



**Kampus  
Merdeka**  
INDONESIA JAYA

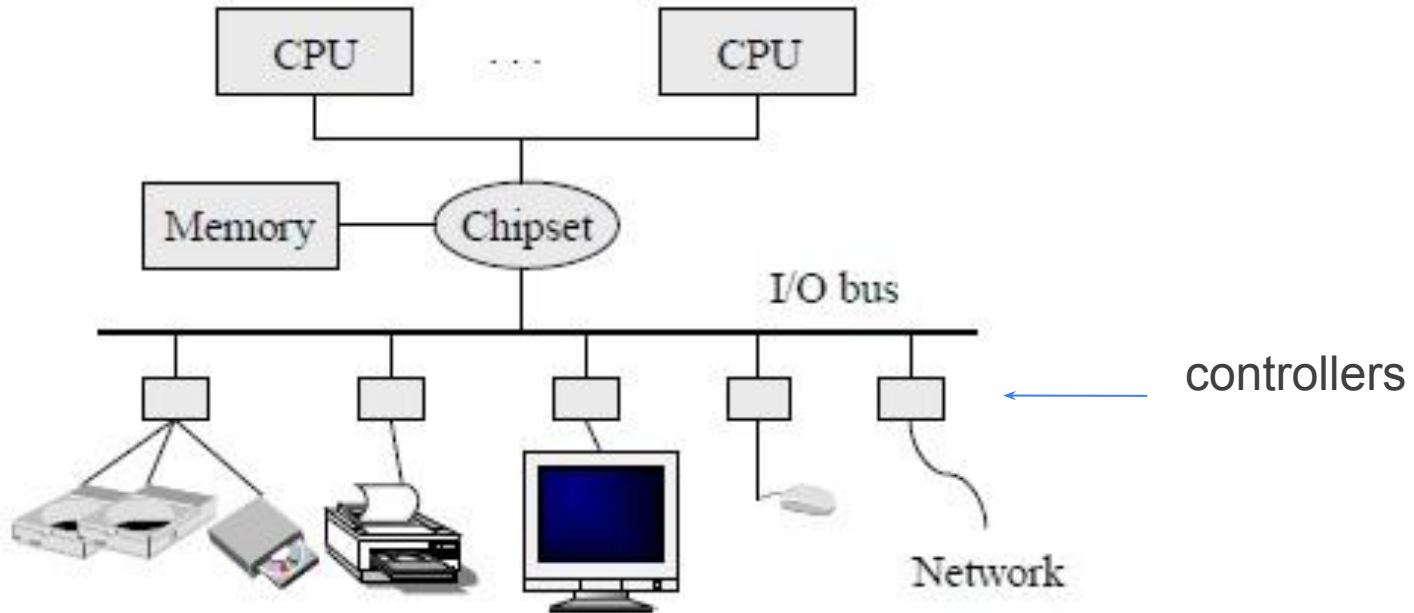
## Sistem Operasi: Perangkat Keras

Septian Cahyadi



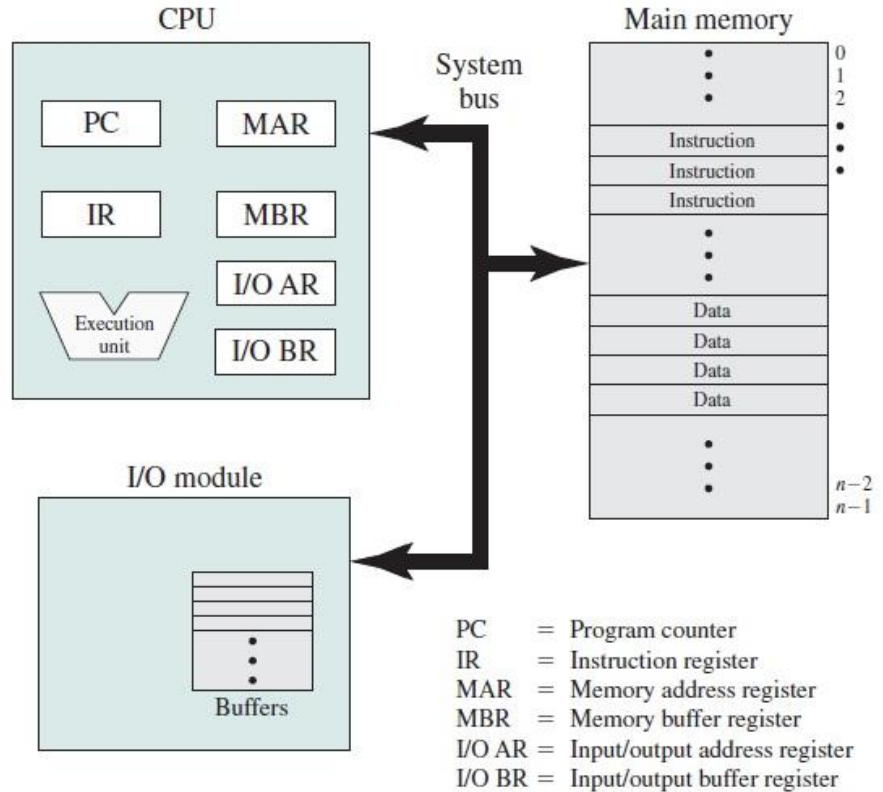
- Bagian-bagian Perangkat Keras
  - CPU
  - Memory
  - Perangkat I/O
  - Bus
- Melakukan booting

# Struktur Perangkat Keras





- Otak komputer
- Mengeksekusi program:  
sekumpulan instruksi  
sekuensial
- Mengendalikan I/O





# Siklus Instruksi

- Siklus: fetch – decode – execute
- Waktu tiap tahap tidak sama
