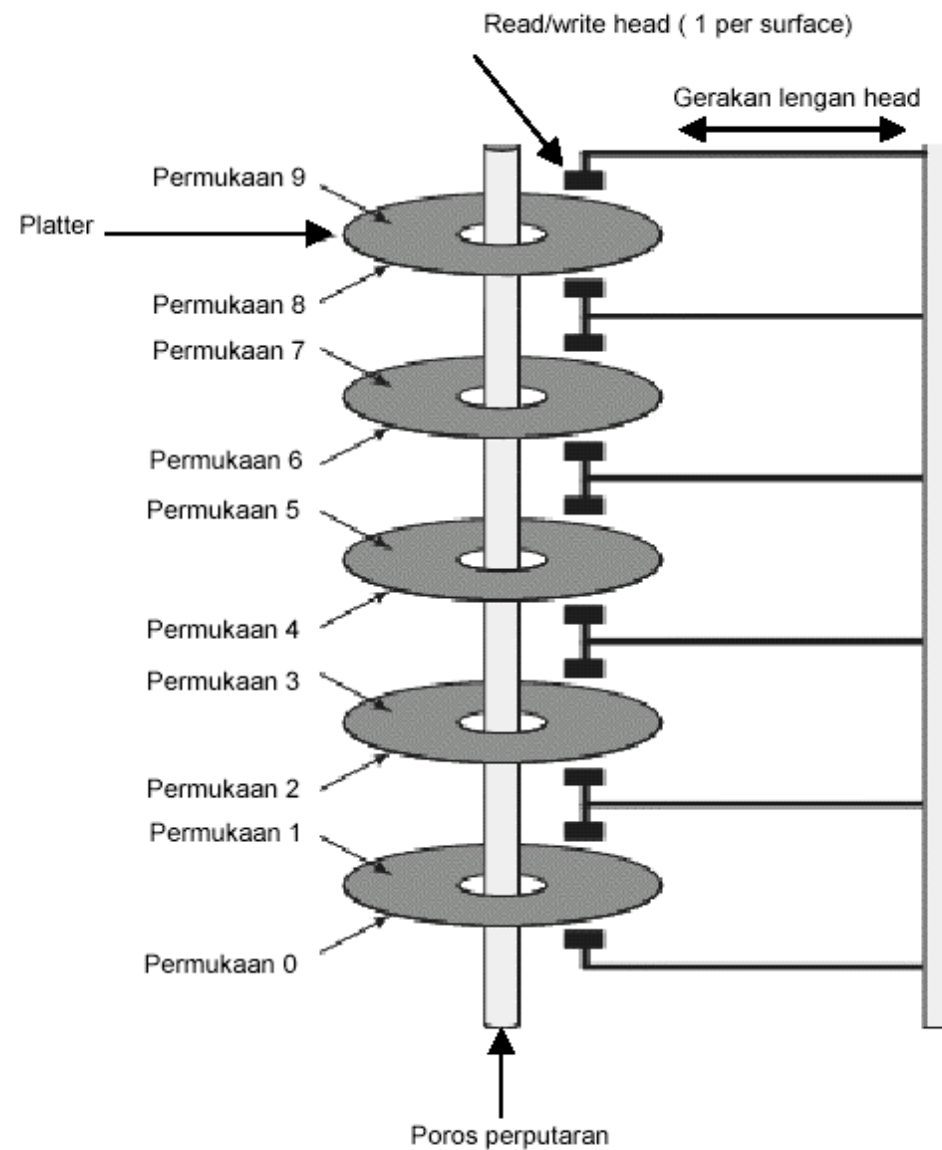


Mekanisme head (cont)

- Istilah Winchester dikenalkan IBM pada model disk 3340-nya. Model ini merupakan removable disk pack dengan head yang dibungkus di dalam pack. Sekarang istilah Winchester digunakan oleh sembarang disk drive yang dibungkus pack dan memakai rancangan head aerodinamis

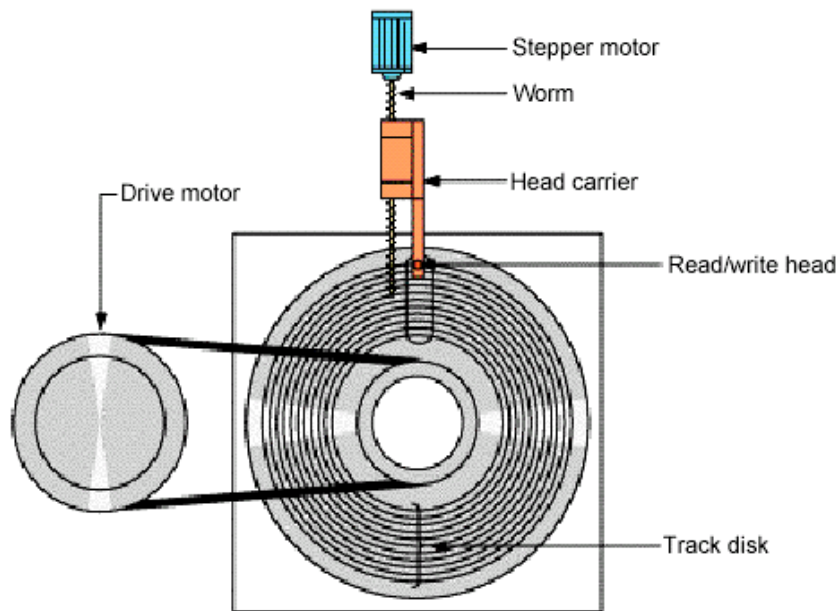


Disk piringan banyak (multiple platters disk)





Floppy Disk



- Karakteristik disket adalah head menyentuh permukaan disk saat membaca ataupun menulis.
- Efeknya disket tidak tahan lama dan sering rusak.
- Maka dibuat mekanisme penarikan head dan menghentikan rotasi disk ketika head tidak melakukan operasi baca dan tulis.
- Efeknya waktu akses disket cukup lama



Karakteristik berbagai macam disket

Parameter	LD 5,25"	HD 5,25"	LD 3,5"	HD 3,5"
Ukuran (inches)	5,25	5,25	3,5	3,5
Kapasitas (byte)	360K	1,2 M	720K	1,44 M
Tracks	40	80	80	80
Sectors/track	9	15	9	18
Heads	2	2	2	2
Rotasi/min	300	500	300	300
Data rate (kbps)	250	500	250	500
Tipe	flexible	flexible	rigid	rigid



IDE Disk (Harddisk)

- Saat IBM mengembangkan PC XT, menggunakan sebuah hardisk Seagate 10 MB untuk menyimpan program maupun data. Harddisk ini memiliki 4 head, 306 silinder dan 17 sektor per track, dicontrol oleh pengontrol disk Xebec pada sebuah kartu plug-in.
- Teknologi yang berkembang pesat menjadikan pengontrol disk yang sebelumnya terpisah menjadi satu paket terintegrasi, diawali dengan teknologi drive IDE (Integrated Drive Electronics) pada tengah tahun 980.
- Teknologi saat itu IDE hanya mampu menangani disk berkapasitas maksimal 528 MB dan mengontrol 2 disk.



IDE Disk (Harddisk) *cont.*

- IDE berkembang menjadi EIDE (Extended Integrated Drive Electronics) mampu menangani harddisk lebih dari 528 MB dan mendukung pengalamatan LBA (Logical Block Addressing), yaitu metode pangalamatan yang hanya memberi nomer pada sektor – sektor mulai dari 0 hingga maksimal 224-1.
- Metode ini mengharuskan pengontrol mampu mengkonversi alamat–alamat LBA menjadi alamat head, sektor dan silinder.
- Peningkatan kinerja lainnya adalah kecepatan tranfer yang lebih tinggi, mampu mengontrol 4 disk, mampu mengontrol drive CD-ROM



SCSI Disk (Harddisk)

- Disk SCSI (Small Computer System Interface) mirip dengan IDE dalam organisasi pengalamatannya.
- Perbedaan pada piranti antarmukanya yang mampu mentransfer data dalam kecepatan tinggi.
- Kecepatan transfernya tinggi, merupakan standar bagi komputer UNIX dari Sun Microsystems, HP, SGI, Macintosh, Intel terutama komputer server jaringan, dan vendor lainnya
- SCSI sebenarnya lebih dari sekedar piranti antarmuka harddisk.
- SCSI adalah sebuah bus karena mampu sebagai pengontrol hingga 7 peralatan seperti: harddisk, CD ROM, rekorder CD, scanner, dll. Masing-masing peralatan memiliki ID unik
- sebagai media pengenalan oleh SCSI



Versi Disk SCSI

Nama	Data bits	Bus MHz	MB/det
SCSI-1	8	5	5
Fast SCSI	8	10	10
Wide Fast SCSI	16	10	20
Ultra SCSI	8	20	20
Wide Ultra SCSI	16	20	40
Ultra-2 SCSI	8	40	40
Wide Ultra-2 SCSI	16	40	80



Waktu Akses Disk

- 1. Seek time**
- 2. Rotational Latency**
- 3. Access Time**



RAID

(Redudancy Array of Independent Disk)

Karakteristik umum disk RAID

- Raid merupakan sekumpulan disk drive yang dianggap oleh sistem operasi sebagai sebuah drive logic tunggal
- Data di distribusikan ke drive fisik array
- Kapasitas redudant disk di gunakan untuk menyimpan informasi paritas yang menjamin recoverability data ketika kegagalan disk
- Raid terdiri dari 6 tingkat



- RAID (Redundancy Array of Independent Disk) merupakan organisasi disk memori yang mampu menangani beberapa disk dengan sistem akses paralel dan redundansi ditambahkan untuk meningkatkan reliabilitas.
- Kerja paralel 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 membacaan kembali



RAID tingkat 0

- Sebenarnya bukan RAID, karena tidak menggunakan redudansi dalam meningkatkan kinerjanya
- Data didistribusikan pada seluruh disk secara array merupakan keuntungan dari pada menggunakan satu disk berkapasitas besar
- RAID-0 menjadi model data strip pada disk dengan suatu management tertentu hingga data sistem data dianggap tersimpan pada suatu disk logik
- Mekanisme transfer data dalam satu sektor sekaligus sehingga hanya baik untuk menangani transfer data besar



RAID tingkat 1

- Redundansi diperoleh dengan cara menduplikasi seluruh data pada disk mirror nya
- Seperti RAID-0, RAID-1 juga menggunakan teknologi stripping
- Perbedaannya adalah dalam tingkat 1 setiap strip logik di-petakan ke dua disk yang secara logika terpisah sehingga setiap disk pada array akan memiliki mirror disk yang berisi data yang sama
- RAID-1 mahal
- RAID-1 memiliki kinerja 2 kali lipat dibandingkan RAID-0 pada operasi baca
- RAID-1 masih bekerja berdasarkan sektor



RAID tingkat 1

- Keuntungan RAID-1:
 1. Permintaan pembacaan dapat dilayani oleh salah satu disk karena ada dua disk berisi data yang sama, tergantung waktu akses yang tercepat
 2. Permintaan penyimpanan dilakukan pada dua disk secara paralel
 3. Terdapat back up data dalam disk mirror-nya



RAID tingkat 2

- RAID-2 menggunakan teknik akses paralel untuk semua disk
- Seluruh disk berpartisipasi dan mengeksekusi setiap permintaan sehingga terdapat mekanisme sinkronisasi perputaran disk dan headnya
- Teknologi stripping digunakan dalam tingkat ini, hanya stripnya berukuran kecil (word/byte)
- Koreksi kesalahan menggunakan sistem bit paritas dengan kode hamming



RAID tingkat 3

- Diorganisasikan mirip dengan RAID-2
- RAID-3 hanya membutuhkan disk redundant tunggal dan tidak bergantung pada jumlah array
- Bit paritas dikomputasikan untuk setiap data word dan ditulis pada disk paritas khusus
- Saat terjadi kegagalan drive, data disusun kembali dari sisa data yang masih baik dan dari informasi paritasnya
- Menggunakan akses paralel dengan data didistribusikan dalam bentuk strip-strip kecil
- Kinerjanya menghasilkan transfer berkecepatan tinggi, namun hanya dapat mengeksekusi sebuah permintaan I/O saja sehingga kalau digunakan pada lingkungan transaksi data tinggi terjadi penurunan kinerja



RAID tingkat 4

- Menggunakan teknik akses yang independen untuk setiap disknya sehingga permintaan baca atau tulis dilakukan secara paralel
- RAID ini cocok untuk menangani sistem dengan kelajuan transfer data yang tinggi
- Tidak memerlukan sinkronisasi disk
- Stripping data dalam ukuran yang besar
- Strip paritas bit per bit dihitung ke seluruh strip yang berkaitan pada setiap disk data
- Paritas disimpan pada disk paritas khusus
- Saat operasi penulisan array management software tidak hanya meng-update data tetapi juga paritas yang terkait
- Disk paritas khusus menjadikan keamanan data lebih terjamin tetapi memperlambat kinerja



RAID tingkat 5

- Merupakan teknologi RAID terbaru
- Menggunakan metode penghitungan dua paritas untuk alasan keakuratan danantisipasi terhadap koreksi kesalahan
- Paritas tersimpan pada disk lainnya
- Memiliki kecepatan transfer yang tinggi



Optical Memori

- Optical memori identik dengan CD (Compact Disk)
- Dikembangkan oleh Philip dan Sony pada tahun 1980
- CD merupakan disk yang tidak dapat di hapus yang dapat menyimpan lebih dari 60 menit informasi audio pada salah satu sisinya
- CD mampu menyimpan data dalam jumlah besar menjadikannya media penyimpan yang fleksibel digunakan diberbagai peralatan seperti komputer, kamera video, MP3 player dan lain-lain



Optical Disk

CD	<i>Compact Disk</i> . Suatu disk yang tidak dapat dihapus yang menyimpan informasi audio yang telah didigitasi. Sistem standar menggunakan disk 12 cm yang dapat merekam lebih dari 60 menit waktu putar tanpa terhenti.
CDROM	<i>Compact Disk Read-Only Memory</i> . Disk yang tidak dapat dihapus untuk menyimpan data komputer. Sistem standar menggunakan disk 12 cm yang dapat menampung lebih dari 550 Mbyte
CD-R	<i>Compact Disk Recordables</i> . Merupakan CD untuk penggunaan khusus, biasanya untuk master CD dan photo CD. Lapisan reflektif terbuat dari emas sehingga berwarna kuning. Kapasitas sama dengan CD lainnya
CD-RW	<i>Digital Video Rewritables</i> . Merupakan generasi CD yang dapat ditulis berulang kali namun belum populer saat ini karena masih relatif mahal
DVD	<i>Digital Versatile Disk</i> . Salah satu jenis CD yang memiliki pit data lebih kecil, spiral data yang lebih rapat sehingga kapasitasnya sangat besar, bisa mencapai 4,7GB untuk sisi tunggal dan berlapis tunggal. Laser optis yang digunakan adalah laser merah yang berukuran lebih kecil dari CD biasa. Kualitas yang dihasilkan juga lebih baik dari CD model lain



CD ROM

- Baik CD Audio maupun CD ROM menggunakan teknologi yang sama.
- Perbedaan utamanya adalah CD ROM player lebih kasar dan memiliki perangkat error correcting untuk menjamin bahwa data di transfer dengan benar dari disk ke komputer.
- Disk terbuat dari resin, seperti polycarbonate, dan di lapiasi dengan permukaan yang sangat reflektif, biasanya alumunium.



- Informasi yang di rekam secara digital di terbitkan sebagai sekumpulan lobang-lubang mikroskopik pada permukaan yang reflektif.
- Hal ini di lakukan pertama-tama dengan menggunakan laser berintensitas tinggi yang di fokuskan dengan teliti untuk membuat master disk.
- Permukaan yang berlubang disk salinan di lindungi dari debu dan gesekan dengan lapisan bening.
- Informasi di lacak dengan laser berintensitas rendah yang di tempatkan di dalam optical disk player atau drive unit



- Laser menyinari lapisan pelindung yang bening sementara motor memutar disk
- Intensitas sinar laser yang direfleksikan akan berubah jika mengenai lubang-lubang tersebut.
- Perubahan ini akan dideteksi oleh fotosensor dan di konversikan menjadi signal digital.



Keuntungan CD ROM

- Kapasitasan informasi jauh lebih besar dibandingkan dengan disk magnetik
- Disk optik bersama-sama dengan datanya dapat di perbanyak dengan biaya murah
- Disk optik dapat dengan mudah di pindah-pindah



Kekurangan CD ROM

- CD ROM hanya dapat di baca saja
- CD ROM mempunyai akses yang lebih lama di bandingkan disk magnetik (lebih lama 1/2 detik)



WORM **(write once read many)**

- Pada WORM disiapkan sebuah disk sedemikian rupa sehingga disk tersebut dapat di tulisi sekali dengan menggunakan sinar laser berintensitas sedang.
- Dengan menggunakan kontroller disk yang harganya lebih mahal dibandingkan dengan CD ROM, pelanggan dapat menulis sekali dan juga membaca disk.



Pita Magnetik

- Menggunakan sistem disk pada pembacaan dan penulisan
- Medianya pita mylar lentur yang di lapisi dengan oksida magnet
- Pita dan drive pita merupakan analog terhadap sistem tape recorder yang sering di gunakan dirumah-rumah



Disk Optis yang dapat dihapus

Disk dapat berulang-ulang ditulisi

Keuntungannya :

- Berkapasitas besar (650 MB/disk)
- Portabilitas dapat di pindahkan dari drive-nya
- Reliabilitas dan tahan lama
- Disk dapat dibaca dan ditulisi berulang-ulang



DISK Disk Masa Depan

1. Format DVD
2. Format High Definition DVD
3. Format Blue Ray



Kesimpulan

- Kebutuhan akan memori utama saja tidak mencukupi maka diperlukan peralatan tambahan untuk menyimpan data yang lebih besar dan dapat dibawa kemana-mana.
- Disk adalah piringan bundar yang terbuat dari bahan tertentu (logam atau plastik) dengan permukaan dilapisi bahan yang dapat di magnetisasi.
- Dengan berkembangnya komputer pribadi maka diperlukan media untuk mendistribusikan software maupun pertukaran data. Solusinya ditemukannya disket atau floppy disk.



PERTEMUAN

5

INPUT OUTPUT



Sistem Komputer

Tiga komponen utama :

- CPU
- Memori (primer dan sekunder)
- Peralatan masukan/keluaran (I/O devices) seperti printer, monitor, keyboard, mouse, dan modem



Modul I/O

Piranti tidak langsung dihubungkan dengan bus sistem komputer , Mengapa ?

