

PERTEMUAN :

STRUKTUR SISTEM KOMPUTER

A. TUJUAN PEMBELAJARAN

Pada bab ini akan dijelaskan mengenai struktur sistem operasi, Anda harus mampu:

- 1.1 Penyebab data hilang
- 1.2 Intruder
- 1.3 Membedakan keamanan sistem dan proteksi

B. URAIAN MATERI

Tujuan Pembelajaran 1.1:

Struktur Sistem Operasi

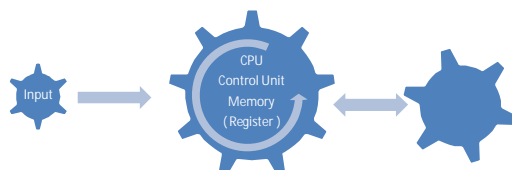
Struktur Komputer

Struktur sebuah sistem komputer dapat dibagi menjadi:

1. Sistem Operasi Komputer.
2. Struktur I/O.
3. Struktur Penyimpanan.
4. Storage Hierarchy.
5. Proteksi Perangkat Keras.

Sistem Operasi Komputer

Ingat Konsep Dasar Sistem Komputer



Struktur I/O

Bagian ini akan membahas struktur I/O, interupsi I/O, dan DMA, serta perbedaan dalam penanganan interupsi.

Interupsi I/O

Untuk memulai operasi I/O, CPU me-load register yang bersesuaian ke device controller. Sebaliknya device controller memeriksa isi register untuk kemudian menentukan operasi apa yang harus dilakukan. Pada saat operasi I/O dijalankan ada dua kemungkinan, yaitu synchronous I/O dan asynchronous I/O.

Struktur DMA

Direct Memory Access (DMA) suatu 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.

Struktur Penyimpanan

Program komputer harus berada di memori utama (biasanya RAM) untuk dapat dijalankan. Memori utama adalah satu-satunya tempat penyimpanan yang dapat diakses secara langsung oleh prosesor.

Memori Utama

Hanya memori utama dan register merupakan tempat penyimpanan yang dapat diakses secara langsung oleh prosesor. Oleh karena itu instruksi dan data yang akan dieksekusi harus disimpan di memori utama atau register.

Magnetic Disk

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.

Storage Hierarchy

Dalam storage hierarchy structure, data yang sama bisa tampil dalam level berbeda dari system penyimpanan. Sebagai contoh integer A berlokasi pada bekas B yang ditambahkan 1, dengan asumsi bekas B terletak pada magnetic disk. Operasi penambahan diproses dengan pertama kali mengeluarkan operasi I/O untuk menduplikat disk block pada A yang terletak pada memori utama. Operasi ini diikuti dengan kemungkinan penduplikatan A ke dalam cache dan penduplikatan A ke dalam internal register.

Proteksi Perangkat Keras

Sistem komputer terdahulu berjenis programmer-operated systems. Ketika komputer dioperasikan dalam konsol mereka (pengguna) harus melengkapi sistem terlebih dahulu. Akan tetapi setelah sistem operasi lahir maka hal tersebut diambil alih oleh sistem operasi. Pengertian multi programming adalah kegiatan menjalankan beberapa program pada memori pada satu waktu.

Pembagian ini memang menguntungkan sebab banyak proses dapat berjalan pada satu waktu akan tetapi mengakibatkan masalah-masalah baru. Ketika tidak di sharing maka jika terjadi kesalahan hanyalah akan membuat kesalahan program. Tapi jika di-sharing jika terjadi kesalahan pada satu proses/ program akan berpengaruh pada proses lainnya.

.

Operasi Dual Mode

Untuk memastikan operasi berjalan baik kita harus melindungi sistem operasi, program, dan data dari program-program yang salah. Proteksi ini memerlukan share resources. Hal ini bisa dilakukan system operasi dengan cara menyediakan pendukung perangkat keras yang mengizinkan kita membedakan mode pengeksekusian program.

Mode yang kita butuhkan ada dua mode operasi yaitu:

- Mode Monitor.
- Mode Pengguna.

Pada perangkat keras akan ada bit atau Bit Mode yang berguna untuk membedakan mode apa yang sedang digunakan dan apa yang sedang dikerjakan. Jika Mode Monitor maka akan bernilai 0, dan jika Mode Pengguna maka akan bernilai 1.

