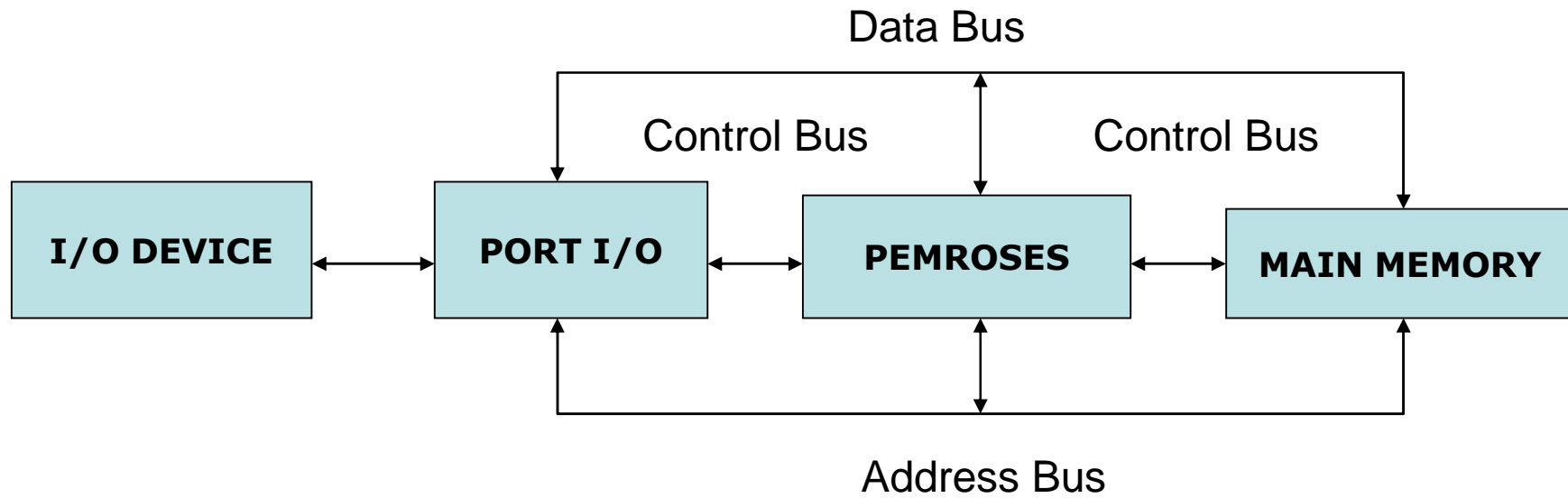


SISTEM OPERASI

Periyadi,ST

Figure 1. COMPUTER SYSTEM



BASIC ELEMENT OF COMPUTER SYSTEMS

- Processor
- Main Memory
 - referred to as real memory or primary memory
 - volatile
- I/O modules
 - secondary memory devices
 - communications equipment
 - terminals
- System bus
 - communication among processors, memory, and I/O modules

Pemroses/Processor

- Fungsi utama mengendalikan operasi komputer (Central Processing Unit)
- Melakukan fungsi pemrosesan data
- Melakukan operasi logik
- Mengelola aliran data

Cara Kerja

- Membaca instruksi dari memory kemudian dieksekusi, dimana eksekusi dituntun oleh clock yang membangkitkan pulsa ke pemroses. Setiap pulsa clock, pemroses melakukan kerja.
 - Mengambil instruksi yang dikodekan secara binner dari memori utama
 - Men-dekode instruksi menjadi aksi-aksi sederhana
 - Melaksanakan aksi-aksi

Operasi-operasi di komputer

- Operasi aritmatika
- Operasi Logika
- Operasi pengendalian

Komponen Pemroses

- ALU (Arithmetic Logic unit) untuk Komputasi, operasi aritmatika dan logika
- CU (Control Unit) untuk pengendalian operasi yang dilakukan sistem komputer
- Register-register, membantu pelaksanaan operasi-operasi pada pemroses

Register

- Terlihat pemakai, pemrogram dapat memeriksa isi dari register
 - Data Register
 - General Purpose Register
 - Difungsikan untuk berbagai keperluan pada instruksi mesin yang melakukan suatu operasi pada data
 - Special Purpose Register
 - Digunakan untuk keperluan tertentu, ex;
 - » Menampung Operasi floating point
 - » Menampung limpahan operasi aritmatika, dsb.

Register

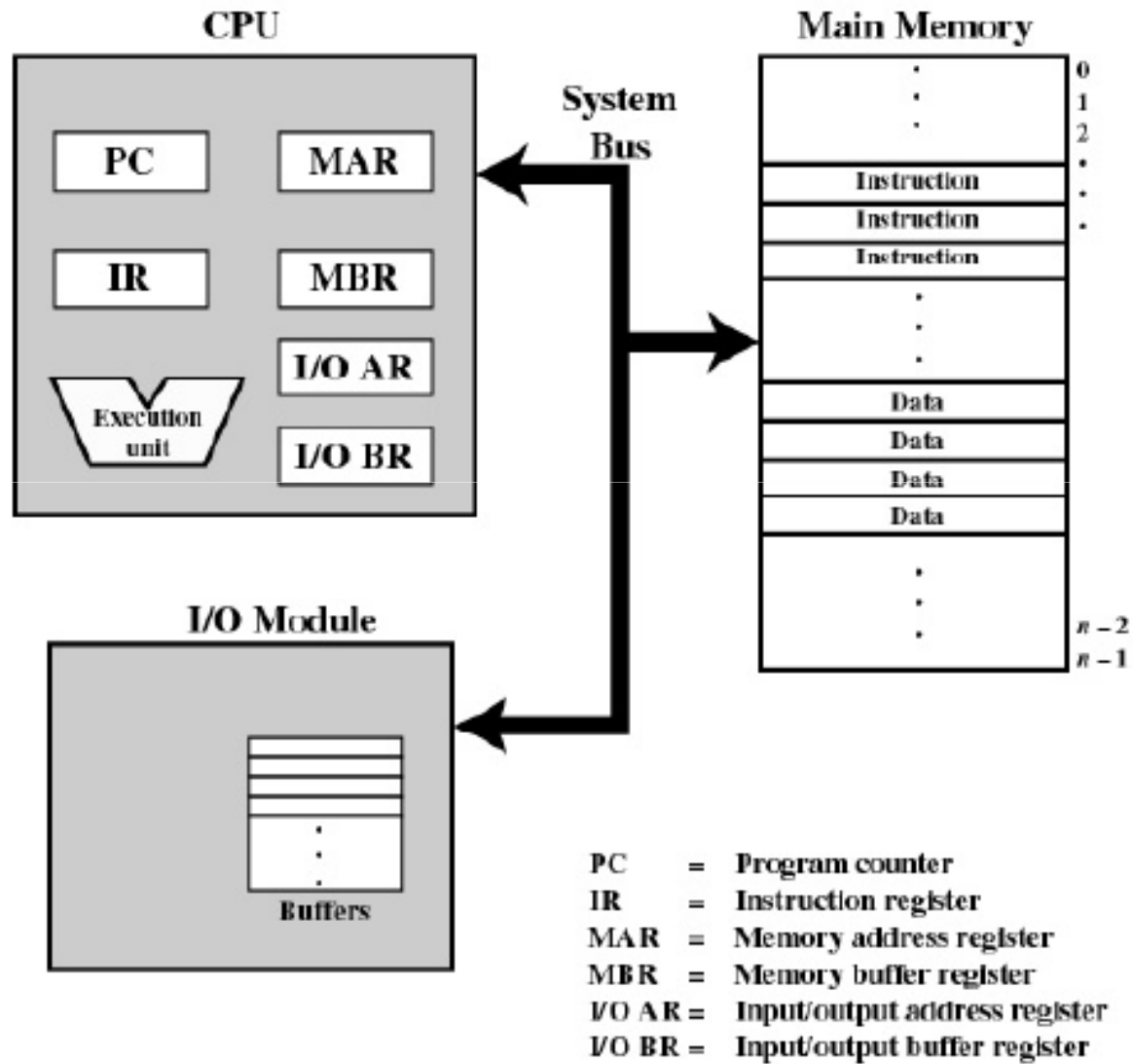
– Address Register

- Alamat data di main memory
- Alamat instruksi di memori utama.
- Bagian alamat yang digunakan dalam penghitungan alamat lengkap.
- Ex;
 - Index register
 - Segmen Pointer Register
 - Stack Point Register
 - Flag register

