

Memori

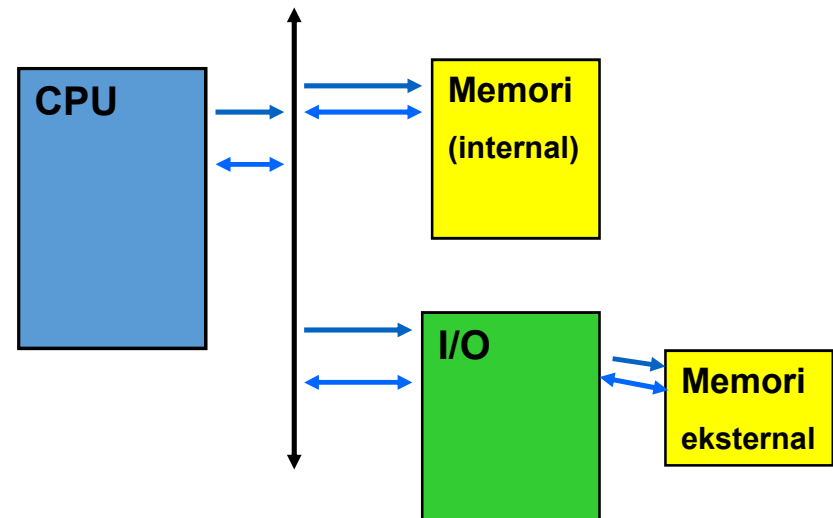
Team Dosen
Telkom University
2016

Karakteristik Sistem Memori Internal

- Lokasi
- Kapasitas (Relatif thd ruang memori)
- Besaran Transfer
- Metoda akses
- Performa
- Jenis Fisik
- Karakteristik Fisik
- Organisasi Memori

Lokasi

- Register di CPU
- Internal (di dalam sistem komputer) contoh: cache & memori utama
- Eksternal (diakses lewat I/O) contoh: USB drive, IEEE1394 (firewire) drive dll



Kapasitas dan lebar bandwidth

- Memori Internal
 - Jumlah dalam byte (ruang memori selalu disebut dalam Byte)
 - Lebar akses : 1 bit (chip), 4 bit (chip), 8 bit (SIMM-30), 32 bit (SIMM-72), 64 bit (DIMM- SDRAM), 64 bit (DIMM-DDR), 64 bit (DIM-DDR2)
- Memori Eksternal
 - Jumlah dalam Byte (contoh HD IDE 40 GB , USB flash drive 128 MB)
 - Lebar akses : 1 bit (USB), 8 bit (Disket), 16 bit (IDE)



SIMM

- SIMM-1 (30 pin)
 - A0-A11 CAS & RAS → 4K x 4k = 16M
 - DQ1-DQ8 → B
 - Kapasitas 16 MB
- SIMM-2 (72 pin)
 - DQ0 – DQ35 (32 bit + 4 parity ECC)
 - A0 – A13
 - CAS 0 – CAS3 & RAS0 – RAS1

DIMM

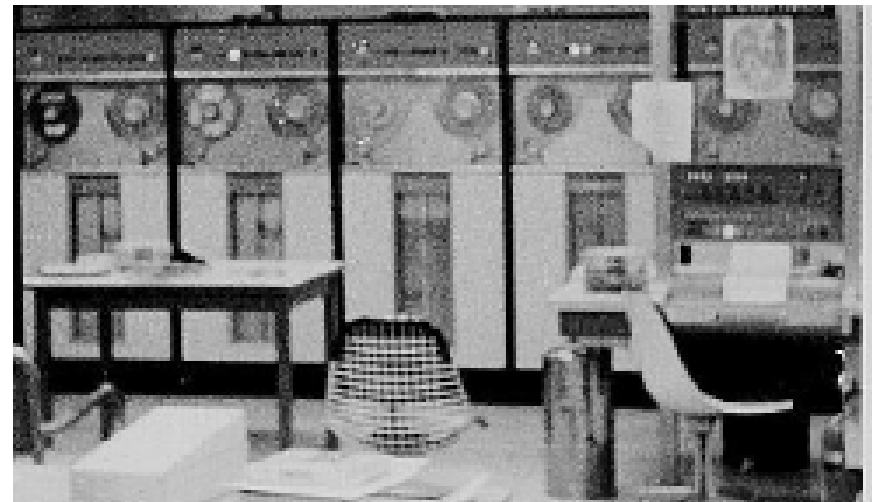
- 72-pin DIMM, used for [FPM](#) DRAM and [EDO](#) DRAM
- 72-pin [SO-DIMM](#), used for [FPM](#) DRAM and [EDO](#) DRAM
- 100-pin DIMM, used for printer [SDRAM](#)
- 144-pin [SO-DIMM](#), used for [SDR](#) SDRAM
- 168-pin DIMM, used for [SDR](#) SDRAM
- 184-pin DIMM, used for [DDR](#) SDRAM
- 200-pin [SO-DIMM](#), used for [DDR](#) SDRAM and [DDR2](#) SDRAM
- 240-pin DIMM, used for [DDR2](#) SDRAM and [FB-DIMM](#) DRAM

Besaran Transfer

- Memori External
 - Jumlah bit dibaca atau ditulis ke memori pada satu waktu (bps, Mbps dll).
- Memori Internal
 - Biasanya sebuah blok yang lebih besar dari word (16bit) (umumnya prosesor terbaru mengambil data dari memori dalam ukuran 32/64 bit, cache mengambil data dari memori dengan ukuran blok 4 byte)
 - Intel P4 (32 bit) → mempunyai bus data 64 bit (8 Byte)

Metoda Akses

- Sekuensial :
 - tape (loose end, kaset (PC saat awal keluar)),
 - (-) tidak bisa mencari data secara random
- Langsung (semirandom) :
 - Disk (HD : bahan disk metal, Disket : bahan disk lentur)
 - Pencarian data random
- Random : memori utama
- Asosiatif : cache



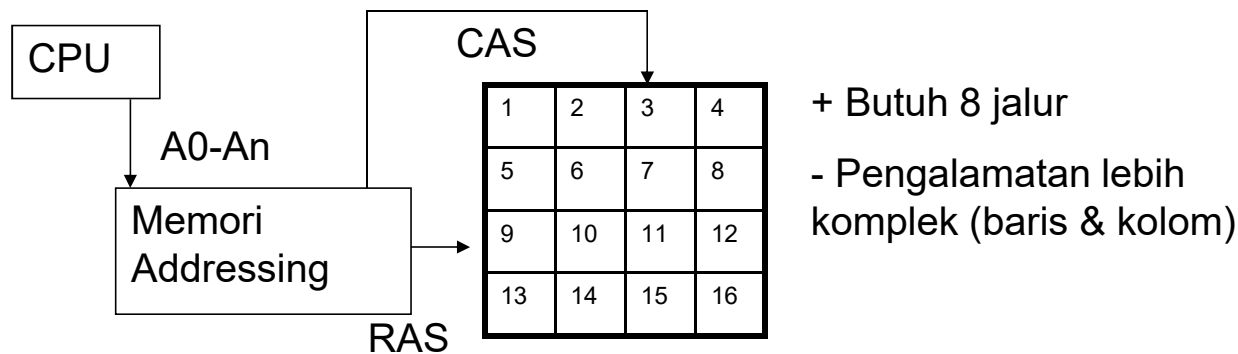
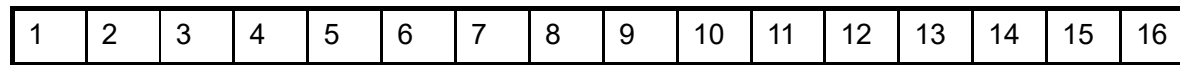
Performa

- Waktu akses (latency)
 - Waktu antara datangnya alamat sampai mendapatkan data valid
- Waktu siklus memori
 - Waktu yang diperlukan oleh memori untuk mempersiapkan akses berikutnya
 - Waktu siklus : akses + persiapan
- Laju Transfer
 - Laju data dapat dipindahkan
 - PC2100 (FSB , PC2700, PC5500(DDR2 1.1GBps) www.hwhell.com

Standard name	Memory clock	Time between signals	I/O Bus clock	Data transfers per second	Module name	Peak transfer rate
DDR-200	100 MHz	10 ns	100 MHz	200 Million	PC-1600	1.600 GB/s
DDR-266	133 MHz	7.5 ns	133 MHz	266 Million	PC-2100	2.133 GB/s
DDR-333	166 MHz	6 ns	166 MHz	333 Million	PC-2700	2.667 GB/s
DDR-400	200 MHz	5 ns	200 MHz	400 Million	PC-3200	3.200 GB/s

