

MODUL SISTEM OPERASI



PENGELOLAAN MEMORI DENGAN PAGING

SISTEM PAGING

Mengimplementasikan ruang alamat besar pada memori kecil menggunakan *index register, base register, dan segment register.*

Virtual Address

Real Address

Page

Page Frame

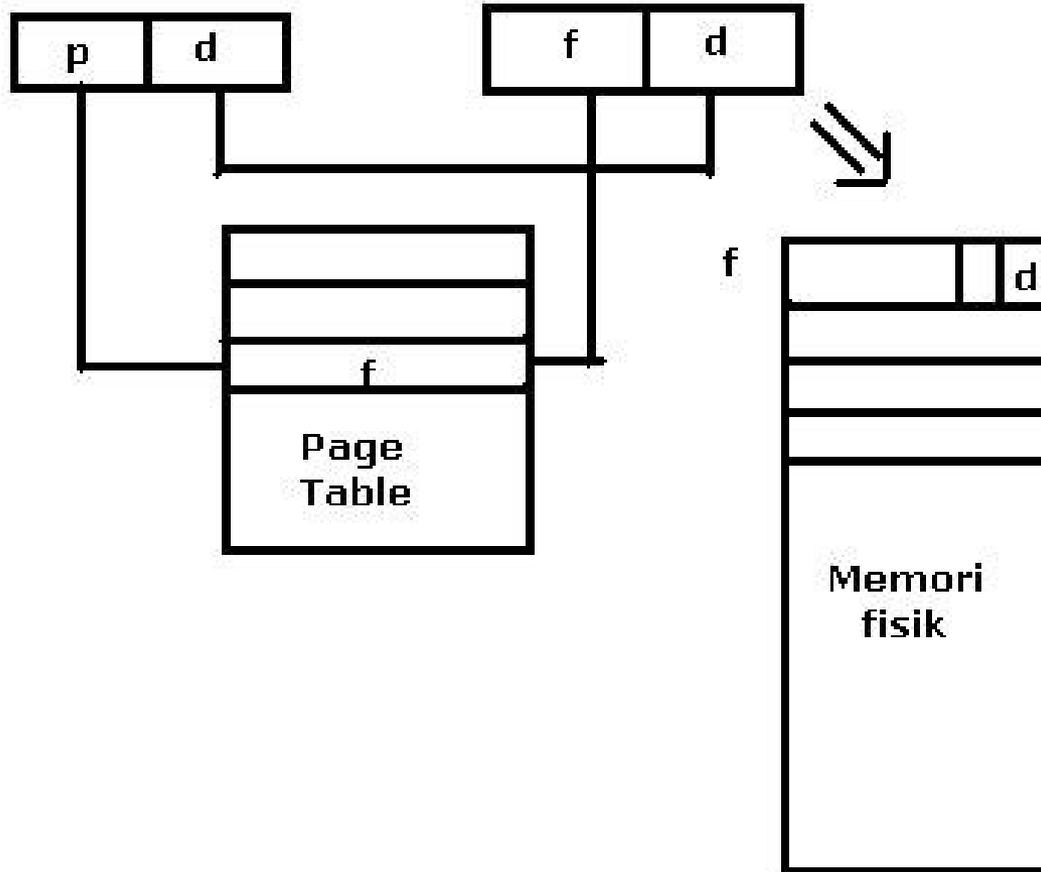
Page Fault

MMU

Sistem Paging

Alamat Logik

Alamat fisik



Sistem Paging

Alamat Maya

Alamat yang dihasilkan perhitungan menggunakan index register, base register, dan segment register.

Alamat Nyata

Alamat di memori fisik.

Page

Unit terkecil pada ruang alamat maya (*virtual address space*).

Sistem Paging

Page Frame

Unit terkecil pada ruang alamat fisik (*real address space*).