Register

- Register untuk kendali dan status
 - Register untuk alamat dan buffer
 - MAR (Memory Address Register)
 - Mencatat alamat memori yang akan di akses (R/W)
 - MBR (Memory Buffer Register)
 - Menampung data yang akan dituliskan ke memori yang alamatnya ditunjuk MAR atau menampung data dari memori yang akan dibaca
 - I/O AR
 - Mencatat alamat port yang akan diakses (R/W)
 - I/O BR
 - Menampung data yang akan dituliskan ke port yang alamatnya ditunjuk oleh I/O AR atau menampung data dari port yang akan dibaca

Figure 2. Top-Level Components



Register

- Register untuk eksekusi instruksi
 - Program Counter
 - Instruction Register
- Register untuk Informasi dan Status
 - Sign
 - Zero
 - Carry
 - Equal
 - Overflow
 - Interrupt enable/disable
 - Supervisor

Interrupt

- An interruption of the normal sequence of execution
- Improves processing efficiency
- Allows the processor to execute other instructions while an I/O operation is in progress
- A suspension of a process caused by an event external to that process and performed in such a way that the process can be resumed

Figure 3. Interrupt Cycle

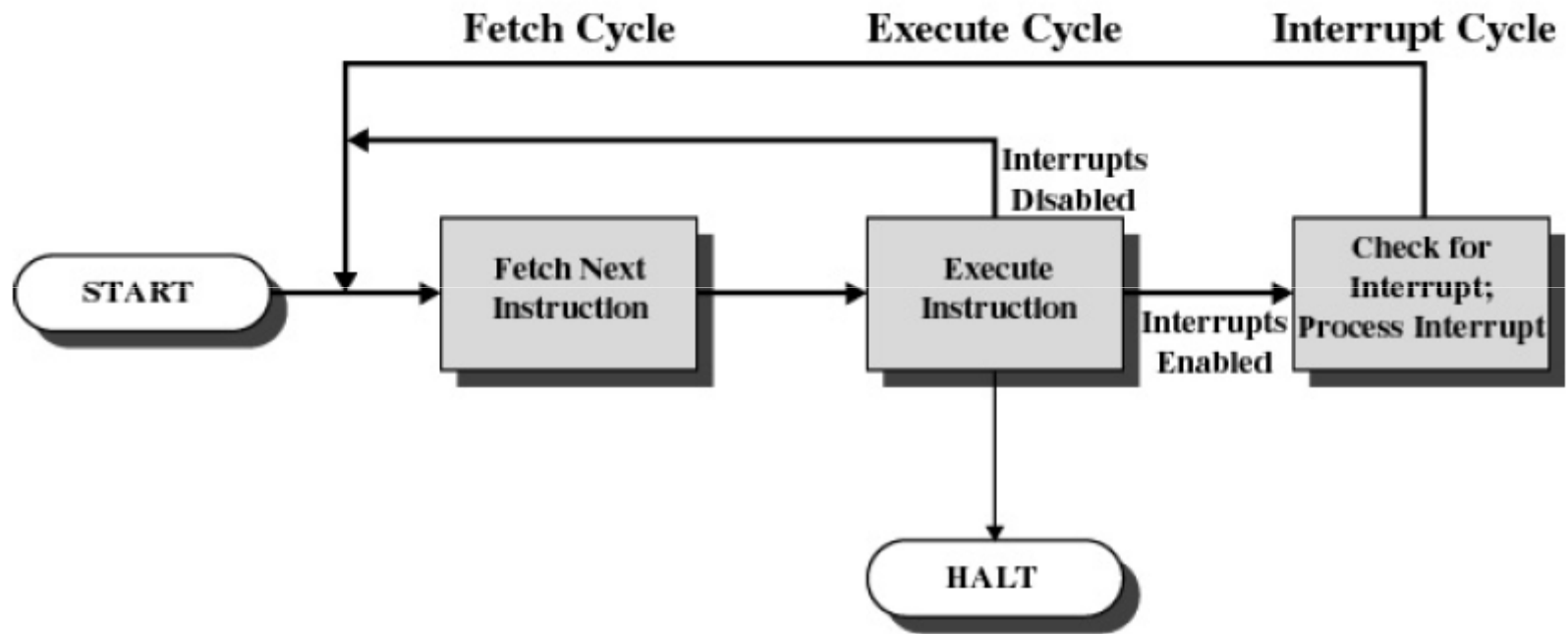
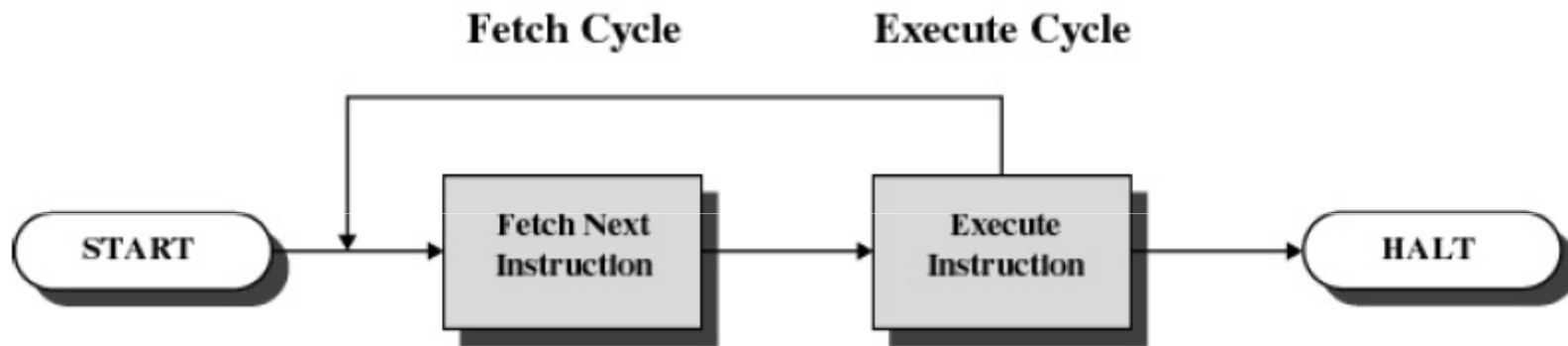


Figure 4. Instructions Cycle



Classes Of Interrupt

- ***Program***
 - arithmetic overflow
 - division by zero
 - execute illegal instruction
 - reference outside user's memory space
- ***Timer***
- ***I/O***
- ***Hardware failure***

Interrupt Handler

- A program that determines nature of the interrupt and performs whatever actions are needed
- Control is transferred to this program
- Generally part of the operating system

