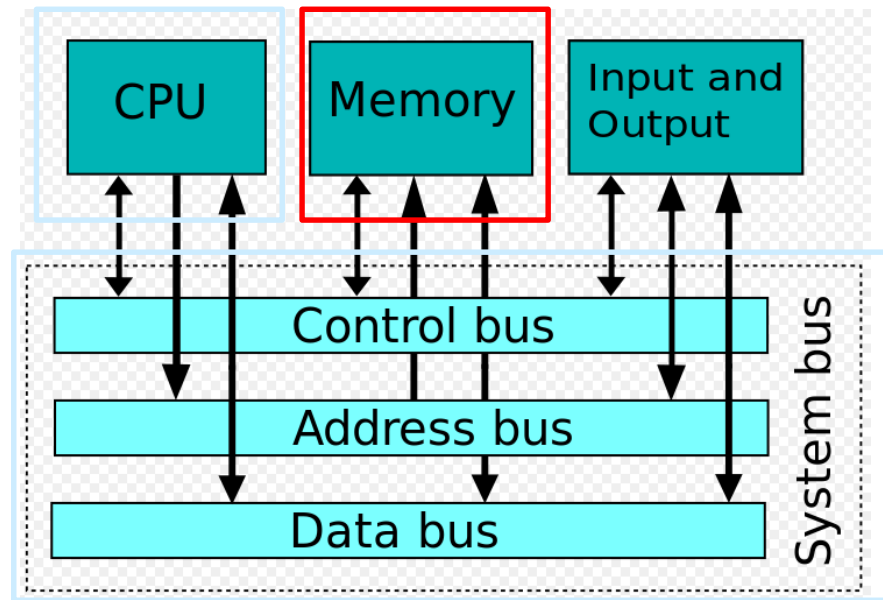


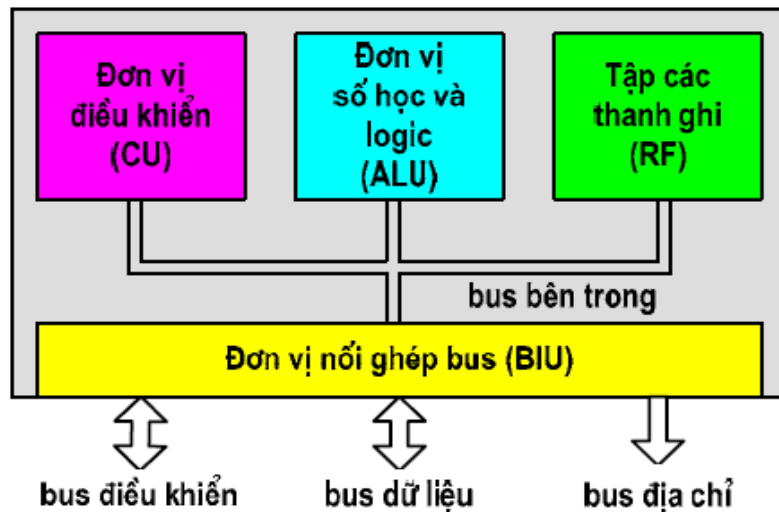
Chương 5

Hệ thống nhớ máy tính



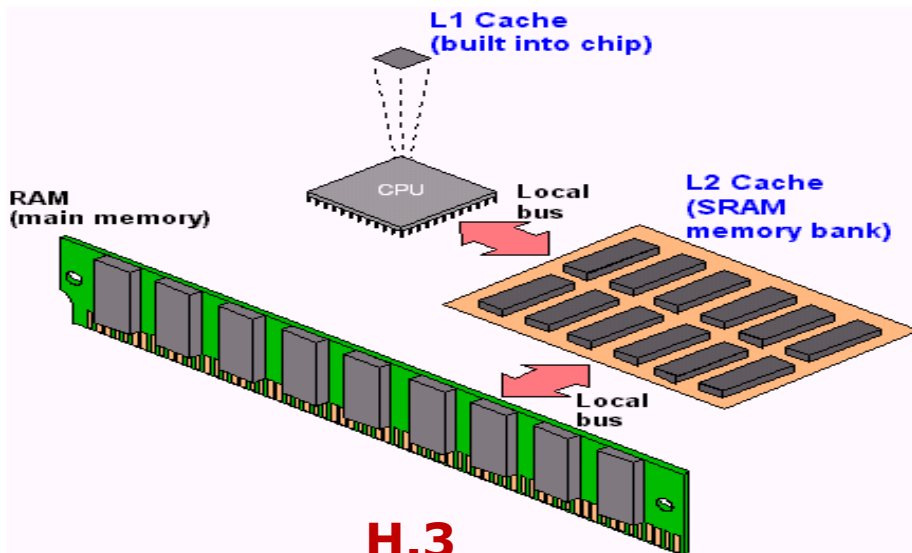
Giảng viên: ThS. Phan Như Minh

Quan sát và định vị các thành phần nhớ?