Proteksi I/O

Pengguna bisa mengacaukan sistem operasi dengan melakukan instruksi I/O ilegal dengan mengakses lokasi memori untuk sistem operasi atau dengan cara hendak melepaskan diri dari prosesor.

Proteksi Memori

Salah satu proteksi perangkat keras ialah dengan proteksi memori yaitu dengan pembatasan penggunaan memori. Disini diperlukan beberapa istilah yaitu:

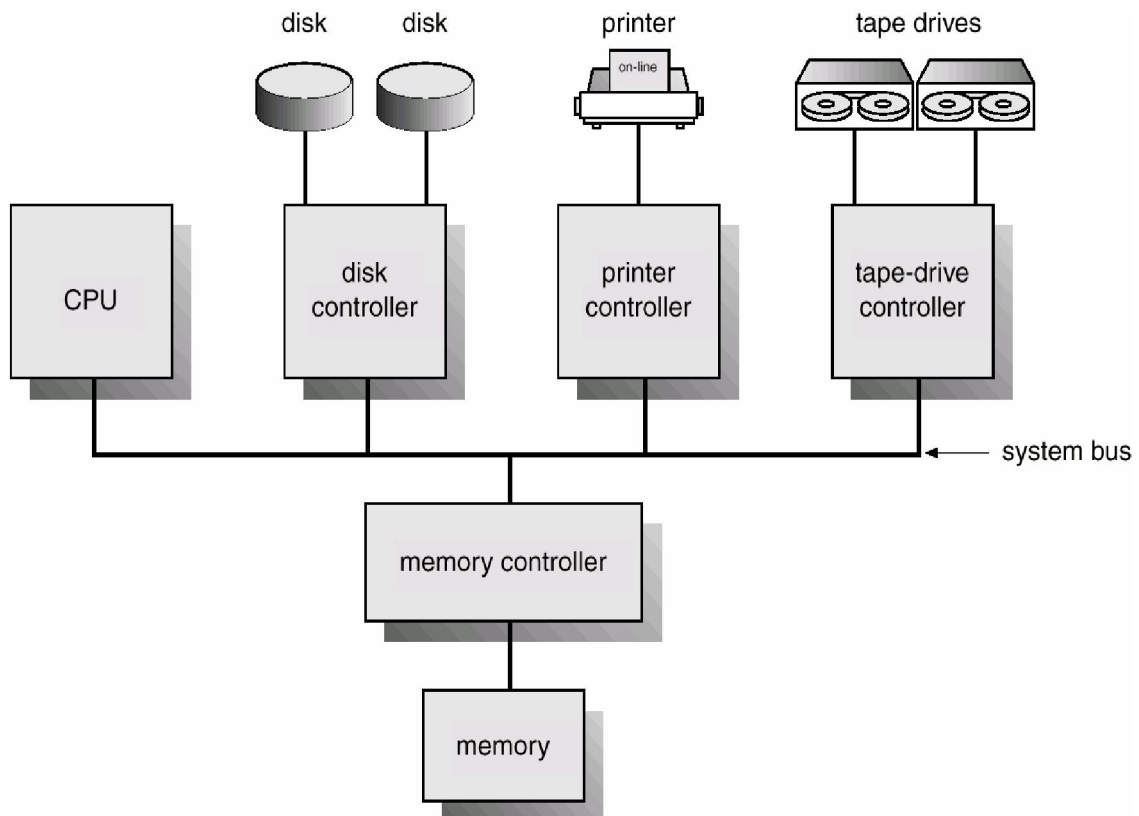
1. Base Register yaitu alamat memori fisik awal yang dialokasikan/ boleh digunakan oleh pengguna.
2. Limit Register yaitu nilai batas dari alamat memori fisik awal yang dialokasikan/boleh digunakan oleh pengguna.
3. Proteksi Perangkat Keras.

Sebagai contoh sebuah pengguna dibatasi mempunyai base register 300040 dan mempunyai limit register 120900 maka pengguna hanya diperbolehkan menggunakan alamat memori fisik antara 300040 hingga 420940 saja.