Figure 5. Program Flow of Control Without Interrupts

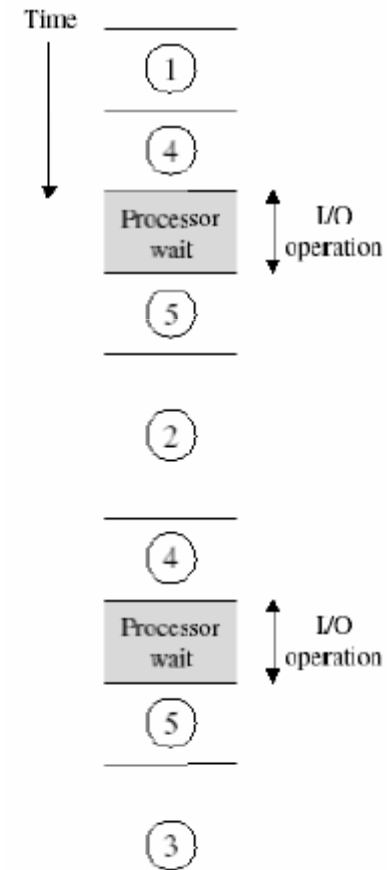
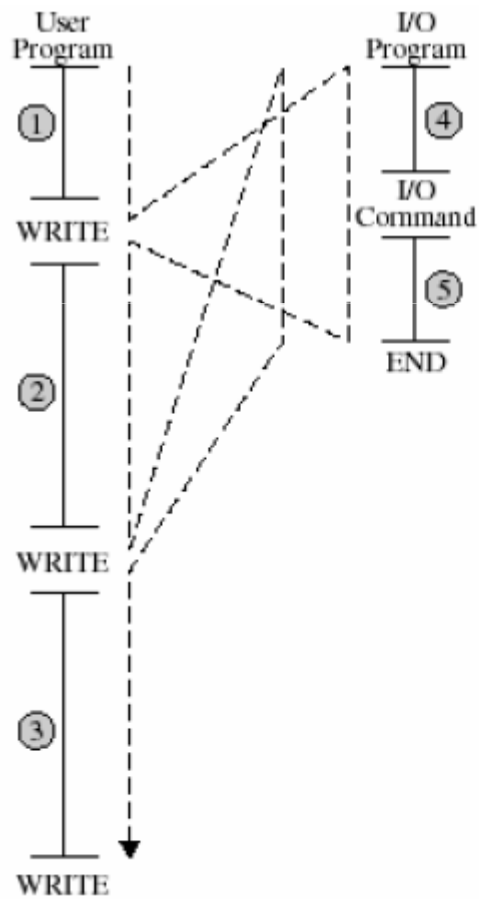
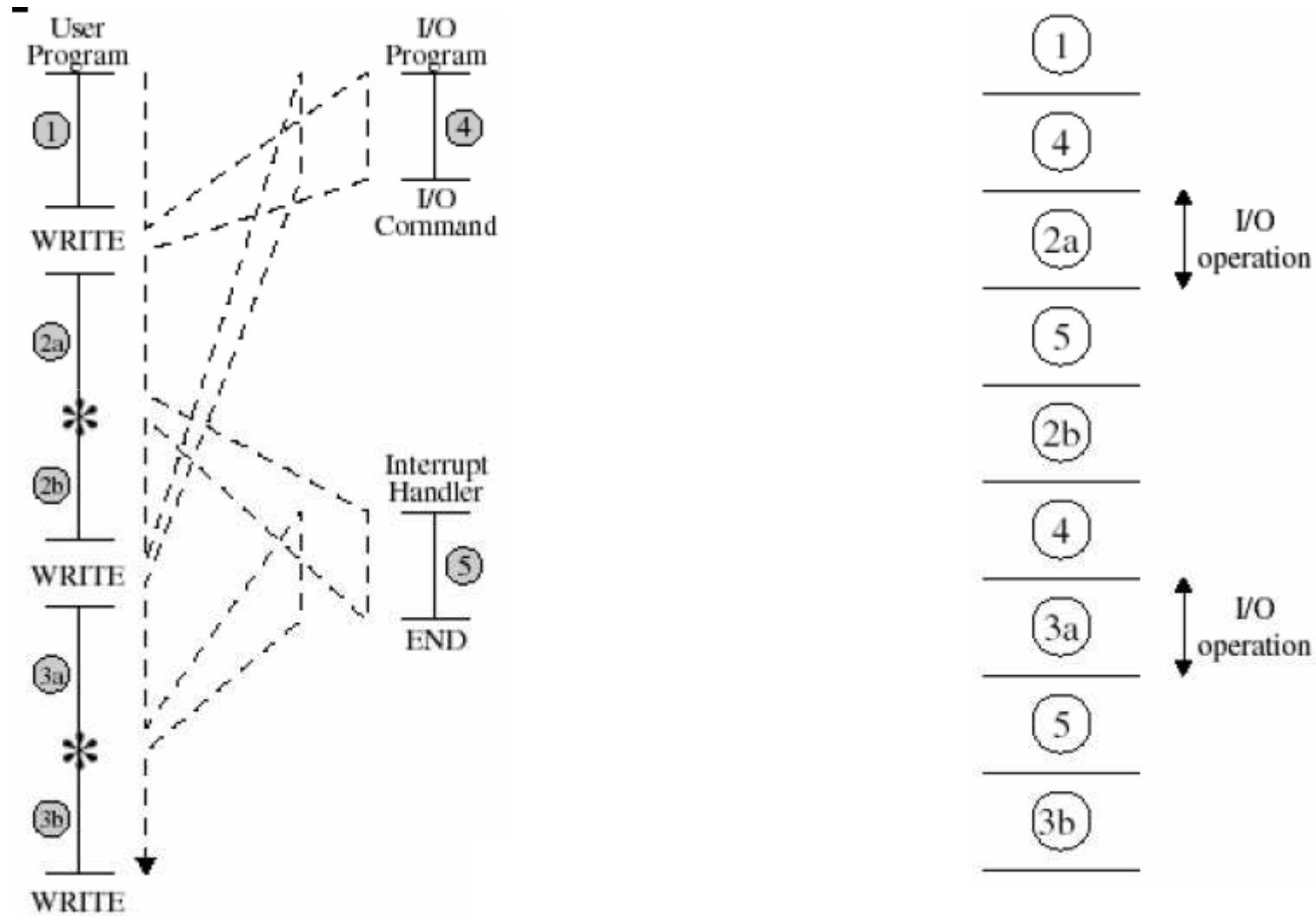
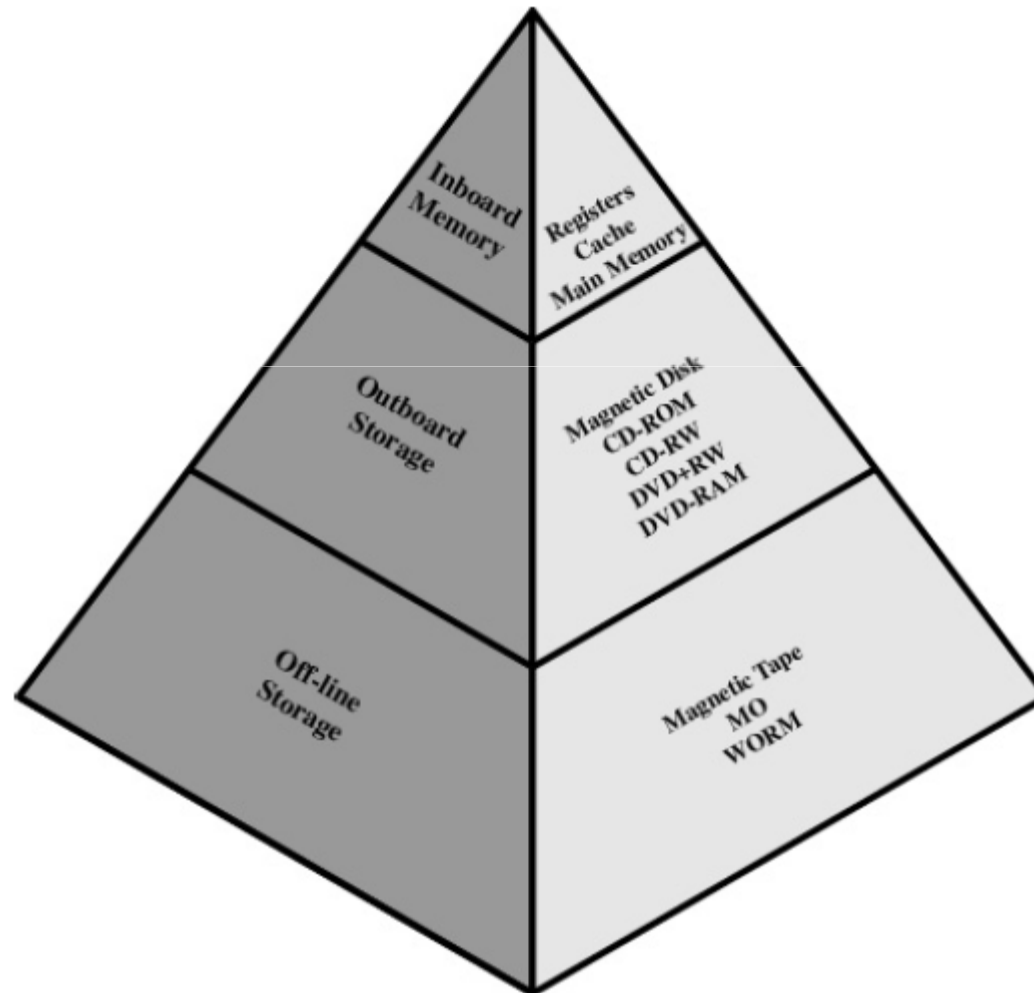


Figure 6. Program Flow of Control Without Interrupts



Memori



Struktur Penyimpanan

- Ukuran memori utama relatif kecil untuk dapat menyimpan data dan program secara keseluruhan.
- Memori utama bersifat *volatile*, tidak bisa menyimpan secara permanen, apabila komputer dimatikan maka data yang tersimpan di memori utama akan hilang.

Going Down the Hierarchy

- Decreasing cost per bit
- Increasing capacity
- Increasing access time
- Decreasing frequency of access of the memory by the processor

Memori

- Memori berfungsi untuk menyimpan data dan program
- Setiap kali pemroses melakukan eksekusi, pemroses melakukan pembacaan instruksi dari memori utama

Konsep Memori 2 level

- **Cache memory,**
memori berkapasitas terbatas dan berkecepatan tinggi. Cache memori ada diantara memori utama dan register pemroses berfungsi agar pemroses tidak langsung mengacu memori utama tetapi di cache memory yang kecepatan aksesnya lebih tinggi. Metode ini meningkatkan kinerja sistem
- **Buffering,**
bagian memori utama untuk menampung data yang akan di trasfer dari/ke perangkat masukan/keluaran dan penyimpan sekunder. Buffering dapat mengurangi frekuensi pengaksesan dari/ke perangkat masukan/keluaran dan penyimpan sekunder sehingga meningkatkan kinerja sistem.