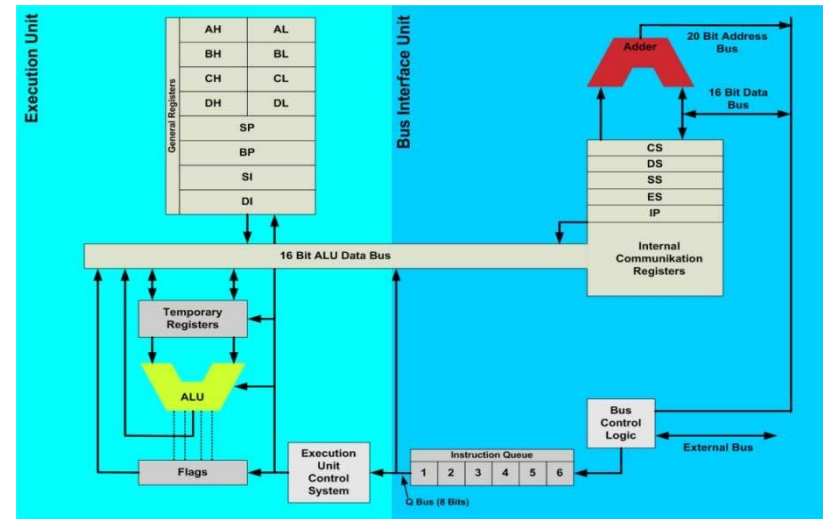


H.1

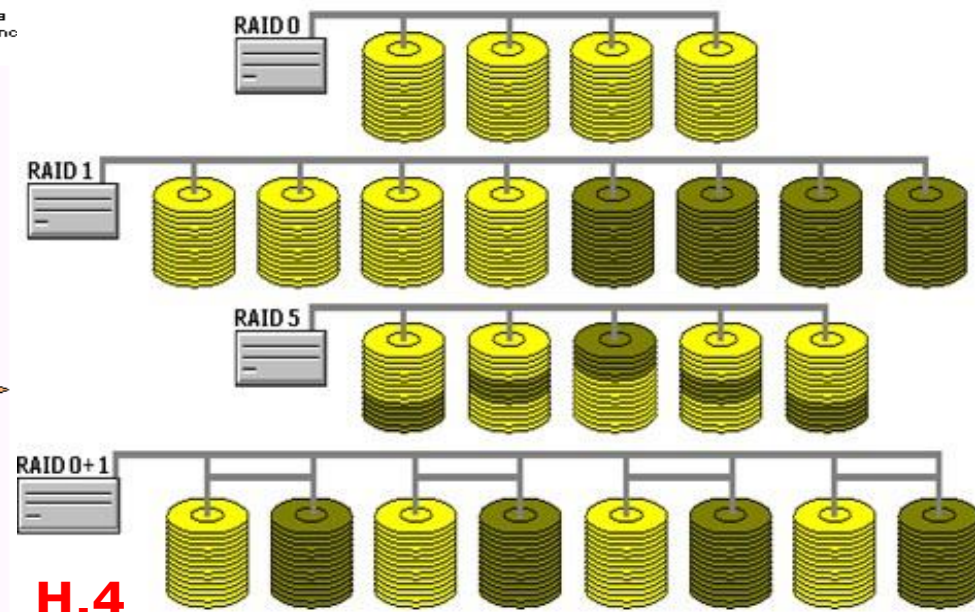
From Computer Desktop Encyclopedia
© 1999 The Computer Language Co., Inc



H.3



H.3



H.4

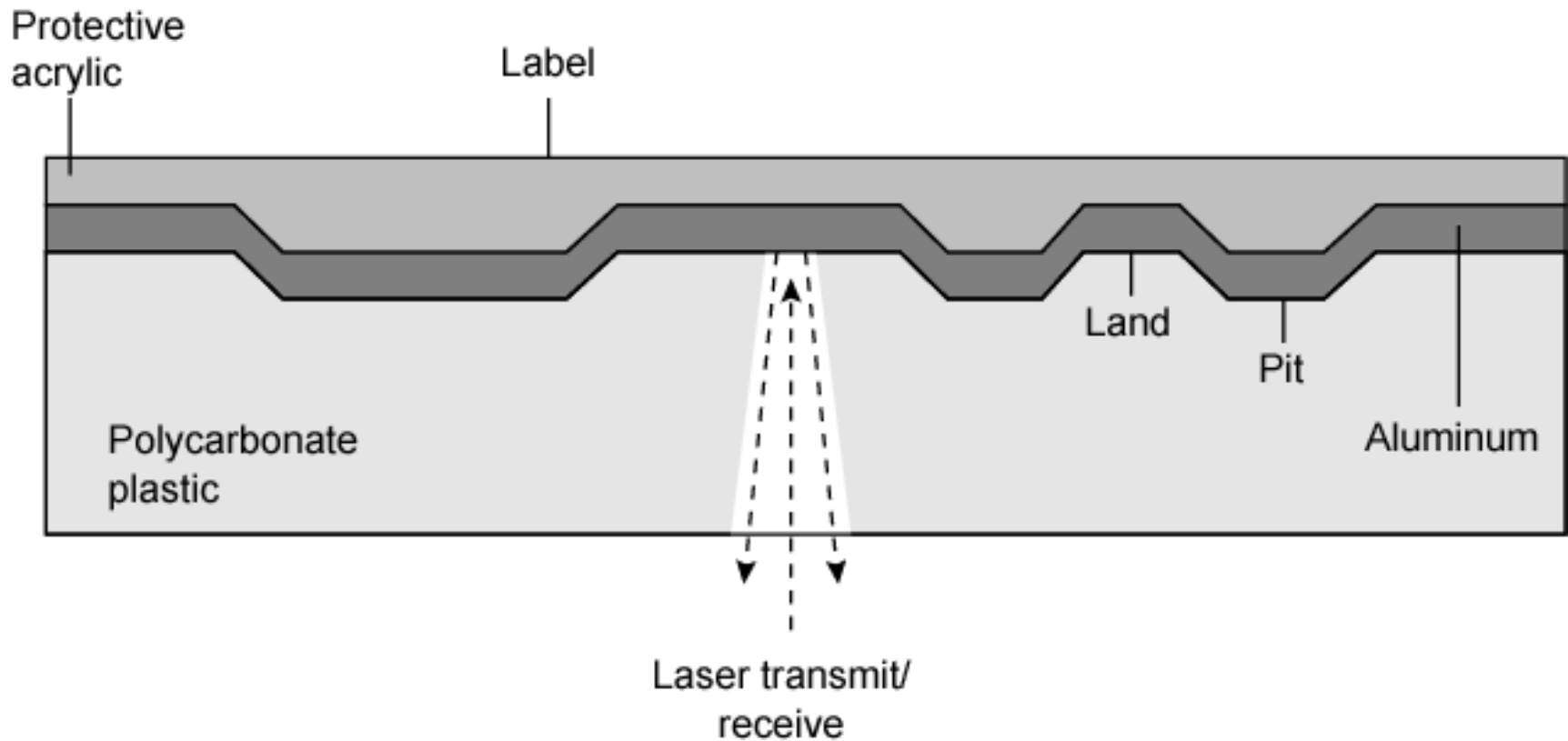
3. Đĩa quang

- ❖ CD-ROM (Compact Disk ROM)
- ❖ CD-R (Recordable CD)
- ❖ CD-RW (Rewriteable CD)
- ❖ Dung lượng thông dụng 650MB
- ❖ Ổ đĩa CD:
 - Ổ CD-ROM
 - Ổ CD-Writer: Ghi một phiên hoặc ghi nhiều phiên
 - Ổ CD-RW
- ❖ Tốc độ đọc cơ sở 150KByte/s.
- ❖ Tốc độ bội, ví dụ: 48x, 52x,...

CD-ROM

- ❖ Ban đầu chỉ dùng cho đĩa audio
- ❖ Dung lượng 650 MBs, thời gian lưu trữ 70 phút
- ❖ Dữ liệu được chứa trên các chỗ khuyết trên mặt đĩa, đọc dựa trên cơ chế phản xạ của tia laser
- ❖ Dữ liệu lưu trữ dưới dạng các bit
- ❖ Dữ liệu được ghi đều nhau, toàn đĩa là một track duy nhất
- ❖ Đĩa quay với tốc độ thẳng cố định.