STRUKTUR SISTEM KOMPUTER

- Operasi Sistem Komputer.
- Struktur I/ O.
- Struktur Storage.
- Proteksi Hardware.

ARSITEKTUR SISTEM KOMPUTER



Gambar arsitektur pada sistem komputer.

OPERASI SISTEM KOMPUTER

- CPU devices dan I/O dapat beroperasi secara serentak (concurrent).
 - Menjadikan efisiensi dalam pemakaian CPU.
- Semua request ke I/ O dikendalikan oleh I/ O systems :
 - Setiap device terdapat controller yang mengendalikan device tertentu, misalkan video display => video card, disk => disk controller.
 - Setiap device controller mempunyai local buffer.
- CPU memindahkan data dari/ke memory ke/dari local buffer.
 - Setelah itu controller akan mengirimkan data dari buffer ke device.
- Bagaimana mekanisme I/O supaya CPU dapat melakukan switch dari satu job ke job lain?

Operasi Sistem Komputer (Cont.)

- Ilustrasi :
 - Instruksi CPU dalam orde : beberapa mikro-detik
 - Operasi read/ write dari disk : 10 – 15 mili-detik.
 - Ratio : CPU ribuan kali lebih cepat dari operasi I/O. Jika CPU harus menunggu (idle) sampai data transfer selesai, maka utilisasi CPU sangat rendah (lebih kecil 1%).
- Solusi : operasi CPU dan I/O harus overlap
 - Concurrent : CPU dapat menjalankan beberapa I/O device sekaligus.
 - CPU tidak menunggu sampai operasi I/O selesai tapi melanjutkan tugas yang lain.
 - Bagaimana CPU mengetahui I/ O telah selesai?

PROGRAMMED I/O (1)

- PROGRAMMED I/O
 - Yaitu mekanisme CPU yang bertanggung jawab memindahkan data dari/ ke memori ke/dari controller.
- CPU bertanggung jawab untuk jenis operasi I/ O :
 - Transfer data dari/ ke buffer.
- Controller melakukan detail operasi I/ O
 - Jika telah selesai memberikan informasi ke CPU => flag
- Bagaimana CPU mengetahui operasi telah selesai?
 - Apakah menguji flag? Seberapa sering?

PROGRAMMED I/O (2)

- CPU harus mengetahui jika I/O telah selesai => hardware flag (controller).
- Polling : CPU secara periodik menguji flag (true or false)
 - Menggunakan instruksi khusus untuk menguji flag
 - Masalah : seberapa sering? “wasted CPU time !”? Antar I/ O device berbeda “speed”!
- Interrupt :
 - Bantuan hardware : melakukan interupsi pada CPU jika flag tersebut telah di-set (operasi I/O telah selesai).

INTERRUPT

- Interrupt:
 - CPU transfer control ke “interrupt service routine”, => address dari service routine yang diperlukan untuk device tsb.
 - Interrupt handler : menentukan aksi/ service yang diperlukan.
- Struktur interrupt harus menyimpan address dari instruksi yang sedang dikerjakan oleh
 - CPU (interrupted). CPU dapat resume ke lokasi tersebut jika service routine telah selesai dikerjakan.
- Selama CPU melakukan service interrupt, maka interrupt selanjutnya tidak akan dilayani “disabled”, karena CPU tidak dapat melayani interrupt (lost).
- Pengoperasian sistem tersebut menggunakan interrupt driven.

INTERRUPT HANDLING

- Hardware dapat membedakan devices mana yang melakukan interupsi.
 - Jenis interupsi :
 - ü polling
 - ü vectored interrupt system
- Tugas sistim operasi menyimpan status CPU (program counter, register dll).
 - Jika service routine telah selesai => CPU dapat melanjutkan instruksi terakhir yang dikerjakan.
 - Sistim operasi akan “load” kembali status CPU tersebut