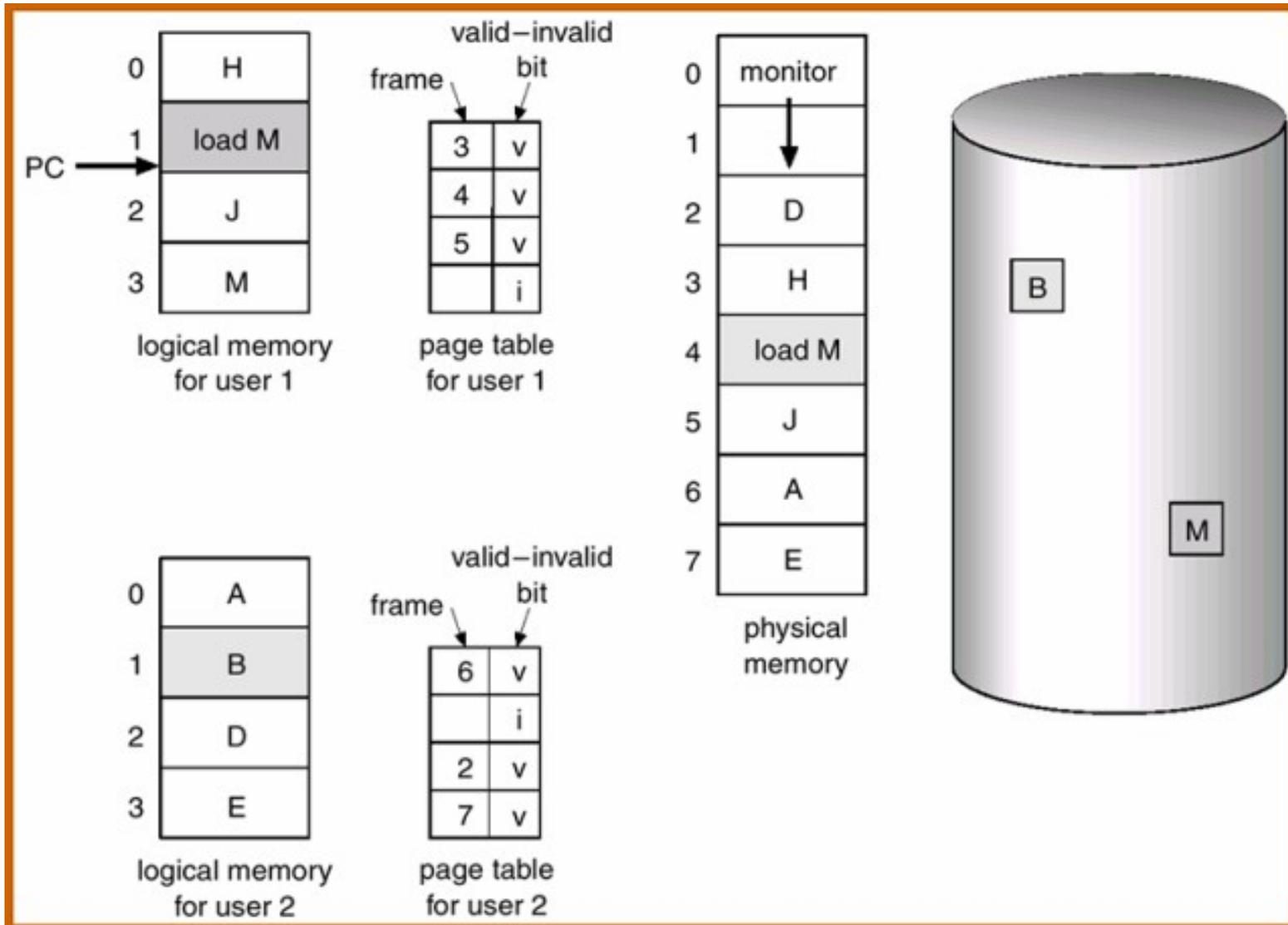
Page Fault

Exception untuk permintaan alokasi “page” ke memori.

Memory Management Unit (MMU)

Kumpulan chip yang memetakan alamat maya ke alamat fisik.

Rutinitas Sistem Paging



Rutinitas Sistem Paging

1. Mencari lokasi dari halaman yang diinginkan pada *disk*.
2. Mencari *frame* yang kosong:
 - a. Jika ada, maka gunakan *frame* tersebut.
 - b. Jika tidak ada, maka :
tentukan *frame* yang tidak sedang dipakai
atau yang tidak akan digunakan
dalam jangka waktu lama, lalu
kosongkan *frame* tersebut.