Hoạt động của đĩa CD



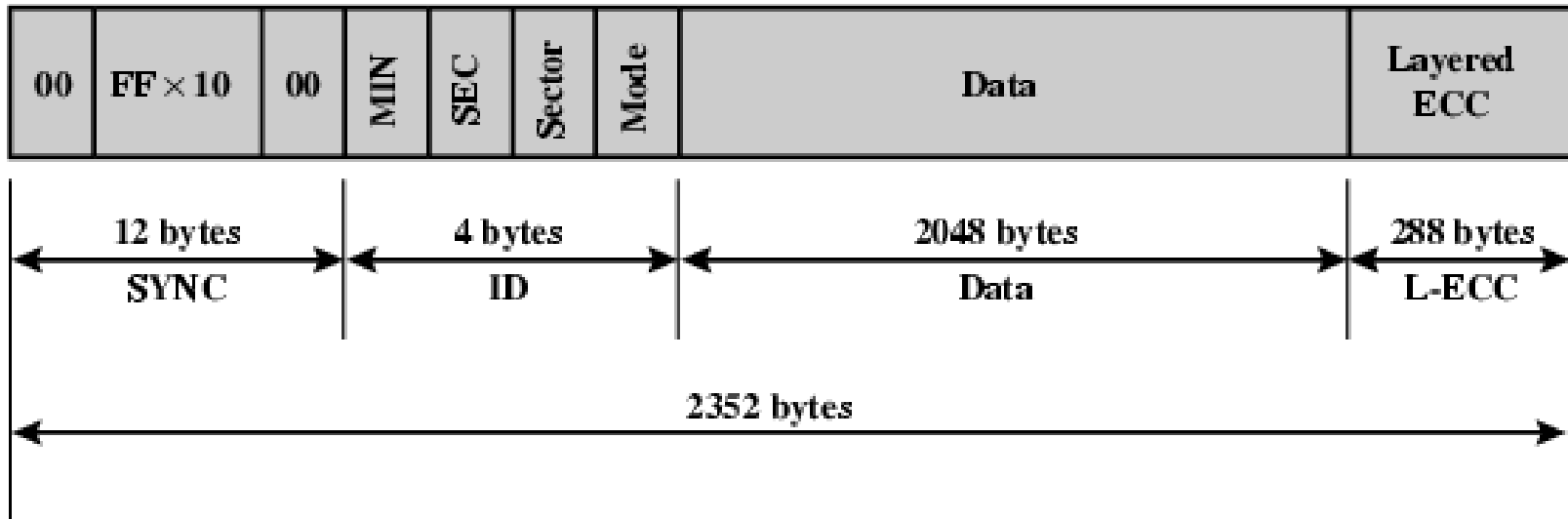
❖ Tốc độ đọc đĩa Audio:

- Tốc độ tuyến tính cố định (1.2 m/s)
- Track theo dạng xoắn ốc (5.27 km)
- Thời gian đọc xong đĩa 4391 giây = 73.2 phút

❖ Tốc độ chuyển dữ liệu: 24X, 32X, 52X

- 1X = 150KB/s

Định dạng đĩa CD



- ❖ Mode 0: dữ liệu trống
- ❖ Mode 1: 2048 byte dữ liệu+ 288 byte mã sửa lỗi
- ❖ Mode 2: 2336 byte dữ liệu

3. Đĩa quang (tiếp)

❖ DVD

- Digital Video Disk: chỉ dùng trên ổ đĩa xem video
- Digital Versatile Disk: ổ trên máy tính
- Ghi một hoặc hai mặt
- Một hoặc hai lớp trên một mặt
- Thông dụng: 4,7GB/lớp

4. Flash Disk

- ❖ Thường kết nối qua cổng USB
- ❖ Không phải dạng đĩa
- ❖ Bộ nhớ bán dẫn cực nhanh (flash memory)
- ❖ Dung lượng tăng nhanh
- ❖ Thuận tiện

5. Băng từ (Magnetic Tape)

- ❖ Truy xuất tuần tự
- ❖ Chậm
- ❖ Giá thành thấp
- ❖ Dùng nhiều trong sao lưu và phục hồi



a. Bộ nhớ thật:

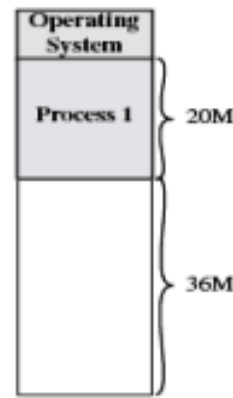
- ❖ Không gian địa chỉ trong chương trình trùng với không gian địa chỉ trong bộ nhớ. Cho phép người lập trình truy cập trực tiếp vào 1 ô nhớ → Khó bảo vệ bộ nhớ.
- ❖ Khi thi hành, hệ điều hành nạp toàn bộ chương trình vào bộ nhớ (nạp trước) → bộ nhớ máy tính phải đủ lớn để chạy các CT lớn
- ❖ Chương trình được cấp phát 1 vùng nhớ có địa chỉ liên tục (cấp phát liên tục). HĐH sẽ thu hồi vùng nhớ sau khi chương trình kết thúc
- ❖ Để thực hiện đa chương, HĐH cần chia BN ra nhiều vùng (partition), mỗi vùng cấp phát cho 1 CT

❖ Khi bộ nhớ đầy

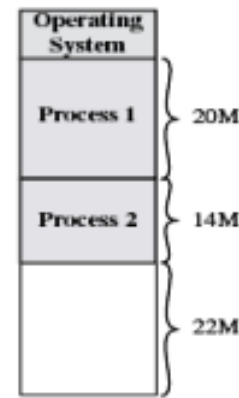
- HĐH không cấp tiếp, các CT phải chờ đến khi có 1 vùng nhớ trống
- HĐH cấp tiếp: Cần kỹ thuật trao đổi (swapping) để ghi tạm vùng nhớ của 1 CT khác ra BN ngoài, lấy chỗ trống cấp cho CT mới



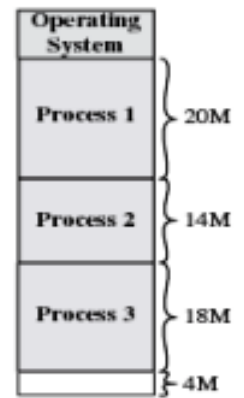
(a)



(b)



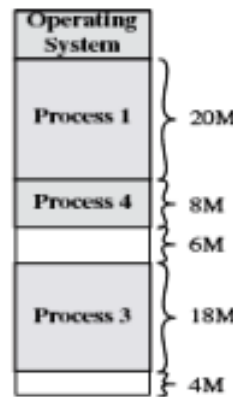
(c)



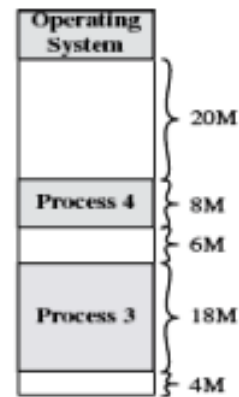
(d)



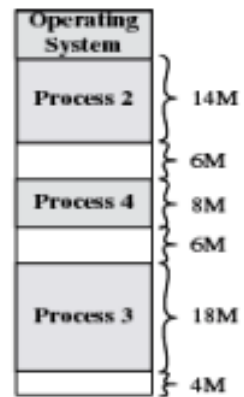
(e)



(f)