Kapasitas RAM yang mencapai xxx MB sd GB (tepatnya xxx Mb sd Gb per chip 1 IC biasanya hanya mempunyai 1 sd 4 pin data) memerlukan ‘kaki’ (pin) yang banyak. Misal : 1 Gb per IC membutuhkan 30 pin alamat (ukuran IC jadi besar, tidak diinginkan). Solusi : Pengalamatan dibagi menjadi kolom dan baris

Butuh 16 jalur untuk mengisi semua sel



Kecepatan Akses Memori

- 2 – 3 – 3 – 7 (– 1T) T=clock
 - CAS Latency 2, 2.5, 3 (DDR) 2,3 (SDR)
 - RAS to CAS Delay 2,3
 - RAS Precharge 2,3
 - (RAS Latency) Active to Precharge 3,4,5,6,7
 - Command Rate
 - (-1T) Delay antara akses
 - Memory Cyclic Time (Total)

Jenis Fisik

- Semikonduktor (2004 : 1ns, 1Gb)
 - RAM, ROM
- Magnetik
 - Disk & Tape
- Optik
 - CD & DVD & Blueray Disk & HD-DVD
- Organik (still in research, 2004 : Microsoft patent)
- Atomik (may be you' ll be the inventor)



IBM Millipede

Hard Disk Sebesar Perangkat

■ IBM memperkenalkan prototipe hard disk masa depan ini pada ajang CeBIT. Hard disk dengan metode nano-teknologi ini dikembangkan di Swiss. Millipede dapat menampung sampai 1 terabit (1.000 Gigabit) per kuadrat inci. Hard disk sebesar perangkat ini setara dengan 25 DVD.

Cara kerjanya, silikon-tip

yang berjumlah 4.096 (64 x 64) membakar data pada film tipis terbuat dari polimer. Silikon-tip ini merupakan pengembangan dari penemuan Gerd Binnig (IBM) yang meraih Nobel untuk fisika pada tahun 1986 dengan nama *Scanning Tunneling Microscopy* (STM). Selain STM, Gerd Binnig juga menemukan *Atomic Force Microscope* yang memung-

Protipe: Hard disk baru IBM terpasang seperti sebuah chip.

kinkan microprobe membaca data seukuran atom. Millipede memiliki ribuan microprobe seperti ini. Agar transfer data tinggi, alat pembaca dan penulis ini menggunakan banyak microprobe yang dapat bekerja secara paralel. Media penyimpanan yang terbuat dari polimer dapat digerakkan sehingga setiap probe dapat membaca dan menulis sepanjang 100 mikrometer. Kecepatan akses sangat tinggi karena jaraknya sangat berdekatan.

Konstruksi Millipede berbeda jauh dengan desain hard disk saat ini. Prototipe-nya berbentuk chip seukuran 10x10 mm. Target IBM sudah terbaca. Apalagi kalau bukan mobile device, kamera atau ponsel.

Info: www.ibm.com

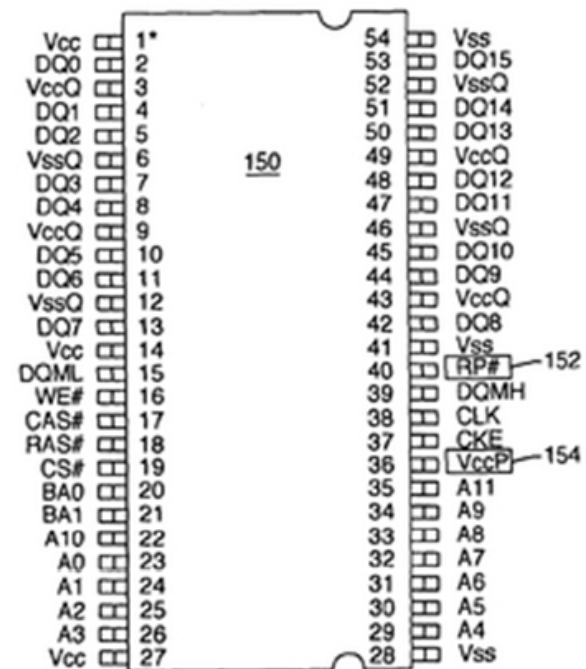
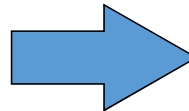


Karakteristik Fisik

- Volatile vs non-volatile
(data hilang/tidak hilang jika tidak ada daya)
- Erasable vs non-erasable

Organisasi

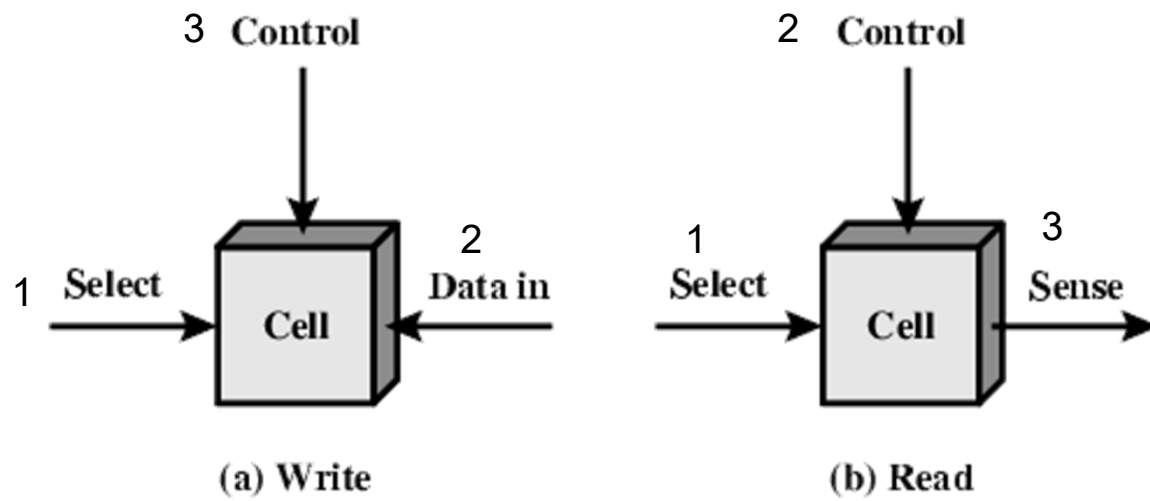
- Pengaturan fisik memori membentuk bit ke word
 - Perlu 8 IC atau 2 IC untuk membentuk byte, 16 IC atau 4 IC untuk membentuk word
 - 1 IC membentuk byte atau word
- Byte = 8 bit
- Word = 16 bit
- Dword = 32 bit



Memori Semikonduktor

- Menyediakan 2 state stabil sehingga dapat digunakan untuk mereprentasikan kondisi logika 0 dan 1
- Bisa ditulis
- Bisa dibaca

Operasi Sel Memori



Jenis Memori Semikonduktor

Memory Type	Category	Erasure	Write Mechanism	Volatility
Random-access memory (RAM)	Read-write memory	Electrically, byte-level	Electrically	Volatile
Read-only memory (ROM)	Read-only memory	Not possible	Masks	Nonvolatile
Programmable ROM (PROM)			Electrically	
Erasable PROM (EPROM)	Read-mostly memory	UV light, chip-level	Electrically	
Electrically Erasable PROM (EEPROM)		Electrically, byte-level		
Flash memory		Electrically, block-level		