STRUKTUR I/ O

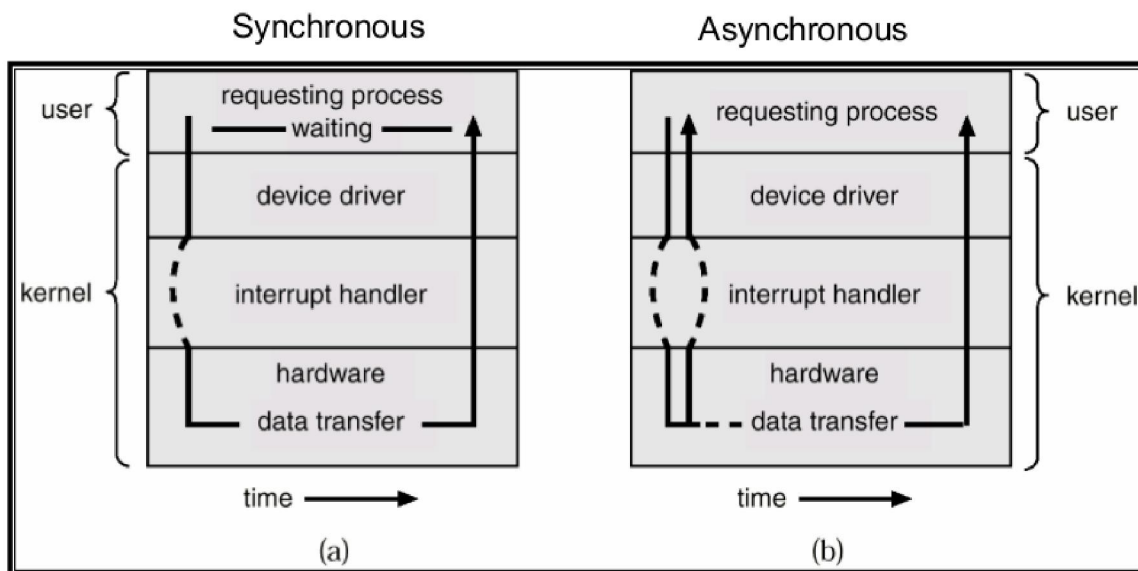
- User request I/O :
 - CPU : load instruksi ke register controller
 - Controller : menjalankan instruksi
- Setelah I/ O mulai, control kembali ke user program jika operasi I/O telah selesai
 - Instruksi khusus : wait => CPU menunggu sampai ada interrupt berikutnya dari I/O
 - tersebut.
 - Paling banyak hanya mempunyai satu I/O request.
 - Keuntungan : CPU mengetahui secara pasti device mana yang melakukan interrupt (operasi I/O selesai).

- Kerugian : operasi I/O tidak dapat serentak untuk semua device.