- 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



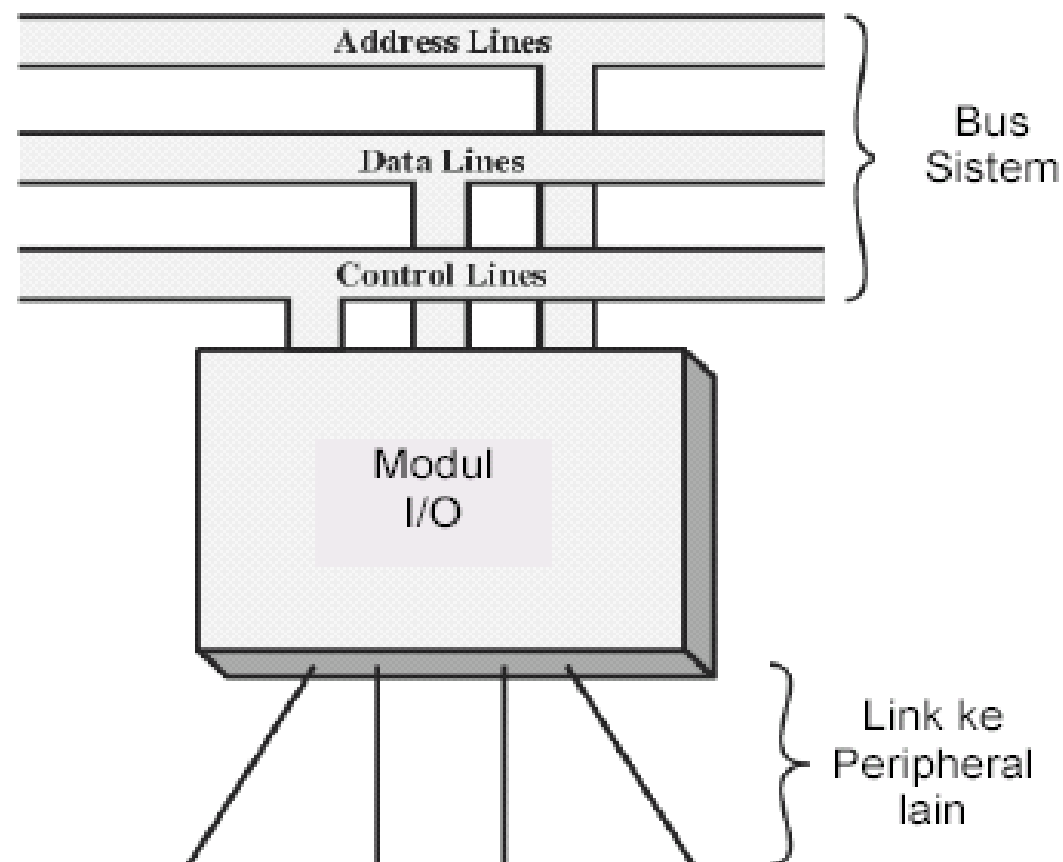
Sistem Masukan & Keluaran Komputer

Inti mempelajari sistem I/O suatu komputer ?

- Menjembatani CPU dan memori dengan dunia luar merupakan hal yang terpenting untuk kita ketahui
- Mengetahui fungsi dan struktur modul I/O



Model Generik Suatu Modul I/O





Modul I/O

- Modul I/O adalah suatu komponen dalam sistem komputer
 - Bertanggung jawab atas pengontrolan sebuah perangkat luar atau lebih
 - Bertanggung jawab pula dalam pertukaran data antara perangkat luar tersebut dengan memori utama ataupun dengan register – register CPU.
- Antarmuka internal dengan komputer (CPU dan memori utama)
- Antarmuka dengan perangkat eksternalnya untuk menjalankan fungsi – fungsi pengontrolan



Fungsi modul I/O

1. Control dan timing
2. Komunikasi CPU
3. Komunikasi perangkat eksternal
4. Data Buffering (pem-buffer-an data)
5. Deteksi error (kesalahan)



Kontrol dan Pewaktuan

- Fungsi kontrol dan pewaktuan (control & timing) merupakan hal yang penting untuk mensinkronkan kerja masing – masing komponen penyusun komputer.
- Dalam sekali waktu CPU berkomunikasi dengan satu atau lebih perangkat dengan pola tidak menentu dan kecepatan transfer komunikasi data yang beragam, baik dengan perangkat internal seperti register – register, memori utama, memori sekunder, perangkat peripheral.
- Proses tersebut bisa berjalan apabila ada fungsi kontrol dan pewaktuan yang mengatur sistem secara keseluruhan
- Transfer data tidak akan lepas dari penggunaan sistem bus, maka interaksi CPU dan modul I/O akan melibatkan kontrol dan pewaktuan sebuah arbitrase bus atau lebih



Langkah-langkah pemindahan data dari peripheral ke CPU melalui sebuah modul I/O

- Permintaan dan pemeriksaan status perangkat dari CPU ke modul I/O.
- Modul I/O memberi jawaban atas permintaan CPU.
- Apabila perangkat eksternal telah siap untuk transfer data, maka CPU akan mengirimkan perintah ke modul I/O.
- Modul I/O akan menerima paket data dengan panjang tertentu dari peripheral.
- Selanjutnya data dikirim ke CPU setelah diadakan sinkronisasi panjang data dan kecepatan transfer oleh modul I/O sehingga paket – paket data dapat diterima CPU dengan baik



Proses fungsi komunikasi antara CPU dan modul I/O

- *Command Decoding*, yaitu modul I/O menerima perintah – perintah dari CPU yang dikirimkan sebagai sinyal bagi *bus* kontrol. Misalnya, sebuah modul I/O untuk disk dapat menerima perintah: Read sector, Scan record ID, Format disk.
- *Data*, pertukaran data antara CPU dan modul I/O melalui *bus* data.
- *Status Reporting*, yaitu pelaporan kondisi status modul I/O maupun perangkat peripheral, umumnya berupa status kondisi *Busy* atau *Ready*. Juga status bermacam – macam kondisi kesalahan (*error*).
- *Address Recognition*, bahwa peralatan atau komponen penyusun komputer dapat dihubungi atau dipanggil maka harus memiliki alamat yang unik, begitu pula pada perangkat peripheral, sehingga setiap modul I/O harus mengetahui alamat peripheral yang dikontrolnya



Buffering

- Tujuan utama adalah mendapatkan penyesuaian data sehubungan perbedaan laju transfer data dari perangkat peripheral dengan kecepatan pengolahan pada CPU.
- Laju transfer data dari perangkat peripheral lebih lambat dari kecepatan CPU maupun media penyimpan

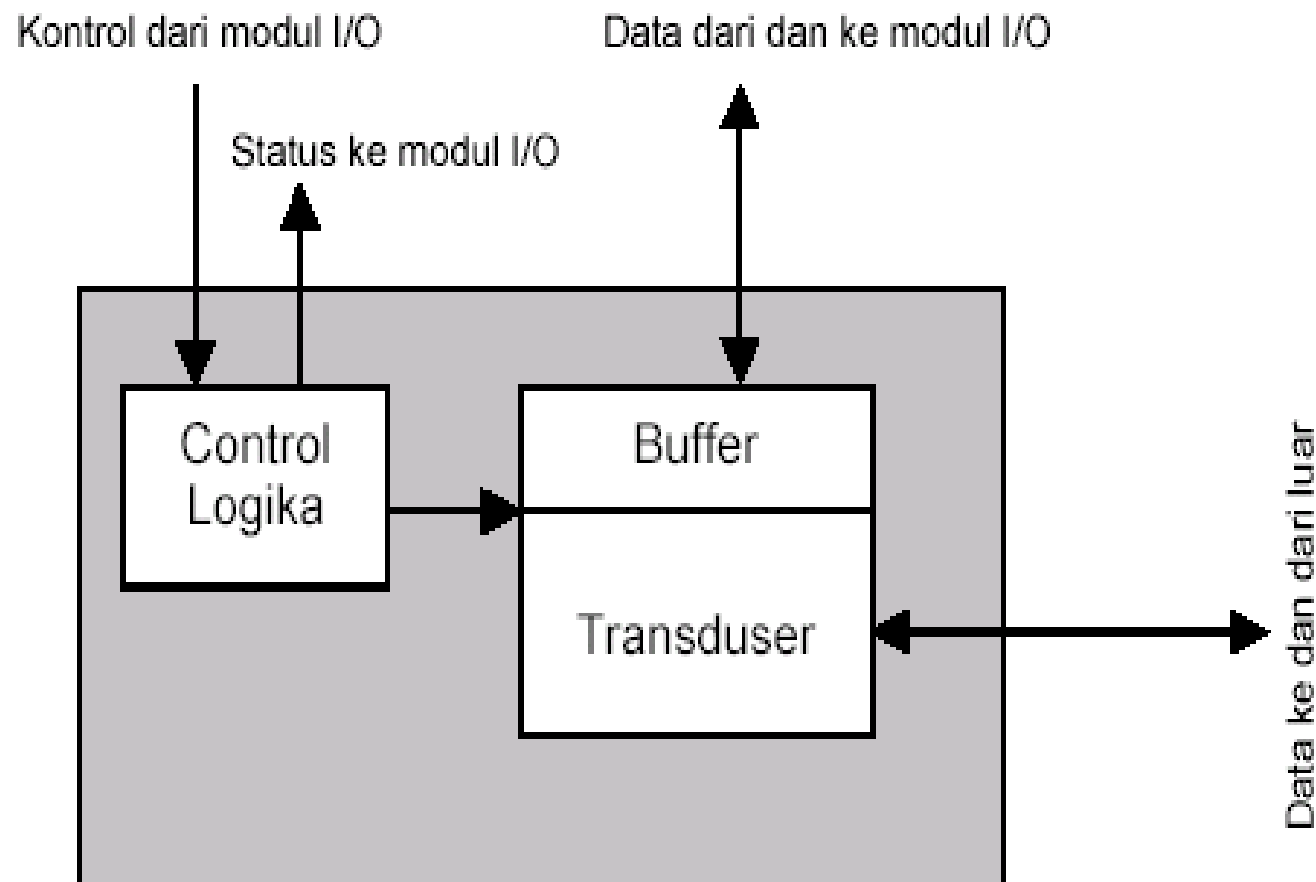


Deteksi Kesalahan

- Bila perangkat peripheral terdapat masalah sehingga proses tidak dapat dijalankan, maka modul I/O akan melaporkan kesalahan tersebut.
 - Misal informasi kesalahan pada peripheral printer seperti: kertas tergulung, tinta habis, kertas habis.
- Teknik yang umum untuk deteksi kesalahan adalah penggunaan bit paritas



Skema Perangkat Peripheral





Suatu perangkat eksternal terdiri dari :

Control Signal

Menentukan fungsi-fungsi yang di lakukan perangkat seperti:

- Mengirimkan data ke modul I/O
- Menerima data dari modul I/O
- Report status/membentuk fungsi kontrol tertentu ke perangkat



Signal Status

Menandai status perangkat misalnya ready / not ready

Control Logic

Berkaitan dengan perangkat yang mengontrol operasi perangkat dalam memberikan respon yang berasal dari modul I/O



Tranducer

Mengubah data dari energi listrik menjadi energi lain atau dari energi tertentu ke energi listrik

Buffer

Menampung sementara data yang di transfer diantara modul I/O dan dunia luar (ukuran buffer yang umum 8-16 bit)



Perangkat Eksternal

Ada 3 kategori perangkat eksternal

1. Human Readable

Contoh : Video Display Terminal (VDT) dan Printer

2. Machine Readable

Contoh : Sistem disk dan pita maghnetik, sistem robot

3. Communications



Keyboard dan Monitor

- Bagi input keyboard ketika sebuah tombol ditekan oleh user maka hal ini akan menghasilkan signal listrik yang di interpretasikan oleh tranducer pada keyboard dan di terjemahkan kedalam pola bit kode ASCII tertentu.
- Pola bit ini kemudian di transmisikan ke modul I/O pada komputer



- Di dalam komputer teks akan disimpan dalam kode ASCII yang sama
- pada output karakter-karakter kode ASCII akan di transmisikan dari modul I/O ke perangkat eksternal
- Transducer pada perangkat akan menginterpretasikan kode ini pada dan mengirimkan signal elektronik yang diperlukan ke perangkat output untuk menampilkan karakter tersebut atau untuk membentuk fungsi kontrol yang diminta



Disk Drive

- Pada fixed head disk transducer mampu melakukan konversi dari pola-pola magnetik pada permukaan disk yang bergerak menjadi bit-bit pada buffer perangkat
- Pada movable head disk juga harus dapat memindahkan lengan disk secara radial menjauhi dan mendekati permukaan disk



Fungsi modul I/O

Modul I/O merupakan suatu entity di dalam komputer yang bertanggung jawab atas pengontrol sebuah perangkat eksternal atau lebih dan untuk pertukaran data antara perangkat-perangkat tersebut dengan memori utama dan / atau register-register CPU

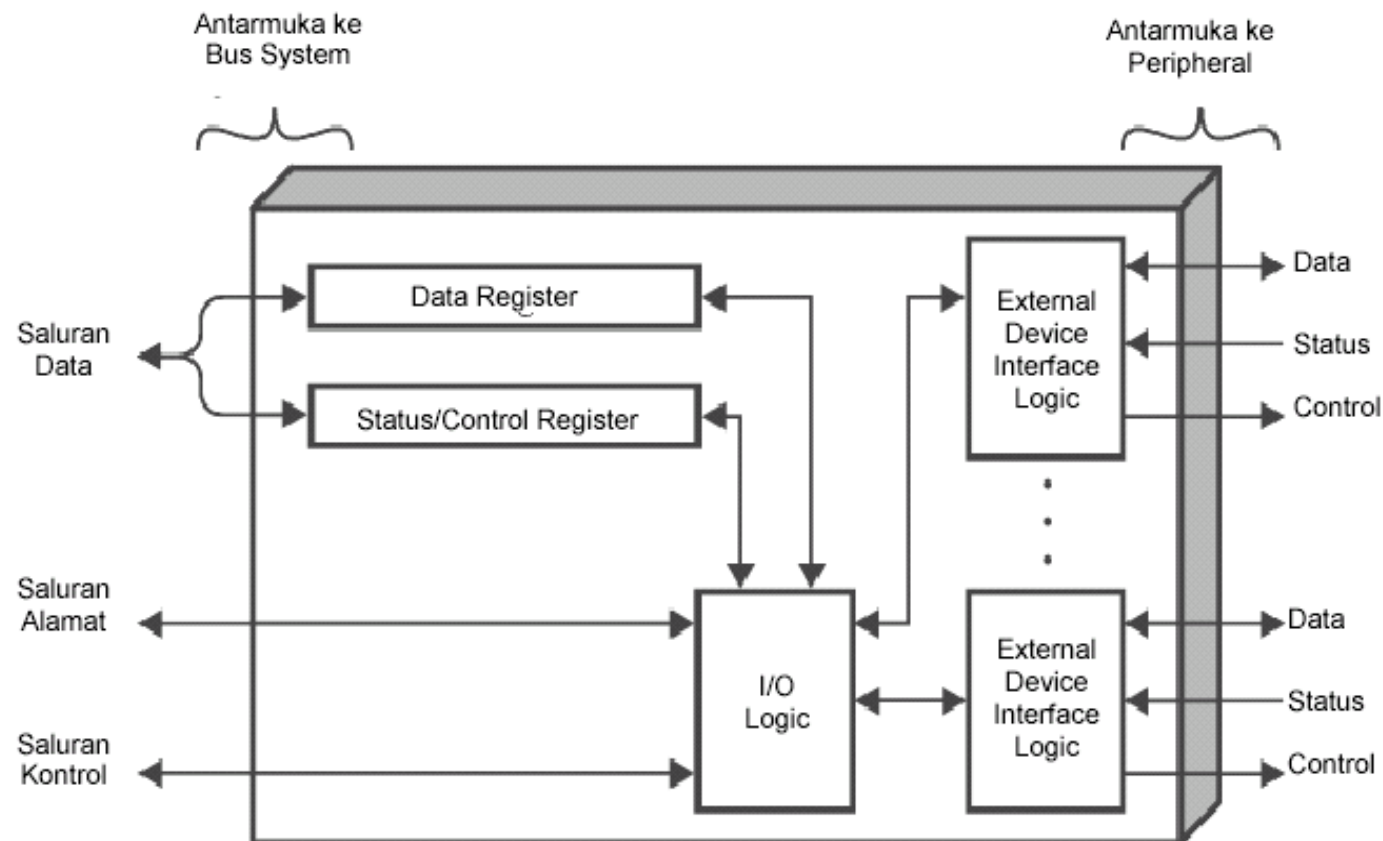


Struktur Modul I/O

- Modul I/O berfungsi untuk memungkinkan CPU dapat mengetahui perangkat yang jumlahnya banyak dengan cara yang sederhana.
- Berbagai macam modul I/O seiring perkembangan komputer.
 - Intel 8255A yang sering disebut PPI (Programmable Peripheral Interface).
- Bagaimanapun kompleksitas suatu modul I/O, terdapat kemiripan struktur.



Struktur Modul I/O



Blok diagram struktur modul I/O



Struktur Modul I/O

- Antarmuka modul I/O ke CPU melalui bus sistem komputer terdapat tiga saluran
 - Saluran data
 - Saluran alamat
 - Saluran kontrol.
- Bagian terpenting adalah blok logika I/O yang berhubungan dengan semua peralatan antarmuka peripheral, terdapat fungsi pengaturan dan switching pada blok ini



I/O Terprogram

- Terdapat 3 buah teknik yang di gunakan dalam operasi I/O
- Dengan menggunakan interrupt driven I/O, CPU mengeluarkan perintah I/O, dilanjutkan dengan mengeksekusi instruksi-instruksi lainnya, dan di interupsi oleh modul I/O apabila instruksi-instruksi tersebut telah selesai di laksanakan.



- Dengan menggunakan I/O terprogram dan I/O interrupt, maka CPU bertanggung jawab atas pengeluaran data dari memori utama untuk keperluan output dan penyimpanan data didalam memori utama untuk keperluan input
- Alternatifnya dengan menggunakan DMA



I/O Terprogram

- Data saling dipertukarkan antara CPU dan modul I/O.
- CPU mengeksekusi program yang memberikan operasi I/O kepada CPU secara langsung
 - Pemindahan data
 - Pengiriman perintah baca maupun tulis
 - Monitoring perangkat



I/O Terprogram

Kelemahan :

- 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



Klasifikasi perintah I/O

1. Perintah *control*.

Perintah ini digunakan untuk mengaktivasi perangkat peripheral dan memberitahukan tugas yang diperintahkan padanya.

2. Perintah *test*.

Perintah ini digunakan CPU untuk menguji berbagai kondisi status modul I/O dan peripheralnya. CPU perlu mengetahui perangkat peripheralnya dalam keadaan aktif dan siap digunakan, juga untuk mengetahui operasi –operasi I/O yang dijalankan serta mendeteksi kesalahannya.

3. Perintah *read*.

Perintah pada modul I/O untuk mengambil suatu paket data kemudian menaruh dalam buffer internal. Proses selanjutnya paket data dikirim melalui *bus* data setelah terjadi *sinkronisasi* data maupun kecepatan transfernya.

4. Perintah *write*.

Perintah ini kebalikan dari *read*. CPU memerintahkan modul I/O untuk mengambil data dari *bus* data untuk diberikan pada perangkat peripheral tujuan data tersebut.



Instruksi-instruksi I/O

- Instruksi-instruksi dapat dengan mudah dipetakan dkedalam perintah-perintah I/O dan sering sekali terdapat korespodensi satu-satu yang sederhana
- Ketika CPU, main memory dan I/O menggunakan bus umum bersama-sama, maka akan dimungkinkan penggunaan dua mode pengalamatan, yaitu :
 - Memory-mapped I/O
 - 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.
- Konsekuensinya adalah diperlukan saluran tunggal untuk pembacaan dan saluran tunggal untuk penulisan.