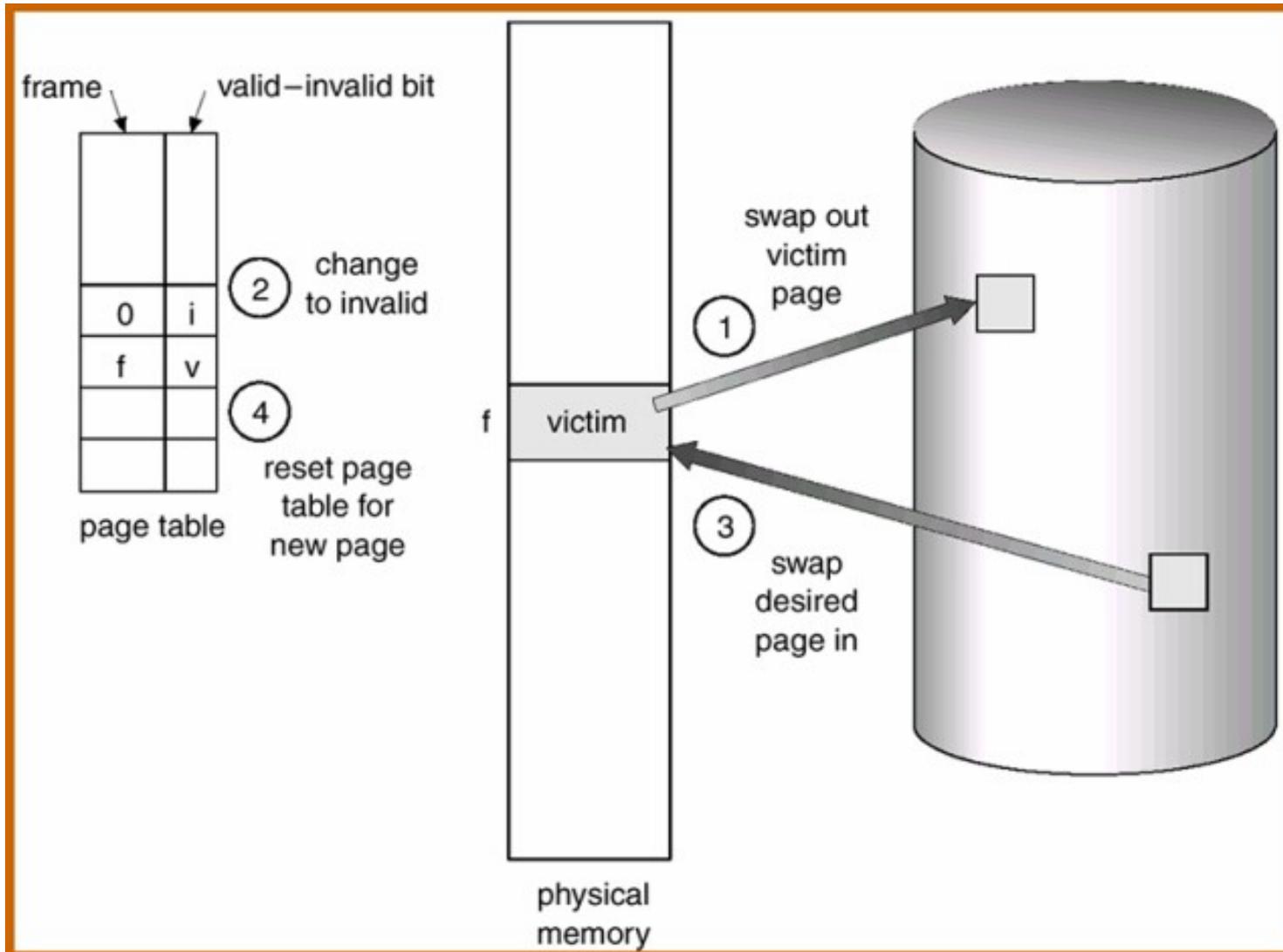
Rutinitas Sistem Paging

- > Gunakan algoritma pemindahan halaman untuk menentukan *frame* yang akan dikosongkan.
- > Usahakan agar tidak menggunakan frame yang akan digunakan dalam waktu dekat.
- > Jika terpaksa, maka sebaiknya segera masukkan kembali frame tersebut agar tidak terjadi *overhead*.