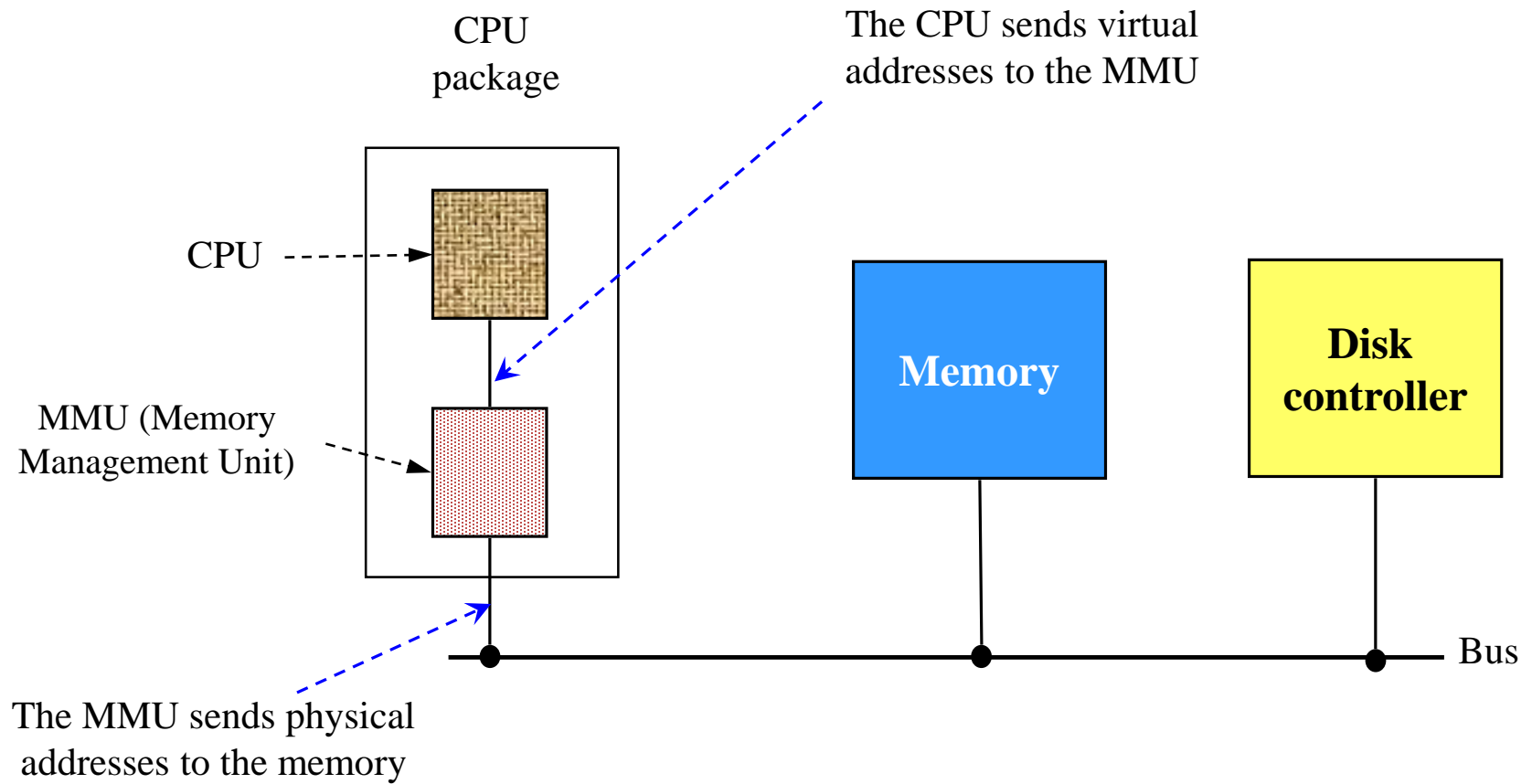
(g)



(h)

b. Bộ nhớ ảo:

- ❖ Không gian địa chỉ trong CT (địa chỉ ảo) được tách biệt với không gian địa chỉ trong BN (địa chỉ thực) → CPU và HĐH sẽ phối hợp để ánh xạ (mapping) địa chỉ ảo trong CT thành địa chỉ thật trong BN
- ❖ Việc ánh xạ và quản lý BN ảo được thực hiện qua đơn vị MMU (Memory Management Unit)

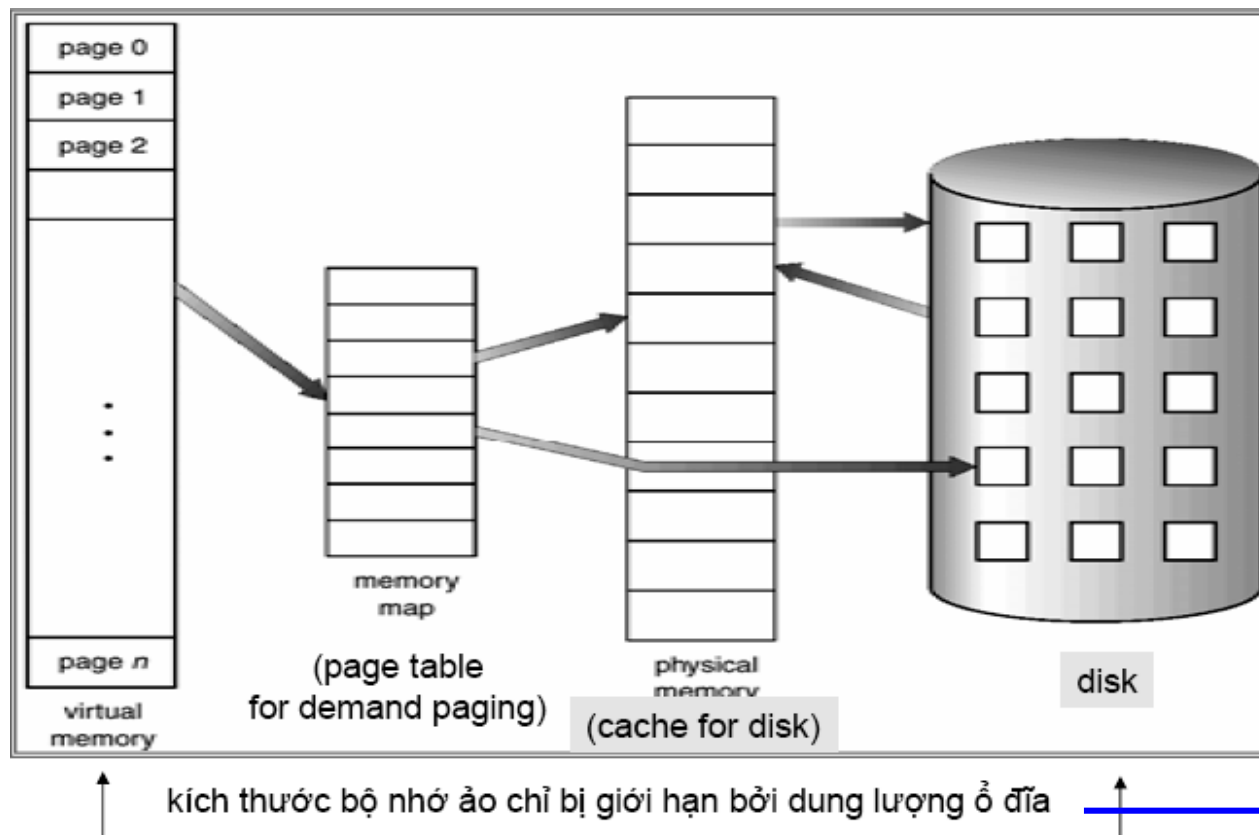


6. VIRTUAL MEMORY

- ❖ Khi thi hành, hệ điều hành chỉ nạp các phần cần thiết của CT vào bộ nhớ (nạp theo yêu cầu), không cần nạp toàn bộ CT → tránh lãng phí BN
- ❖ Các CT được cấp phát nhiều vùng nhớ có địa chỉ tách biệt nhau (cấp phát không liên tục).
 - Sử dụng kỹ thuật trao đổi (swapping) để ghi tạm thời các vùng nhớ chưa cần đến ra BN ngoài (swap-out) để lấy chỗ trống nạp thông tin cần thiết vào BN (swap-in) khi cần đến
- ❖ BN ngoài thông dụng là đĩa cứng
- ❖ Có 2 kỹ thuật BN ảo:
 - Kỹ thuật phân trang: Kích thước các vùng nhớ cố định
 - Kỹ thuật phân đoạn: Kích thước các vùng nhớ thay đổi