Keuntungan : efisien dalam pemrograman, namun memakan banyak ruang memori alamat

Kerugiannya : ruang memory alamat yang berharga akan habis terpakai



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.
- Port-port I/O hanya dapat di akses dengan perintah-perintah I/O khusus yang akan mengaktifasi saluran perintah I/O pada bus
- **Keuntungan** : sedikitnya instruksi I/O



Interrupt – Driven I/O

- Proses tidak membuang – buang waktu
- Prosesnya :
 - 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



Interrupt – Driven I/O

- Kendali perintah masih menjadi tanggung jawab CPU, baik pengambilan perintah dari memori maupun pelaksanaan isi perintah tersebut.
- Terdapat selangkah kemajuan dari teknik sebelumnya
 - CPU melakukan multitasking beberapa perintah sekaligus
 - Tidak ada waktu tunggu bagi CPU sehingga prosesnya cepat



Interrupt – Driven I/O

- Cara kerja teknik interupsi di sisi modul I/O
 - 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
 - Modul siap menerima perintah selanjutnya



Interrupt

- Pengolahan interupsi saat perangkat I/O telah menyelesaikan sebuah operasi I/O :
- Perangkat I/O akan mengirimkan sinyal interupsi ke CPU.
- CPU menyelesaikan operasi yang sedang dijalkannya kemudian merespon interupsi.
- CPU memeriksa interupsi tersebut, kalau valid maka CPU akan mengirimkan sinyal *acknowledgment* ke perangkat I/O untuk menghentikan interupsinya.
- CPU mempersiapkan pengontrolan transfer ke routine interupsi. Hal yang dilakukan adalah menyimpan informasi yang diperlukan untuk melanjutkan operasi yang tadi dijalankan sebelum adanya interupsi. Informasi yang diperlukan berupa:
 - Status prosesor, berisi register yang dipanggil PSW (*program status word*).
 - Lokasi intruksi berikutnya yang akan dieksekusi.
- Informasi tersebut kemudian disimpan dalam stack pengontrol sistem.



Interrupt

- Pengolahan interupsi saat perangkat I/O telah menyelesaikan sebuah operasi I/O :
 - ☐ CPU akan menyimpan PC (*program counter*) eksekusi sebelum interupsi ke stack pengontrol bersama informasi PSW.
 - ☐ Mempersiapkan PC untuk penanganan interupsi.
 - ☐ CPU memproses interupsi sampai selesai
 - ☐ Bila pengolahan interupsi selesai, CPU akan memanggil kembali informasi yang telah disimpan pada stack pengontrol untuk meneruskan operasi sebelum interupsi .



Kesimpulan

- Modul I/O merupakan peralatan antarmuka (interface) bagi sistem bus atau switch sentral dan mengontrol satu atau lebih perangkat peripheral.
- Modul I/O adalah suatu komponen dalam sistem komputer yang bertanggung jawab atas pengontrolan sebuah perangkat luar atau lebih dan bertanggung jawab pula dalam pertukaran data antara perangkat luar tersebut dengan memori utama ataupun dengan register –register CPU.



PERTEMUAN

6

INPUT OUTPUT



Sistem Komputer

Tiga komponen utama :

- ❑ CPU

- ❑ Memori (primer dan sekunder)

- ❑ Peralatan masukan/keluaran (I/O devices) seperti printer, monitor, keyboard, mouse, dan modem



Modul I/O

- 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



Interrupt Driven I/O

- Masalah yang di jumpai dalam I/O terprogram adalah bahwa CPU harus menunggu modul I/O yang di inginkan agar siap baik untuk menerima maupun untuk mengirimkan data dalam waktu yang relatif lama.
- Pada saat menunggu CPU harus berulang-ulang menanyakan status modul I/O.



- Akibatnya tingkat kinerja sistem secara keseluruhan menurun tajam
- Alternatifnya adalah CPU mengeluarkan perintah I/O ke modul dan kemudian mengerjakan perintah lainnya.
- Kemudian modul I/O akan menginterrupt CPU untuk meminta layanan ketika modul telah siap untuk saling bertukar data dengan CPU
- Setelah itu CPU akan mengeksekusi pengiriman data, seperti sebelumnya dan dilanjutkan dengan menyelesaikan proses sebelumnya.



Interrupt

Teknik yang digunakan CPU dalam menangani program interupsi

- ☐ *Multiple Interrupt Lines.*
- ☐ *Software poll.*
- ☐ *Daisy Chain.*
- ☐ *Arbitrasi bus*



Multiple Interrupt Lines.

- Teknik yang paling sederhana
- Menggunakan saluran interupsi berjumlah banyak
- Tidak praktis untuk menggunakan sejumlah saluran bus atau pin CPU ke seluruh saluran interupsi modul – modul I/O



Software Poll

- CPU mengetahui adanya sebuah interupsi, maka CPU akan menuju ke routine layanan interupsi yang tugasnya melakukan poll seluruh modul I/O untuk menentukan modul yang melakukan interupsi

Kerugian software poll

- memerlukan waktu yang lama karena harus mengidentifikasi seluruh modul untuk mengetahui modul I/O yang melakukan interupsi



Daisy Chain

- Teknik yang lebih efisien
- Menggunakan hardware poll
- Seluruh modul I/O tersambung dalam saluran interupsi CPU secara melingkar (chain)
- Apabila ada permintaan interupsi, maka CPU akan menjalankan sinyal acknowledge yang berjalan pada saluran interupsi sampai menjumpai modul I/O yang mengirimkan interupsi



Arbitrasi bus

- Modul I/O memperoleh kontrol bus sebelum modul ini menggunakan saluran permintaan interupsi
- Hanya akan terdapat sebuah modul I/O yang dapat melakukan interupsi



Pengontrol Interrupt Intel 8259A

- Intel mengeluarkan chips 8259A
- Sebagai interrupt arbiter pada mikroprosesor Intel 8086
- Manajemen interupsi modul - modul I/O
- Chips ini dapat diprogram untuk menentukan prioritas modul I/O yang lebih dulu ditangani CPU apabila ada permintaan interupsi yang bersamaan



Mode pada Interrupt Intel 8259A

- *Fully Nested*

Permintaan interupsi dengan prioritas mulai 0 (IR0) hingga 7 (IR7).

- *Rotating*

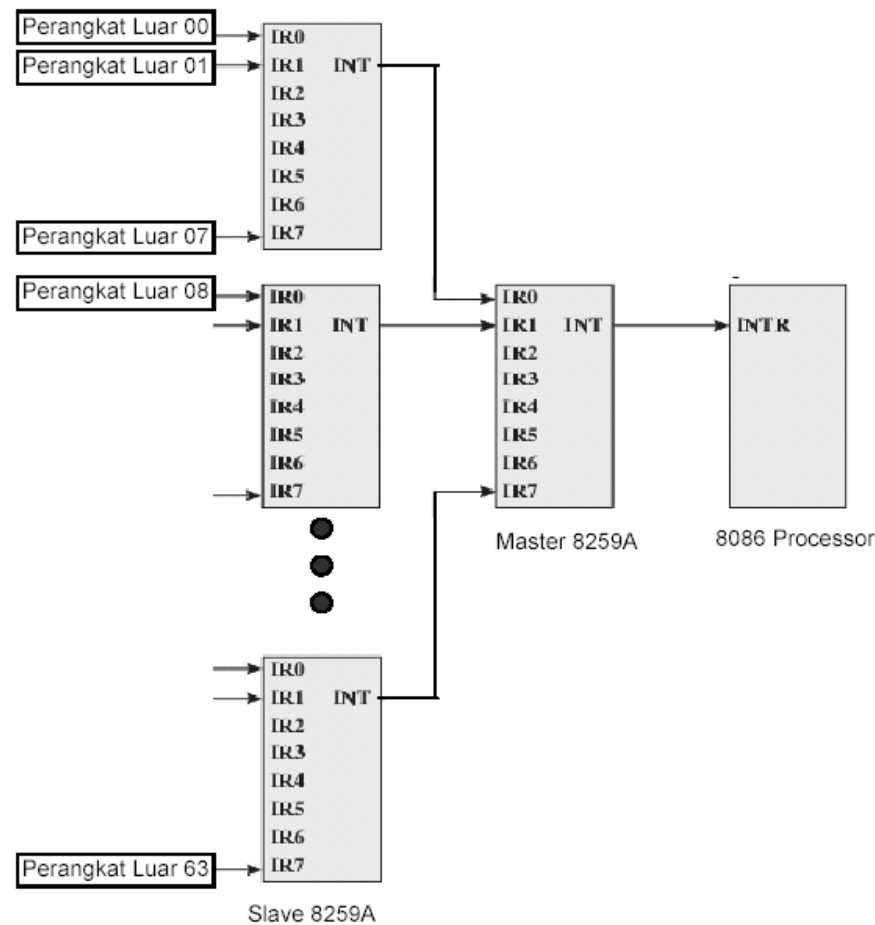
Bila sebuah modul telah dilayani interupsinya akan menempati prioritas terendah.

- *Special Mask*

Prioritas diprogram untuk modul I/O tertentu secara spesial.



Pemakaian pengontrol interupsi 8559A pada 8086





Programmable Peripheral Interface Intel 8255A

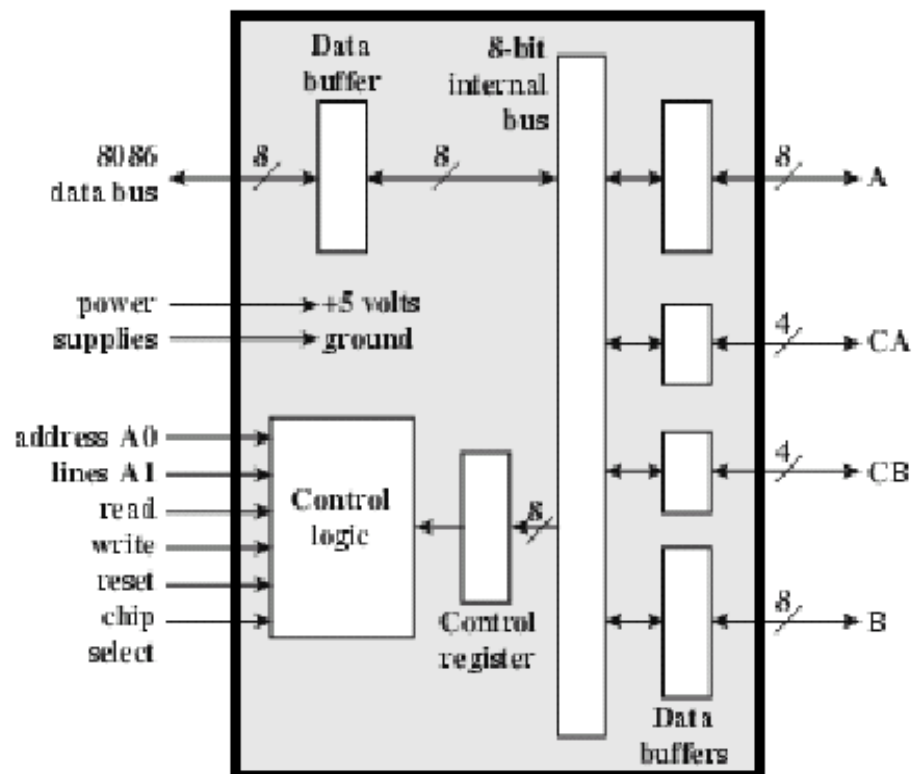
- Menggunakan I/O terprogram
- Interrupt driven I/O
- Dirancang untuk keperluan mikroprosesor 8086



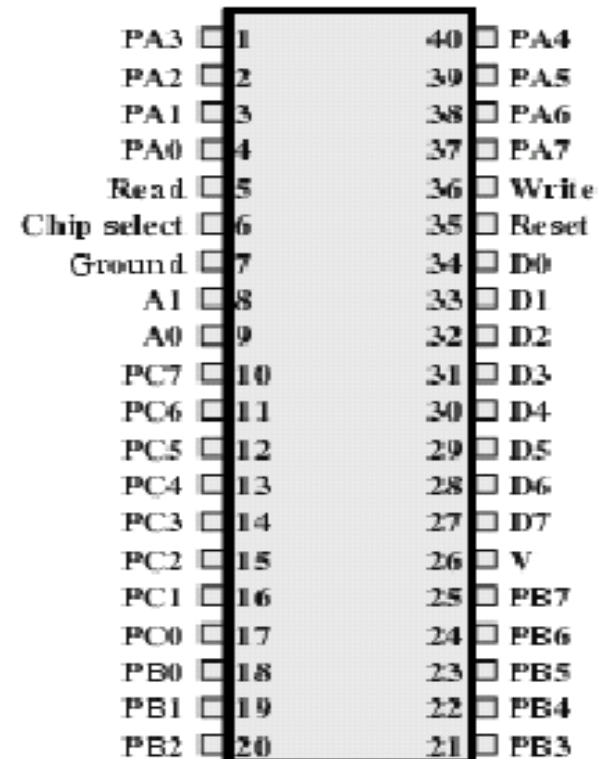
Modul I/O PPI 8255

Bagian kanan dari blok diagram Intel 8255A

- 24 saluran antarmuka luar
 - 8 bit port A
 - 8 bit port B
 - 4 bit port CA dan 4 bit port CB
- Saluran tersebut dapat diprogram dari mikroprosesor 8086 dengan menggunakan register kontrol untuk menentukan bermacam–macam mode operasi dan konfigurasinya.



(a) Blok diagram



(b) Layout pin



Modul I/O PPI 8255

Bagian kiri blok diagram merupakan interface internal dengan mikroprosesor 8086.

- 8 bus data dua arah (D0 – D7)
- bus alamat
- bus kontrol yang terdiri atas saluran CHIP SELECT, READ, WRITE, dan RESET



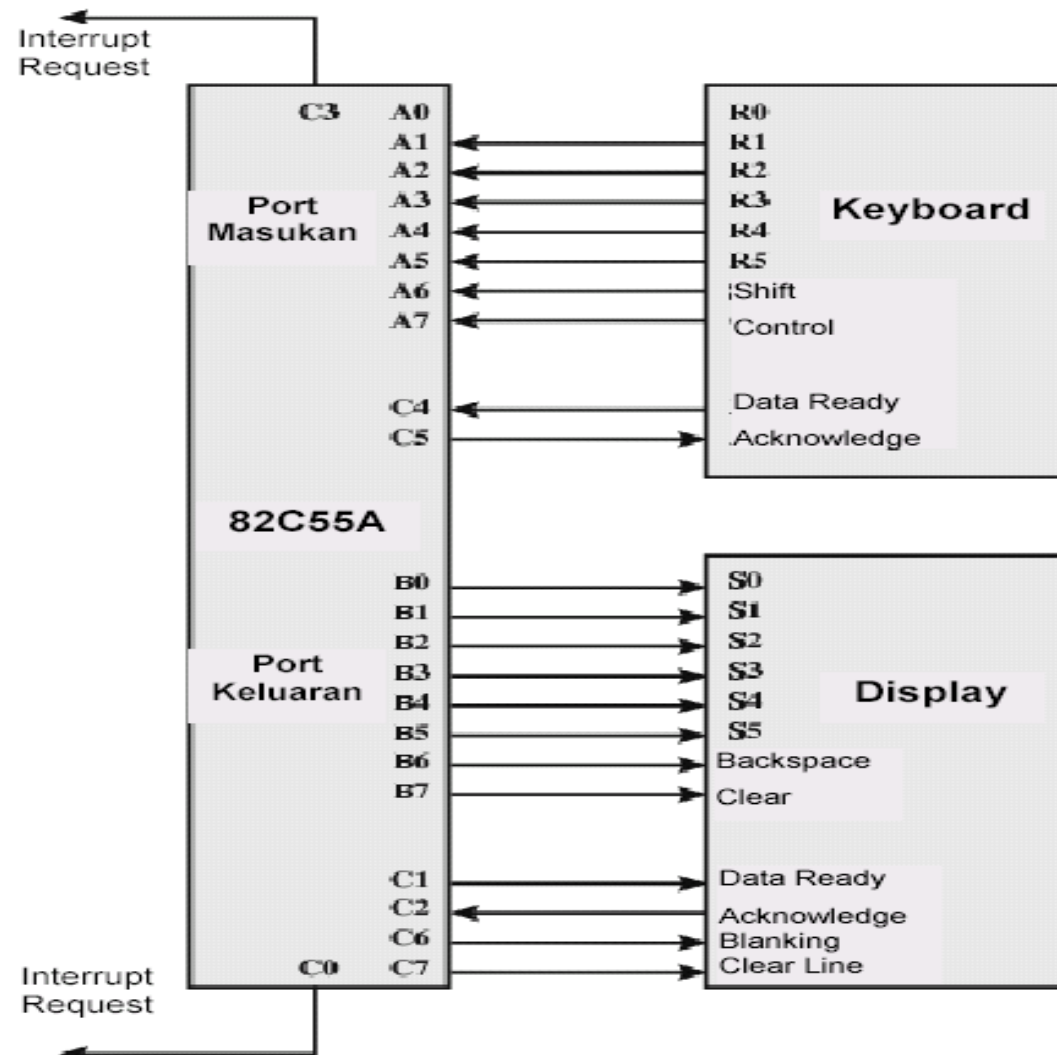
Modul I/O PPI 8255

- Pengaturan mode operasi pada register kontrol dilakukan oleh mikroprosesor
- Mode 0, ketiga port berfungsi sebagai tiga port I/O 8 bit
- Mode lain dapat port A dan port B sebagai port I/O 8 bit, sedangkan port C sebagai pengontrol saluran port A dan B



Interface keyboard dan display Intel 8255A

PPI Intel 8255A
dapat diprogram
untuk mengontrol
berbagai
peripheral
sederhana





Direct Memory Access (DMA)

Kelemahan I/O terprogram dan Interrupt-Driven I/O

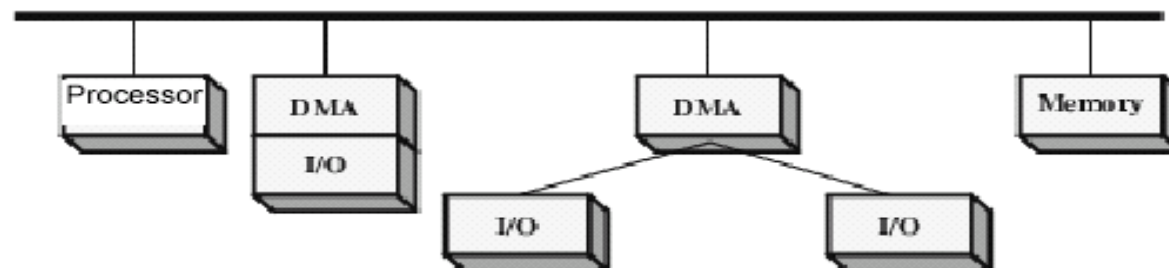
- Proses yang terjadi pada modul I/O masih melibatkan CPU secara langsung, berimplikasi pada :
 - ✓ Kelajuan transfer I/O yang tergantung kecepatan operasi CPU.
 - ✓ Kerja CPU terganggu karena adanya interupsi secara langsung



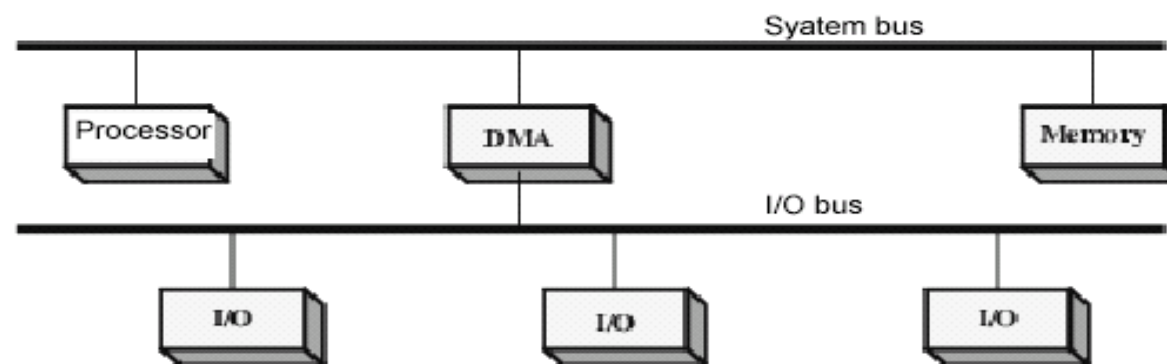
Konfigurasi Modul DMA



(a) Single bus



(b) Single bus, Integrated DMA-I/O



(c) I/O bus



Direct Memory Access (DMA)

- DMA meliputi modul-modul tambahan pada bus sistem
- Modul DMA mampu menirukan CPU, bahkan mengambil alih kontrol sistem dari CPU.