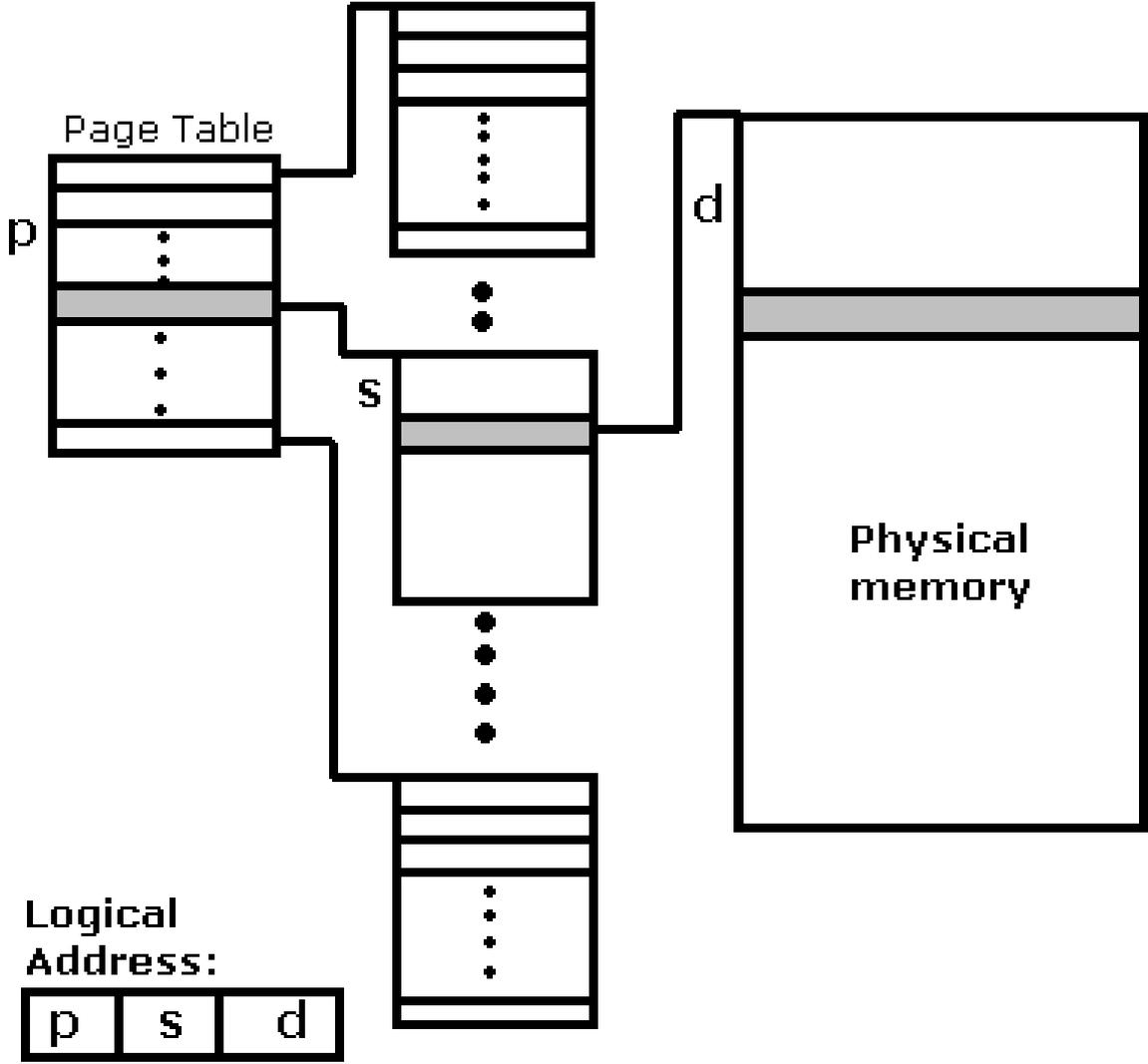
Rutinitas Sistem Paging

- c. Tulis halaman yang dipilih ke *disk*, ubah tabel halaman dan tabel *frame*.
3. Membaca halaman yang diinginkan ke dalam *frame* kosong yang baru.
4. Mengulangi proses pengguna dari awal.