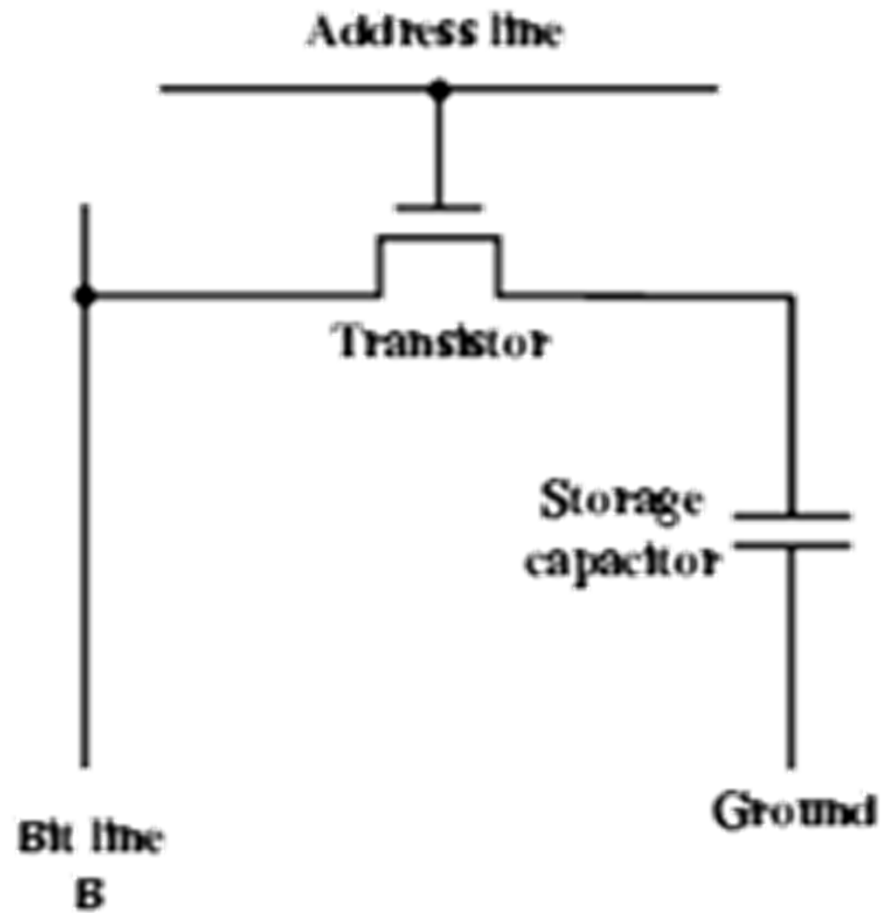
Random Access Memory

- Read/Write
- Volatile
- Penyimpanan Sementara
- Statik atau dinamis

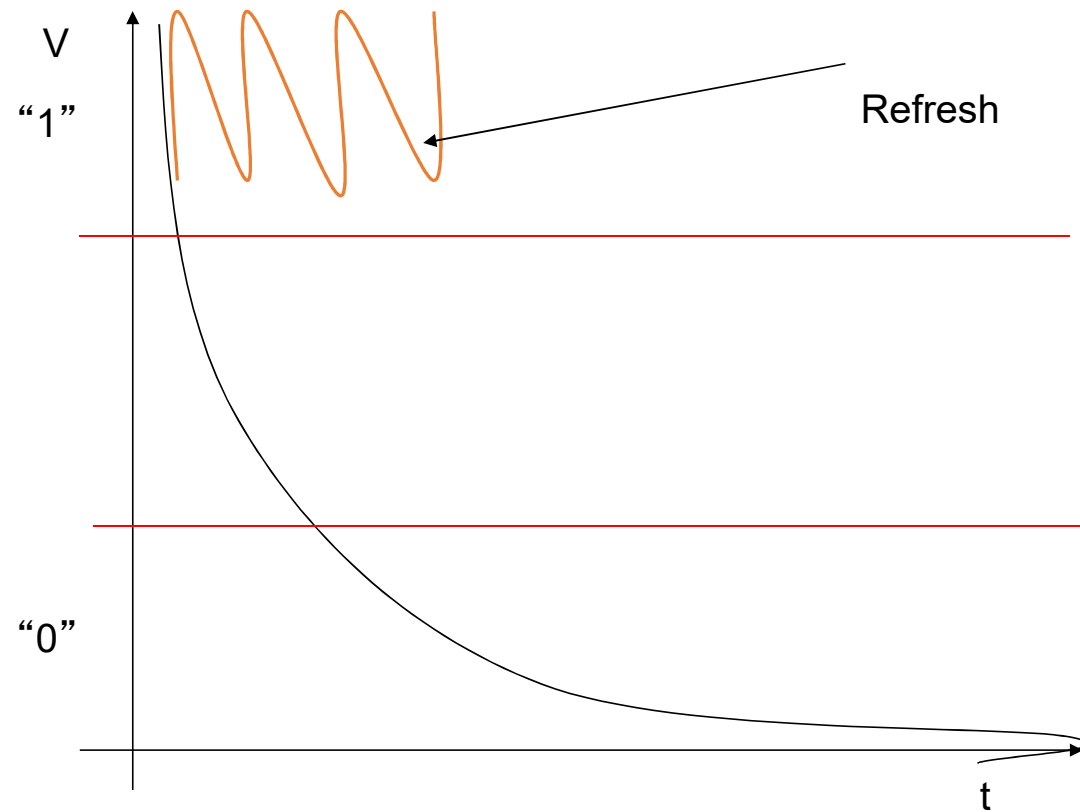
RAM Dinamis (DRAM)

- Bit disimpan sebagai muatan di *kapasitor*
 - Muatan bocor sedikit demi sedikit
 - Memerlukan **refresh**
 - Konstruksi sederhana
 - Ukuran perbit kecil (kapasitas Gb/IC)
 - Murah (1990: 0.5 bit /Rp ;2004: 640 bit/Rp ;2006: 8888.89 bit/Rp;2008: 40000 bit/Rp)
 - Lambat (1985: 200 ns; 2004-2006 : 7.5 ns;)
- Memori Utama
- Pada dasarnya analog (diisi sebagai level tegangan), level muatan menentukan nilai logika (TTL : '0' = 0 sd 1 volt, '1' = 2.5 sd 5 volt)

Struktur RAM Dinamis



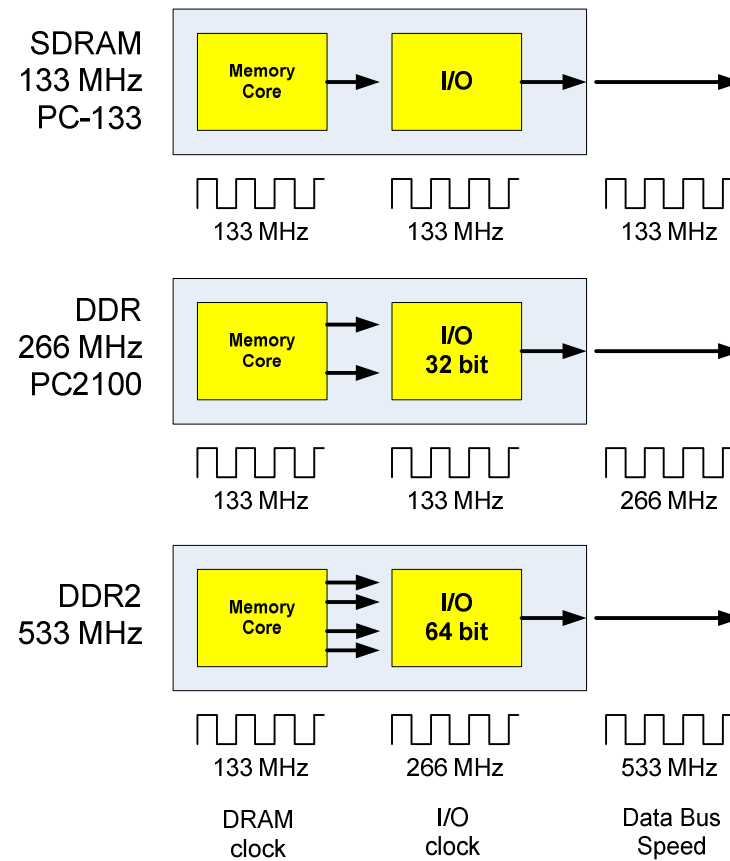
Muatan Kapasitor



Operasi DRAM

- Jalur alamat aktif ketika bit dibaca atau ditulis
 - Switch Transistor menutup (arus mengalir)
- Tulis
 - Tegangan ke jalur bit
 - Tinggi untuk 1, rendah untuk 0
 - Beri jalur alamat sinyal
 - Menyimpan muatan ke kapasitor
- Baca
 - Jalur alamat dipilih
 - transistor hidup
 - Muatan dari kapasitor dimuat ke jalur bit melalui sirkit perasa
 - Bandingkan dengan nilai referensi untuk menentukan 0 atau 1
 - Muatan kapasitor harus disimpan ulang