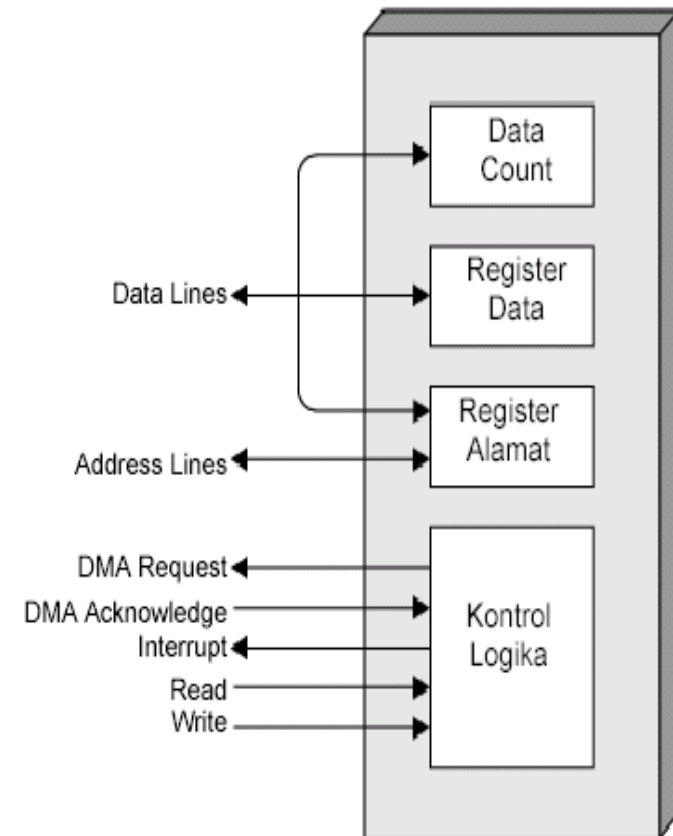
Prinsip Kerja DMA

1. Pada saat CPU ingin membaca atau menulis blok data, CPU mengeluarkan perintah ke modul DMA dengan mengirimkan informasi berikut :
 - Read atau write yang di minta
 - Alamat perangkat I/O yang di libatkan
 - Lokasi awal dalam memori untuk membaca atau menulis
 - jumlah word yang akan di baca atau di tulis
2. Kemudian CPU melanjutkan pekerjaan lainnya
3. CPU telah mendelegasikan kepada modul DMA, dan modul ini akan menjalankan tugasnya.



Prinsip kerja DMA

4. Modul DMA memindahkan seluruh blok data, word per-word secara langsung ke memori atau dari memori tanpa harus melalui CPU.
5. Ketika pemindahan ini selesai modul DMA akan mengirimkan signal interrupt ke CPU sehingga CPU hanya akan dilibatkan pada awal dan akhir saja
6. CPU dapat menjalankan proses lainnya tanpa banyak terganggu dengan interupsi



Blok diagram modul DMA



Direct Memory Access (DMA)

- Melaksanakan transfer data secara mandiri
 - DMA memerlukan pengambil alihan kontrol bus dari CPU
 - DMA akan menggunakan bus bila CPU tidak menggunakannya atau DMA memaksa CPU untuk menghentikan sementara penggunaan bus
 - Teknik cycle-stealing, modul DMA mengambil alih siklus bus
- Penghentian sementara penggunaan bus bukanlah bentuk interupsi, tetapi penghentian proses sesaat yang berimplikasi hanya pada kelambatan eksekusi CPU saja



Saluran I/O dan Prosesor

Evolusi fungsi I/O

1. CPU mengontrol peripheral secara langsung
2. Ditambahkannya sebuah pengontrol atau modul I/O. CPU menggunakan I/O terprogram tanpa menggunakan interrupt



3. CPU menggunakan interrupt
4. menggunakan DMA
5. modul I/O ditingkatkan kemampuannya menjadi sebuah prosesor yang memiliki tugasnya sendiri, yang menggunakan instruksi tertentu untuk I/O tertentu
6. Modul I/O memiliki memori lokalnya sendiri dan merupakan sebuah komputer yang memiliki tugasnya sendiri



Karakteristik Saluran I/O

- Saluran I/O merepresentasikan pengembangan konsep DMA.
- Saluran I/O memiliki kemampuan untuk mengeksekusi instruksi I/O yang memberikan kontrol sepenuhnya terhadap operasi-operasi I/O
- Instruksi instruksi itu dapat disimpan dalam memori utama untuk selanjutnya dapat di eksekusi oleh prosesor yang terdapat dalam saluran I/O itu sendiri

Terdapat dua jenis saluran I/O yang umum dipakai :

1. selector channel
2. Multiplexor channel



Perangkat Eksternal Komputer

- ☐ Disebut juga peripheral
- ☐ Ada perangkat pengendalinya (Modul I/O)
- ☐ Memiliki nilai apabila bisa berinteraksi dengan dunia luar
- ☐ Tidak akan berfungsi apabila tidak dapat berinteraksi dengan dunia luar
 - Tidak ada keyboard.
 - Tidak ada monitor.
 - Keyboard dan monitor tergolong dalam perangkat eksternal komputer



Interface Eksternal

Jenis-jenis Interface

Karakteristik utama interface adalah serial dan paralel

☐ Paralel

Terdapat sejumlah saluran yang terhubung kemodul I/O dan peripheral dan sejumlah bit di pindahkan secara simultan melalui bus data

☐ Serial

- Hanya terdapat saluran yang digunakan untuk mentransmisikan data dan bit bit harus di transmisikan satu persatu
- Paralel umumnya digunakan untuk peripheral ber kecepatan tinggi seperti pita dan disk
- Serial umum digunakan untuk printer dan terminal



Konfigurasi Point to point dan multi point

Koneksi diantara modul I/O di dalam sebuah sistem komputer dan perangkat eksternal dapat berbentuk point to point atau multi point

- Point to point

memiliki saluran dedicated antara modul I/O dan perangkat eksternal

Contoh : keyboard, printer dan modem eksternal

- Multi point : di gunakan untuk mendukung perangkat penyimpanan berukuran besar (disk dan pita) serta multi media (CD ROM, Video, Audio).



Small Computer System Interface (SCSI)

- Sebuah contoh interface yang baik untuk perangkat peripheral eksternal
- Dipopulerkan oleh Macintosh tahun 1984
- SCSI merupakan interface standar untuk drive CDROM, peralatan audio dan perangkat penyimpanan eksternal berukuran besar



Klasifikasi Perangkat Eksternal

- ❖ *Human Readable*, yaitu perangkat yang berhubungan dengan manusia sebagai pengguna komputer. Contoh: monitor, keyboard, mouse, printer, joystick, disk drive.
- ❖ *Machine readable*, yaitu perangkat yang berhubungan dengan peralatan. Biasanya berupa modul sensor dan transduser untuk monitoring dan kontrol suatu peralatan atau sistem.
- ❖ Communication, yaitu perangkat yang berhubungan dengan komunikasi jarak jauh. Contoh: NIC dan modem



Klasifikasi berdasar arah data

- Perangkat output
 - Perangkat input
 - Kombinasi output-input.
-
- Contoh perangkat output : monitor, proyektor dan printer.
 - Contoh perangkat input : keyboard, mouse, joystick, scanner, mark reader, bar code reader.



Kesimpulan

1. PPI 8255 merupakan salah satu modul I/O yang dirancang untuk keperluan I/O mikroprosessor 8086
2. Terdapat tiga buah teknik dalam operasi I/O, yaitu: I/O terprogram, interrupt – driven I/O, dan DMA (Direct Memory Access).
3. Perangkat eksternal atau lebih umum disebut peripheral tersambung dalam sistem CPU melalui perangkat pengendalinya, yaitu modul I/O.
4. Perangkat eksternal diklasifikasikan Human Readable, Machine readable, Communication



Selesai



PERTEMUAN

9

CENTRAL PROCESSING UNIT



ALU

(Arithmetic and Logic Unit)

- ALU merupakan bagian komputer yang berfungsi membentuk operasi-operasi aritmatika dan logik terhadap data
- Semua elemen lain sistem komputer (control unit, register, memori, I/O) berfungsi terutama untuk membawa data ke ALU untuk selanjutnya di proses dan kemudian mengambil kembali hasilnya.



Representasi Integer

$$- 1101.0101 = -11.3125$$



Representasi Nilai Tanda

- Bentuk yang paling sederhana representasi yang memakai bit tanda adalah representasi nilai tanda.
- Misal :
 $+18 = 00010010$
 $-18 = 10010010$
(sign magnitude/nilai tanda)
- Terdapat kekurangan pada cara diatas



Komplement-2

- $+7 = 0111$ $+18 = 00010010$
- $-7 = 1001$ $-18 = 11101101$
- Dapat di simpulkan bahwa hasil akan berbeda dengan nilai tanda



Representasi fixed point

Semua representasi di atas dapat pula disebut dengan fixed point, karena radix pointnya (binary pointnya) tetap dan di asumsikan akan berada di sebelah kanan.



6. Aritmatika Integer

A. Negasi

Untuk membuat negasi gunakan komplement dua (dianjurkan)

Penjumlahan negasi :

$$+7 = 0111$$

$$-7 = 1001$$

maka bila ada soal $(-7) + (+5) = 1001$

$$\begin{array}{r} 0101 \\ \hline \end{array}$$

$$1110$$



Hasil = 1110 adalah bilangan negatif maka positifnya adalah = komplement 2-kan bilangan tersebut : 0010 = +2 maka bilangan 1110 adalah negatif dari 2 atau (-2)

Aturan overflow = Bila dua buah bilangan di tambahkan, dan keduanya positif atau keduanya negatif maka over flow akan terjadi jika dan hanya jika hasilnya memiliki tanda yang berlawanan.



B. Pengalian :

$$\begin{array}{r} 1011 \\ \times 1101 \\ \hline 1011 \\ 0000 \\ 1011 \\ 1011 \\ \hline 10001111 \end{array}$$



- Perkalian dengan bilangan negatif juga akan sama cuma negatif tersebut harus dihasilkan dari komplement 2
- Karena hasil kali $(-)$ dengan $(+)$ = $(-)$ maka hasil kali tersebut komplement duakan untuk mengetahui hasilnya.



C. Pembagian

Keterangan :

1011 = divisor

10010011 = dividend

Hasil = quotient

Sisa = remainders

$$\begin{array}{r} 1011 \overline{) 10010011} \\ \underline{1011} \\ 1110 \\ \underline{1011} \\ 1111 \\ \underline{1011} \\ 100 \end{array}$$



Representasi Floating Point

Misal :

$$976.000.000.000 = 9,76 \times 10^{11}$$

MENJADI

$$0,0000000000976 = 9,76 \times 10^{-10}$$



Aritmetika Floating Point

Penambahan dan pengurangan

- a. periksa bilangan-bilangan nol
- b. ratakan significand
- c. tambahkan atau kurangkan significand
- d. normalisasi hasilnya



contoh :

$$\begin{array}{rcl} 123 \times 100 & \longrightarrow & 123 \quad \times 100 \\ 456 \times 10^{-2} + & \longrightarrow & \frac{4,56 \times 100 +}{127,56 \times 100} \end{array}$$



Perkalian dan Pembagian

- a. Kalikan atau bagi significand
 - b. tambahkan atau kurangkan eksponensial
- contoh :

$$123 \times 10^2$$

$$3 \times 10^{-4}$$

$$369 \times 10^{-2}$$



Operasi Mikro

- Fungsi dari sebuah komputer adalah untuk eksekusi program
- Setiap siklus yang lebih kecil akan terdiri dari sejumlah langkah yang masing-masing langkah tersebut terdiri dari register-register CPU. Dapat di sebut langkah-langkah tersebut sebagai operasi mikro.
- Operasi mikro adalah operasi fungsional atau atomik suatu CPU.



SIKLUS PENGAMBILAN

- MAR dihubungkan dengan alamat bus sistem. MAR menspesifikasikan alamat di dalam memori untuk operasi read dan write.
- MBR dihubungkan dengan saluran data bus sistem. MBR berisi nilai yang akan disimpan di memori atau nilai terakhir yang di baca dari memori



- PC
Menampung alamat instruksi berikutnya yang akan di ambil.
- IR
Menampung instruksi terakhir yang diambil.



Siklus Pengambilan :

T1 : MAR	—————→	(PC)
T2 : MBR	—————→	Memori
PC	—————→	(PC) + 1
T3 : IR	—————→	(MBR)



Siklus Tak Langsung

T1 : MAR → (IR(Alamat))
T2 : MBR → Memori
T3 : IR (Alamat) → (MBR(Alamat))



Siklus Interrupt

T1 : MBR —————> (PC)
T2 : MAR —————> Alamat-simpan
 PC —————> Alamat –rutin
T3 : Memori —————> (MBR)



Siklus Eksekusi

Penambahan (ADD)

1. ADD R1,X = Menambahkan isi lokasi X ke register R1

T1 : MAR \longrightarrow (IR(alamat))

T2 : MBR \longrightarrow memori

T3 : R1 \longrightarrow (R1)+(MBR)



2. ISZ X = Isi lokasi X ditambahkan dengan 1. Apabila hasilnya sama dengan nol, maka instruksi berikutnya dilompati.

T1 : MAR \longrightarrow (IR(alamat))

T2 : MBR \longrightarrow Memori

T3 : MBR \longrightarrow (MBR) + 1

T4 : Memori \longrightarrow (MBR)

If (MBR=0) then (PC \longrightarrow PC+1)



- 3. BSA X : Alamat instruksi yang berada setelah instruksi BSA disimpan di lokasi X, dan eksekusi dilanjutkan pada lokasi X+1. Alamat yang di simpan akan di gunakan kemudian untuk keperluan return.

T1 : MAR \longrightarrow (IR(alamat))

MBR \longrightarrow (PC)

T2 : PC \longrightarrow (IR(Alamat))

Memori \longrightarrow (MBR)

T3 : PC \longrightarrow (PC) + 1



Siklus Instruksi

- Setiap fase siklus instruksi dapat di uraikan menjadi operasi mikro elementer.
- Ada empat buah kode siklus instruksi (ICC).
- ICC menandai status CPU dalam hal bagian tempat siklus tersebut berada.



Kode ICC :

- 00 : fetch
- 01 : Indirect
- 10 : execute
- 11 : interrupt



KONTROL CPU

Karakterisasi Unit Kontrol :

1. Menentukan elemen dasar CPU
2. Menjelaskan operasi mikro yang akan dilakukan CPU
3. Menentukan fungsi-fungsi yang harus dilakukan unit kontrol agar menyebabkan pembentukan operasi mikro



Elemen Dasar Fungsional CPU

1. ALU
2. Register-register
3. Lintasan data internal
4. Lintasan data eksternal
5. Unit Kontrol



Unit Kontrol melakukan dua tugas dasar :

- Pengurutan
- Eksekusi



Sinyal Kontrol

Input sinyal kontrol :

- Clock
- register Instruksi
- sinyal kontrol dari bus kontrol
- flag



Output sinyal kontrol :

- Sinyal kontrol di dalam CPU
- Sinyal kontrol bagi bus kontrol



selesai



PERTEMUAN

10

SET INSTRUKSI MODE DAN FORMAT PENGALAMATAN



Karakteristik Instruksi Mesin

- Instruksi mesin (machine instruction) yang dieksekusi membentuk suatu operasi dan berbagai macam fungsi CPU.
- Kumpulan fungsi yang dapat dieksekusi CPU disebut set instruksi (instruction set) CPU.
- Mempelajari karakteristik instruksi mesin, meliputi
 - Elemen – elemen intruksi mesin
 - Representasi instruksinya
 - Jenis – jenis instruksi
 - Penggunaan alamat
 - Rancangan set instruksi



Elemen Instruksi Mesin

Untuk dapat dieksekusi suatu instruksi harus berisi elemen informasi yang diperlukan CPU secara lengkap dan jelas , Apa saja elemennya ?

1. Operation code (Op code)
Menspesifikasi operasi yang akan dilakukan. Kode operasi berbentuk kode biner
2. Source Operand reference
Operasi dapat berasal dari lebih satu sumber. Operand adalah input operasi
3. Result Operand reference
Merupakan hasil atau keluaran operasi
4. Next Instruction Reference
Elemen ini menginformasikan CPU posisi instruksi berikutnya yang harus diambil dan dieksekusi



Operand dari Operasi

Melihat dari sumbernya, operand suatu operasi dapat berada di salah satu dari ketiga daerah berikut ini :

- Memori utama atau memori virtual
- Register CPU
- Perangkat I/O



Representasi Instruksi

- Instruksi komputer direpresentasikan oleh sekumpulan bit. Instruksi dibagi menjadi beberapa field.
- Field – field ini diisi oleh elemen – elemen instruksi yang membawa informasi bagi operasi CPU.
- Layout instruksi dikenal dengan format instruksi



Format Instruksi

Opcode	Alamat
--------	--------

- Kode operasi (opcode) direpresentasikan dengan singkatan – singkatan, yang disebut mnemonic.
- Mnemonic mengindikasikan suatu operasi bagi CPU.
- Contoh mnemonic adalah :
 - ADD = penambahan
 - SUB = subtract (pengurangan)
 - LOAD = muatkan data ke memori



- Contoh representasi operand secara simbolik :
 - ADD X, Y artinya : tambahkan nilai yang berada pada lokasi Y ke isi register X, dan simpan hasilnya di register X.
- Programmer dapat menuliskan program bahasa mesin dalam bentuk simbolik.
- Setiap opcode simbolik memiliki representasi biner yang tetap dan programmer dapat menetapkan lokasi masing – masing operand



Jenis – Jenis Instruksi

Contoh suatu ekspresi bilangan :

$X = X + Y ;$

X dan Y berkorespondensi dengan lokasi 513 dan 514.

Pernyataan dalam bahasa tingkat tinggi tersebut mengintruksikan komputer untuk melakukan langkah berikut ini :

- Muatkan sebuah register dengan isi lokasi memori 513.
- Tambahkan isi lokasi memori 514 ke register.
- Simpan isi register ke lokasi memori 513



Korelasi

- Terlihat hubungan antara ekspresi bahasa tingkat tinggi dengan bahasa mesin.
- Dalam bahasa tingkat tinggi, operasi dinyatakan dalam bentuk aljabar singkat menggunakan variabel.
- Dalam bahasa mesin hal tersebut diekspresikan dalam operasi perpindahan antar register



- Dapat ditarik kesimpulan bahwa instruksi – instruksi mesin harus mampu mengolah data sebagai implementasi keinginan – keinginan kita.
- Terdapat kumpulan unik set instruksi, yang dapat digolongkan dalam jenis–jenisnya, yaitu
 - Pengolahan data (data processing),
meliputi operasi – operasi aritmetika dan logika. Operasi aritmetika memiliki kemampuan komputasi untuk pengolahan data numerik. Sedangkan instruksi logika beroperasi terhadap bit – bit word sebagai bit, bukannya sebagai bilangan, sehingga instruksi ini memiliki kemampuan untuk pengolahan data lain
 - Perpindahan data (data movement),
berisi instruksi perpindahan data antar register maupun modul I/O. Untuk dapat diolah oleh CPU maka diperlukan instruksi -
instruksi yang bertugas memindahkan data operand yang diperlukan



- Penyimpanan data (data storage),
berisi instuksi – instruksi penyimpanan ke memori. Instruksi penyimpanan sangat penting dalam operasi komputasi, karena data tersebut akan digunakan untuk operasi berikutnya, minimal untuk ditampilkan pada layar harus diadakan penyimpanan walaupun sementara
- Kontrol aliran program (program flow control),
berisi instruksi pengontrolan operasi dan pencabangan. Instruksi ini berguna untuk pengontrolan status dan mengoperasikan pencabangan ke set instruksi lain



Jumlah Alamat

- Jumlah register atau alamat yang digunakan dalam operasi CPU tergantung format operasi masing – masing CPU.
- Ada format operasi yang menggunakan 3, 2, 1 dan 0 register.
- Umumnya yang digunakan adalah 2 register dalam suatu operasi. Desain CPU saat ini telah menggunakan 3 alamat dalam suatu operasi, terutama dalam MIPS (million instruction per secon).