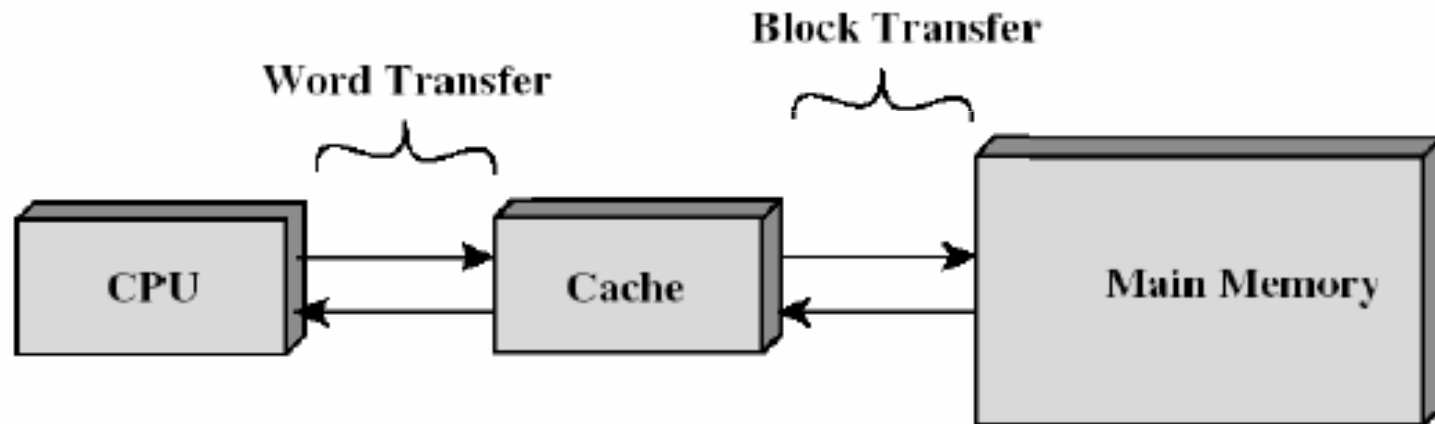
Disk Cache

- A portion of main memory used as a buffer to temporarily to hold data for the disk
- Disk writes are *clustered*
- Some data written out may be referenced again. The data are retrieved rapidly from the software cache instead of slowly from disk

Cache Memory

- Processor speed is faster than memory speed
- Increase the speed of accessing memory
- Invisible to operating system

Figure 7. cache memori



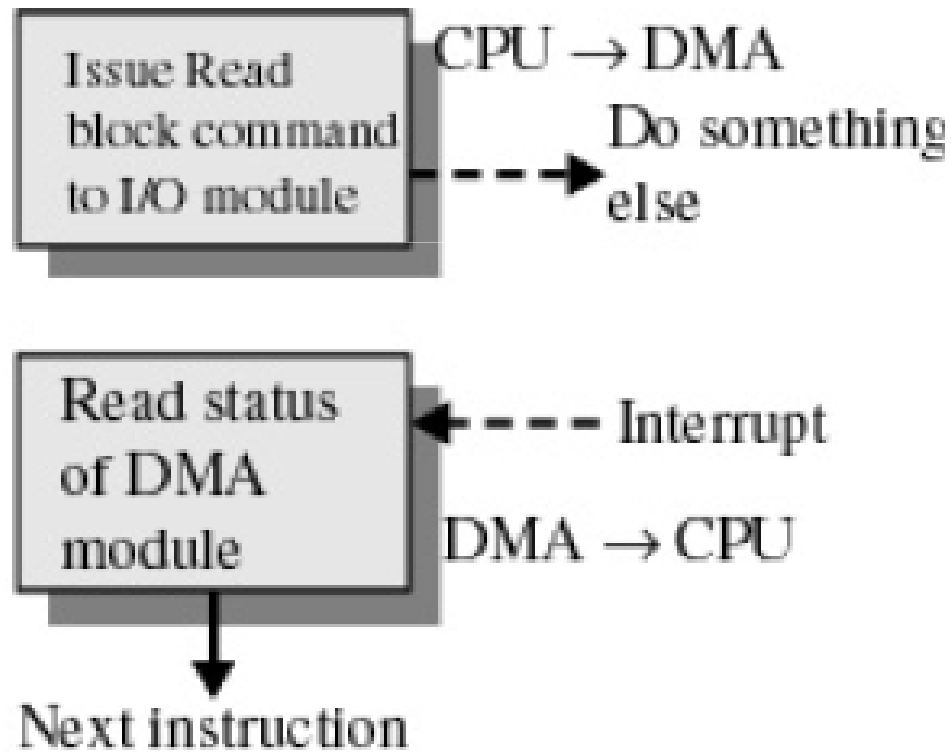
DMA

- Metoda penanganan I/O dimana *device controller* langsung berhubungan dengan memori tanpa campur tangan CPU. Setelah men-set *buffers*, *pointers*, dan *counters* untuk perangkat I/O, *device controller* mentransfer blok data langsung ke penyimpanan tanpa campur tangan CPU. DMA digunakan untuk perangkat I/O dengan kecepatan tinggi. Hanya terdapat satu interupsi setiap blok, berbeda dengan perangkat yang mempunyai kecepatan rendah dimana interupsi terjadi untuk setiap *byte (word)*.

Direct Memory Access (DMA)

- I/O exchanges occur directly with memory
- Processor grants I/O module authority to read from or write to memory
- Relieves the processor responsibility for the exchange
- Processor is free to do other things
- Transfers a block of data directly to or from memory
- An interrupt is sent when the task is complete
- The processor is only involved at the beginning and end of the transfer

Figure 8. DMA



Magnetic Disk

- Berperan sebagai *secondary storage* pada sistem komputer modern. *Magnetic Disk* disusun dari piringan-piringan seperti CD. Kedua permukaan piringan diselimuti oleh bahan-bahan magnetik. Permukaan dari piringan dibagi-bagi menjadi *track* yang memutar, yang kemudian dibagi lagi menjadi beberapa sektor.

I/O Device Components

- Komponen mekanis adalah perangkat itu sendiri
- Komponen elektronik yaitu pengendali perangkat berupa chip controller

How it's works

- Perangkat nyata yang dikendalikan chip controller diboard sistem atau card.
- Controller dihubungkan dengan pemroses dan komponen-komponen lain lewat bus.
- Controller biasanya mempunyai register-register untuk mengendalikannya. Status register berisi status yang mendeskripsikan kode kesalahan.
- Tiap controller dibuat agar dapat dialamati secara individu oleh pemroses sehingga perangkat lunak device driver dapat menulis ke register-registernya dan dengan demikian mengendalikannya.

System Bus

- **Bus alamat (address bus),**
Bus alamat berisi 16,20,24 jalur sinyal paralel atau lebih. CPU mengirimkan alamat lokasi memori atau port yang ingin ditulis atau dibaca pada bus ini. Jumlah lokasi memori yang dilayani ditentukan jumlah jalur alamat. Jika CPU mempunyai N jalur alamat maka dapat mengamati pangkat N (2^N) lokasi memori dan atau port secara langsung. Saat CPU membaca atau menulis data mengenai port, alamat port dikirimkan di bus alamat.
- **Bus data (data bus),**
Bus data berisi 8, 16,32 jalur sinyal paralel atau lebih. Jalur-jalur data adalah dua arah (bidirectional). CPU dapat membaca dan mengirim dari dari/ke memori atau port. Terdapat banyak perangkat yang di tempelkan ke bus data tapi hanya satu perangkat pada satu saat dapat memakainya. Untuk melakukan prosedur ini perangkat harus mempunyai tiga state (tristate) agar bisa dipasang pada bus data

System Bus

– **Bus kendali (control bus),**

Bus ini berisi 4-10 jalur sinyal paralel. CPU mengirimkan sinyal-sinyal pada bus kendali untuk memerintahkan memori atau port. Sinyal bus kendali antara lain :

- Memory read untuk memerintahkan melakukan pembacaan memori
- Memory Write untuk memerintahkan melakukan penulisan memori
- I/O read untuk memerintahkan melakukan pembacaan I/O port
- I/O write untuk memerintahkan melakukan penulisan I/O port

Mode Eksekusi Instruksi

- Program bagian dari SO
- Program pemakai
- Instruksi-instruksi yang memerlukan kewenangan tinggi, antara lain :
 - Membaca atau memodifikasi register kendali (bit-bit register PSW)
 - Instruksi-instruksi primitif perangkat I/O
 - Instruksi-instruksi untuk manajemen memori
 - Bagian memori tertentu hanya dapat diakses dalam mode kewenangan tinggi

Sistem Operasi

Periyadi, ST

Sistem Operasi

- Sistem operasi merupakan sebuah penghubung antara pengguna dari komputer dengan perangkat keras komputer.
- Pengelola seluruh sumber-daya yang terdapat pada system komputer dan menyediakan sekumpulan layanan (*system calls*) ke pemakai sehingga memudahkan dan menyamankan penggunaan serta pemanfaatan sumber-daya sistem komputer .