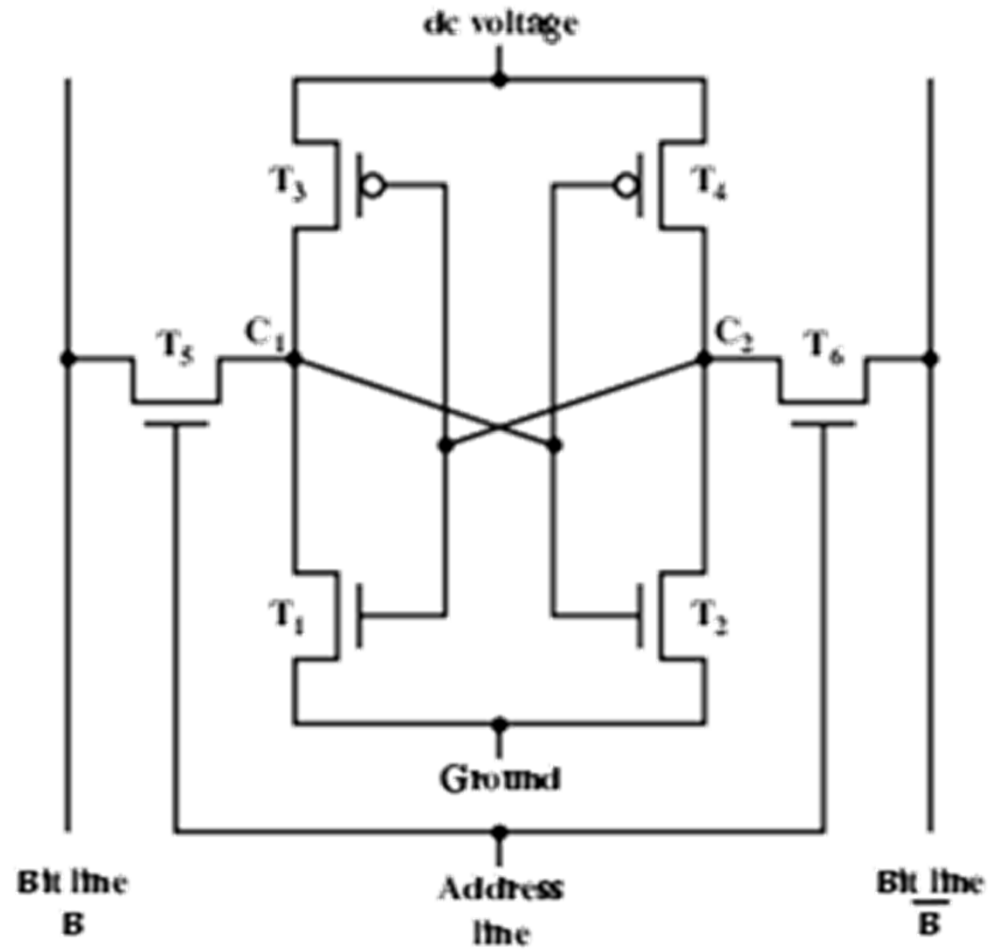
DRAM, DDR dan DDR2



RAM Statik (SRAM)

- Bit disimpan sebagai switch on/off
- Tidak ada kebocoran muatan
- Tidak perlu refresh
- Lebih kompleks
- Ukuran perbit lebih besar
- Lebih mahal
- 5516AFL-25 2k*8 static ram 250ns, 6116 SMT \$10.00
- 5517AL-200 2k*8 static ram 200ns, 6116 SMT \$10.00
- 55417J-25 16k*4 static ram 25ns cmos SMT \$10.00
- Lebih Cepat
- Cache
- Digital
 - Menggunakan Flip-flop

Struktur RAM Statik



Operasi RAM Statik

- State 1
 - C_1 high, C_2 low
 - $T_1 T_4$ off, $T_2 T_3$ on
- State 0
 - C_2 high, C_1 low
 - $T_2 T_3$ off, $T_1 T_4$ on
- Transistor jalur alamat $T_5 T_6$ adalah switch
- Tulis – menerapkan nilai ke B & B
- Baca – nilai on jalur B

SRAM v DRAM

- Keduanya volatile
 - Diperlukan daya untuk mempertahankan data
- Sel Dinamik
 - Sederhana
 - Lebih padat
 - Lebih murah
 - Perlu refresh
- Static
 - Lebih cepat
 - Lebih mahal
 - Jumlah komponen per bit lebih banyak
 - Pengguna : Cache

Read Only Memory (ROM)

- Permanen, hanya baca, tidak bisa ditulis
 - Nonvolatile
- Penggunaan:
 - Microprogramming (control unit)
 - Library subroutines (os embedded)
 - Systems programs (BIOS)
 - Function tables (look up table)

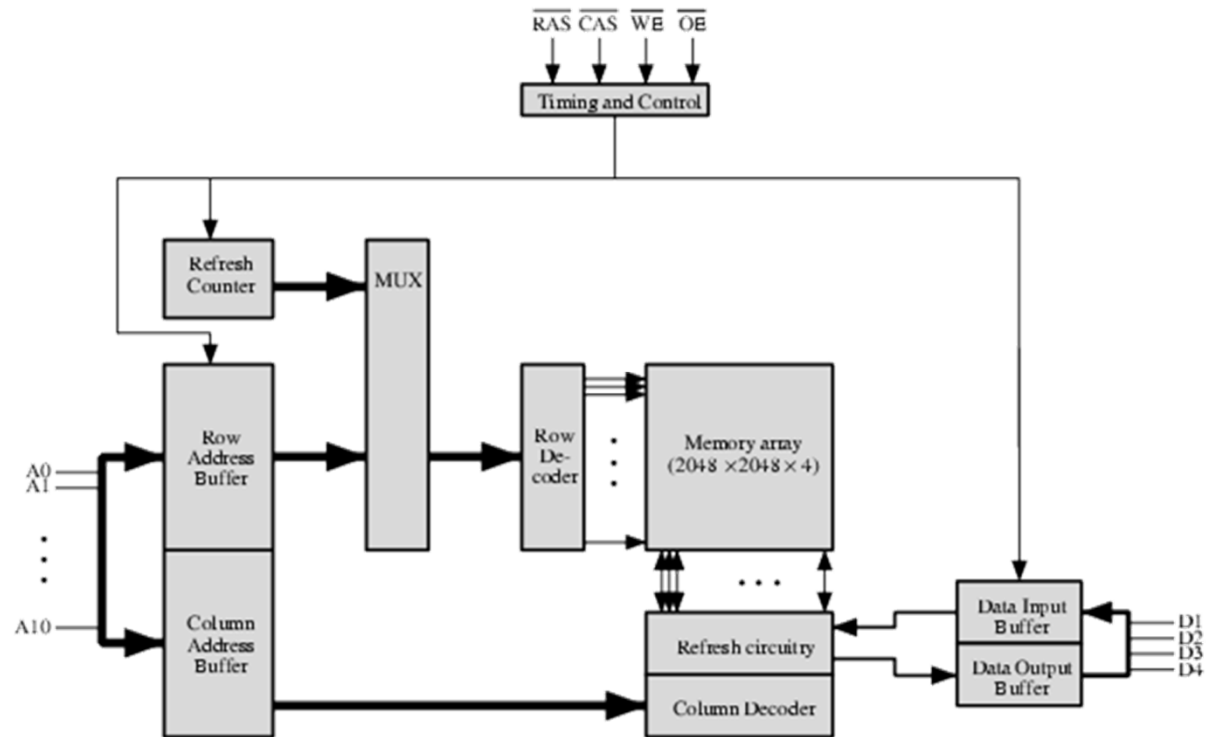
Jenis-jenis ROM

- Ditulis saat dibuat (di isi di pabrik)
 - Mahal untuk jumlah sedikit
- Programmable (sekali)
 - PROM
 - Perlu alat khusus untuk memprogram
- Baca “sering”
 - Erasable Programmable (EPROM)
 - Dihapus dengan UV
 - Electrically Erasable (EEPROM)
 - Waktu tulis lebih lama dari baca
 - Flash memory
 - Menghapus seluruh memori secara elektrik