- Alamat per instruksi yang lebih sedikit akan membuat instruksi lebih sederhana dan pendek, tetapi lebih sulit mengimplementasikan fungsi-fungsi yang kita inginkan.
- Karena instruksi CPU sederhana maka rancangan CPU juga lebih sederhana.
- Jumlah bit dan referensi per instruksi lebih sedikit sehingga fetch dan eksekusi lebih cepat.
- Jumlah instruksi per program biasanya jauh lebih banyak
- Pada jumlah alamat per instruksi banyak, jumlah bit dan referensi instruksi lebih banyak sehingga waktu eksekusi lebih lama.
- Diperlukan register CPU yang banyak, namun operasi antar register lebih cepat.



- Lebih mudah mengimplementasikan fungsi – fungsi yang kita inginkan.
- Jumlah instruksi per program jauh lebih sedikit.
- Untuk lebih jelas perhatikan contoh instruksi – instruksi dengan jumlah register berbeda untuk menyelesaikan persoalan yang sama
- Contoh penggunaan set instruksi dengan alamat 1, 2, dan 3 untuk menyelesaikan operasi hitungan

$$Y = (A - B) \div (C + D * E)$$



Contoh instruksi 2 dan 3 alamat

Instruksi 3 alamat :

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

Instruksi 2 alamat :

Instruksi		Komentar
MOVE	Y, A	$Y = A$
SUB	Y, B	$Y = Y - B$
MOVE	T, D	$T = D$
MPY	T, E	$T = T \times E$
ADD	T, C	$T = T + C$
DIV	Y, T	$Y = Y \div T$



Instruksi 1 alamat

Instruksi		Komentar
LOAD	D	AC = D
MPY	E	AC = AC * E
ADD	C	AC = AC + C
STOR	Y	Y = AC
LOAD	A	AC = A
SUB	B	AC = AC - B
DIV	Y	AC = AC ÷ Y
STOR	Y	Y = AC

Instruksi	Keterangan	isi stack
PUSH	B	B
PUSH	A	B, A
SUB	A-B	(A-B)
PUSH	E	(A-B), E
PUSH	D	(A-B), E, D
MUL	D * E	(A-B), (D * E)
PUSH	C	(A-B), (D * E), C
ADD	C + (D * E)	(A-B), (C + D * E)
DIV	(A-B) / (C + (D * E))	(A-B) / (C + (D * E))



Spesifikasi instruksi 3 alamat :

- Simbolik : $a = b + c$.
- Format alamat : hasil, operand 1, operand 2
- Digunakan dalam arsitektur MIPS.
- Memerlukan word panjang dalam suatu instruksi.



Spesifikasi instruksi 2 alamat :

- Simbolik : $a = a + b$.
- Satu alamat diisi operand terlebih dahulu kemudian digunakan untuk menyimpan hasilnya.
- Tidak memerlukan instruksi yang panjang.
- Jumlah instruksi per program akan lebih banyak dari pada 3 alamat.
- Diperlukan penyimpanan sementara untuk menyimpan hasil.



Spesifikasi instruksi 1 alamat :

- Memerlukan alamat implisit untuk operasi.
- Menggunakan register akumulator (AC) dan digunakan pada mesin lama.

Spesifikasi instruksi 0 alamat :

- Seluruh alamat yang digunakan implisit.
- Digunakan pada organisasi memori, terutama operasi stack



Rancangan Set Instruksi

- Aspek paling menarik dalam arsitektur komputer adalah perancangan set instruksi, karena rancangan ini berpengaruh banyak pada aspek lainnya.
- Set instruksi menentukan banyak fungsi yang harus dilakukan CPU.
- Set instruksi merupakan alat bagi para pemrogram untuk mengontrol kerja CPU.
- Pertimbangan : Kebutuhan pemrogram menjadi bahan pertimbangan dalam merancang set instruksi



Masalah rancangan yang fundamental meliputi :

- Operation repertoire :
 - Berapa banyak dan operasi – operasi apa yang harus tersedia
 - Sekompleks apakah operasi itu seharusnya
- Data types :
 - Jenis data
 - Format data
- Instruction format
 - Panjang instruksi,
 - Jumlah alamat,
 - Ukuran field
- Registers
 - Jumlah register CPU yang dapat direferensikan oleh instruksi, dan fungsinya
- Addressing
 - mode untuk menspesifikasi alamat suatu operand



Tipe Operasi

- Dalam perancangan arsitektur komputer, jumlah kode operasi akan sangat berbeda untuk masing – masing komputer, tetapi terdapat kemiripan dalam jenis operasinya



Jenis Operasi Komputer

- Transfer data. – Konversi
- Aritmetika. - Input/Output
- Logika. - Kontrol sistem dan transfer kontrol



- Operasi set instruksi secara umum

Jenis	Nama Operasi	Keterangan
Pemindahan Data	Move	Memindahkan word atau blok dari sumber ke tujuan.
	Store	Memindahkan word dari prosesor ke memori.
	Load	Memindahkan word dari memori ke prosesor.
	Exchange	Menukar isi sumber dengan tujuan.
	Clear (reset)	Memindahkan word 0 ke tujuan.
	Set	Memindahkan word 1 ke tujuan.
	Push	Memindahkan word dari sumber ke bagian paling atas stack.
	Pop	Memindahkan word dari stack teratas ke tujuan.



- Operasi set instruksi secara umum

Logika	AND	Melakukan operasi logika tertentu terhadap bit.
	OR	
	NOT	
	Exclusive-OR	
	Test	Menguji kondisi tertentu.
	Compare	Membandingkan dua operand (secara logika maupun aritmetika).
	Set Variabel Kontrol	Instruksi untuk menyetel kontrol bagi keperluan proteksi, interrupt, kontrol timer.
	Shift	Melakukan penggeseran bit – bit operand.
	Rotate	Melakukan pemutaran bit – bit operand.



• Operasi set instruksi secara umum

Pemindahan Kontrol	Jump (cabang)	Perpindahan tidak bersyarat; memuatkan PC dengan alamat tertentu.	Pemindahan Kontrol	Execute	Mengambil operand dari lokasi tertentu dan mengeksekusinya sebagai instruksi.
	Jump bersyarat	Menguji persyaratan tertentu; melakukan aktivitas tergantung persyaratannya.		Skip	Menambah PC sehingga melompati instruksi berikutnya.
	Jump ke subrutin	Menempatkan informasi data kontrol program saat itu di lokasi yang ditentukan; melompat ke alamat tertentu.		Skip bersyarat	Melompat berdasarkan syarat tertentu.
	Return	Mengganti ini PC dan register lainnya yang berasal dari lokasi tertentu.		Halt	Menghentikan eksekusi program.
				Wait (hold)	Menghentikan eksekusi program dan menguji persyaratan.
				No operation	Tidak ada operasi yang dilakukan, namun eksekusi program dilanjutkan.



- Operasi set instruksi secara umum

Input/Output	Input (read)	Memindahkan data dari port I/O ke tujuan.
	Output (write)	Memindahkan data dari prosesor ke port atau modul I/O.
	Start I/O	Memindahkan instruksi ke prosesor I/O untuk memulai proses I/O.
	Test I/O	Memindahkan informasi status dari sistem I/O ke tujuan.



- Operasi set instruksi secara umum

Konversi	Translate	Menerjemahkan nilai – nilai dalam suatu bagian memori berdasarkan tabel korespondensi.
	Convert	Mengkonversi isi word ke bentuk lain.



Transfer Data

Instruksi tranfer data harus menetapkan :

- Lokasi operand sumber
- Lokasi operand tujuan
- Panjang data yang akan dipindahkan
- Mode pengalamatannya

Apabila sebuah atau kedua operand berada di dalam memori, maka CPU harus melakukan sebagian atau seluruh tindakan berikut :

1. Menghitung alamat memori, yang didasarkan pada mode alamatnya.
2. Apabila alamat mengacu pada virtual memori harus dicari alamat memori sebenarnya.
3. Menentukan apakah alamat berada dalam cache memori.
4. Bila di cache tidak ada, dikeluarkan perintah ke modul memori



Mode Pengalamatan

Mengatasi keterbatasan format instruksi

- Dapat mereferensi lokasi memori yang besar
- Mode pengalamatan yang mampu menangani keterbatasan tersebut
 - Masing – masing prosesor menggunakan mode pengalamatan yang berbeda – beda.
 - Memiliki pertimbangan dalam penggunaannya.
 - Ada beberapa teknik pengalamatan
 - a. Immediate Addressing
 - b. Direct Addressing
 - c. Indirect Addressing
 - d. Register Addressing
 - e. Register Indirect Addressing
 - f. Displacement Addressing
 - g. Stack Addressing

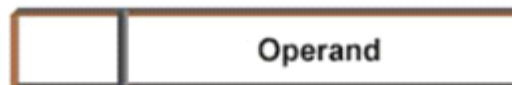


Immediate Addressing (1)

Bentuk pengalamatan ini yang paling sederhana ?

- Operand benar – benar ada dalam instruksi atau bagian dari instruksi = Operand sama dengan field alamat.
- Umumnya bilangan akan disimpan dalam bentuk komplement dua.
- Bit paling kiri sebagai bit tanda.
- Ketika operand dimuatkan ke dalam register data, bit tanda akan digeser ke kiri hingga maksimum word data
- Contoh :

ADD 5 ; tambahkan 5 pada akumulator





Immediate Addressing (+)&(-)

Keuntungan

- Mode ini adalah tidak adanya referensi memori selain dari instruksi yang diperlukan untuk memperoleh operand.
- Menghemat siklus instruksi sehingga proses keseluruhan akan cepat.

Kerugiannya

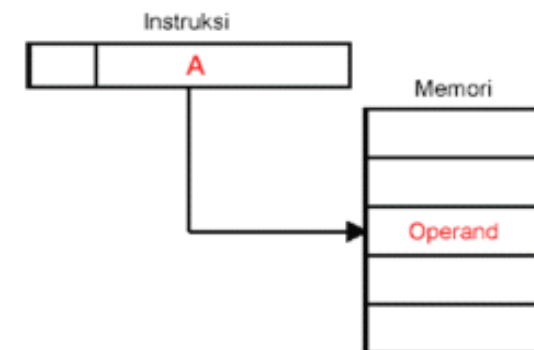
- Ukuran bilangan dibatasi oleh ukuran field alamat



Direct Addressing (2)

Pengalamatan langsung

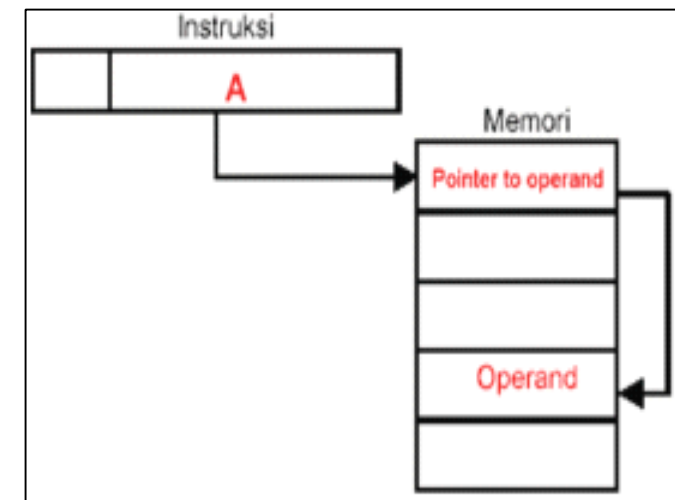
- Kelebihan :
 - Field alamat berisi efektif address sebuah operand.
- Teknik ini banyak digunakan pada komputer lama dan komputer kecil.
- Hanya memerlukan sebuah referensi memori dan tidak memerlukan kalkulasi khusus.
- Kelemahan :
 - Keterbatasan field alamat karena panjang field alamat biasanya lebih kecil dibandingkan panjang word
- Contoh :
 - ADD A ; tambahkan isi pada lokasi alamat A ke akumulator





Indirect Addressing (3)

- Mode pengalamatan tak langsung
 - Field alamat mengacu pada alamat word di dalam memori, yang pada gilirannya akan berisi alamat operand yang panjang
 - Contoh :
ADD (A) ; tambahkan isi memori yang ditunjuk oleh isi alamat A ke akumulator





Indirect Addressing (+)&(-)

Keuntungan

- Ruang bagi alamat menjadi besar sehingga semakin banyak alamat yang dapat referensi.

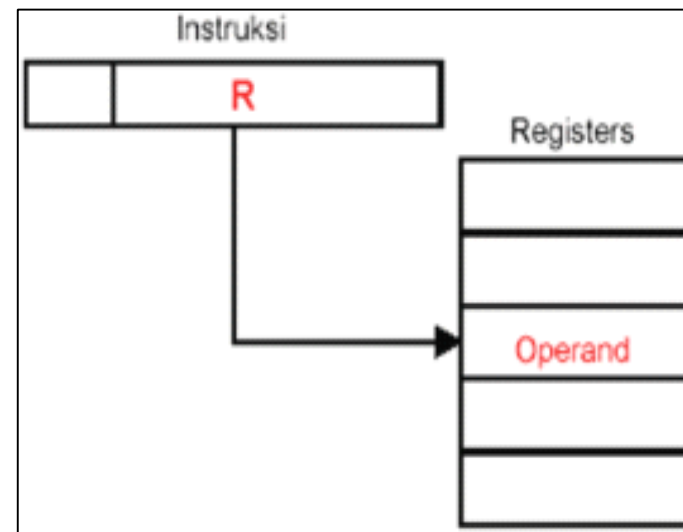
Kerugian

- Diperlukan referensi memori ganda dalam satu fetch sehingga memperlambat proses operasi



Register Addressing (4)

- Metode pengalamatan register mirip dengan mode pengalamatan langsung.
- Perbedaannya terletak pada field alamat yang mengacu pada register, bukan pada memori utama.
- Field yang mereferensi register memiliki panjang 3 atau 4 bit, sehingga dapat mereferensi 8 atau 16 register general purpose.





Register Addressing (+)&(-)

Keuntungan pengalamatan register

- Diperlukan field alamat berukuran kecil dalam instruksi dan tidak diperlukan referensi memori.
- Akses ke register lebih cepat daripada akses ke memori, sehingga proses eksekusi akan lebih cepat.

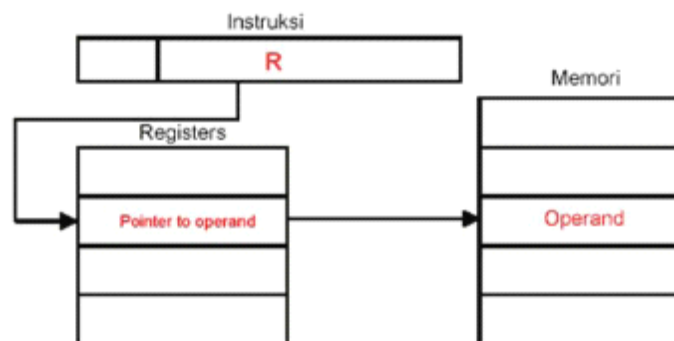
Kerugian

- Ruang alamat menjadi terbatas



Register Indirect Addressing (5)

- Metode pengalamatan register tidak langsung mirip dengan mode pengalamatan tidak langsung.
- Perbedaannya adalah field alamat mengacu pada alamat register.
- Letak operand berada pada memori yang ditunjuk oleh isi register.
- Keuntungan dan keterbatasan pengalamatan register tidak langsung pada dasarnya sama dengan pengalamatan tidak langsung.
 - Keterbatasan field alamat diatasi dengan pengaksesan memori yang tidak langsung sehingga alamat yang dapat direferensi makin banyak.
 - Dalam satu siklus pengambilan dan penyimpanan, mode pengalamatan register tidak langsung hanya menggunakan satu referensi memori
 - utama sehingga lebih cepat daripada mode pengalamatan tidak langsung



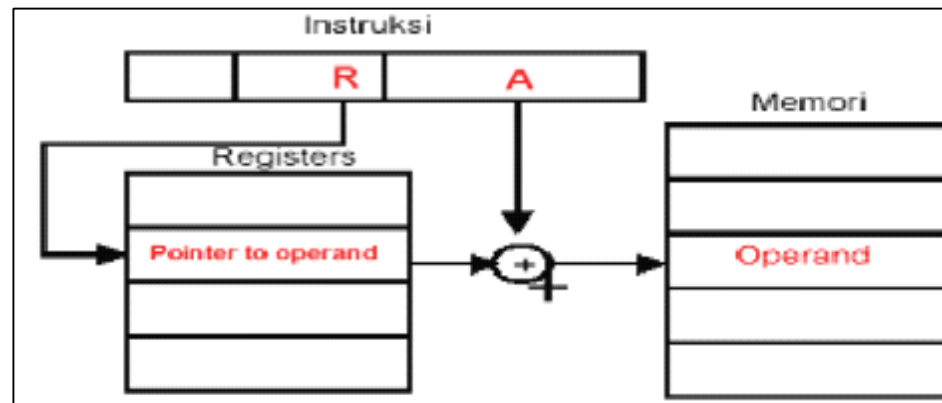


Displacement Addressing (6)

- Menggabungkan kemampuan pengalamatan langsung dan pengalamatan register tidak langsung.
- Mode ini mensyaratkan instruksi memiliki dua buah field alamat, sedikitnya sebuah field yang eksplisit.
 - Field eksplisit bernilai A dan field implisit mengarah pada register



Displacement Addressing (6)



- ☐ Operand berada pada alamat A ditambah isi register.
- ☐ Tiga model displacement
 - Relative Addressing
 - Base Register Addressing
 - Indexing



Displacement Addressing (6)

- ❑ Relative addressing, register yang direferensi secara implisit adalah program counter (PC).
 - Alamat efektif didapatkan dari alamat instruksi saat itu ditambahkan ke field alamat.
 - Memanfaatkan konsep lokalitas memori untuk menyediakan operand-operand berikutnya.
- ❑ Base register addressing, register yang direferensikan berisi sebuah alamat memori, dan field alamat berisi perpindahan dari alamat itu.
 - Referensi register dapat eksplisit maupun implisit.
 - Memanfaatkan konsep lokalitas memori.
- ❑ Indexing adalah field alamat mereferensi alamat memori utama, dan register yang direferensikan berisi pemindahan positif dari alamat tersebut.
 - Merupakan kebalikan dari model base register.
 - Field alamat dianggap sebagai alamat memori dalam indexing.
 - Manfaat penting dari indexing adalah untuk eksekusi program-program iteratif



Stack Addressing (7)