Sistem Operasi

- Layers of a Computer System
 - Hardware
 - Operating System
 - Utilities
 - Application Programs
- Services typically provided by an O.S.
 - Program Creation (editor, compiler)
 - Program Execution
 - Access to I/O devices
 - Controlled access to files
 - System Access (logging in)
 - Error detection/response
 - Accounting

Sistem Komputer

- Sistem komputer adalah sekumpulan sumber daya untuk memindahkan, menyimpan, dan memproses data, serta mengendalikan fungsi-fungsinya. Sistem operasi bertanggung jawab mengelola sumber daya-sumber daya sistem komputer

- Must manage system resources
 - Kernel – Most frequently used items
- O.S. will evolve over time
 - Hardware upgrades, New services, Fixes

Fungsi

- mengatur dan mengawasi penggunaan perangkat keras oleh berbagai program aplikasi serta para pengguna
- Untuk menghindari konflik yang terjadi pada saat pengguna menggunakan sumber-daya yang sama, sistem operasi mengatur pengguna mana yang dapat mengakses suatu sumber-daya. Sistem operasi juga sering disebut *resource allocator*.
- program pengendali yang bertujuan untuk menghindari kekeliruan (*error*) dan penggunaan komputer yang tidak perlu.

Fungsi

- **Pengelola Sumber Daya Sistem komputer (resource manager)**

SO sebagai EM/VM

- Abstraksi mesin tingkat tinggi yang lebih sederhana dan menyembunyikan kerumitan perangkat keras
- Basis untuk program lain

- **Penyedia layanan (extended/virtual machine)**
 - menyediakan sekumpulan layanan (system calls) ke pemakai sehingga memudahkan dan menyamankan penggunaan atau pemanfaatan sumber daya sistem komputer. Melakukan pemrograman terhadap sumber daya SK

Sasaran So

- Kenyamanan
- Efisiensi
- Mampu berevolusi

- **Sumber daya Fisik**

- Antara lain : Keyboard, bar-code reader ; Mouse, joystick, light-pen, track-ball, touch-screen (pointing devices); Floppy Disk Drive, harddisk, tape drive, optical disk, CD ROM drive (storage device); Layar monitor CRT, LCD (display devices); Printer; Modem, NIC, PCMCIA (communication devices); RAM, cache memory, register dan memori-memori volatile lain; Multimedia devices; Grafis Devices; Device Controls

- **Sumber Daya Abstrak**

- Data
- Program

Fungsi Fungsi Minor SO

- Mengimplementasi interface untuk pemakai
- Memungkinkan pemakaian bersama hardware
- Memungkinkan pemakaian data bersama
- Mencegah users saling mengganggu satu sama lain
- Menjadwalkan pemakaian sumber daya
- Memberi fasilitas I/O
- Memulihkan error
- Menghitung penggunaan sumber daya
- Mengorganisasi data agar pengamanan dan cepaat diakses
- Menangani komunikasi jaringan

OS History

- Serial Processing
 - Sign-up sheet for computer time
 - Each user would set up the job, run it, and collect output
- Simple Batch System (mid 1950s)
 - Try to improve scheduling/setup time
 - Have a **monitor** that loads jobs, when a job finishes it jumps to the monitor that loads the next job.
 - Use JCL (Job Control Language) to submit jobs
 - Desirable hardware items:
 - Memory protection
 - Timer
 - Privileged instructions (I/O)
 - Interrupts

- Multiprogrammed Batch System
 - Several programs in memory at one time
 - One program can do I/O while another computes
 - Needs memory management, scheduling

OS History-Time Sharing System

- Allow several users to interact with the system at the same time
- Emphasizes response time over processor use
- Compatible time sharing system (CTSS) – 1962
 - Machine had 32K 36-bit words
 - Switched users every 0.2 second
 - Supported up to 32 users
- MULTICS (1965)
 - Intended as the computer service for Boston
 - Strong influence on later systems
 - Good security, user interface

- Note systems tend to grow over time as user requirements expand
 - MS-DOS 1.0 (8k, 1981)
 - DOS 2.0 – Subdirectories (1983)
 - DOS 3.1 – Networking (1984)
 - Windows 3.1 – Graphical interface (1990)
 - Windows 95 – 32-bit internals (1995)
- Single-user multitasking (windows 95)

- **Generasi Pertama (1945-1955)**
 - Generasi pertama merupakan awal perkembangan sistem komputasi elektronik sebagai pengganti sistem komputasi mekanik
- **Generasi Kedua (1955-1965)**
 - Generasi kedua memperkenalkan *Batch Processing System*, yaitu Job yang dikerjakan dalam satu rangkaian, lalu dieksekusi secara berurutan
- **Generasi Ketiga (1965-1980)**
 - Pada generasi ini perkembangan sistem operasi dikembangkan untuk melayani banyak pemakai sekaligus, dimana para pemakai interaktif berkomunikasi lewat terminal secara on-line ke komputer, maka sistem operasi menjadi *multi-user* (di gunakan banyak pengguna sekali gus) dan *multi-programming* (melayani banyak program sekaligus).
- **Generasi Keempat (Pasca 1980an)**

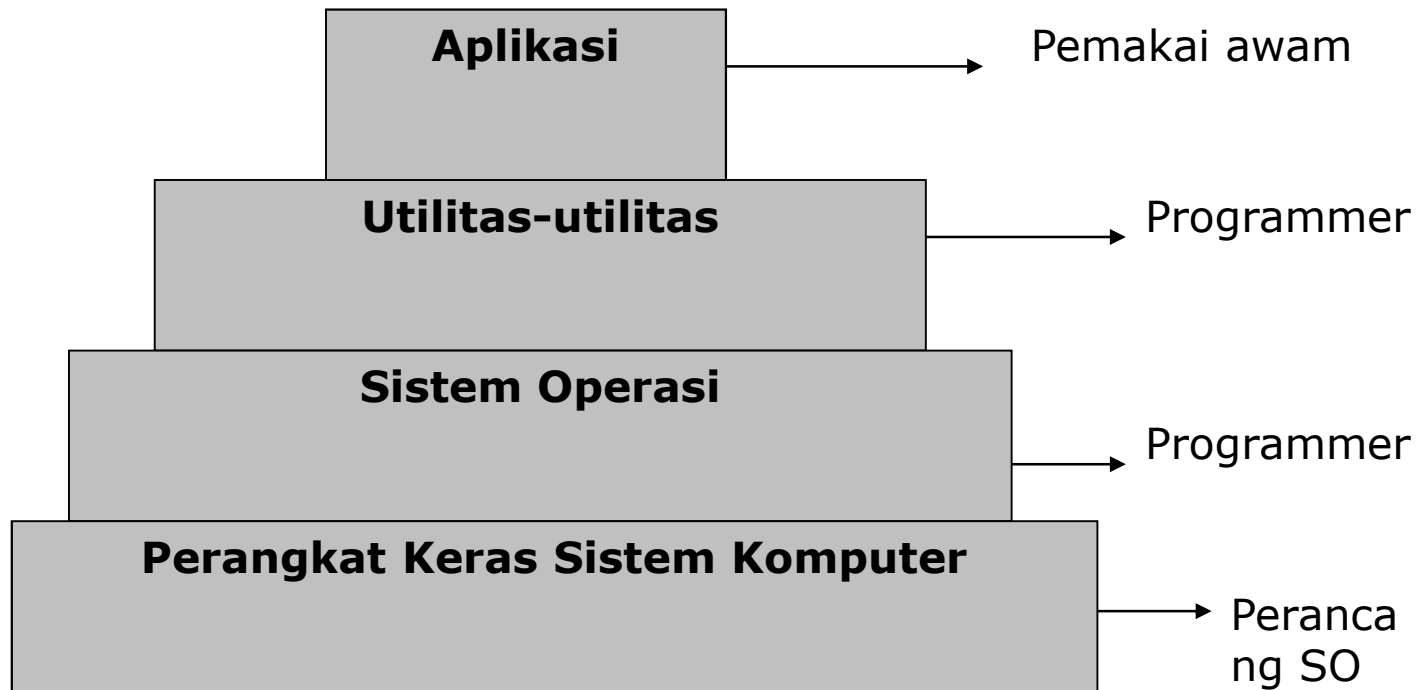
Point of View Computer Sistem

- **Pemakai,**
 - Pemakai tidak ada kaitannya dengan arsitektur komputer. User hanya sebatas menggunakan command language SO untuk meminta layanan-layanan SO. Command language tersedia dilapisan yang disebut **shell**, shell dapat berupa; **Text-based shell dan GUI (Graphical User Interface)**
- **Pemrogram,**
 - programmer membuat aplikasi untuk pemakai awam dengan menggunakan bahasa pemrograman. Jika ini secara langsung dikembangkan dengan intruksi bahasa mesin maka programmer bertanggung jawab mengelola dan mengendalikan seluruh perangkat komputer.