Rutinitas Sistem Paging

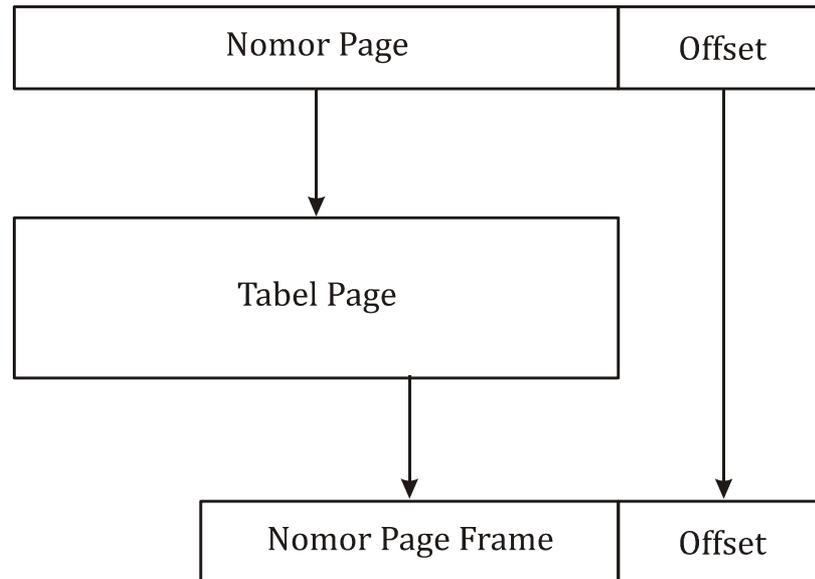


Hirarki Sistem Paging



Manajemen memory unit

Pemetaan memori maya ke memori fisik dan menerbitkan *exception* adanya *page fault* yang melewatkan ke sistem operasi yang menanganinya.