- ❑ Stack adalah array lokasi yang linier = pushdown list = last-in-first-out-queue.
- ❑ Stack merupakan blok lokasi yang terbalik.
 - Butir ditambahkan ke puncak stack sehingga setiap saat blok akan terisi secara parsial.
- ❑ Yang berkaitan dengan stack adalah pointer yang nilainya merupakan alamat bagian paling atas stack.
- ❑ Dua elemen teratas stack dapat berada di dalam register CPU, yang dalam hal ini stack pointer mereferensi ke elemen ketiga stack.
- ❑ Stack pointer tetap berada di dalam register.
- ❑ Dengan demikian, referensi – referensi ke lokasi stack di dalam memori pada dasarnya merupakan pengalamatan register tidak langsung



Perbandingan Mode pengalamatan

Mode	Algoritma	Keuntungan Utama	Kerugian utama
Immediate	Operand = A	Tidak ada referensi memori	Besaran operand terbatas
Direct	EA = A	Sederhana	Ruang alamat terbatas
Indirect	EA = (A)	Ruang alamat besar	Referensi memori berganda
Register	EA = R	Tidak ada referensi memori	Ruang alamat terbatas
Register Indirect	EA = (R)	Ruang alamat besar	Referensi memori ekstra
Displacement	EA = A + (R)	Fleksibilitas	Kompleksitas
Stack	EA = Puncak Stack	Tidak ada referensi	Aplikasi memori terbatas



Keterangan :

- A = isi suatu field alamat dalam instruksi
- EA = alamat aktual (efektif) sebuah lokasi yang berisi operasi yang di referensikan
- (X) = isi lokasi X



Mode Pengalamatan Pentium

- Pentium dilengkapi bermacam – macam mode pengalamatan untuk memudahkan bahasa – bahasa tingkat tinggi mengeksekusinya secara efisien (C/Fortran)



Mode pengalamatan pentium

Mode	Algoritma
Immediate	$\text{Operand} = A$
Register	$eA = R$
Displacement	$eA = (\text{SR}) + A$
Base	$eA = (\text{SR}) + (B)$
Base with displacement	$eA = (\text{SR}) + (B) + A$
Scaled index with displacement	$eA = (\text{SR}) + (B) + (I) + A$
Base with scaled index and displacement	$eA = (\text{SR}) + (I) \times S + (B) + A$
Relative	$eA = (\text{PC}) + A$

Keterangan :

- SR = register segment
- PC = program counter
- A = isi field alamat
- B = register basis
- I = register indeks
- S = faktor skala



Mode pengalamatan pentium

- Mode immediate
 - Operand berada di dalam instruksi.
 - Operand dapat berupa data byte, word maupun doubleword
- Mode operand register, operand adalah isi register.
 - Beberapa macam jenis register
 - register 8 bit (AH, BH, CH, DH, AL, BL, CL, DL)
 - register 16 bit (AX, BX, CX, DX, SI, DI, SP, BP)
 - register 32 bit (EAX, EBX, ECX, EDX, ESI, EDI, ESP, EBP)
 - register 64 bit yang dibentuk dari register 32 bit secara berpasangan.
 - register 8, 16 dan 32 merupakan register untuk penggunaan umum (general purpose register).
 - register 64 bit biasanya untuk operasi floating point.
 - register segmen (CS, DS, ES, SS, FS, GS)



- Mode displacement
 - ☐ Alamat efektif berisi bagian – bagian instruksi dengan displacement 8, 16, atau 32 bit.
 - ☐ Dengan segmentasi, seluruh alamat dalam instruksi mengacu ke sebuah offset di dalam segmen.
 - ☐ Dalam Pentium, mode ini digunakan untuk mereferensi variabel – variabel global



2. Format-Format Instruksi

- Format instruksi menentukan layout bit suatu instruksi.
- Format instruksi harus mencakup opcode dan secara implisit atau eksplisit, nol operand atau lebih



- Seluruh operand eksplisit direferensikan dengan menggunakan salah satu mode pengalamatan yang ada
- Secara implisit atau eksplisit format harus dapat mengindikasikan mode pemngalamata seluruh operandnya.
- Pada sebagian besar set instruksi digunakan lebih dari satu format instruksi.



A. Panjang Instruksi

- Umumnya pemrograman menginginkan opcode, operand, dan mode pengalamatan yang lebih banyak serta range alamat yang lebih besar
- Dengan adanya opcode dan operand yang lebih banyak akan memudahkan pekerjaan pemrograman



- Mode pengalamatan yang lebih banyak akan memberikan fleksibilitas yang lebih besar terhadap pemogram dalam mengimplementasikan fungsi-fungsi tertentu, seperti manipulasi table dan pencabangan yang berjumlah banyak.
- Dengan bertambahnya ukuran memori utama dan semakin banyaknya pemakaian memori virtual, pemogram akan dapat mengalami jangkauan memori yang lebih besar.
- Kecepatan perpindahan memori tidak dapat diatasi dengan penambahan kecepatan processor



- Karena memori akan dapat menjadi sebuah bottleneck apabila prosessor dapat mengeksekusi instruksi lebih cepat dari pada kecepatan untuk mengambil instruksi itu.
- Salah satu cara mengatasi masalah ini adalah dengan menggunakan cache memori atau dengan menggunakan instruksi-instruksi yang lebih pendek
- Instruksi 16 bit akan dapat diambil dua kali lebih cepat di bandingkan instruksi 32 bit namun mungkin akan dieksekusi dua kali lebih lambat



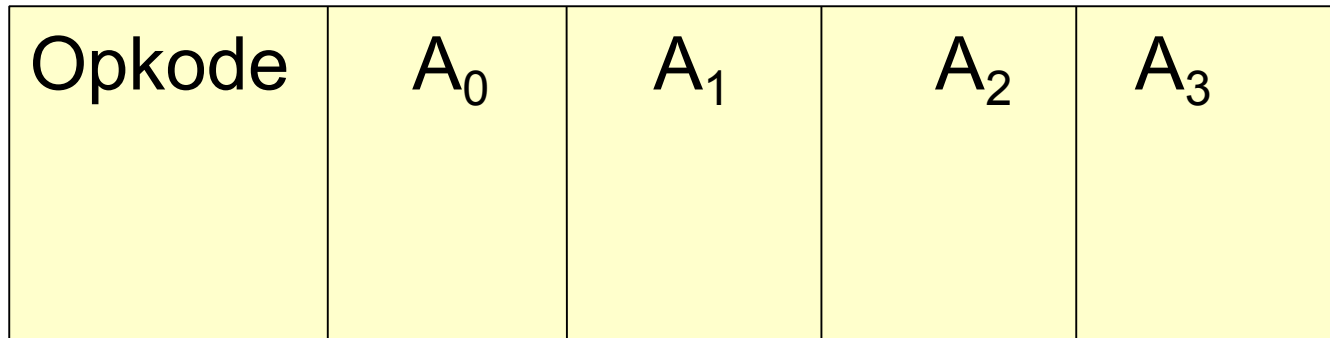
B. Format Instruksi

- Organisasi internal pada komputer dinyatakan oleh instruksi-instruksi yang dapat dijalankannya.
- Suatu instruksi merupakan suatu tata cara yang digunakan oleh komputer untuk menyatakan operasi-operasi seperti ADD, STORE, LOAD, MOVE dan BRANCH beserta untuk menentukan lokasi data di mana suatu operasi akan dikerjakan.



C. Format Alamat

- Pengkodean biner pada setiap komputer memiliki format kode instruksi tersendiri.
- Pada komputer terdahulu, setiap instruksi terdiri atas sebuah upcode dan empat field alamat.



- A_0 = alamat operand pertama
- A_1 = alamat operand kedua
- A_2 = alamat di mana hasil operand disimpan
- A_3 = Alamat dari instruksi berikutnya



D. Mode Pengalamatan

- Suatu mode pengalamatan dapat digunakan untuk menentukan suatu alamat tempat untuk dimana operand akan di fetch
- Beberapa teknik semacam ini dapat meningkatkan kecepatan pelaksanaan instruksi dengan menurunkan jumlah referensi pada memori utama dan meningkatkan jumlah referensi pada register kecepatan tinggi



- Mode pengalamatan ini menjabarkan suatu aturan untuk menginterpretasikan atau memodifikasi field alamat dari instruksi sebelum operand di referensikan
- Pada semua mode pengalamatan lainnya, operand yang sesungguhnya tidak disimpan pada field alamat tetapi beberapa nilai di jabarkan dan di gunakan untuk menentukan operasi operand.



E. Kode Instruksi (KI)

- Selain dari representasi data, kode biner juga digunakan untuk membuat instruksi kontrol dalam komputer, yang disebut kode instruksi.
- Kode instruksi merupakan kelompok bit yang memberitahukan kepada komputer untuk menunjukkan suatu operasi tertentu.



- Kode Instruksi dibagi dalam bagian-bagian, yang masing-masing bagian mempunyai interpretasi sendiri
- Bagian yang paling pokok adalah kode operasi (Operation Code / Opcode)
- Opcode adalah sekelompok bit yang menunjukkan operasi seperti ADD, SUBTRACT, SHIFT, dan COMPLEMENT
- Bagian lain dari instruksi mencakup satu operasi (operand) atau lebih



- Operand adalah suatu nama yang digunakan untuk obyek instruksi dan mungkin data atau alamat yang mengatakan dimana data tersebut
- Untuk membuat kode instruksi dalam komputer harus kode biner. (seperti operasi LOAD dan Store)
- Load adalah meng-copy bilangan dari lokasi memori kedalam register
- Store adalah meng-copy bilangan dari register kedalam lokasi memori



Selesai



PERTEMUAN

11

STRUKTUR DAN FUNGSI CPU



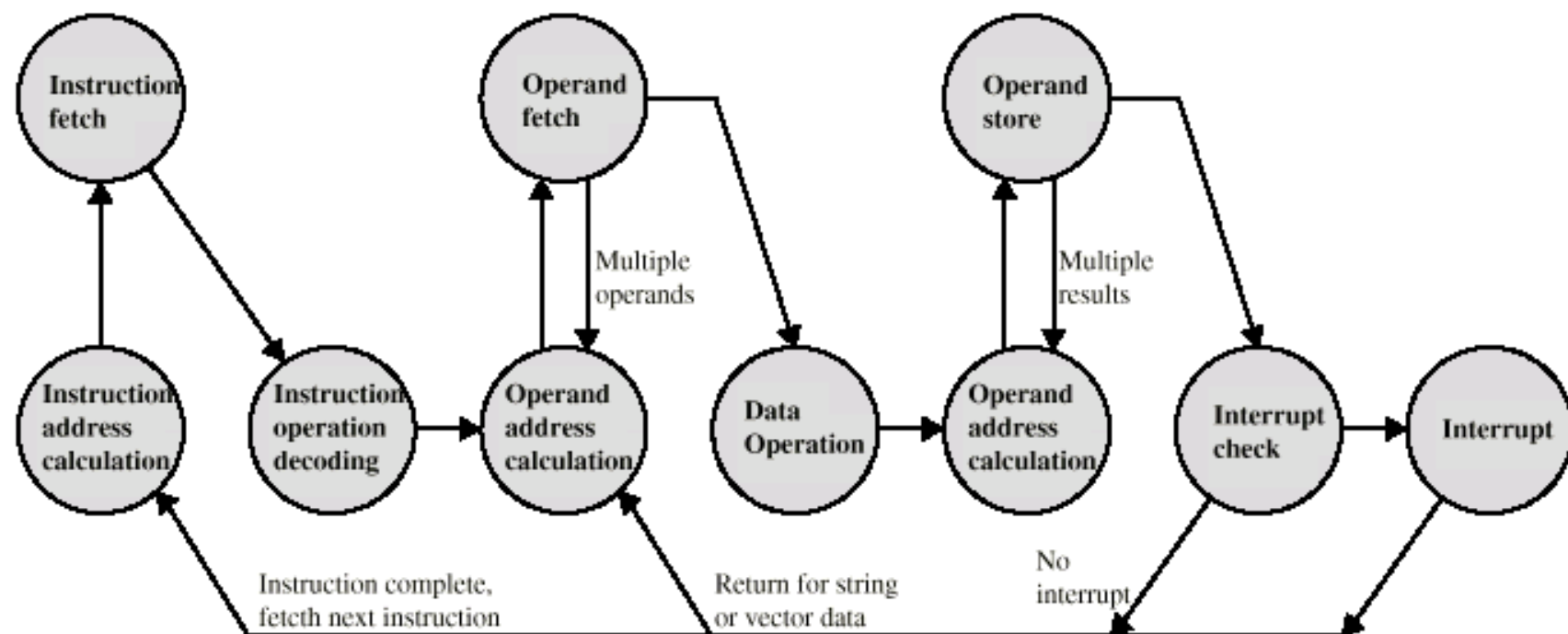
1. Organisasi Processor

Hal-hal yang perlu dilakukan CPU adalah :

1. Fetch Instruction = mengambil instruksi
2. Interpret Instruction = Menterjemahkan instruksi
3. Fetch Data = mengambil data
4. Process Data = mengolah data
5. Write data = menulis data



- Agar dapat melakukan hal-hal diatas maka CPU perlu menyimpan data untuk sementara waktu
- CPU harus dapat mengingat lokasi instruksi terakhir sehingga CPU akan dapat mengambil instruksi berikutnya.
- CPU perlu menyimpan instruksi dan data untuk sementara waktu pada saat instruksi sedang di eksekusi.
- Dengan kata lain CPU memerlukan memori internal berukuran kecil.





2. Organisasi Register

- Register dalam CPU memiliki dua fungsi :
- 1. User-visible Register
- 2. Control and status Register



A. User-Visible Register

- Adalah register yang dapat direferensikan dengan menggunakan bahasa mesin yang dieksekusi CPU



- Terdiri dari :
 - General Purpose = dapat berisi operand sembarang op-code
 - Data = hanya dapat dipakai untuk menampung data
 - Alamat = dapat di gunakan untuk mode pengalamatan tertentu
 - Kode-kode kondisi (flag) = bit-bit yang di setel perangkat keras CPU sebagai hasil operasi



B. Control dan Status Register

- Adalah register-register yang digunakan oleh unit kontrol untuk mengontrol operasi CPU dan oleh program sistem operasi untuk mengontrol eksekusi program



Terdapat empat buah register yang penting

1. **Program Counter (PC)**

berisi alamat instruksi yang akan di ambil

2. **Instructions Register (IR)**

berisi instruksi yang terakhir diambil

3. **Memory Address Register (MAR)**

Berisi alamat sebuah lokasi di dalam memori

4. **Memory Buffer Register (MBR)**

Berisi sebuah word data yang akan dituliskan kedalam memori atau word yang terakhir di baca



- Semua rancangan CPU mencakup sebuah register atau sekumpulan register sering kali disebut Program Status Word (PSW) yang berisi informasi status.
- Biasanya PSW berisi kode kondisi dan informasi status lainnya



Common field atau flag meliputi hal-hal berikut ini

- Sign = berisi bit tanda hasil operasi aritmetika terakhir
- Zero = disetel bila hasil sama dengan nol
- Carry = disetel apabila operasi yang dihasilkan didalam carry (penambahan) kedalam bit yang lebih tinggi atau borrow (pengurangan) dari bit yang lebih tinggi



- Equal = disetel apabila hasil perbandingan logikanya sama
- Overflow = digunakan untuk mengindikasikan overflow aritmetika
- Interrupt enable / disable = di gunakan untuk mengizinkan atau mencegah interrupt
- Supervisor = mengindikasikan apakah CPU sedang mengeksekusi dalam mode supervisor atau dalam mode user.



3. Siklus Instruksi



A. Proses Tak Langsung

- Eksekusi sebuah instruksi melibatkan sebuah operand atau lebih di dalam memori, yang masing-masing operand memerlukan akses memori
- Apabila digunakan pengalamatan tak langsung, maka diperlukan akses memori tambahan



- Kita dapat menganggap pengambilan alamat-alamat tak langsung sebagai sebuah sub instruksi atau lebih.
- Kegiatan utamanya terdiri dari aktifitas-aktifitas pengambilan instruksi dan pengeksekusian instruksi.
- Operand yang dibutuhkan diambil dengan menggunakan pengalamatan tak langsung
- Setelah eksekusi dilakukan, interrupt dapat diproses sebelum mengambil instruksi berikutnya



Aliran data :

- Pada saat pengambilan instruksi di baca dari memori
- Dalam CPU, PC berisi alamat berikutnya yang akan diambil
- Alamat ini di pindahkan ke MAR dan ditaruh di bus alamat
- Unit control meminta pembacaan memori dan hasilnya disimpan di bus data dan disalin ke MBR dan kemudian di pindahkan ke IR



- Sementara itu PC naik nilainya 1, sebagai persiapan untuk pengambilan selanjutnya.
- Bila siklus pengambilan telah selesai, Unit kontrol memeriksa isi IR untuk menentukan apakah IR berisi operand specifier yang menggunakan pengalamatan tak langsung
- Apabila berisi operand maka dilakukan siklus tak langsung



4. Pipelining Instruksi

- Efisiensi sebuah sistem komputer dinilai berdasarkan kecepatan perangkat keras dan fasilitas-fasilitas perangkat lunak.
- Penilaian ini disebut dengan throughput = jumlah pemrosesan yang dapat dikerjakan dalam suatu interval waktu tertentu.



- Salah satu teknik yang mendorong peningkatan suatu sistem throughput yang cukup hebat disebut sebagai pemrosesan PIPELINE
- Pemrosesan Pipeline dalam suatu komputer diperoleh dengan membagi suatu fungsi yang akan dijalankan menjadi beberapa sub fungsi yang lebih kecil dan merancang perangkat keras yang terpisah (STAGE), untuk setiap sub fungsi.
- Stage-stage itu kemudian bersama-sama membentuk pipeline tunggal untuk menjalankan fungsi asli



Sinkronisasi Pipeline

- Pada semua baris perakitan industri, efisiensi suatu pipeline dapat berkurang jauh akibat suatu bottleneck.
- Bottleneck terjadi sewaktu pemrosesan pada suatu stage, menghabiskan waktu lebih lama dari stage yang lain



- Karena itu idealnya kita menginginkan pada semua stage menghabiskan waktu yang sama.
- Untuk menyamakan waktu yang diperlukan pada setiap stage maka stage-stage tersebut harus di sinkronisasi.
- Dapat dilakukan dengan menyisipkan kunci-kunci sederhana antara stage-stage tersebut



Efisiensi Pipeline

- Suatu fungsi pipeline hampir selalu lebih baik dari pada fungsi non-pipeline
- Namun pada kenyataannya, harga pipeline tersebut sangat mempengaruhi kapan dan apakah fungsi-fungsi yang di pipeline-kan lebih baik atau tidak.



Beberapa penyebab harga pipeline :

1. Tambahan kunci (latch) perangkat keras yang dibutuhkan.
2. Kendali yang diperlukan untuk penjadwalan input tersebut.
3. Waktu yang dihabiskan oleh data dalam latch untuk menyesuaikan diri dengan suatu penangguhan clock yang seragam
4. Jumlah rata-rata input yang tersedia



Klasifikasi Pipeline :

- **A. Klasifikasi berdasarkan Fungsi**
- **Pipelining Aritmatika** = Proses segmentasi fungsi dari ALU dari sistem yang muncul dalam kategori ini.



Pipelining Instruksi

- Proses fetch pada instruksi tidak akan dimulai sampai eksekusi tidak akan dimulai sampai eksekusi instruksi sebelumnya selesai.
- Untuk mem-pipelinekan fungsi ini, instruksi-instruksi yang berdampingan di fetch dari memori ketika instruksi yang sebelumnya di decode dan di jalankan.
- Proses pipelining instruksi disebut instructions



Pipelining Processor

- Sewaktu stage dari suatu stage merupakan merupakan prosesor aktual dan latch-latch saling berbagi memori antara prosessor-prossor tersebut maka pipeline itu disebut sebagai pipeline prossor.