6. Bộ nhớ ảo (Virtual Memory)

Bộ nhớ ảo lớn hơn bộ nhớ vật lý



Vì vẫn cần bộ nhớ ngoài để lưu trữ ctr & dữ liệu khi chưa được đưa vào bộ nhớ trong

6. Bộ nhớ ảo (Virtual Memory)

Bộ nhớ ảo dạng phân trang:

- Tổ chức phân trang
- Chuyển đổi địa chỉ
- Nạp trang
- Thay thế trang

6. Bộ nhớ ảo (Virtual Memory)

❖ Tổ chức phân trang

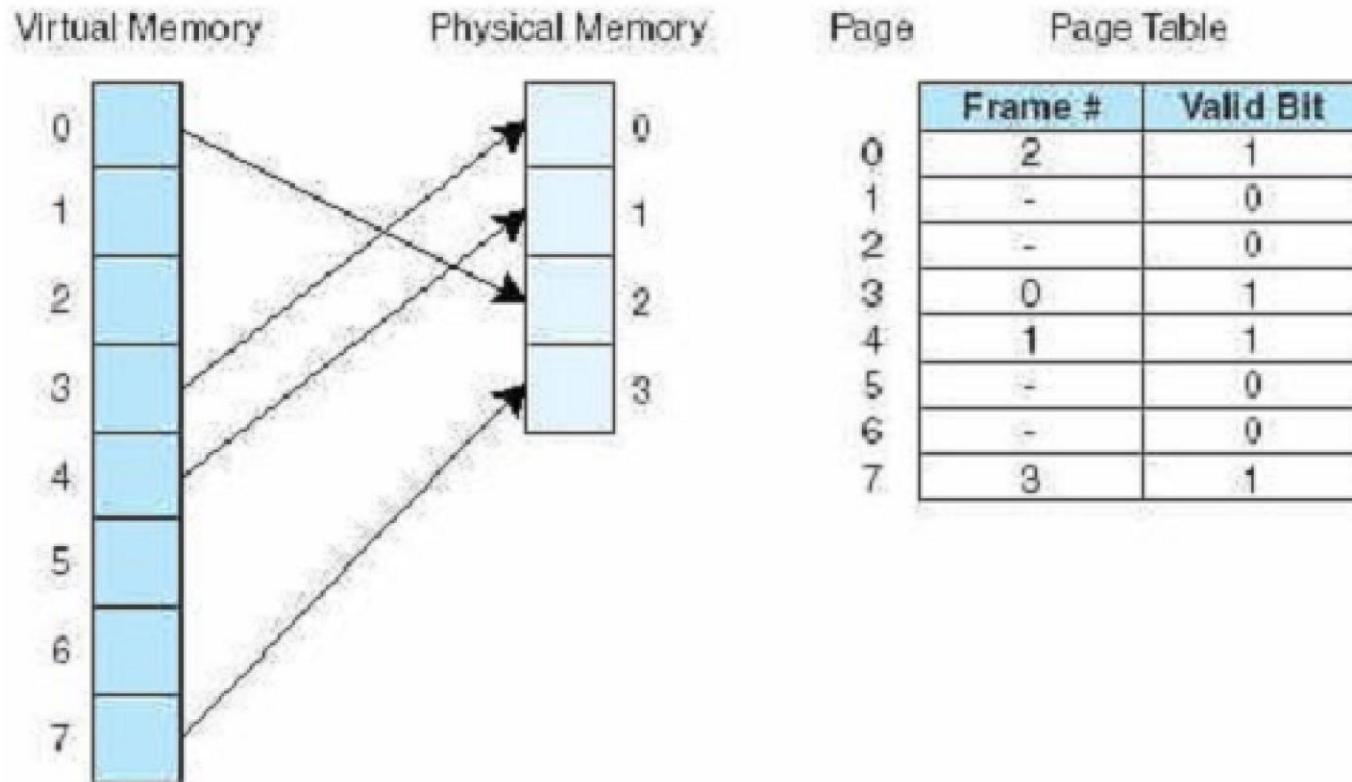
- Không gian địa chỉ ảo chia thành các trang (page)
- Bộ nhớ được chia thành các khung trang (page frame), chứa được một trang
 - Các trang của một chương trình khi ở trên bộ nhớ không cần ở các vị trí liên tục
 - Số trang >> số khung
- Có bảng trang (page table) quản lý các trang

6. Bộ nhớ ảo (Virtual Memory)

❖ Bảng trang

- Có N dòng, với N là số trang
 - → mỗi dòng tương ứng với 1 trang
- Cấu trúc 1 dòng:
 - Valid bit
 - Valid = 0 nếu trang chưa có trên bộ nhớ
 - Valid = 1 nếu trang đang có trên bộ nhớ
 - Frame Number (Frame #)
 - Số thứ tự khung trang đang chứa trang

6. Bộ nhớ ảo (Virtual Memory)



Bộ nhớ ảo có 8 trang, bộ nhớ vật lý có 4 khung
Bảng trang có 8 phần tử

6. Bộ nhớ ảo (Virtual Memory)

❖ Chuyển đổi địa chỉ

- Địa chỉ ảo theo phân trang
 - Page Number: số thứ tự trang
 - Offset: địa chỉ trong trang