- Fetch: mengambil instruksi pada alamat di PC dan menyalinnya ke IR
- Decode: mengartikan instruksi pada IR
- Execute: menjalankan instruksi pada IR



Instruction format

0001 = Load AC from memory  
0010 = Store AC to memory  
0101 = Add to AC from memory

Partial list of opcodes



**Kampus  
Merdeka**  
INDONESIA JAYA

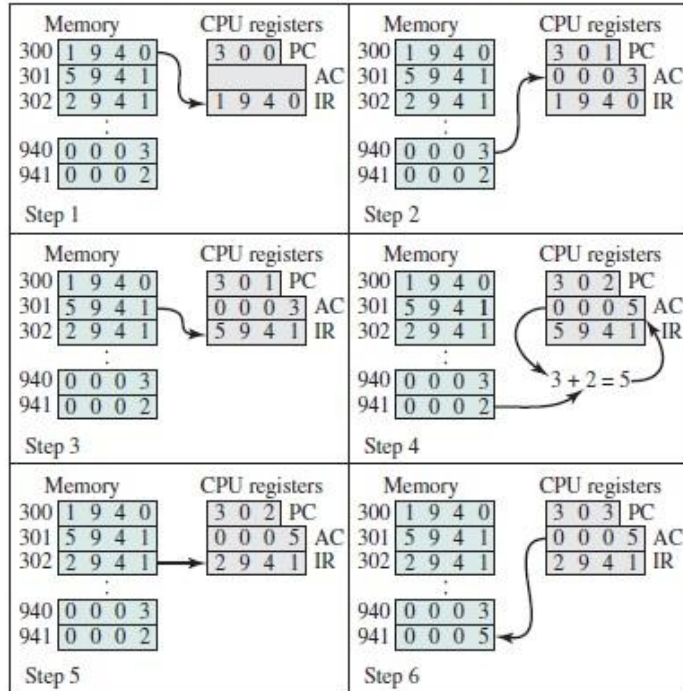
# Jenis Instruksi

- Processor – Memory
- Processor – I/O
- Arithmetic or Logical Operations
- Control