- Programmer dapat mengendalikan sistem komputer melalui beragam level, antara lain :
 - **Mempergunakan program utilitas**, sekumpulan program sistem disediakan untuk untuk mempermudah tugas pemrograman. Kumpulan ini disebut utilitas, utilitas sudah menjadi ssatu paket dengan sistem operasi yang akan diterjemahkan menjadi layanan-layanan SO. Utilitas yang bisa disediakan adalah :
 - Utilitas untuk membantu penciptaan program
 - Utilitas untuk manajemen berkas
 - Utilitas untuk mengendalikan perangkat I/O
 - Utilitas-utilitas untuk tugas dasar lainnya
 - **Mempergunakan fasilitas sistem melalui interface layanan (service interface)**
 - **Mempergunakan panggilan sistem (system calls)**

- **Perancang SO,**

SO bertugas untuk me-*masking* perangkat keras komputer agar tampil menarik, mudah dan nyaman bagi programmer. SO menghindari rincian operasi perangkat keras dan menyediakan antarmuka untuk programmer dalam menggunakan sistem. SO bertindak sebagai mediator, mempermudah programmer daaan program aplikasi mengakses dan menggunakan fasilitas dan sumber daya sistem komputer.



Manajemen Proses

Periyadi,ST
Operating System

PROSES

- Proses adalah entitas dinamis. Proses berisi instruksi dan data. program counter dan semua register pemroses, dan stack berisi data sementara seperti parameter rutin, alamat pengiriman dan variabel-variabel lokal.
- Salah satunya adalah program yang sedang dieksekusi yang merupakan unit kerja terkecil yang secara individu memiliki sumber daya-sumber daya dan dijadwalkan sistem operasi. Sistem operasi mengelola semua proses di sistem dan mengalokasikan sumber daya ke proses-proses sesuai kebijaksanaan untuk memenuhi sasaran sistem.

Istilah berkaitan dengan proses

- *Multiprogramming (multitasking).*
- *Multiprocessing.*
- *Distributed processing/computing.*

Multiprogramming (multitasking).

- Manajemen banyak proses pada satu pemroses. Istilah yang digunakan multiprogramming (multitasking) bukan multiprocessing.
- Program-program yang dijalankan sebenarnya bersifat:
 - Saling tak bergantung (independen).
 - Satu program pada satu saat (one program at any instant).

Multiprocessing

- Manajemen banyak proses di komputer multiprocessor (banyak pemroses didalamnya).

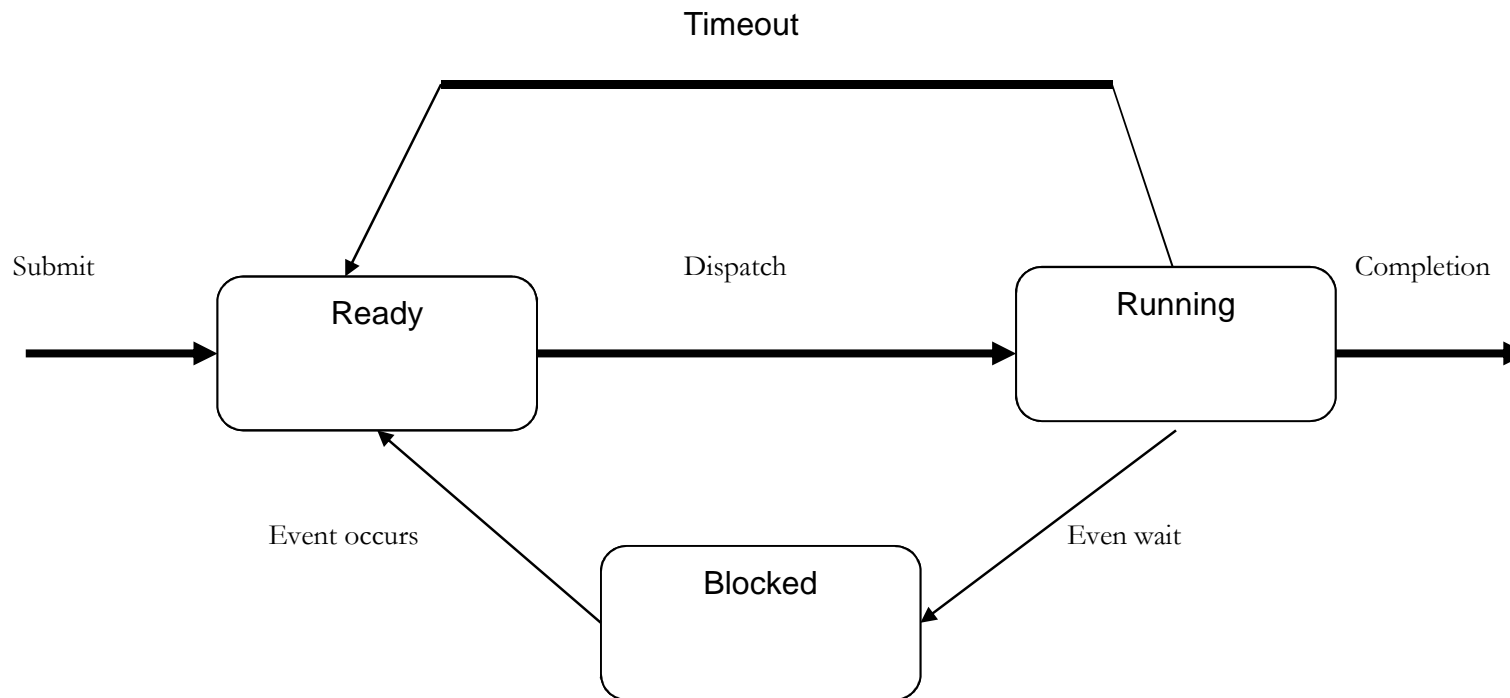
Distributed Processing

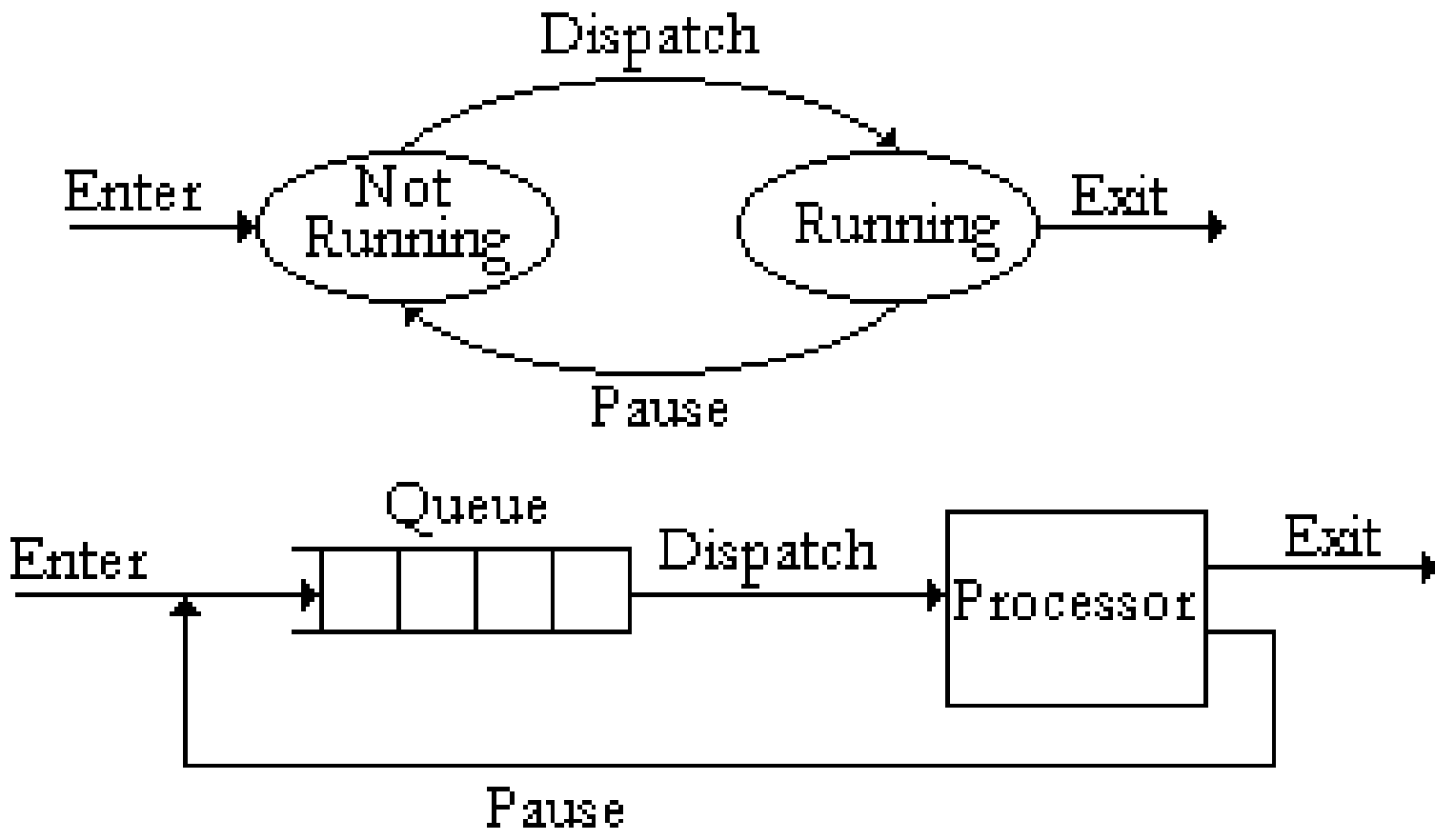
- Manajemen banyak proses yang dieksekusi di banyak sistem komputer yang tersebar (terdistribusi).