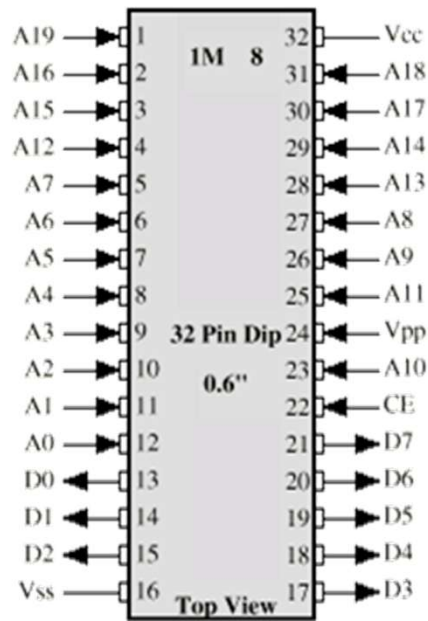
Logic Chip

- IC 16Mbit dapat disusun sebagai 1M dari word 16 bit
- Sistem 1 bit per IC memerlukan 16 IC untuk membentuk word 16 bit
- IC 16Mbit dapat disusun sebagai array 2048 x 2048 x 4bit
 - Mengurangi jumlah pin alamat
 - Multiplex alamat baris dan kolom
 - Alamat 11 pin ($2^{11}=2048$)
 - Menambah 1 pin menyebabkan kapasitas naik 4x

16 Mb DRAM (4M x 4)

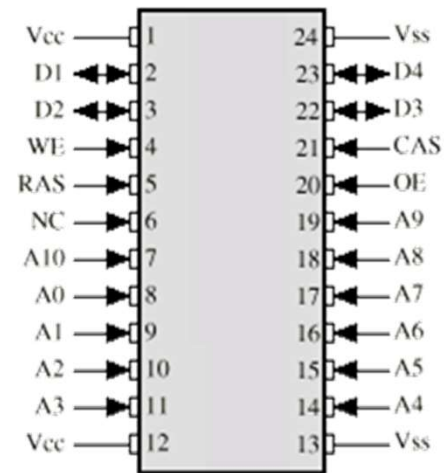


Kemasan



xROM 1 MB

1 MB = A0 sd A19 &
D0 sd D7



RAM 4x4 Mb

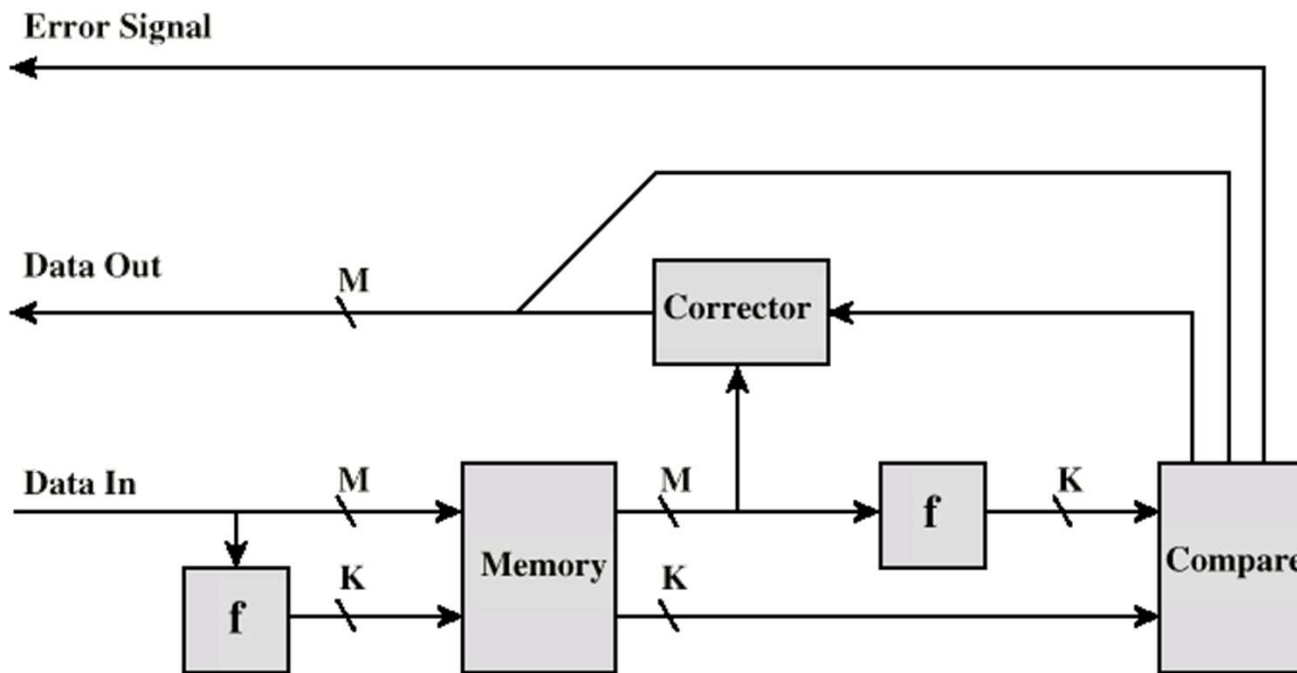
4 → D1 sd D4

$4Mb = 2^{11} \times 2^{11}$ (A0
sd A10)

Koreksi Error

- Hard Failure
 - Kerusakan fisik permanen
 - Kesalahan pembuatan, usia devais
- Soft Error
 - Random, non-destructive
 - Tidak menimbulkan kerusakan permanen pada memori
 - Disebabkan power supply, partikel alpha
- Digunakan metoda Hamming untuk membetulkannya

Fungsi Koreksi Kesalahan



Organisasi Lanjut DRAM

- DRAM dasarnya masih sama dengan RAM tahun 1970an
- Enhanced DRAM
 - SDRAM
 - RDRAM
 - CDRAM

Synchronous DRAM (SDRAM)

- Pertukaran data dengan prosesor disinkronkan dengan clock eksternal
- Alamat diberikan pada RAM
- RAM menemukan data (CPU biasanya menunggu pada DRAM konvensional)
- Dikarenakan SDRAM memindahkan data setelah sejumlah clock, CPU akan tahu kapan data tersedia
- CPU tidak perlu menunggu, dapat melakukan hal lain

Rambus DRAM (tidak dipakai)

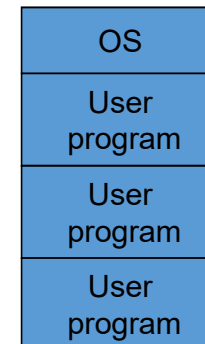
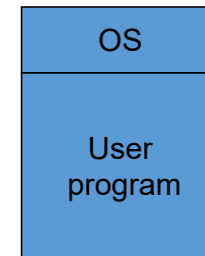
- Mahal
- Diadopsi Intel th 2000 untuk Pentium & Itanium (sebelum akhirnya dipilih DDR-SDRAM)
- Kemasan Vertical (SIP) – semua pin pada satu sisi
- Pertukaran data melewati 28 kawat < 12cm
- Bus mengalami sampai 320 RDRAM di 1.6Gbps

Cached DRAM

- Dikembangkan Mitsubishi
- Mengintegrasikan cacheSRAM kecil ke DRAM
- SRAM dapat berfungsi sebagai cache atau buffer

Manajemen Memori

- Uni-program
 - Memori dibagi 2 bagian
 - Untuk OS (monitor)
 - Untuk eksekusi program pada saat yang sama (concurrent)
- Multi-program
 - Bagian pemakai dibagi-bagi lagi untuk process yang sedang aktif
 - (sedang dijalankan), bagi process yang tidak aktif (menunggu dijalankan) akan disimpan sementara di memori sekunder (swapping)



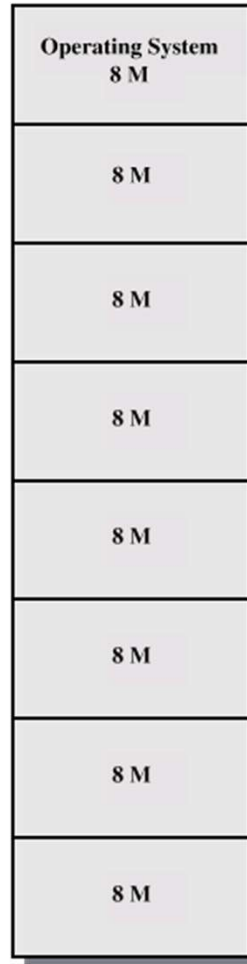
Manajemen Memori

- Masalah : I/O sangat lambat, bahkan dengan multiprogramming pun masih tersisa waktu idle CPU (menunggu I/O selesai) yang cukup panjang
- Solusi:
 - Menambah kapasitas memori utama (agar semakin banyak program yang aktif)
 - Mahal
 - Kecenderungan program baru lebih besar memerlukan memori
 - Swapping