# Contoh Instruksi

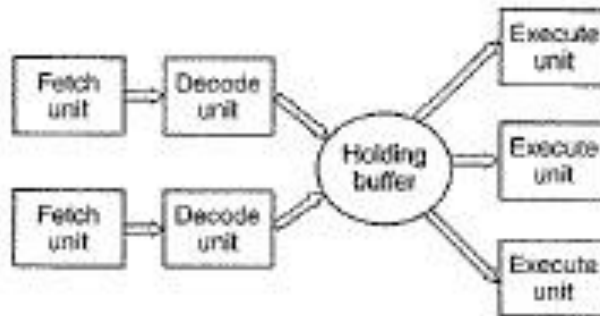
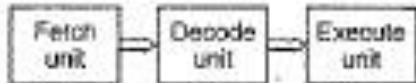
## Fetch stage

## Execute stage



# Pipeline dan Superscalar (1)

- Beberapa teknologi untuk mempercepat proses dengan mengeksekusi lebih dari 1 instruksi tiap saat
- Pipeline: menyediakan 3 unit (dalam 1 line) dan dapat bekerja paralel
- Superscalar: membagi menjadi beberapa unit







# Pipeline dan Superscalar (2)

- Masalah pipeline: instruksi yang sudah diambil dari memory harus dijalankan, padahal ada instruksi percabangan
- Masalah superscalar: ketidakterurutan eksekusi instruksi



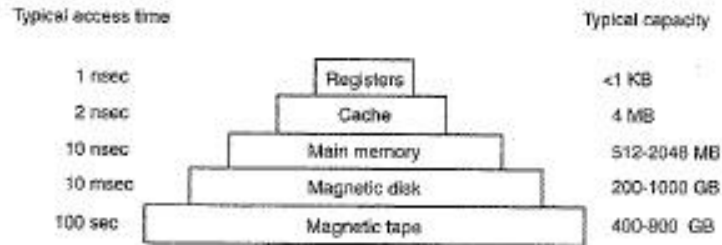
# Instruksi Khusus

- 2 mode operasi: mode kernel dan user
- Pada mode kernel, semua jenis instruksi bisa dijalankan
- Pada mode user, hanya sebagian jenis instruksi saja
- Program dari user berjalan di mode user, SO pada mode kernel
- Jika program user membutuhkan lebih, minta kepada SO lewat system call, untuk sementara pindah ke mode kernel
- Jenis instruksi khusus lainnya diperlukan saat terjadi kasus khusus seperti pembagian dengan 0



# Memory

- Memory ideal: cepat, kapasitas besar dan murah
- Biasanya hanya sebagian saja yang terpenuhi
- Solusi: hirarki memory



- Fungsi SO untuk mengatur transfer antar hirarki
- Data yang sering diakses diletakkan di memory yang lebih cepat



# Instruksi Khusus

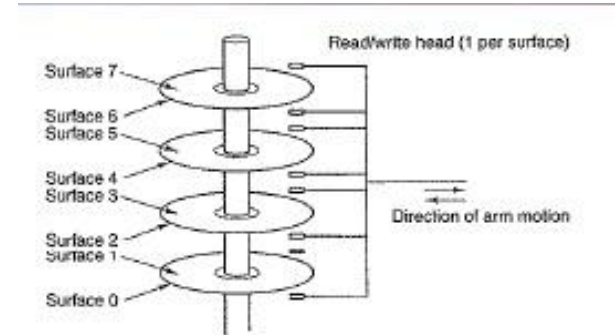
- Jika suatu memory word dibutuhkan SO akan memeriksa apakah ada di cache
- Jika ada → cache hit, maka memory utama tidak perlu diakses
- Jika tidak → cache miss, perlu mengakses memory utama yang jauh lebih lambat
- Cache juga dibuat berlevel, masing-masing level berbeda kecepatannya