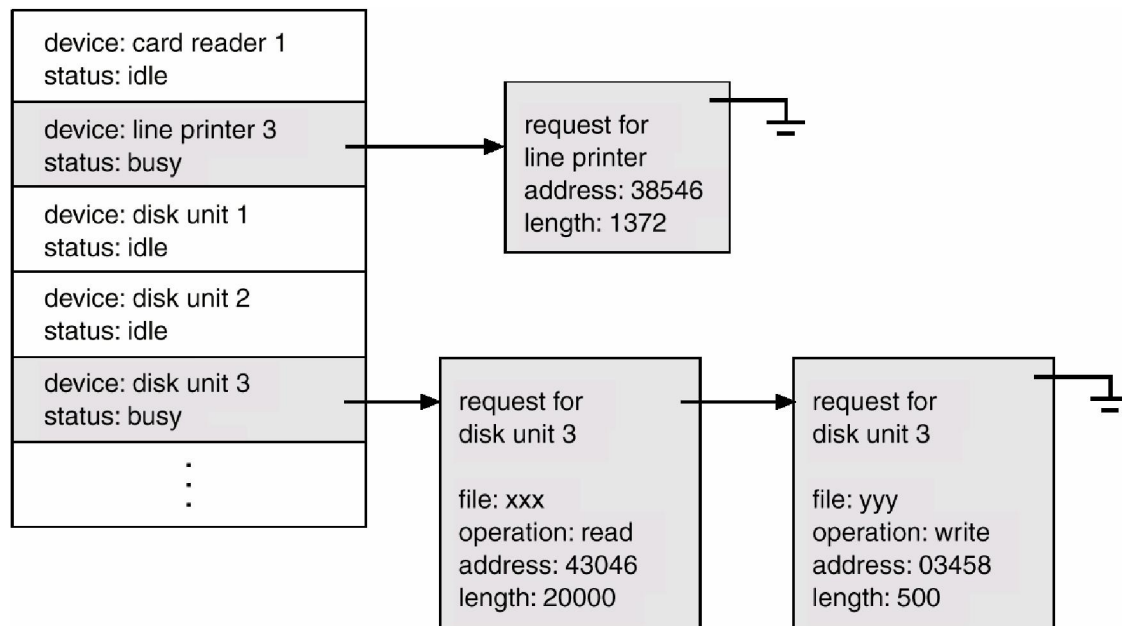
I/ O INTERRUPT

- Pilihan lebih baik adalah asynchronous I/ O
- Setelah I/ O mulai, kendali langsung kembali ke user program tanpa menunggu I/ O selesai.
- CPU dapat melanjutkan operasi I/O untuk device yang lain.
- User program dapat menjalankan program tanpa menunggu atau harus menunggu sampai I/ O selesai.
- System call-request ke OS untuk operasi I/ O dan menunggu sampai I/O selesai.
- Potensi lebih dari satu device
 - User hanya dapat menggunakan I/ O melalui system call.
 - Device-status table memuat informasi untuk setiap I/O device : tipe, alamat, status, dll.
- OS mengatur tabel ini dan mengubah isinya sesuai dengan status device (interrupt).

DUA METODE I/O



DEVICE-STATUS TABLE



DIRECT MEMORY ACCESS (DMA)

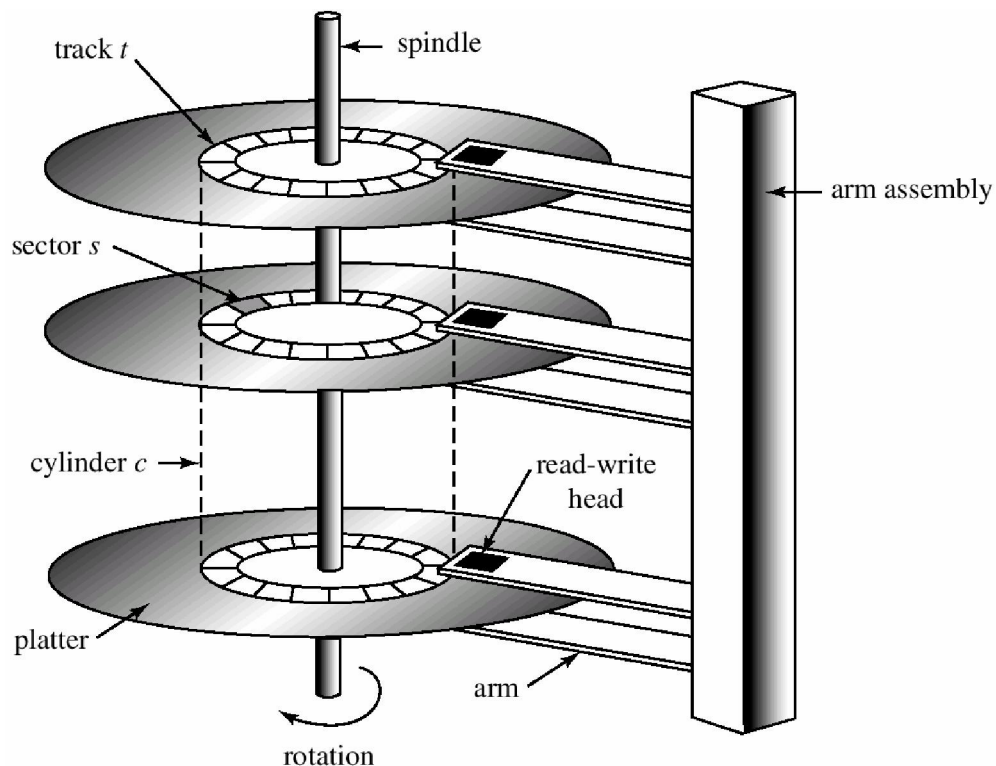
- Jika I/O devices sangat cepat (“high-speed”), beban CPU menjadi besar harus mengawasi transfer data dari controller ke memory dan sebaliknya.
- Hardware tambahan => DMA controller dapat memindahkan blok data dari buffer langsung ke memory tanpa mengganggu CPU.
- CPU menentukan lokasi memory dan jika DMA controller telah selesai => interrupt ke CPU.
- Hanya satu interrupt ke CPU untuk sekumpulan data (blok).