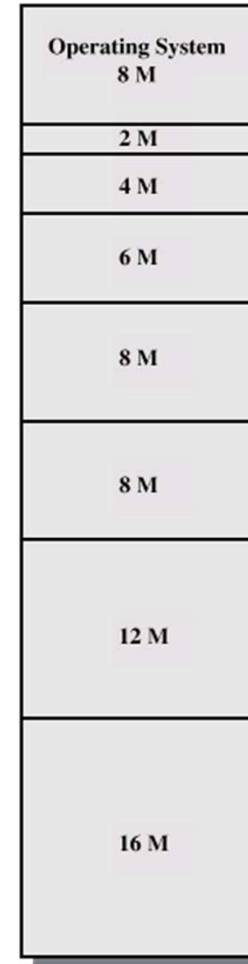
Partisi

- Membagi memori ke bagian-bagian kecil untuk dialokasikan bagi process
- Partisi tetap
 - Mungkin tidak sama besar
 - Process dimasukan ke tempat sekecil mungkin (best fit)
 - Terdapat memori yang tidak terpakai (wasted space)

Partisi Tetap



(a) Equal-size partitions



(b) Unequal-size partitions

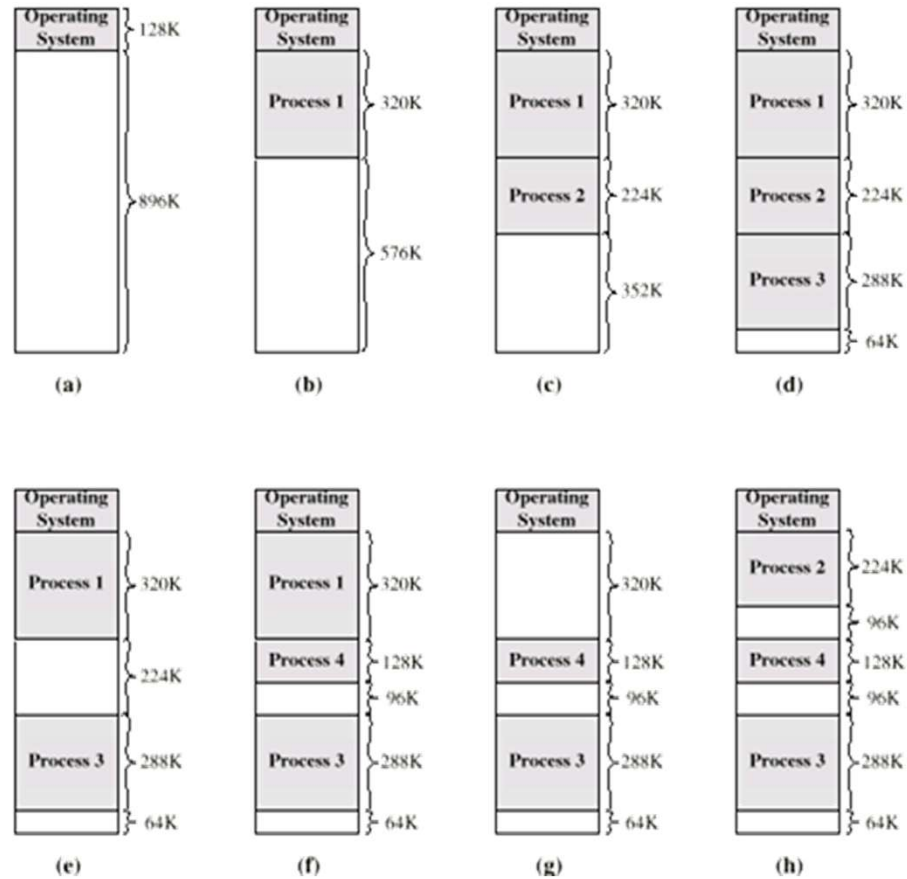
Partisi Ukuran Variabel (1)

- Alokasikan memori dengan jumlah sesuai yang diminta proses
- Membuat lubang (sisa) diujung memori, terlalu kecil untuk digunakan
 - Hanya satu lubang – kehilangan sedikit sekali
- Ketika semua proses diblok, swap out sebuah proses dan ganti dengan yang lain
- Proses yang baru mungkin lebih kecil, sehingga terdapat lubang baru

Partisi Ukuran Variabel (2)

- Bisa terdapat banyak lubang
- Solusi:
 - Defragmentasi memori – buat program bergeser mengisi lubang memori

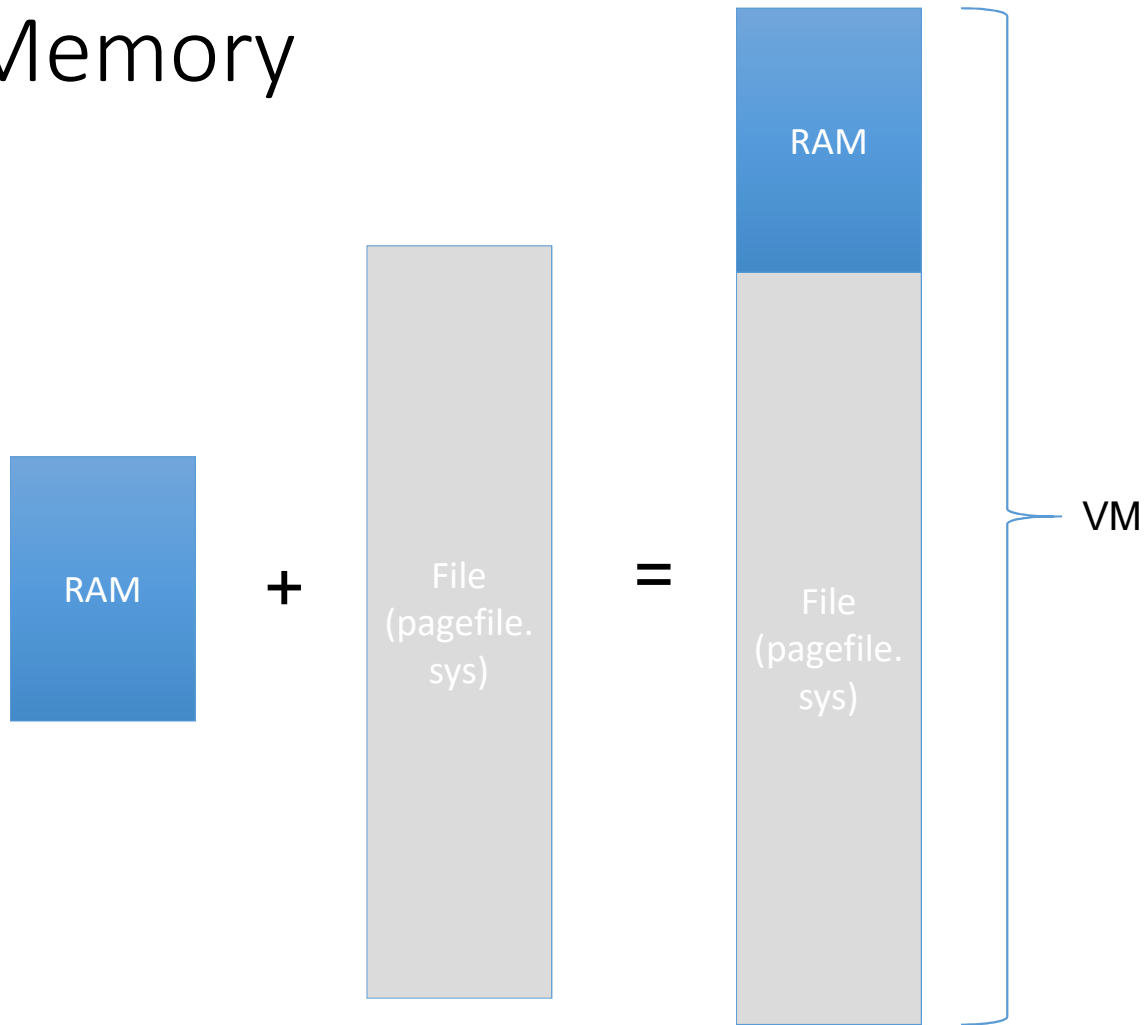
Effek dari Partisi Dinamis



Apa itu Swapping?