Kebutuhan utama pengendalian proses oleh sistem operasi yang mengacu ke proses, yaitu: [STA-95]

- Saling melanjutkan (Interleave)
- Mengikuti kebijaksanaan tertentu.
- Mendukung komunikasi antar proses dan penciptaan proses.

Diagram State 3 Keadaan





Tabel Tiga State Dasar Proses

Status	Deskripsi
Running	Pemroses sedang mengeksekusi instruksi proses itu
Ready	Proses siap (ready) dieksekusi, tapi tidak tersedia untuk eksekusi proses ini
Blocked	Proses menunggu kejadian untuk melengkapinya tugasnya Contoh : Proses menunggu <ul style="list-style-type: none">▪ Selesainya operasi perangkat I/O▪ Tersedianya memori▪ Tibanya pesan jawaban▪ Dsb.

Siklus hidup proses.

- Proses yang baru diciptakan akan segera mempunyai state ready.
- Proses berstate running menjadi blocked karena sumber daya yang diminta belum tersedia atau meminta layanan perangkat masukan/keluaran sehingga menunggu kejadian muncul. Proses menunggu kejadian alokasi sumber daya atau selesainya layanan perangkat masukan/keluaran (event wait).
- Proses berstate running menjadi ready karena penjadwal memutuskan eksekusi proses lain karena jatah waktu untuk proses tersebut telah habis (time-out).

Siklus hidup proses.

- Proses berstate blocked menjadi ready saat sumber daya yang diminta/diperlukan telah tersedia atau layanan perangkat masukan/keluaran selesai (event occurs).
- Proses berstate ready menjadi running karena penjadwal memutuskan penggunaan pemroses untuk proses itu karena proses yang saat itu running berubah statenya (menjadi ready atau blocked) atau telah menyelesaikan sehingga disingkirkan dari sistem Proses menjadi mendapatkan jatah pemroses.

PCB (Program Control Block)

- Sistem operasi memerlukan banyak informasi mengenai proses guna pengelolaan proses, Informasi ini berada di PCB
 - Informasi identifikasi proses.
 - Informasi status pemroses.
 - Informasi kendali proses.

Elemen PCB

Identifikasi Process
Identififier
Identififier numerik meliputi <ul style="list-style-type: none">▪ Identififier proses▪ Identififier proses yang menciptakan▪ Identififier pemakai
Informasi Status Pemroses
Register-register yang terlihat pemakai
Register-register yang dapat ditunjuk instruksi bahasa assembly untuk diproses pemroses
Register-register kendali dan status
Register-register yang digunakan untuk mengendalikan operasi proses, antara lain : <ul style="list-style-type: none">▪ Program counter▪ PSW▪ Dsb
Pointer Stack
Tiap proses mempunyai satu stack atau lebih. Stack digunakan untuk parameter atau alamat prosedur pemanggil dan system call. Ponter stack menunjuk posisi paling atas dari stack.

Elemen PCB

Informasi Kendali Proses

Informasi penjadwakan dan status

Informasi-informasi yang digunakan untuk menjalankan fungsi penjadwalan antara lain :

- Status proses : mendefinisikan keadaan/status proses (running, ready, blocked, dsb)
- Prioritas : menjelaskan prioritas proses
- Informasi berkaitan dengan penjadwalan : berkaitan dengan informasi penjadwalan seperti lama menunggu, lama proses terakhir dieksekusi, dsb.
- Kejadian : identitas kejadian yang ditunggu proses.

Penstrukturan data

Satu proses dapat dikaitkan dengan proses lain dalam satu antrian atau ring, atau struktur lainnya. PCB harus memiliki pointer untuk mendukung struktur ini.

Komunikasi antar proses

Beragam flag, sinyal dan pesan dapat diasosiasikan dengan komunikasi antara dua proses yang terpisah. Informasi ini disimpan dalam PCB

Elemen PCB

Kewenangan proses

Proses dapat mempunyai kewenangan berkaitan dengan memori dan tipe instruksi yang dapat dijalankan

Manajemen memori

Bagian ini berisi pointer ke tabel segmen atau page yang menyatakan memori maya (virtual memori) proses.

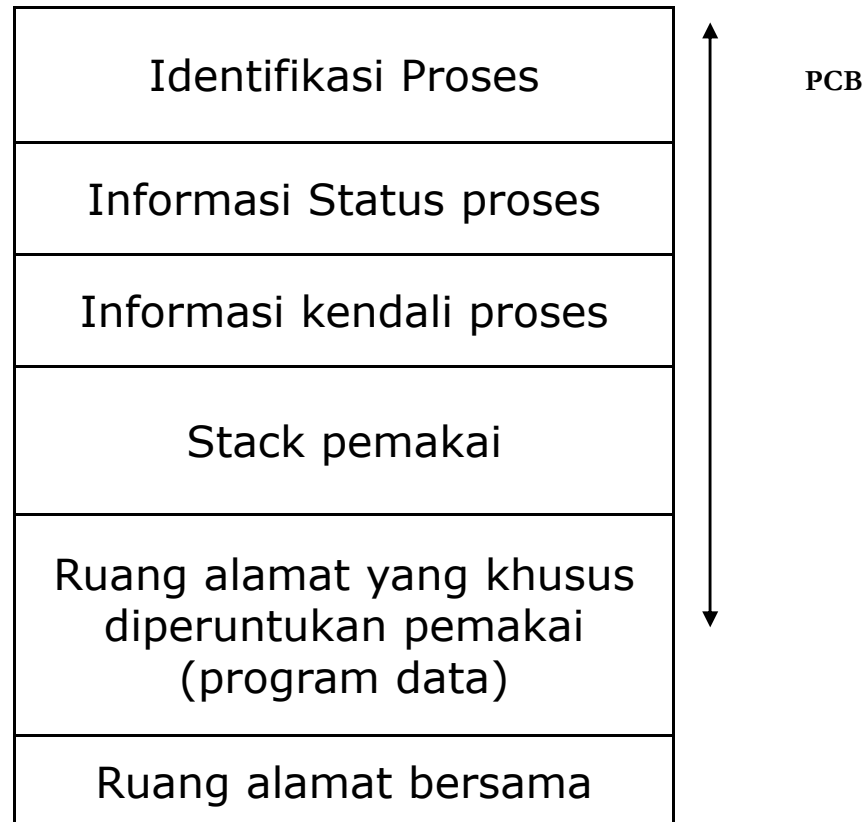
Kepemilikan dan utilisasi sumber daya

Sumber daya yang dikendalikan proses harus diberi tanda, misalnya :

- Berkas yang dibuka
- Pemakaian pemroses
- Pemakaian sumber daya lainnya

Informasi ini diperlukan oleh penjadwal

Proses Pemakai



Citra proses

- mempunyai struktur berisi PCB, stack pemakai (user stack), ruang alamat proses eksklusif dan ruang alamat yang dipakai bersama proses lain. Struktur ini ditunjukkan pada tabel sebelumnya. Pada tabel diatas, struktur citra proses digambarkan kontigu (berturutan) di satu ruang alamat. Implementasi penempatan citra proses yang sesungguhnya bergantung skema manajemen memori yang digunakan dan organisasi struktur kendali sistem operasi.