Komponen Internal MMU

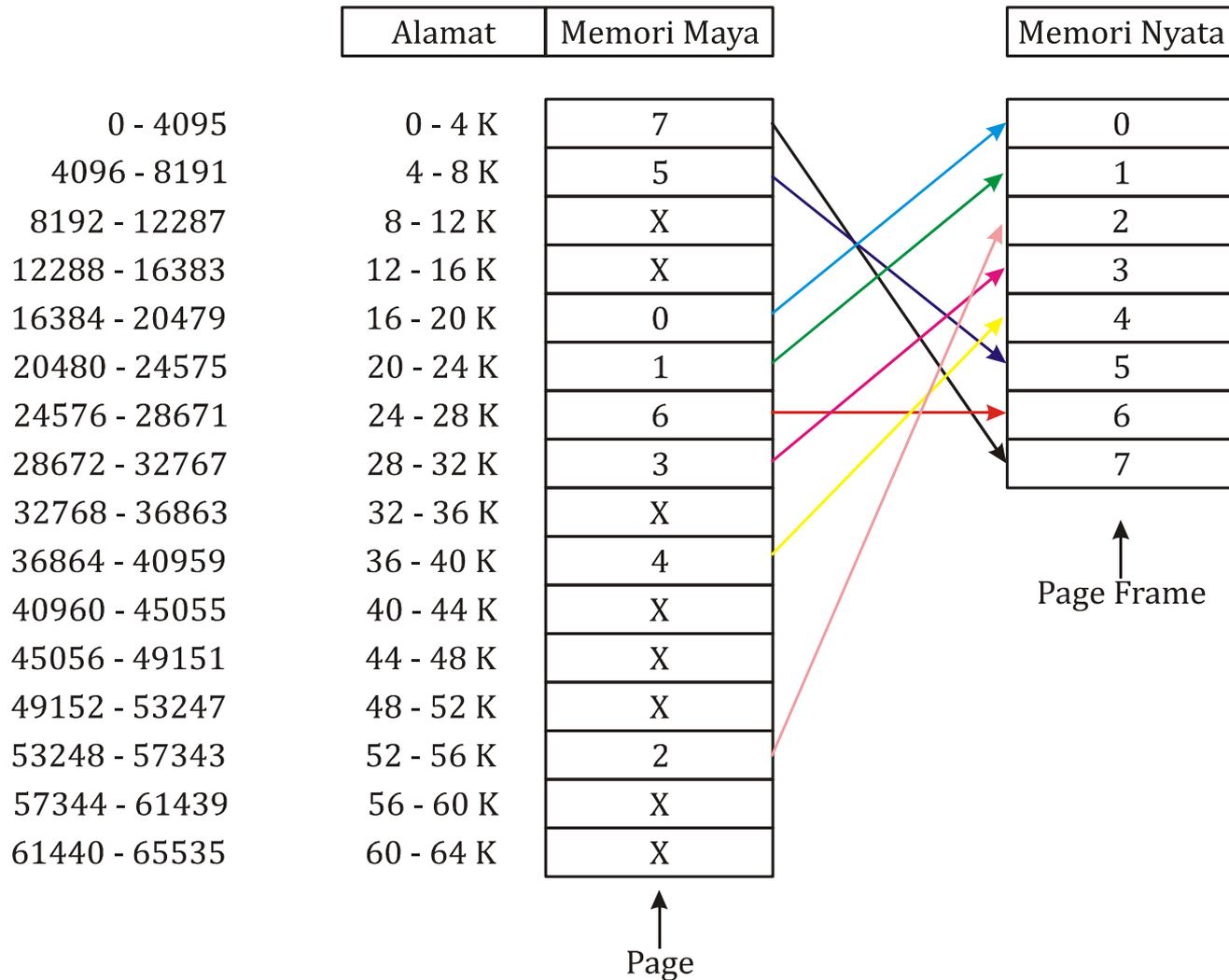
Manajemen memory unit

1. Nomor page maya digunakan sebagai indeks ke tabel page untuk menemukan isian page maya.
2. Dari isian tabel page dapat diketahui, apakah page dipetakan ke memori fisik (dengan memeriksa presen/absent bit).

Manajemen memory unit

3. Apabila alamat terdapat di memori fisik maka isian tabel page memuat nomor page frame. Nomor page frame di tabel page dikopi sebagai bit-bit berorder tinggi di register alamat fisik dan ditambah offset di alamat maya.
4. Bila alamat tidak ada di memori fisik maka MMU menerbitkan page fault.