STRUKTUR STORAGE

- Main Memory : Yaitu media penyimpanan, dimana CPU dapat melakukan akses secara langsung.
- Secondary storage : yaitu tambahan dari main memory yang memiliki kapasitas besar dan bersifat nonvolatile.
- Magnetic disks :

- Metal keras atau piringan yang terbungkus material magnetik.
- Permukaan disk terbagi secara logikal dalam track, yang masing-masing terbagi lagi dalam sector.
- Disk controller menentukan interaksi logikal antara device dan komputer.

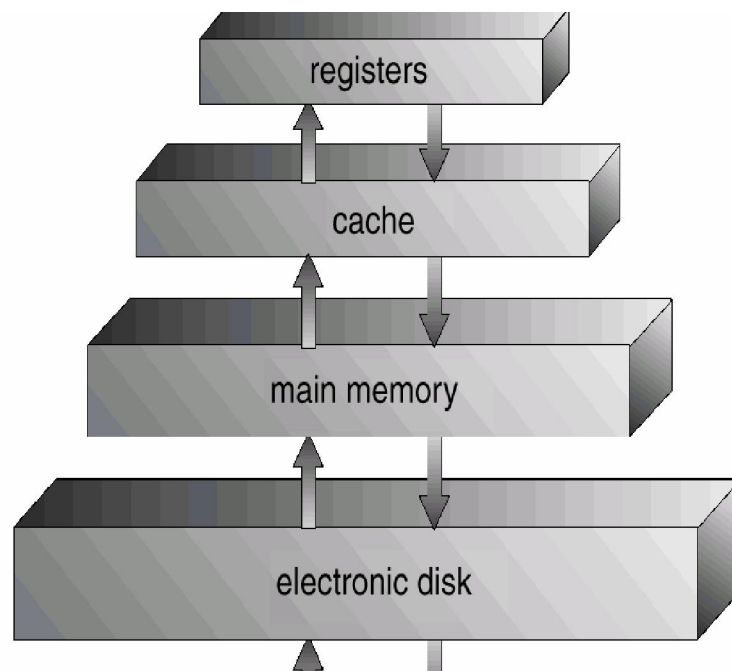
MEKANISME PERGERAKAN HEAD-DISK



HIRARKI STORAGE

- Hirarki sistem storage, diorganisasikan dalam bentuk :
 1. Kecepatan
 2. Biaya
 3. Volatilitas
- Caching : Yaitu penduplikasian informasi ke dalam sistem storage yang cepat dapat dilakukan melalui cache pada secondary storage.

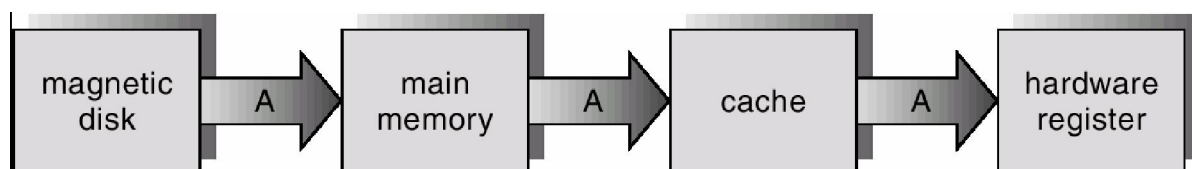
HIRARKI STORAGE-DEVICE



CACHING

- Menggunakan memori berkecepatan tinggi untuk menangani akses data saat itu juga (yang terbaru).
- Membutuhkan manajemen cache.
- Caching mengenalkan tingkatan lain dalam hirarki storage, dimana data secara serentak disimpan pada lebih dari satu tingkatan secara konsisten.

MIGRASI DARI DISK KE REGISTER



PROTEKSI HARDWARE

Jenis proteksi hardware diantaranya adalah :

- Dual-Mode Operation
- Proteksi I/ O
- Proteksi Memory

- Proteksi CPU

DUAL-MODE OPERATION

- Penggunaan resource sharing membutuhkan sistem operasi yang menjamin suatu program yang salah tidak menyebabkan program lain tidak terpengaruh.
- Menyediakan dukungan hardware yang dibedakan ke dalam dua mode operasi :
 1. User mode – eksekusi dilakukan untuk kepentingan user.
 2. Monitor mode (disebut juga kernel mode atau system mode) eksekusi dilakukan untuk kepentingan sistem operasi.