Operasi-operasi pada Proses

- Penciptaan proses (create a process).
- Penghancuran/terminasi proses (destroy a process).
- Penundaan proses (suspend a process).
- Pelanjutan kembali proses (resume a process).
- Pengubahan prioritas proses.
- Mem-block proses.
- Membangunkan proses.
- Menjadwalkan proses.
- Memungkinkan proses berkomunikasi dengan proses lain.

Penciptaan Proses

- Penciptaan proses melibatkan banyak aktivitas, yaitu:
- Menamai (memberi identitas) proses.
- Menyisipkan proses pada senarai proses atau tabel proses.
- Menentukan prioritas awal proses.
- Menciptakan PCB.
- Mengalokasikan sumber daya awal bagi proses.

Kejadian yang dapat menyebabkan penciptaan proses, antara lain: [STA-95]

- Pada lingkungan batch, sebagai tanggapan atas pemberian satu kerja (job).
- Pada lingkungan interaktif, ketika pemakai baru berusaha logon.
- Sebagai tanggapan suatu aplikasi, seperti permintaan pencetakan file, sistem operasi dapat menciptakan proses yang akan mengelola pencetakan itu.
- Proses penciptaan proses lain (proses anak).

alasan-alasan penciptaan proses.

Penyebab Penciptaan	Deskripsi
Terdapat batch baru	SO dengan kendali batch job, setelah menciptakan proses baru, kemudian melanjutkan membaca job selanjutnya
Satu pemakai interaktif logon	Seorang pemakai pada satu terminal sedang melakukan logon ke sistem
Sistem operasi menciptakan proses untuk memberi layanan	SO menciptakan proses untuk memenuhi satu fungsi pada program pemakai, tanpa mengharuskan pemakai menunggu
Proses menciptakan proses anak	Untuk mencapai modularitas atau mengeksploitasi kongkurensi, program pemakai memerintahkan pembuatan sejumlah proses.

- Proses dapat menciptakan proses baru yaitu proses anak (child process). Proses yang menciptakan proses disebut proses induk (parent process). Proses anak-pun kembali dapat menciptakan proses-proses anak. Proses-proses dapat membentuk pohon hirarki proses.

Penghancuran Proses

- Penghancuran proses melibatkan pembebasan proses dari sistem, yaitu:
- Sumber daya-sumber daya yang dipakai dikembalikan .
- Proses dihancurkan dari senarai atau tabel sistem
- PCB dihapus (ruang memori PCB dikembalikan ke pool memori bebas)

Penghancuran lebih rumit bila proses telah menciptakan proses-proses lain. Terdapat dua pendekatan, yaitu:

- Pada beberapa sistem, proses-proses turunan dihancurkan saat proses induk dihancurkan secara otomatis.
- Beberapa sistem lain menganggap proses anak independen terhadap proses induk. Proses anak tidak secara otomatis dihancurkan saat proses induk dihancurkan.

Alasan-alasan Penghancuran proses

Selesainya proses secara normal	Proses mengeksekusi panggilan layanan So untuk menandakan bahwa proses telah berjalan secara lengkap.
Batas waktu telah terlewati	Proses telah berjalan melebihi batas waktu total yang dispesifikasikan. Terdapat banyak kemungkinan untuk tipe waktu yang diukur, termasuk waktu total yang dijalani ("walk clock time") jumlah waktu yang dipakai untuk eksekusi, dan jumlah waktu sejak pemakai terakhir kali memberi masukan (pada proses interaktif) .
Memori tidak tersedia	Proses memerlukan memori lebih banyak daripada yang dapat disediakan oleh sistem.
Pelanggaran terhadap batas memori	Proses mencoba mengakses lokasi memori yang tidak diijinkan untuk diakses

Alasan-alasan Penghancuran proses

Terjadi kesalahan karena pelanggaran proteksi	Proses berusaha menggunakan sumber daya atau file yang tidak diijinkan dipakainya, atau proses mencoba menggunakannya tidak untuk peruntukannya, seperti menulis file read only
Terjadi kesalahan aritmatika	Proses mencoba perhitungan terlarang, seperti pembagian dengan nol, atau mencoba menyimpan angka yang lebih besar daripada yang dapat diakomodasi oleh H/W
Waktu telah kadaluwarsa	Proses telah menunggu lebih lama daripada maksimum yang telah ditentukan untuk terjadinya suatu kejadian spesifik
Terjadi kegagalan masukan/keluaran	Kesalahan muncul pada masukan atau keluaran, seperti ketidakmampuan menemukan file, kegagalan membaca atau menulis setelah sejumlah maksimum percobaan yang ditentukan (misalnya area rusak didapatkan pada tape, atau operasi tidak valid seperti membaca dari line printer)

Alasan-alasan Penghancuran proses

Intruksi yang tidak benar	Proses berusaha mengeksekusi instruksi yang tidak ada (sering sebagai akibat pencabangan ke daerah data dan berusaha mengeksekusi data tersebut)
Terjadi usaha memakai instruksi yang tidak diijinkan	Proses berusaha mengeksekusi instruksi yang disimpan untuk So
Kesalahan penggunaan data	Bagian data adalah tipe yang salah atau tidak diinisialisasi
Diintervensi oleh SO atau operator	Untuk suatu alasan, operator atau sistem operasi mengakhiri proses (misalnya terjadi deadlock)
Berakhirnya proses induk	Ketika parent berakhir. So mungkin dirancannng secara otomatis mengakhiri semua anak proses dari parent itu
Atas permintaan proses induk	Parent process biasanya mempunyai otoritas mengakhiri suatu anak proses

Diagram State Lanjut (Lima Keadaan)

- Penundaan (suspension) adalah operasi penting dan telah diterapkan dengan beragam cara.
- Penundaan biasanya berlangsung singkat dan sering dilakukan sistem untuk memindahkan proses-proses tertentu guna mereduksi beban sistem selama beban puncak.
- Proses yang ditunda (suspended process) tidak berlanjut sampai proses lain me-resume. Untuk jangka panjang, sumber daya-sumber daya proses dibebaskan.

- Memori utama seharusnya segera dibebaskan begitu proses tertunda agar dapat dimanfaatkan proses lain.
- Resuming (pengaktifan kembali) proses yaitu menjalankan proses dari titik (instruksi) dimana proses ditunda.

Operasi suspend dan resume penting, sebab:

- Jika sistem berfungsi secara buruk dan mungkin gagal maka proses-proses dapat di-suspend agar di-resume setelah masalah diselesaikan.
- Pemakai yang ragu/khawatir mengenai hasil proses dapat men-suspend proses [bukan membuang (abort) proses]. Saat pemakai yakin proses akan berfungsi secara benar maka dapat me-resume (melanjutkan kembali di instruksi saat di-suspend) proses yang di-suspend.
- Sebagai tanggapan terhadap fluktuasi jangka pendek beban sistem, beberapa proses dapat di-suspend dan di-resume saat beban kembali ke tingkat normal.

- Penundaan dapat diinisialisasi oleh proses itu sendiri atau proses lain
- Pada sistem monoprocessor, proses running dapat men-suspend dirinya sendiri karena tidak ada proses lain yang juga running yang dapat memerintahkan suspend.
- Pada sistem multiprocessor, proses running dapat di-suspend proses running lain pada pemroses berbeda. Proses ready hanya dapat di-suspend oleh proses lain.

- Pada proses blocked terdapat transisi menjadi suspendedblocked. Pilihan ini dirasa aneh. Apakah tidak cukup menunggu selesainya operasi masukan/keluaran atau kejadian yang membual proses ready atau suspendedready? Bukankah state blocked, readyblocked, suspendedblocked sama-sama tidak mendapatjajah waktu pemroses ?Kenapa dibedakan ?

Jawaban

- Karena penyelesaian operasi masukan/keluaran bagi proses blocked mungkin tak pernah terjadi atau dalam waktu tak terdefinisikan sehingga lebih baik di-suspend agar sumber daya-sumber daya yang dialokasikan untuk proses tersebut dapat digunakan proses-proses lain.
- Proses blocked di-suspend sistem atau secara manual menjadi suspendedblocked. Bila akhirnya operasi masukan/keluaran berakhir maka segera proses suspendedblocked mengalami transisi. Karena resume dan suspend mempunyai prioritas tinggi maka transisi segera dilakukan. Suspend dan resume dapat digunakan untuk menyeimbangkan beban sistem saat mengalami lonjakan di atas normal.

Diagram State Lanjut