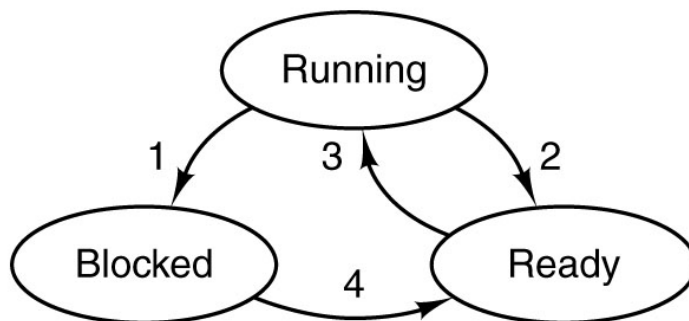
- Antrian jangka panjang process yang disimpan di disk / memori sekunder
- Process di“swap” jika ruang memori telah tersedia (swap in)
- Jika sebuah process selesai, di keluarkan dari memori utama (job out)
- Jika tidak ada process di memori yang siap (misal semua sedang I/O blocked)
 - Swap out sebuah process yang diblok ke antrian intermediate
 - Swap in sebuah process yang siap atau sebuah process baru

Virtual Memory



Proses - State

- Proses mempunyai state eksekusi
 - ready: menunggu untuk dijalankan CPU
 - running: eksekusi di CPU
 - waiting: menunggu sebuah event, misal: I/O



1. Process blocks for input
2. Scheduler picks another process
3. Scheduler picks this process
4. Input becomes available

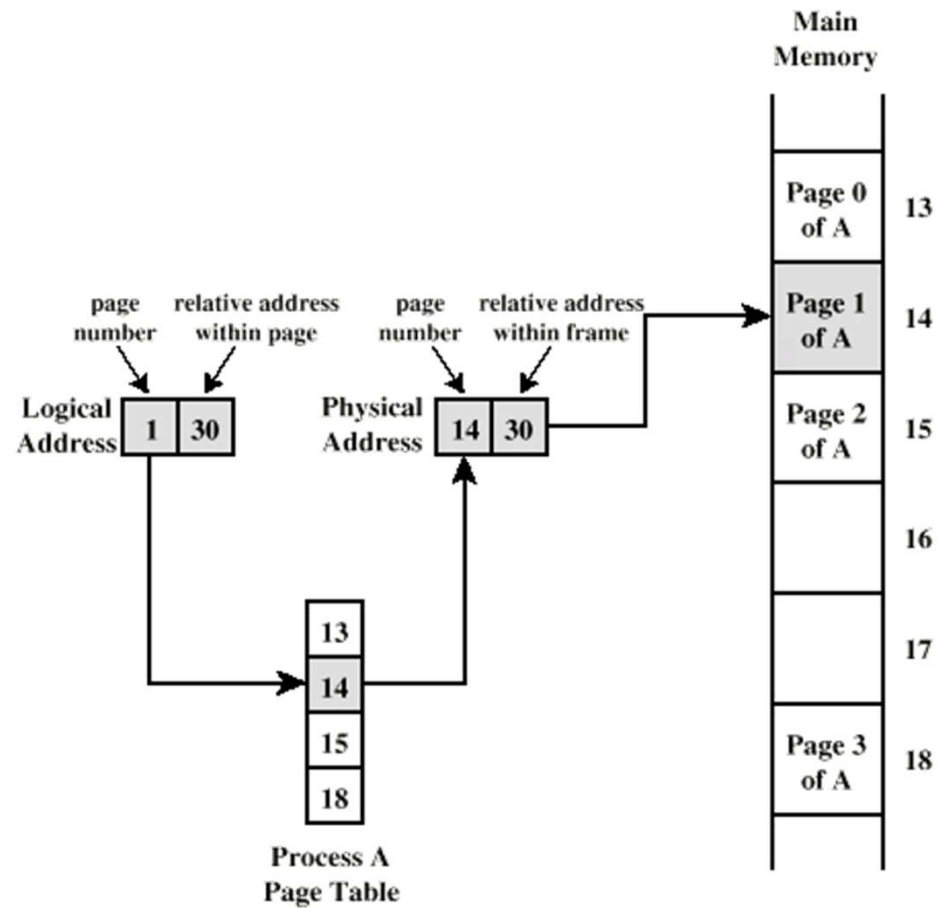
Relokasi

- Tidak ada jaminan proses bisa menempati lokasi memori sebelumnya
- Instruksi mengandung alamat
 - Lokasi data
 - Alamat instruksi (branching)
- Alamat Logik – relatif terhadap awal program
- Alamat Fisik – lokasi fisik saat ini
- Konversi otomatis menggunakan alamat dasar

Paging

- Membagi memori secara sama rata, ukuran kecil – halaman frame
- Bagi program (proses) ke halaman-halaman dg ukuran sama
- Alokasikan nomor halaman frame ke sebuah proses
- OS memelihara daftar dari frame bebas
- Sebuah proses tidak memerlukan halaman yang kontinyu
- Menggunakan tabel halaman untuk memelihara urutan halaman

Paging – Alamat Logik dan Fisik



Lokalitas

- Prinsip dari lokalitas
 - Program cenderung menggunakan kembali data dan instruksi yang baru digunakan sebelumnya
 - Program menggunakan 90% waktu eksekusinya di 10% dari code (aturan 90/10)
 - Lokalitas temporal
 - Item yang baru digunakan kemungkinan besar digunakan sebentar lagi
 - Lokalitas spatial
 - Item yang alamatnya berdekatan cenderung diakses pada waktu yang berdekatan

Latar Belakang Adanya Virtual Memory

- Demand paging
 - Tidak semua halaman dari sebuah proses perlu ada di memori (prinsip lokalitas)
 - Hanya halaman yang diperlukan yang dimuat ke memori
- Kesalahan Halaman (missing page)
 - Halaman yang diperlukan tidak ada di memori (misal operasi jump ke alamat jauh atau operasi branch)
 - OS harus men-swap halaman yang diperlukan
 - Mungkin perlu untuk men-swap out sebuah halaman untuk menyediakan tempat di memori
 - Pemilihan halaman untuk dibuang berdasarkan sejarah pemakaian sebelumnya

Pembuangan (Thrashing)

- Terlampaui banyak proses di memori dengan jumlah page yang terlampaui sedikit untuk setiap proses.
- Akibatnya :
 - OS menggunakan banyak waktu untuk swapping
 - Sedikit waktu untuk bekerja
 - Mengakses disk terus menerus
- Solusi:
 - Algoritma penukaran halaman yang baik
 - Kurangi proses yang berjalan (agar ada tempat di memori untuk swap in, tidak perlu swap out terlebih dahulu)
 - Tambah kapasitas memori (solusi mahal dan belum tentu bisa dilakukan)

Virtual Memory

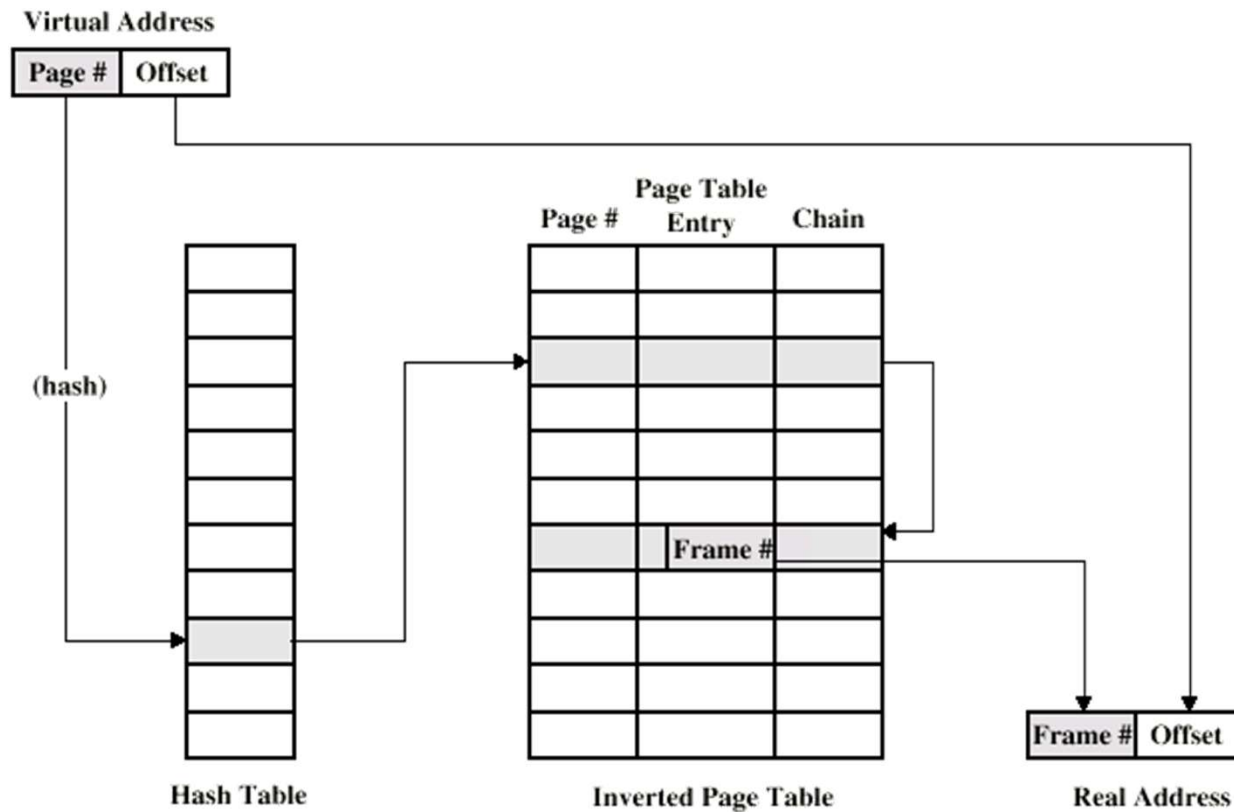
- Penggunaan memori sekunder sebagai perluasan dari memori utama
 - Tidak perlu semua bagian dari suatu proses ada di memori fisik
 - Dapat men-swap halaman jika diperlukan
 - Sehingga dapat menjalankan proses yang lebih besar dari kapasitas memori fisik
-
- Memori utama = memori real
 - Pemakai/programmer mempunyai virtual memori yang lebih besar