- Địa chỉ vật lý

Page Number	Offset
-------------	--------

6. Bộ nhớ ảo (Virtual Memory)

❖ Chuyển đổi địa chỉ

- Địa chỉ ảo theo phân trang

- Page Number: số thứ tự trang
- Offset: địa chỉ trong trang

Page Number	Offset
-------------	--------

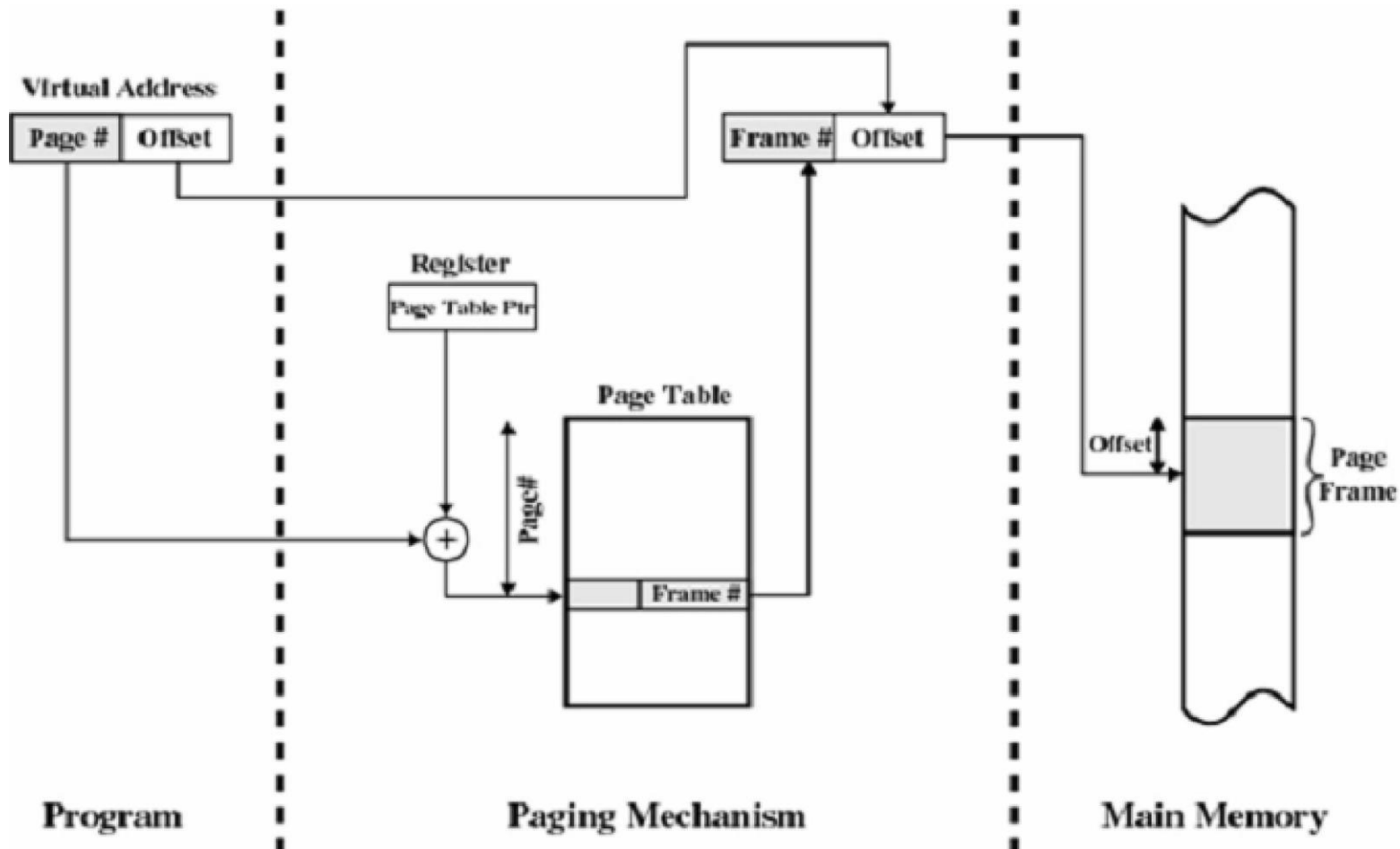
- Địa chỉ vật lý

- Page Number: số thứ tự khung trang
- Offset: địa chỉ trong khung trang

Frame Number	Offset
--------------	--------

6. Bộ nhớ ảo (Virtual Memory)

❖ Sơ đồ chuyển đổi địa chỉ



Cấu trúc bảng trang

Virtual Address

Page #	Offset
--------	--------

(hash)

Hash Table

Page Table
Page # Entry Chain

Inverted Page Table

Frame #	Offset
---------	--------

Real Address

(hash)

Hash Table

Page Table
Page # Entry Chain

Inverted Page Table

Frame #	Offset
---------	--------

Real Address

6. Bộ nhớ ảo (Virtual Memory)

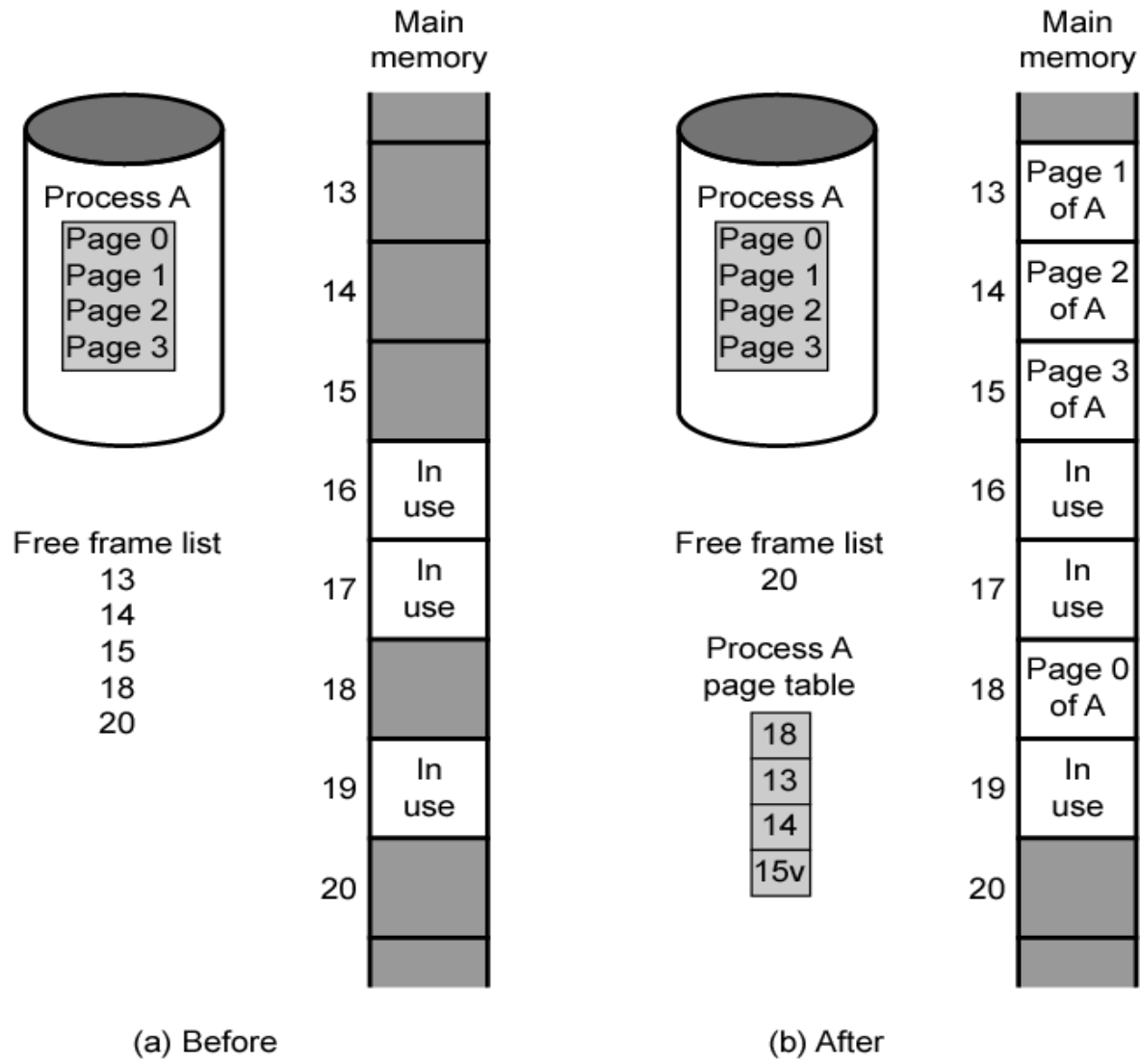
❖ Truy xuất dữ liệu theo địa chỉ ảo

- Tách page # và offset từ địa chỉ ảo
 - Chuyển page # thành frame # bằng cách truy xuất bảng trang
 - A. Tìm phần tử quản lý trang
 - B. Kiểm tra valid bit
1. Valid = 1
- a. Thay page # bằng frame #
 - b. Truy xuất dữ liệu trên khung với vị trí offset