4. Register-register Processor Pentium



UNIT INTEGER

Jenis	Jumlah	Panjang (Bit)	Kegunaan
General	8	32	General purpose user register
Segment	6	16	Berisi pemilih pemilih segment
Flag	1	32	Bit-bit status dan kontrol
Pointer Instruksi	1	32	Pointer instruksi

UNIT FLOATING POINT

Jenis	Jumlah	Panjang (Bit)	Kegunaan
Numerik	8	80	Menampung bilangan-bilangan floating point
Kontrol	1	16	Bit-bit kontrol
Status	1	16	Bit-bit status
Tag Word	1	16	Men-spesifikasi-kan register numerik
Pointer Instruksi	1	48	Menunjuk ke instruksi yang di interupsi oleh exception
Pointer Data	1	48	Menunjuk ke operand yang di interupsi oleh exception



selesai



PERTEMUAN

12

REDUCED INSTRUCTIONS

SET COMPUTER

(RISC)



- Arsitektur RISC merupakan kemajuan yang sangat dramatis dalam frase sejarah arsitektur CPU. Dan merupakan tantangan bagi arsitektur konvensional



- Walaupun sistem RISC telah ditentukan dan dirancang dalam berbagai cara berdasarkan kelompok-kelompoknya, elemen penting yang di gunakan oleh sebagian rancangan adalah sebagai berikut :
 - Set instruksi yang terbatas dan sederhana
 - Register general purpose yang berjumlah banyak, atau penggunaan teknologi kompiller untuk mengoptimalkan pemakaian registernya
 - Penekanan pada pengoptimalan pipeline instruksi



1. Karakteristik karakteristik Eksekusi Instruksi

- Salah satu bentuk evolusi komputer yang paling di rasakan adalah adalah evolusi bahasa pemrograman



Karakteristik dari beberapa prosesor CISC, RISC dan Superskalar

Karakteristik	CISC (Complex Instruction Set Computer)		
	IBM 370/168	VAX 11/780	Intel 80486
Tahun dibuat	1973	1978	1989
Jumlah instruksi	208	303	235
Instruksi (Bytes)	2-6	2-57	1-11
Mode pengalamatan	4	22	11
Jumlah register general purpose	16	16	8
Ukuran memori kontrol (Kbits)	420	480	246
Ukuran cache (Kbytes)	64	64	8



Karakteristik	RISC	
	Motorola 88000	MIPS R4000
Tahun dibuat	1988	1991
Jumlah instruksi	51	94
Instruksi (Bytes)	4	32
Mode pengalamatan	3	1
Jumlah register general purpose	32	32
Ukuran memori kontrol (Kbits)	-	-
Ukuran cache (Kbytes)	16	128



Karakteristik	Super Skalar	
	IBM RS/System 6000	Intel 80960
Tahun dibuat	1990	1989
Jumlah instruksi	184	62
Instruksi (Bytes)	4	4,8
Mode pengalamatan	2	11
Jumlah register general purpose	32	23-256
Ukuran memori kontrol (Kbits)	-	-
Ukuran cache (Kbytes)	32-64	0,5



- Untuk memahami RISC harus diawali dengan tinjauan singkat tentang karakteristik eksekusi instruksi
- Aspek-aspek komputasi yang dimaksud adalah sebagai berikut :
 - A. Operasi-operasi yang di lakukan
 - B. Operand-operand yang di gunakan
 - C. Pengurutan eksekusi



A. Operasi

	Kejadian Dinamik		Instruksi Mesin Berbobot		Referensi Memori Berbobot	
	Pascal	C	Pascal	C	Pascal	C
Assign	45	38	13	13	14	15
Loop	5	3	42	32	33	26
Call	15	12	31	33	44	45
If	29	43	11	21	7	13
Goto	-	3	-	-	-	-
Other	6	1	1	1	2	1



B. Operand

	Pascal	C	Rata-rata
Konstanta Integer	16	23	20
Variabel skalar	58	53	55
Array/Struktur	26	24	25



2. Penggunaan File Register Besar

- Alasan diperlukannya penyimpanan register adalah dengan melihat kenyataan bahwa menyimpan register merupakan perangkat penyimpan paling cepat, yang lebih cepat dibandingkan dengan memori utama dan memori cache.
- File register secara fisik berukuran kecil, dan umumnya berada pada satu keping dengan ALU dan Control Unit dan hanya memakai alamat yang lebih pendek dibandingkan dengan alamat-alamat cache dan memori



- Dengan demikian, di perlukan strategi yang dapat menjaga operand-operand yang paling sering di akses tetap di dalam register dan untuk meminimalkan operasi-operasi register memori
- Dalam mamaksimalkan register dipakai 2 pendekatan
 1. Pendekatan perangkat lunak
 2. Pendekatan perangkat keras



3. Register Windows

- Penggunaan register dalam jumlah yang besar akan mengurangi kebutuhan mengakses memori
- Dalam hal ini tugas perancang adalah mengatur register-register sedemikian rupa sehingga tujuan dapat tercapai



4. File Register Berukuran Besar VS Cache

- File register yang diorganisasikan menjadi dua jendela, berfungsi sebagai buffer kecil yang cepat untuk menampung subset seluruh variabel yang memiliki kemungkinan besar akan banyak di pakai
- File register berfungsi lebih menyerupai cache memori
- File register dapat tidak efisien dalam menggunakan ruang, karena tidak semua prosedur akan memerlukan ruang jendela sepenuhnya yang telah diberikan



- File register dapat tidak efisien dalam menggunakan ruang, karena tidak semua prosedur akan memerlukan ruang jendela sepenuhnya yang telah diberikan
- Sebaliknya cache memiliki ketidak efisienan lainnya yaitu
 - Data akan di baca kedalam cache dalam bentuk blok-blok.
 - sedang file register hanya berisi variabel-variabel yang sedang di gunakan.
 - Cache membaca suatu blok data, yang mungkin sebagian darinya tidak akan di gunakanCache memiliki kemampuan untuk menangani variabel global dan juga variabel lokal



File Register Besar	Cache
Semua skalar lokal	Skalar lokal yang baru di pakai
Variabel-variabel individual	Sekelompok memori
Variabel-variabel global yang di assign kompiler	Variabel global yang baru di pakai
Save/restore tergantung kedalam pertimbangan prosedur	Save/restore tergantung algoritma penggantian cache
Pengalamatan register	Pengalamatan memori



5. Otimasi Register Berbasis Kompiler

- Pada mesin RISC hanya tersedia register dalam jumlah yang sedikit (16-32 buah)
- Disini penggunaan register yang telah di optimalkan tersebut merupakan tanggung jawab kompiler
- Fungsi kompiler adalah untuk menjaga operand bagi komputasi sebanyak mungkin di dalam register dan bukannya di dalam memori utama.
- Hal itu ditujukan untuk meminimalkan operasi load dan store



6. Karakteristik CISC Vs RISC

- Rancangan RISC dapat memperoleh keuntungan dengan mengambil sejumlah feature CISC
- Rancangan CISC dapat memperoleh keuntungan dengan mengambil sejumlah feature RISC



Ciri-ciri RISC :

1. Instruksi berukuran tunggal
2. Ukuran yang umum adalah 4 Byte
3. Jumlah mode pengalamatan data yang sedikit (< 5 buah)
4. Tidak terdapat pengalamatan tak langsung
5. Tidak terdapat operasi yang menggabungkan operasi load / store dengan operasi aritmetika



6. Tidak terdapat lebih dari satu operand beralamat memori per instruksi
7. Tidak mendukung perataan sembarang, bagi data untuk operasi load/store
8. Jumlah maksimum pemakaian memori management Unit (MMU) bagi suatu alamat data adalah sebuah instruksi
9. Jumlah bit bagi register specifier sama dengan lima atau lebih
10. Jumlah bit floating point register specifier empat atau lebih



7. Kontroversi RISC dan CISC

- Tidak terdapat mesin-mesin RISC dan CISC yang sebanding dalam hal harga, tingkat teknologi, kompleksitas gate, kecanggihan kompilar dsb
- Tidak terdapat pengujian program yang pasti. Kinerja bervariasi sesuai dengan programnya
- Kesulitan dalam mengumpulkan akibat-akibat yang disebabkan perangkat keras yang berkaitan dengan keterampilan dalam membuat kompilar



- Kesulitan dalam mengumpulkan akibat-akibat yang disebabkan perangkat keras yang berkaitan dengan keterampilan dalam membuat kompiler
- Sebagian besar analisis komparatif tentang RISC dilakukan pada mesin-mesin “mainan” bukannya pada mesin-mesin komersial. Selain itu, sebagian besar mesin yang tersedia secara komersial dan diiklankan sebagai RISC memiliki karakteristik campuran antara RISC dan CISC. Dengan demikian, perbandingan yang adil dengan mesin CISC komersial dan murni sangatlah sulit dilaksanakan
- Dalam beberapa tahun terakhir, kontroversi RISC dengan CISC semakin berkembang.



- Hal ini disebabkan karena terjadinya semakin konvergensi teknologi.
- Dengan semakin bertambahnya kepadatan keping dan semakin cepatnya perangkat keras, maka sistem RISC menjadi semakin kompleks
- Bersamaan dengan hal itu, untuk mencapai kinerja yang maksimum, Rancangan CISC telah difokuskan terhadap masalah-masalah tradisional yang berkaitan dengan RISC, seperti misalnya penambahan jumlah register general purpose dan penekanan pada rancangan pipeline instruksi



selesai



PERTEMUAN

13

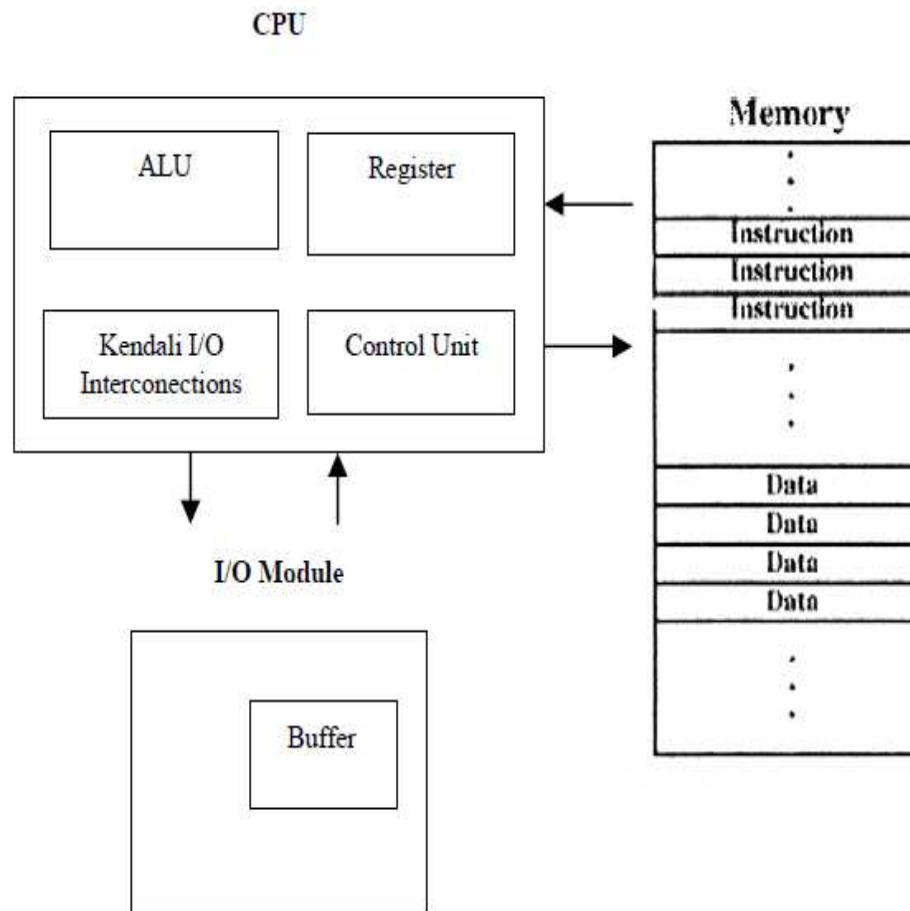
REDUCED INSTRUCTIONS

SET

ARSITECTURE



ARSITEKTUR KOMPUTER



ALU

Tugas utama adalah melakukan semua perhitungan aritmatika dan melakukan keputusan dari suatu operasi logika.

Register

Alat penyimpanan kecil yang mempunyai kecepatan akses cukup tinggi yang digunakan untuk menyimpan data dan instruksi yang sedang diproses sementara data dan instruksi lainnya menunggu giliran untuk diproses masih disimpan di dalam memori utama.



Register dalam CPU diantaranya :

Register untuk alamat dan buffer :

- MAR (Memory Address Register)
 - Untuk mencatat alamat memori yang akan diakses (baik yang akan ditulisi maupun dibaca)
- MBR (Memory Buffer Register)
 - Untuk menampung data yang akan ditulis ke memori yang alamatnya ditunjuk MAR atau untuk menampung data dari memori (yang alamatnya ditunjuk oleh MAR) yang akan dibaca.
- I/O AR (I/O Address Register)
 - Untuk mencatat alamat port I/O yang akan diakses(baik akan ditulisi / dibaca).
- I/O BR (I/O Buffer Register)
 - Untuk menampung data yang akan dituliskan ke port yang alamatnya ditunjuk I/O AR atau untuk menampung data dari port (yang alamatnya ditunjuk oleh I/O AR) yang akan dibaca.

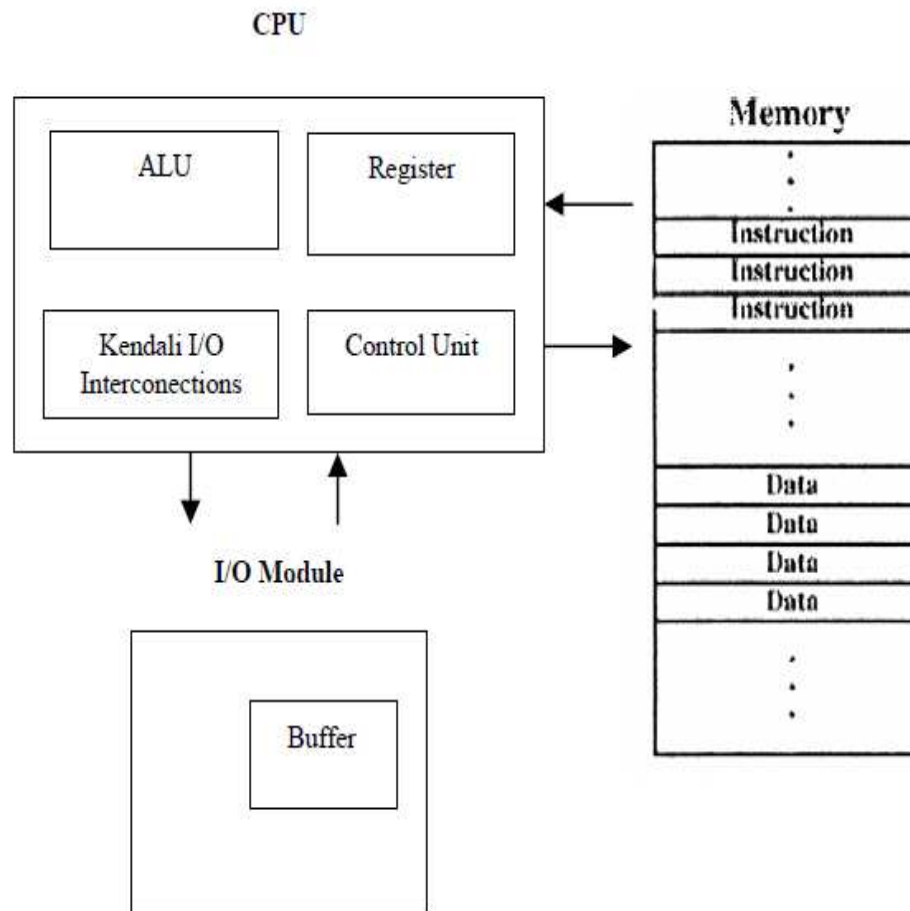


Register untuk eksekusi instruksi

- PC (Program Counter)
 - Mencatat alamat memori dimana instruksi di dalamnya akan dieksekusi
- IR (Instruction Register)
 - Menampung instruksi yang akan dilaksanakan
- AC (Accumulator)
 - Menyimpan data sementara baik data yang sedang diproses atau hasil proses.



ARSITEKTUR KOMPUTER



Control Unit

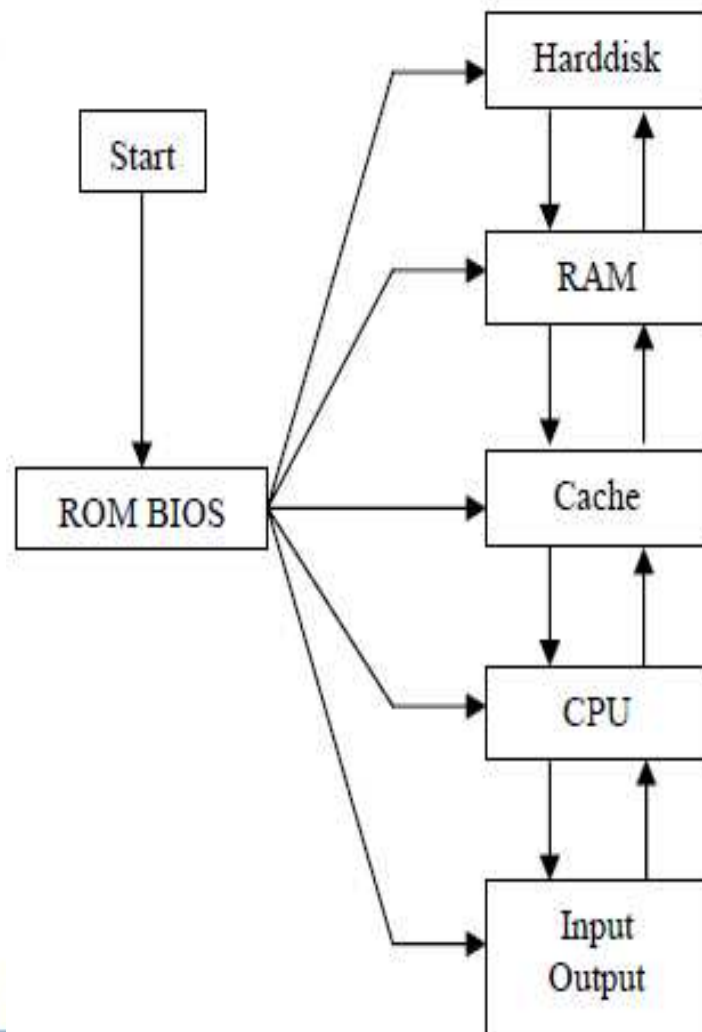
Bertugas mengatur dan mengendalikan semua peralatan yang ada di sistem komputer.

I/O Interconnection

Input-Output (/O) Interconnection merupakan sistem koneksi yang menghubungkan antar komponen internal dalam sebuah CPU, yaitu ALU, unit kontrol, dan register serta menghubungkan CPU dengan bus-bus eksternal diluar CPU.