DUAL-MODE OPERATION (CONT.)

- Mode bit ditambahkan pada computer hardware (CPU) untuk indikasi mode sekarang : monitor (0) atau user (1).
- Jika terjadi interrupt/ fault/ error => hardware mengubah mode ke monitor

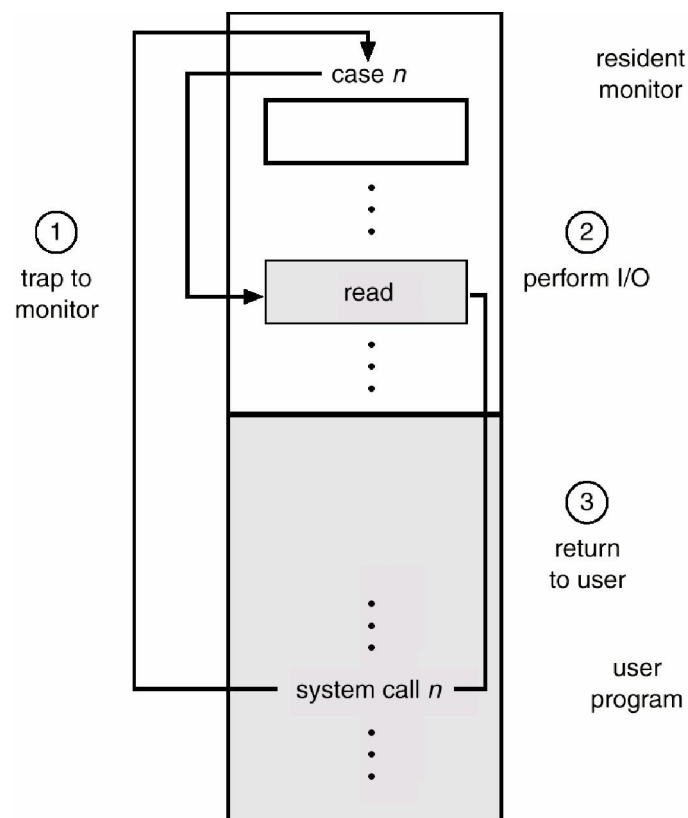


Instruksi Privileged hanya dapat diberikan dalam mode monitor

PROTEKSI I/ O

- Semua instruksi I/O adalah instruksi privileged :
 - Hanya dapat dilakukan melalui OS.
 - OS dapat mencegah “request” ke I/ O dengan melihat mode saat ini.
- OS menjaga supaya program user tidak dapat menjadi “monitor mode” untuk mencegah user program melakukan :
 - Menangani interrupt: dengan mengubah alamat interrupt vector.
 - Mengubah status dan data pada “device table”.