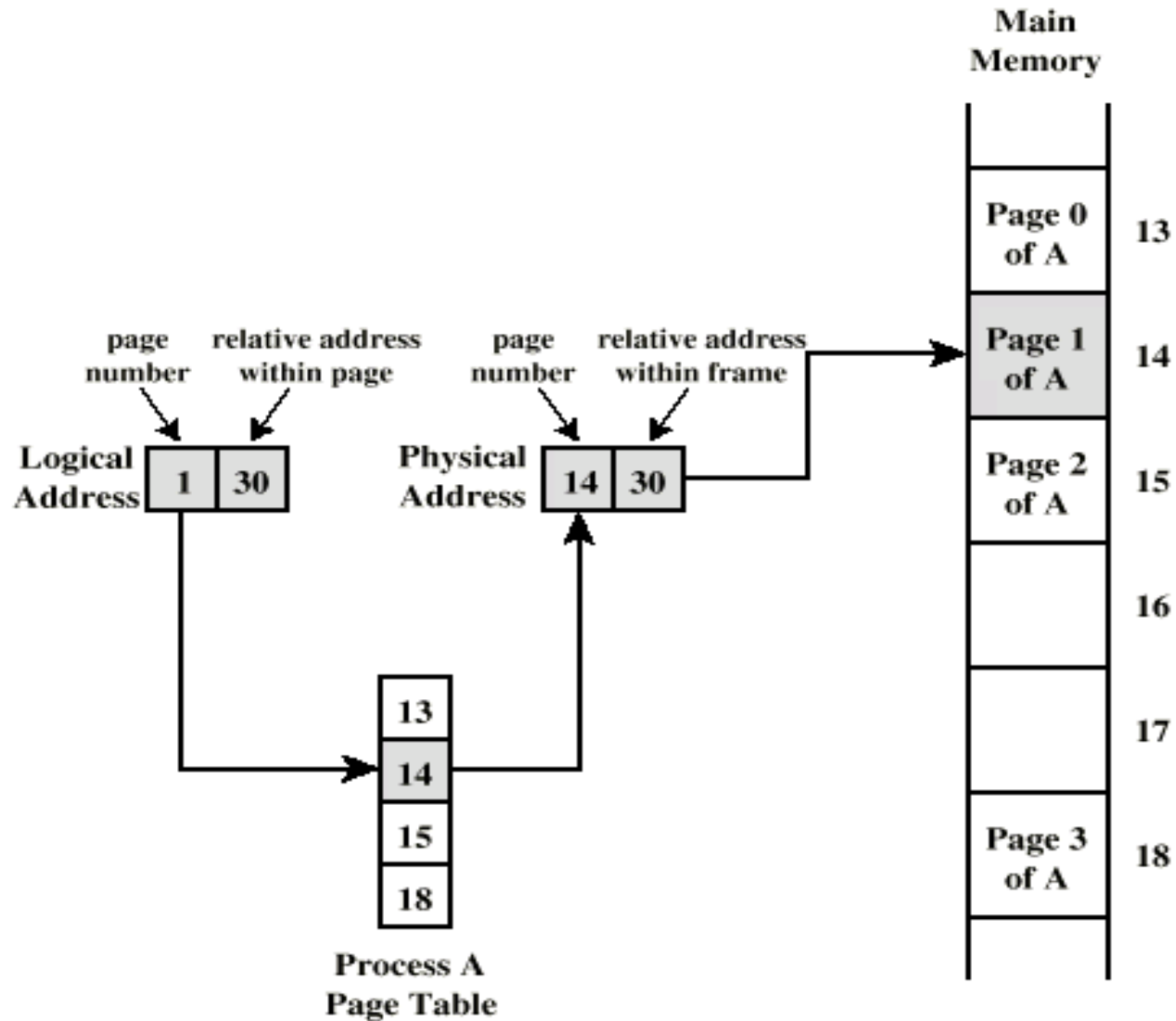
6. Bộ nhớ ảo (Virtual Memory)

❖ Truy xuất dữ liệu theo địa chỉ ảo (tt)

- 2. Valid = 0 \rightarrow lỗi trang
 - a. Tìm trang trên đĩa
 - b. Tìm một khung trống (có thể phải thay thế trang nếu các khung đầy)
 - c. Sao chép trang vào khung trống
 - d. Cập nhật bảng trang (valid = 1, frame # mới)
 - e. Thực hiện truy xuất như bước 1