# Memory Utama

- Volatility → hilangnya data yang disimpan setelah tiada arus listrik
- RAM, volatile
- ROM, nonvolatile namun tidak bisa diubah datanya
- EEPROM dan flash memory, nonvolatile dan bisa diubah datanya
- Flash memory memiliki keterbatasan banyaknya penghapusan data
- CMOS (Complementary Metal–Oxide–Semiconductor), volatile namun bertenaga baterai yang tahan lama

- Kapasitas besar, relatif murah
- Namun lambat, karena mekanis, 5400 – 10800 rpm
- Suatu jalur melingkar pada tiap surface disebut track yang terdiri dari sejumlah sector
- Gabungan track dengan radius sama membentuk cylinder
- Head harus digeser ke cylinder yang tepat, surface diputar sampai ke sector yang tepat, lalu akses data 50 – 160 MB/s
- SO berfungsi memetakan alamat logical menjadi akses fisik, diperumit dengan virtual memory





- Sangat lambat namun murah
- Untuk backup data dalam jumlah besar
- Disimpan terpisah



# Perangkat I/O

- Perangkat dan controller
- Tugas controller untuk memetakan alamat logical menjadi alamat fisik
- Perangkat I/O pun punya interface agar standar
- Device driver adalah s/w yang berkomunikasi dengan controller
- Device driver perlu berjalan di mode kernel dengan cara
  - Relink driver kepada SO lalu reboot
  - Memberitahu adanya driver kepada SO, reboot dan SO mencari driver yg dibutuhkan
  - On-the-fly





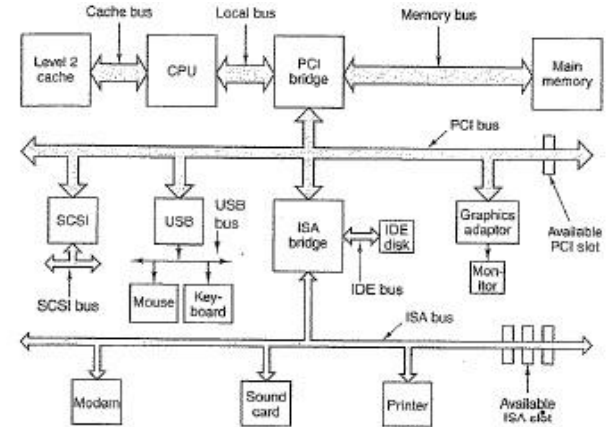
# Perangkat I/O(2)

- Controller dilengkapi register
- Register bisa dipetakan ke address space dari SO, sehingga bisa diakses oleh program tanpa instruksi khusus
- Masalah I/O: lebih lambat dari processor
  - Programmed I/O, processor harus terus-menerus memeriksa I/O module dengan busy waiting
  - I/O module memberitahu driver via interrupt bila proses I/O sudah selesai
  - Menggunakan h/w khusus: DMA (Direct Memory Access) yang bekerja paralel tanpa processor harus terlibat

- Melayani transfer data antar perangkat
- ISA (Industrial Standard Architecture), teknologi lama (8 – 16 bit)
- PCI (Peripheral Component Interconnect), lebih cepat (32 -64bit)
- IDE bus, peripheral standar spt. Disk dan CD

ROM

- USB bus
- SCSI bus untuk perangkat yang cepat
- Conflict terjadi jika interrupt yang sama dipakai 2 perangkat berbeda





# Booting

- Memanggil BIOS (Basic Input Output System)
- Memeriksa kapasitas RAM
- Memeriksa perangkat I/O yang terhubung ke bus (ISA dan PCI)
- Memeriksa boot device sesuai urutan yang tersimpan di BIOS
- Membaca boot sector, yaitu tempat SO disimpan, dan meload ke memory (oleh bootstrap loader)
- SO akan memeriksa device driver
- Diakhiri dengan login



Sekian & terima kasih

Ada pertanyaan ?