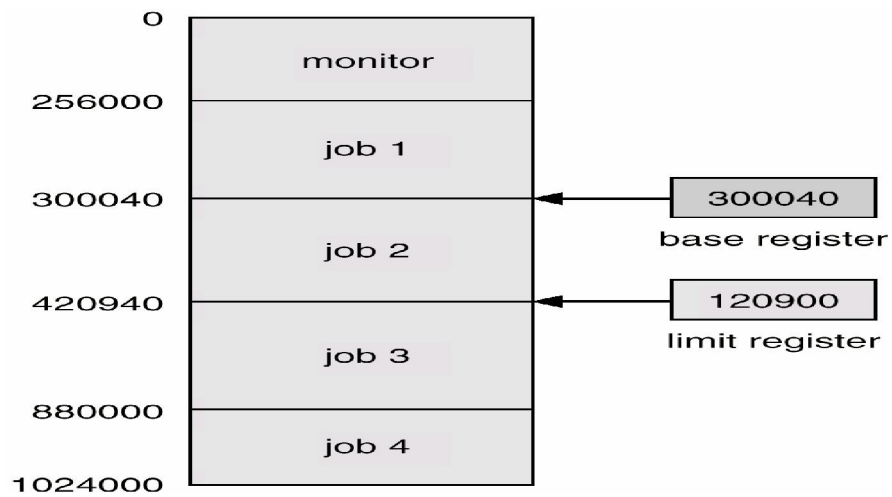
PENGUNAAN SYSTEM CALL UNTUK PENGOPRASIAN I/ O



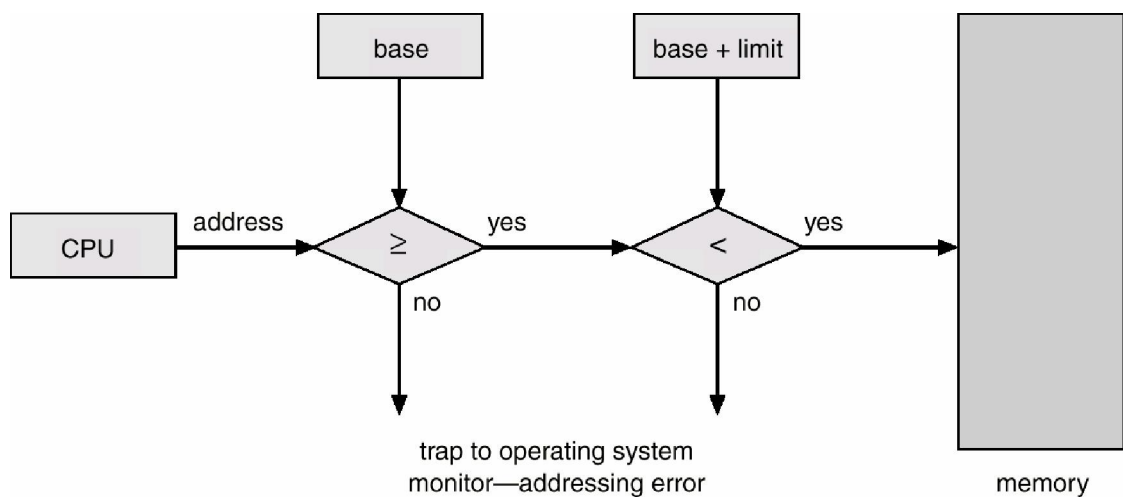
PROTEKSI MEMORY

- Melindungi memori terutama untuk isi : Interrupt vector dan interrupt service routines.
- Cara umum adalah setiap user program hanya dapat mengakses lokasi memori yang telah dibatasi (disediakan untuk program tsb).
 - Range address : Alamat yang valid.
 - Base register : Menyimpan alamat terkecil memori secara fisik.
 - Limit register : Besarnya jangkauan memori yang diijinkan.
- Memori diluar range tersebut tidak dapat diakses oleh user program tsb.

PENGUNAAN BASE DAN LIMIT REGISTER



PROTEKSI ALAMAT HARDWARE



- Ketika eksekusi pada mode monitor, OS dapat mengakses semua lokasi memori.
- Pemuatan instruksi ke base dan limit register tergantung instruksi privileged.

PROTEKSI CPU

- Timer

- Interupsi secara berkala oleh hardware => transfer control ke OS.
- Nilai timer akan berkurang sesuai “clock tick” dari hardware komputer.
- Saat nilai timer menjadi 0, interrupt terjadi.
- Housekeeping: melakukan CPU scheduling (jajah CPU), status device table dll.
- Timer digunakan untuk system time.

C. SOAL LATIHAN/TUGAS

1. Jelaskan penggunaan base dan limit register?
2. Jelaskan dual mode operation?
3. Jelaskan hirarki storage device ?

D. DAFTAR PUSTAKA

Buku

Bambang Hariyanto. 1997. Sistem Operasi, Bandung: Informatika Bandung.

Dali S. Naga. 1992. Teori dan Soal Sistem Operasi Komputer, Jakarta: Gunadarma.

Silberschatz Galvin. 1995. 4 Edition Operating System Concepts: Addison Wesley.

Sri Kusumadewi. 2000. Sistem Operasi. Yogyakarta: J&J Learning.

Tanenbaum, A. 1992. Modern Operating Systems. New York: Prentice Hall

Link and Sites:

<http://www.ilmukomputer.com>

<http://vlsm.bebas.org>

<http://www.wikipedia.com>