Cara Kerja Komputer

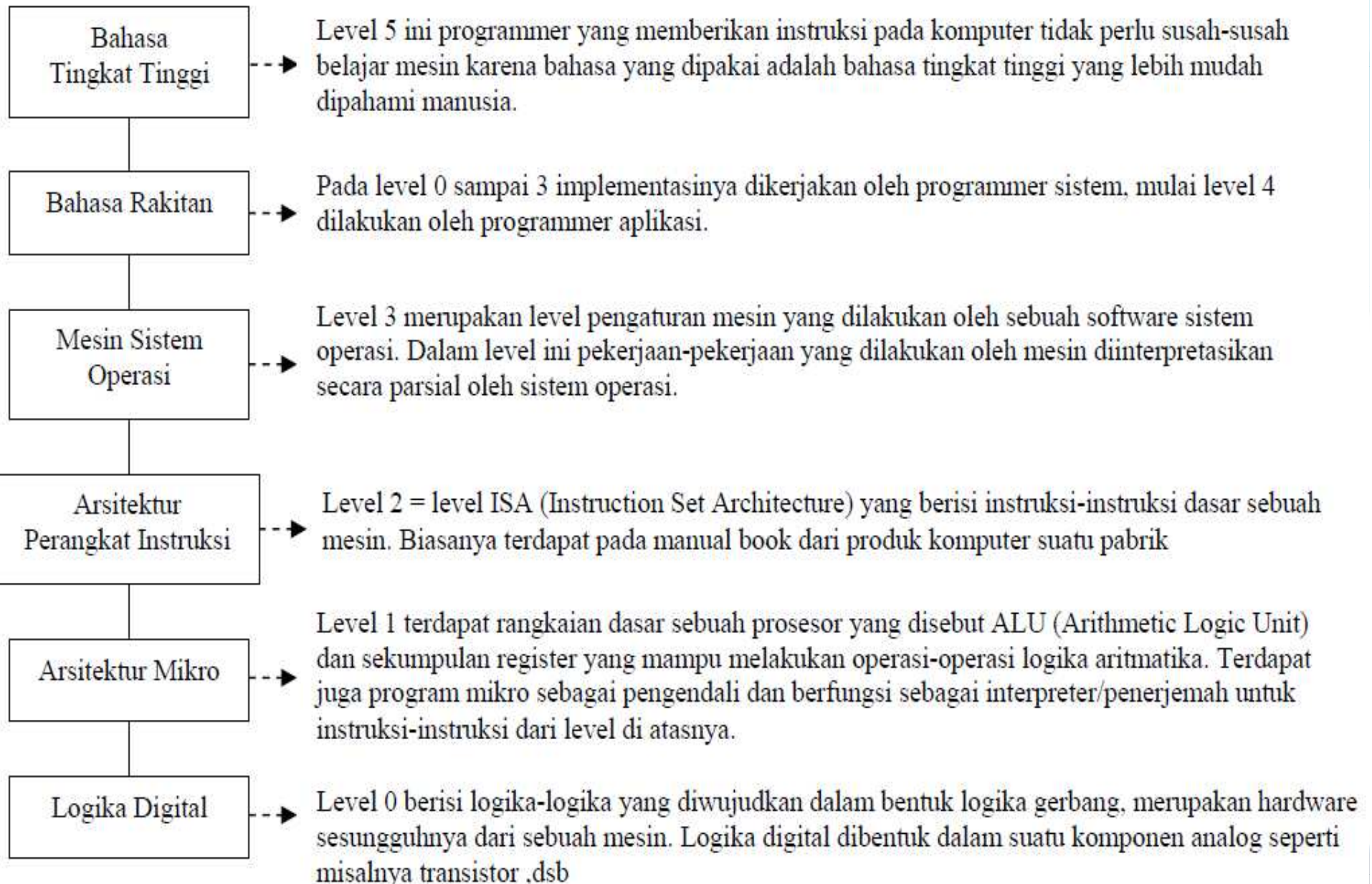


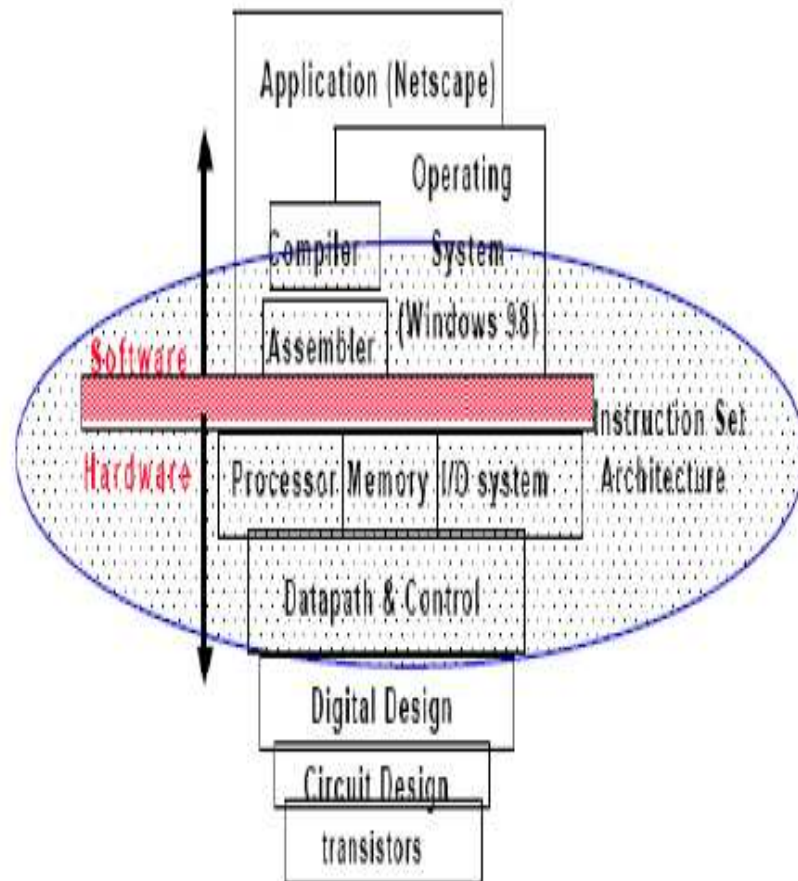
Harddisk menyimpan data dan program yang bersifat permanen. RAM mengcopy data/program dari harddisk untuk diproses oleh CPU

Dari RAM, data atau program yang akan diolah oleh CPU tidak semua langsung diproses CPU tetapi dicopy ke cache memori untuk mengatasi kesenjangan kecepatan CPU-memori

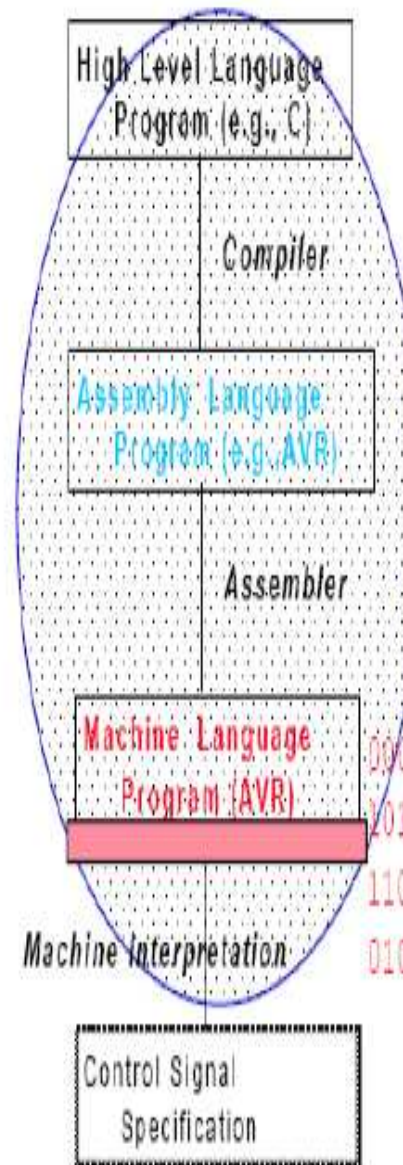
CPU melakukan komunikasi dengan modul I/O untuk menerima input atau menampilkan output dari proses yang dihasilkan. Output akan ditampilkan di komponen-komponen output

KOMPUTER SEBAGAI MESIN 6 LEVEL





° Koordinasi dari **berbagai tingkat abstraksi**



temp = v[k];

v[k] = v[k+1];

v[k+1] = temp;

ldi r1, 0x12

ldi r2, 0x34

add r1, r2

st Y+, r1

0010 1001 1100 0110 1010 1111 0101 1000
 1010 1111 0101 1000 0000 1001 1100 0110
 1100 0110 1010 1111 0101 1000 0000 1001
 0101 1000 0000 1001 1100 0110 1010 1111



SIKLUS INSTRUKSI

Program yang ada di memori komputer terdiri dari sederetan instruksi. Setiap instruksi dieksekusi melalui suatu siklus. Setiap siklus instruksi terdiri dari tahap-tahap :

1. Instruction fetch, yaitu mengambil instruksi dari memori dan mentransfernya ke unit kontrol.
2. Mengartikan (decode) instruksi dan menentukan apa yang harus dikerjakan serta data apa yang digunakan.
3. Baca alamat efektif, jika instruksi beralamat indirect.
4. Proses eksekusi instruksi dengan memilih operasi yang diperlukan dan mengendalikan perpindahan data yang terjadi.
5. Terdapat register dalam CPU yang berfungsi mengawasi dan menghitung instruksi selanjutnya yaitu Program Counter
6. PC akan menambah satu hitungan setiap kali CPU membaca instruksi
7. Instruksi-instruksi yang dibaca akan dibuat dalam register instruksi (IR)



Sejarah

- **Reduced Instruction Set Computing (RISC)** atau "Komputasi set instruksi yang disederhanakan" pertama kali digagas oleh John Cocke, peneliti dari IBM di Yorktown, New York pada tahun 1974 saat ia membuktikan bahwa sekitar 20% instruksi pada sebuah prosesor ternyata menangani sekitar 80% dari keseluruhan kerjanya.
- Komputer pertama yang menggunakan konsep RISC ini adalah IBM PC/XT pada era 1980-an. Istilah RISC sendiri pertama kali dipopulerkan oleh David Patterson, pengajar pada University of California di Berkely.



- RISC, yang jika diterjemahkan berarti "Komputasi Kumpulan Instruksi yang Disederhanakan", merupakan sebuah [arsitektur komputer](#) atau arsitektur komputasi modern dengan instruksi-instruksi dan jenis eksekusi yang paling sederhana. Arsitektur ini digunakan pada komputer dengan kinerja tinggi, seperti komputer vektor.
- Selain digunakan dalam komputer vektor, desain ini juga diimplementasikan pada prosesor komputer lain, seperti pada beberapa mikroprosesor [Intel 960](#), [Itanium \(IA64\)](#) dari [Intel Corporation](#), [Alpha AXP](#) dari [DEC](#), [R4x00](#) dari [MIPS Corporation](#), [PowerPC](#) dan [Arsitektur POWER](#) dari [International Business Machine](#). Selain itu, RISC juga umum dipakai pada [Advanced RISC Machine \(ARM\)](#) dan [StrongARM](#) (termasuk di antaranya adalah Intel [XScale](#)), [SPARC](#) dan [UltraSPARC](#) dari [Sun Microsystems](#), serta [PA-RISC](#) dari [Hewlett-Packard](#).
- Selain RISC, desain [Central Processing Unit](#) yang lain adalah [CISC](#) (*Complex Instruction Set Computing*), yang jika diterjemahkan ke dalam Bahasa Indonesia berarti Komputasi Kumpulan Instruksi yang kompleks atau rumit.



8. REDUCE INSTRUCTION SET ARCHITECTURE

- **Teknologi RISC relatif sangat baru, karena itu saat ini tidak terjadi perdebatan dalam menggunakan CISC atau RISC.**
- **Pendapat-pendapat yang ada hanya membahas kekurangan pendekatan RISC serta memberikan pemahaman motivasi pendukung RISC**
- **Alasan pertama yaitu, penyederhanaan kompilasi**



■ Tugas pembuat kompiler adalah menghasilkan rangkaian instruksi mesin bagi semua pernyataan HLL (high level language).

■ Apabila ada instruksi mesin yang menyerupai HLL, maka tugas ini akan disederhanakan.

■ Pekerjaan mengoptimalkan kode yang dihasilkan utk meminimalkan ukuran kode, mengurangi hitungan eksekusi instruksi, dan meningkatkan pipeline jauh lebih sulit apabila menggunakan CISC



Pipelining

- Merupakan suatu konsep pelaksanaan instruksi yang dibagi dalam banyak bagian, dimana masing-masing bagian ditangani oleh hardware khusus dan keseluruhan bagian dapat beroperasi secara paralel.



On board cache

- Cache adalah memori kecil berkapasitas kecil tetapi berkecepatan tinggi yang dipasang antara prosesor dan memori utama.
- Cache dibuat karena adanya kesenjangan perbedaan kecepatan yang sangat besar antara prosesor dan memori utama.
- Perkembangan kecepatan prosesor tidak diimbangi peningkatan kecepatan memori sehingga proses pembacaan data dari memori relatif lebih lambat bila dibandingkan dengan kecepatan prosesor, sehingga prosesor harus menunggu data dari memori dan menjadi inefisiensi kinerja prosesor.

Contoh :

- RAM : 128 MB DDR 333 □□ clock speed 333 MHz
- Processor : Athlon 1800 MHz □□ clock speed 1800 MHz \approx 1,8 GHz



On board L1 dan L2 cache

- L1 cache = level 1 cache = CPU internal cache = cache yang terletak di inti processor
- L2 cache = level 2 cache = CPU external cache = cache yang terletak di motherboard.
- Pada prosesor generasi baru seperti Pentium II – IV, Duron, Thunderbird L2 cache diletakkan di dalam prosesor (= tidak diletakkan di inti prosesor tapi dimasukkan dalam kemasan prosesor sehingga lebih dekat dengan inti prosesor).



- **Namun CISC akan menghasilkan program yang lebih kecil dan lebih cepat dari RISC**
- CISC cenderung menggunakan instruksi-instruksi yang sederhana.
- CISC cenderung menekankan pada referensi register dibandingkan pada referensi memori, dan referensi register memerlukan bit yang jumlahnya lebih sedikit



Karakteristik Arsitektur RISC

- RISC harus tidak boleh lebih kompleks dan harus dapat mengeksekusi secepat mikro instruksi pada mesin-mesin CISC
- Dengan menggunakan instruksi sederhana atau instruksi satu siklus, hanya di butuhkan sedikit mikro kode



- Instruksi mesin dapat di hardware, instruksi ini akan lebih cepat di bandingkan instruksi-instruksi lainnya yang sejenis pada mesin lainnya, karena instruksi tersebut tidak perlu mengakses penyimpanan kontrol mikroprogram pada saat eksekusi instruksi berlangsung.



Karakteristik RISC

- + Satu instruksi per siklus
- + Operasi register ke register
- + Address mode sederhana
- + Format Instruksi sederhana



Pipelining RISC

- Sebagian besar instruksi merupakan operasi register ke register, dan sebuah siklus instruksi memiliki dua buah fase :

1. **I : Instruction Set**
pengambilan
instruksi

2. **E : Execute Set**
melakukan operasi
ALU dengan input
Register dan output
register



Bagi operasi Load dan store
diperlukan 3 buah fase

1. **I** = Instruction Set

2. **E** = Execute = menghitung alamat
memori

3. **D** = Memori = operasi register ke
memori atau memori ke register



PERKEMBANGAN DESAIN PROSESOR

Tanenbaum mengemukakan adanya prinsip-prinsip penting dalam melakukan desain prosesor komputer modern yaitu prinsip RISC (Reduced Instruction Set Computer), yaitu :

1. Memaksimalkan kecepatan dimana instruksi-instruksi dikeluarkan

Prinsip ini menekankan pengembangan jumlah instruksi yang dapat diproses per detik pada sebuah prosesor, yaitu MIPS (Million of Instruction per Second), mengakibatkan muncul teknologi paralelisme prosesor yang akan dapat meningkatkan kinerja komputer

2. Memperbanyak instruksi yang secara langsung dapat dijalankan hardware untuk mempercepat kinerja

3. Instruksi-instruksi harus mudah untuk di-dekode-kan

Batas kritis pada tingkat kecepatan adalah dekode dari setiap instruksi. Semakin sedikit format instruksi maka akan semakin baik kinerja dan kecepatan sebuah eksekusi instruksi.

4. Hanya instruksi LOAD dan STORE yang diakses ke memori dan berusaha memperkecil instruksi yang langsung diakses dari memori utama.

5. Menyiapkan banyak register, sekarang rata-rata CPU memiliki 32 register.



Pendekatan CISC

- Tujuan utama dari arsitektur CISC adalah melaksanakan suatu perintah cukup dengan beberapa baris bahasa mesin sedikit mungkin. Hal ini bisa tercapai dengan cara membuat perangkat keras prosesor mampu memahami dan menjalankan beberapa rangkaian operasi.
- Untuk tujuan contoh kita kali ini, sebuah prosesor CISC sudah dilengkapi dengan sebuah instruksi khusus, yang kita beri nama MULT. Saat dijalankan, instruksi akan membaca dua nilai dan menyimpannya ke 2 register yang berbeda, melakukan perkalian operasi di unit eksekusi dan kemudian mengembalikan lagi hasilnya ke register yang benar. Jadi instruksi-nya cukup satu saja...
- **MULT 2:3, 5:2**
- MULT dalam hal ini lebih dikenal sebagai “complex instruction”, atau instruksi yang kompleks. Bekerja secara langsung melalui memori komputer dan tidak memerlukan instruksi lain seperti fungsi baca maupun menyimpan.



Pendekatan RISC

- Prosesor RISC hanya menggunakan instruksi-instruksi sederhana yang bisa dieksekusi dalam satu siklus.
- Dengan demikian, instruksi 'MULT' sebagaimana dijelaskan sebelumnya dibagi menjadi tiga instruksi yang berbeda, yaitu "LOAD", yang digunakan untuk memindahkan data dari memori ke dalam register, "PROD", yang digunakan untuk melakukan operasi produk (perkalian) dua operand yang berada di dalam register (bukan yang ada di memori) dan "STORE", yang digunakan untuk memindahkan data dari register kembali ke memori.
- Berikut ini adalah urutan instruksi yang harus dieksekusi agar yang terjadi sama dengan instruksi "MULT" pada prosesor RISC (dalam 4 baris bahasa mesin):
- **LOAD A, 2:3**
LOAD B, 5:2
PROD A, B
STORE 2:3, A



Perbandingan

CISC

Penekanan pada perangkat keras

Termasuk instruksi kompleks multi-clock

Memori-ke-memori: "LOAD" dan "STORE" saling bekerjasama

Ukuran kode kecil, kecepatan rendah

Transistor digunakan untuk menyimpan instruksi2 kompleks

RISC

Penekanan pada perangkat lunak

Single-clock, hanya sejumlah kecil instruksi

Register ke register: "LOAD" dan "STORE" adalah instruksi2 terpisah

Ukuran kode besar, kecepatan (relatif) tinggi

Transistor banyak dipakai untuk register memori



Persamaan Unjuk-kerja (*Performance*)

- Persamaan berikut biasa digunakan sebagai ukuran unjuk-kerja suatu komputer:

$$\frac{\text{time}}{\text{program}} = \frac{\text{time}}{\text{cycle}} \times \frac{\text{cycles}}{\text{instruction}} \times \frac{\text{instructions}}{\text{program}}$$

- Pendekatan CISC bertujuan untuk meminimalkan jumlah instruksi per program, dengan cara mengorbankan kecepatan eksekusi sekian siklus/detik.
- Sedangkan RISC bertolak belakang, tujuannya mengurangi jumlah siklus/detik setiap instruksi dibayar dengan bertambahnya jumlah instruksi per program.



selesai



PERTEMUAN

14

Quiz and Review



selesai