Manajemen memory unit



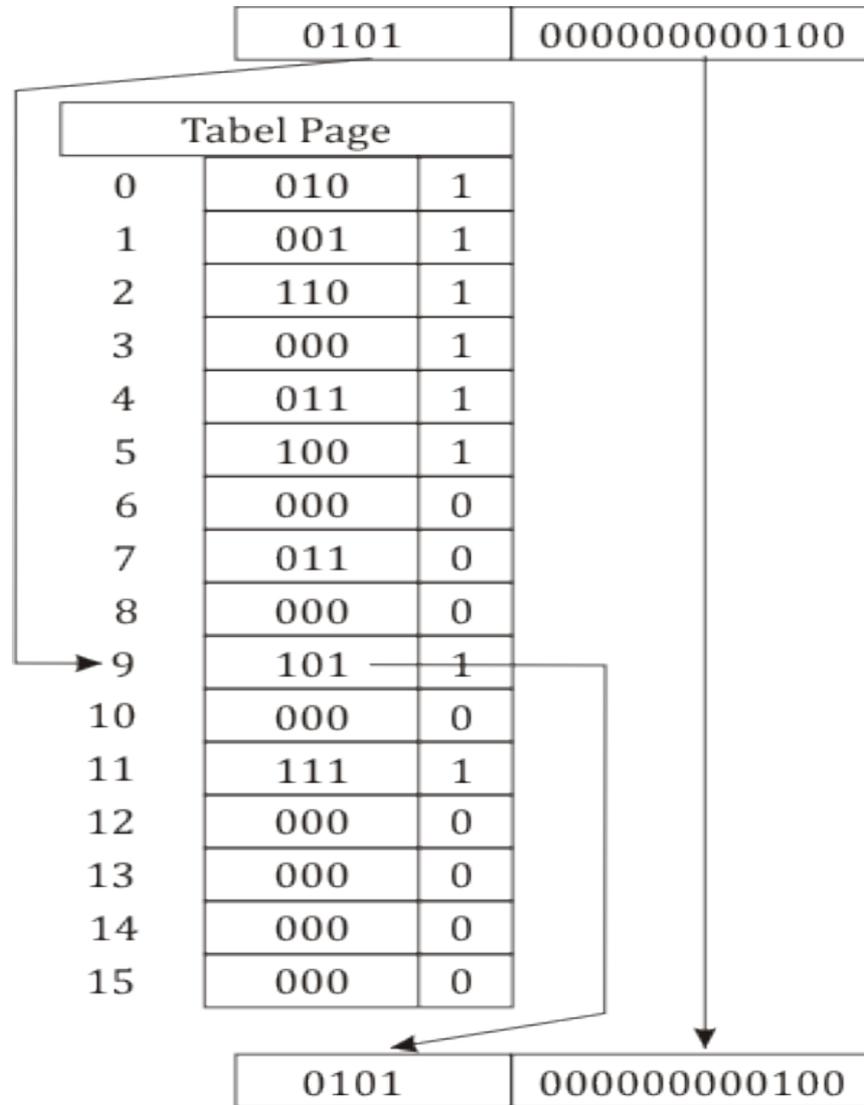
Relasi antara alamat maya dan alamat fisik

Manajemen memory unit

Contoh instruksi : **MOV REG, 0x08**

- Alamat maya 8 dikirim ke MMU
- MMU mengetahui alamat 8 di page 0 (page 0 memuat alamat maya 0-4095)
- Dari tabel, page 0 dipetakan ke frame 7 (page 7 adalah alamat fisik 28672-32768)
- MMU mentransformasikan alamat 8 sebagai (28672+8=28680)
- MMU mengeluarkan alamat 28680 ke bus

Manajemen memory unit



Cara Kerja Pemetaan MMU

Komponen MMU

1. Register Alamat Maya

- ❑ Menyimpan alamat maya yang diacu
- ❑ Nilai di register alamat maya dibagi dua :
 - Bit berorder tinggi menyatakan nomor **page maya**
 - Bit-bit sisa adalah **offset** alamat maya

Komponen MMU

2. Page Table

- Present/absent bit
(bernilai 1 jika page di memori fisik, bernilai 0 jika tidak)
- Nomor page frame
Berisi nomor page frame dimana page berada

Komponen MMU

3. Register Alamat Fisik

- ⦿ Menyimpan alamat fisik yang disinyalkan ke bus.
- ⦿ Nilai di register alamat fisik dibagi menjadi :
 - ✓ Bit berorder tinggi menyatakan nomor page frame
 - ✓ Bit-bit sisa adalah offset alamat fram