Totals		Physical Memory (K)	
Handles	11102	Total	1038768
Threads	480	Available	582416
Processes	48	System Cache	214216

Commit Charge (K)		Kernel Memory (K)	
Total	354104	Total	81692
Limit	2500148	Paged	53976
Peak	547964	Nonpaged	27716

Processes: 48 | CPU Usage: 0% | Commit Charge: 345M / 2441M

Struktur Tabel Halaman



Segmentasi

- Paging tidak (biasanya) terlihat/dikendalikan oleh programmer (urusan OS dan prosesor)
- Segmentasi terlihat/dikendalikan oleh programmer
- Biasanya segmen yg berbeda dialokasikan untuk program dan data (contoh : intel x86 → CS untuk code dan DS untuk data)
- Bisa terdapat banyak segmen program dan data (setiap proses bisa mempunyai segmen code dan datanya sendiri-sendiri)

Kelebihan dari Segmentasi

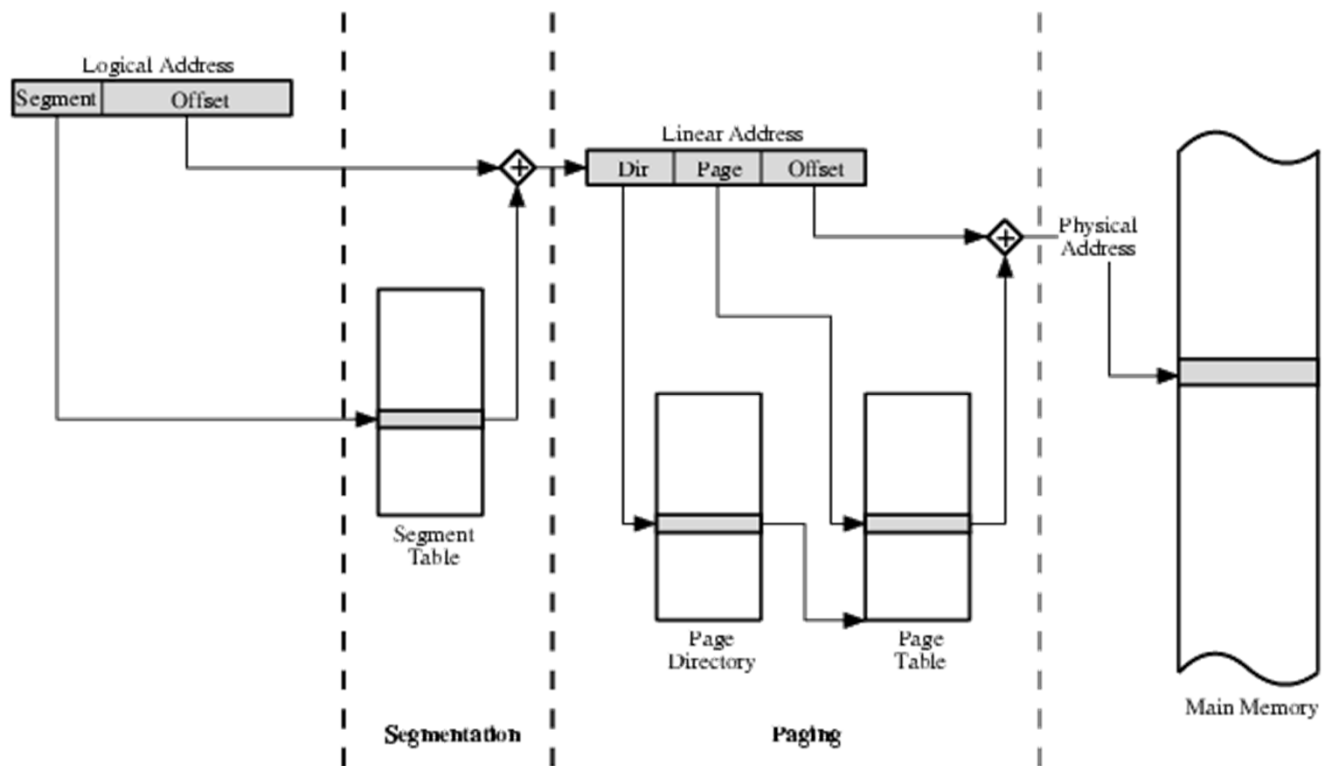
- Kemudahan penanganan struktur data yang terus berkembang
- Membuat program dapat dijalankan dan di recompile secara mandiri tanpa harus melakukan re-linking dan re-loading

- Beberapa sistem menggabungkan segmentasi dengan paging

Pentium II

- Hardware untuk segmentasi dan paging
- Unsegmented unpaged
 - Alamat virtual = alamat fisik
 - Sederhana
 - Performa tinggi
- Unsegmented paged
 - Memori terlihat sebagai ruang alamat dengan halaman linier
 - Proteksi dan manajemen menggunakan paging
 - Berkeley UNIX
- Segmented unpaged
 - Kumpulan ruang alamat lokal
 - Proteksi sampau ke level byte tunggal
- Segmented paged
 - Segmentasi digunakan untuk menentukan subjek partisi memori logik yang Segmentation used to kendali akses
 - Paging mengatur alokasi memori di partisi
 - Unix System V

Mekanisme Translasi Alamat Pentium II



Segmentasi Pentium II

- Setiap alamat virtual terdiri dari 16-bit segmen dan 32-bit offset
- 2 bits dari segmen adalah mekanisme proteksi
- 14 bit menyatakan segmen (ada 16k kemungkinan segmen)
- VM tidak tersegmen $2^{32} = 4\text{Gbyte}$
- VM tersegmen $2^{46} = 64\text{ terabyte}$
 - Bisa lebih besar – tergantung pada proses yang aktif
 - Setengahnya (8K segmen masing-masing 4Gbyte) adalah global

Proteksi Pentium II

- Proteksi bit terdiri dari 4 level
 - 0 most protected, 3 least
 - Penggunaan level tergantung software
 - Level 3 biasanya untuk aplikasi, level 1 untuk OS dan level 0 untuk kernel (level 2 tidak digunakan)
 - Level 2 dapat digunakan aplikasi yang mempunyai keamanan sendiri. Mis : database
 - Beberapa instruksi bekerja hanya di level 0

Paging Pentium II

- Segmentasi dapat tidak digunakan dalam kasus ruang alamat linier
- Tabel lookup 2 level
 - Pertama, daftar halaman
 - 1024 masukan maks
 - Membagi memori linier 4G ke 1024 grup halaman masing-masing 4Mbyte
 - Setiap tabel halaman mempunyai masukan 1024 yang setara dengan halaman 4Kbyte
 - Dapat menggunakan satu halaman daftar untuk semua proses, satu per proses atau campuran
 - Daftar halaman untuk proses yang sedang dijalankan selalu berada di memori
 - Menggunakan TLB yang memegang 32 halaman tabel masukan
 - 2 ukuran halaman 4k atau 4M

Translasi Alamat 32 bit PowerPC