Penggantian Page

Algoritma Penggantian Random

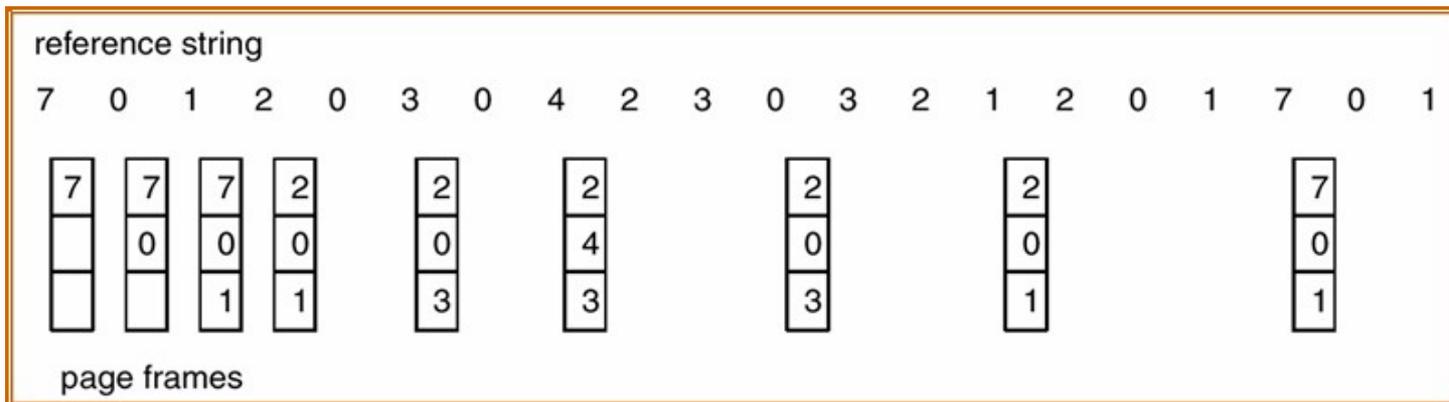
- ➔ Setiap terjadi page fault, penggantian page dipilih secara acak.
- ➔ Tidak memakai informasi apapun untuk menentukan page yang akan diganti.
- ➔ Semua page di memori utama dianggap memiliki bobot yang sama.
- ➔ Dapat memilih sembarang page termasuk page yang sedang diacu.

Penggantian Page

Algoritma Penggantian Optimal

➔ Memilih page yang berpeluang dipakai kembali di masa datang yang paling kecil.

➔ Strategi ini menghasilkan jumlah page fault sedikit tapi tidak mungkin diterapkan.

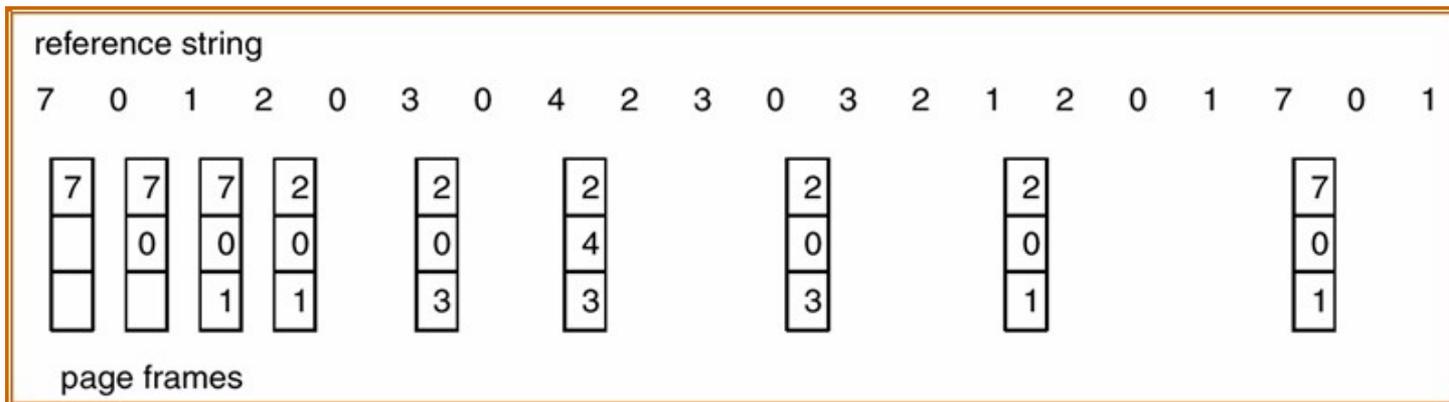


Penggantian Page

Algoritma Penggantian Optimal

➔ Memilih page yang berpeluang dipakai kembali di masa datang yang paling kecil.

➔ Strategi ini menghasilkan jumlah page fault sedikit tapi tidak mungkin diterapkan.



KUIS

1. Sebutkan faktor-faktor yang mempengaruhi proses Swapping?
2. Bagaimanakah konsep dasar dari page replacement dan peristiwa Page Fault ?
3. Jelaskan cara kerja unit manajemen memori (MMU)?