Địa chỉ logic và địa chỉ vật lý của phân trang

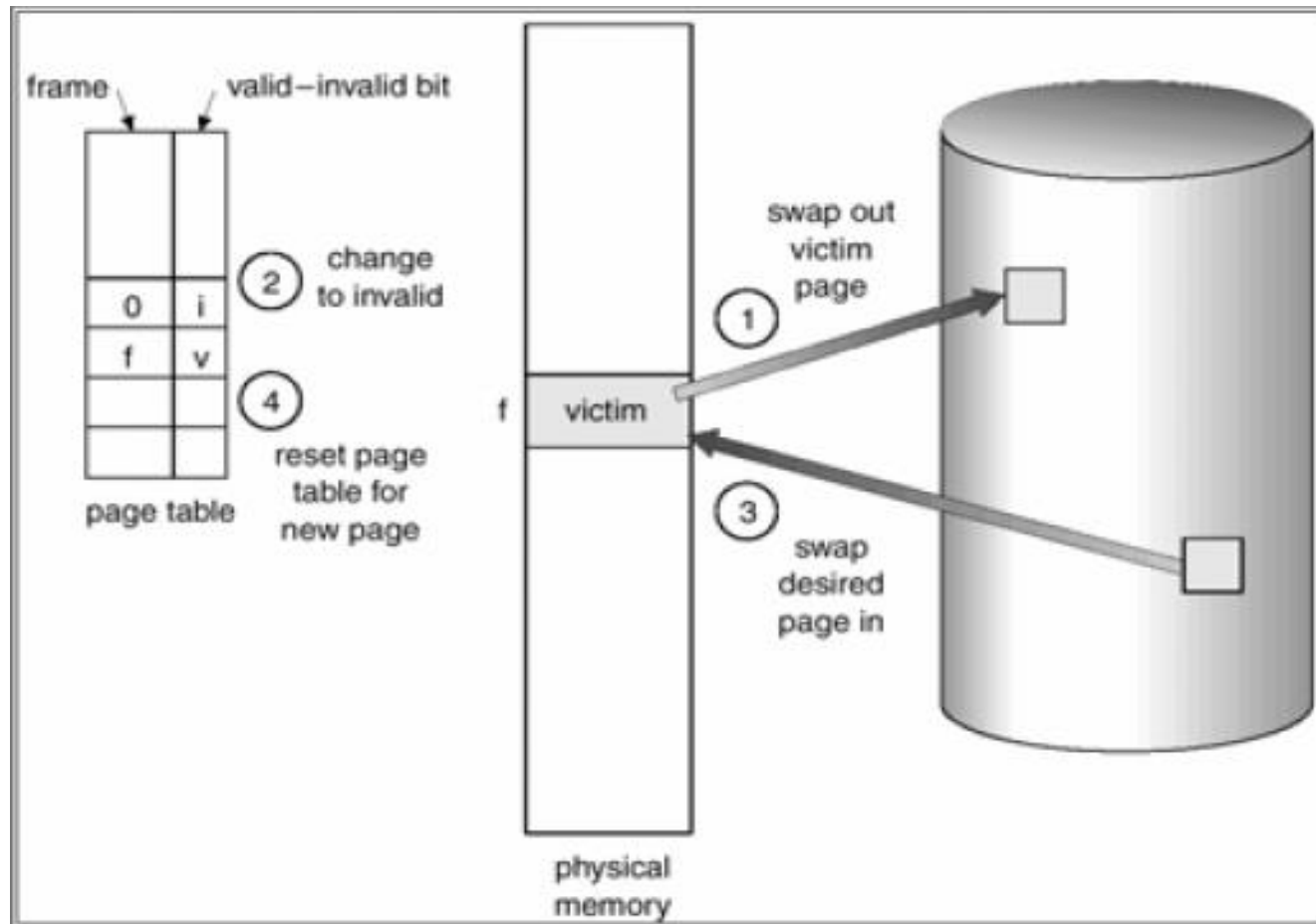


- Có 2 phương pháp:
 - Nạp theo yêu cầu: Nạp trang khi có lỗi trang
 - Nạp trước: Nạp trước các trang theo các điều kiện xác định

Các bước thay trang:

1. Tìm vị trí của trang được yêu cầu trên đĩa.
2. Tìm một frame rồi:
Nếu có frame rồi thì sử dụng nó.
Nếu không có, sử dụng một giải thuật thay trang để giải phóng một frame *nạn nhân(victim)*.
3. Đọc trang được yêu cầu vào frame rồi. Cập nhật bảng phân trang và bảng quản lý frame rồi.
4. Khởi động lại tiến trình.

Thay trang



Thay trang: Các giải thuật thay trang

Mục đích: tỷ lệ page-fault thấp nhất.

Đánh giá giải thuật bằng cách chạy nó trên một chuỗi riêng biệt các **tham chiếu bộ nhớ** và tính số page faults trên chuỗi đó.

Một số giải thuật:

FIFO(First In First Out)

LRU(Least Recently Used)

Giải thuật tương tự LRU

Thay trang: Giải thuật FIFO

Ý tưởng: trang nào được đưa vào bộ nhớ sớm nhất sẽ được giải phóng để nạp trang khác

Ví dụ: chuỗi tham chiếu các trang được nạp vào bộ nhớ có 3 frame

reference string

7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1

7	7	7	2
	0	0	0
		1	1

2	2	4	4	4	0
3	3	3	2	2	2
1	0	0	0	3	3

0	0
1	1
3	2

7	7	7
1	0	0
2	2	1

page frames

Thay trang: Least Recently Used (LRU)

Ý tưởng: Thay trang có khoảng thời gian không dùng lâu nhất

Sự thực hiện của Bộ đếm (Counter)

- ✓ Mọi phần tử bảng có một bộ đếm, mọi thời điểm **trang được tham chiếu** qua phần tử bảng này, copy clock vào trong bộ đếm.
- ✓ Khi trang cần được hoán đổi, tìm trong bộ đếm để xác định trang nào làm nạn nhân.

Thay trang: LRU Page Replacement

reference string

7 0 1 2 0 3 0 4 2 3 0 3 2 1 2 0 1 7 0 1

7	7	7	2		2		4	4	4	0			1		1		1		
	0	0	0		0		0	0	3	3			3		0		0		
		1	1		3		3	2	2	2			2		2		7		

page frames

Thay trang: LRU Page Replacement

Sự thực hiện của Stack

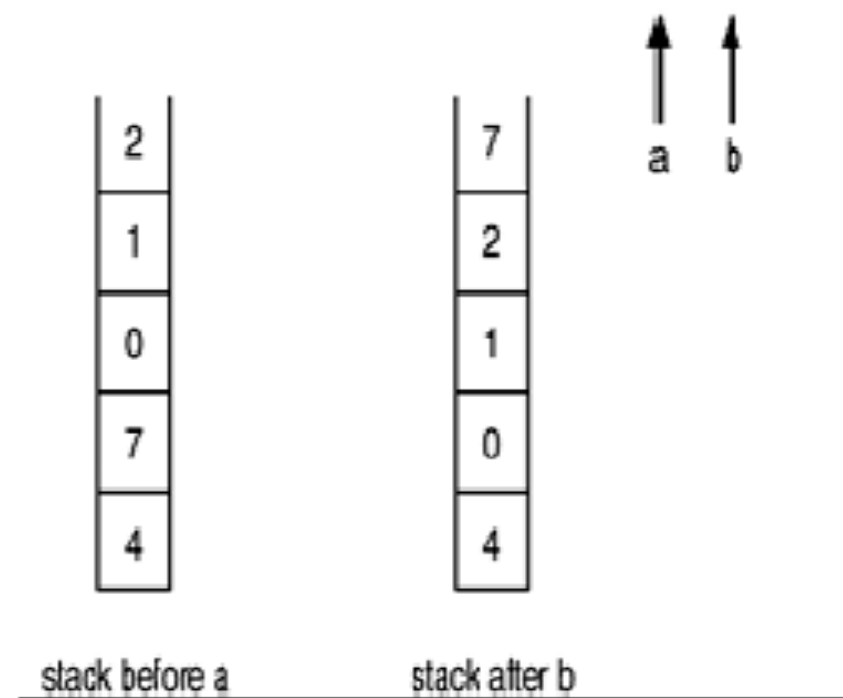
Dành ra một stack của các số tra
trong một danh sách liên kết kép

Khi trang được tham chiếu:

Chuyển nó lên đỉnh nếu đã có trong
nhớ; nếu chưa, chuyển lên đỉnh và l
bỏ trang ở đáy stack

Cần 6 con trỏ để đổi trang

Không cần tìm kiếm để thay thế



Thay trang: Các giải thuật tương tự LRU

❖ Giải thuật dùng thêm bit tham chiếu

Gắn một bit vào mỗi trang, khởi tạo = 0

Khi trang được tham chiếu, bit đó được thiết lập = 1.

Thay trang có bit tham chiếu = 0, nếu có tồn tại trong bộ nhớ

❖ Giải thuật cơ hội thứ hai (Second chance)

Cần có bit tham chiếu. Nếu bit tham chiếu = 0 thì thay trang. Trái lại cho trang đó cơ hội thứ hai và chuyển đến trang tiếp sau

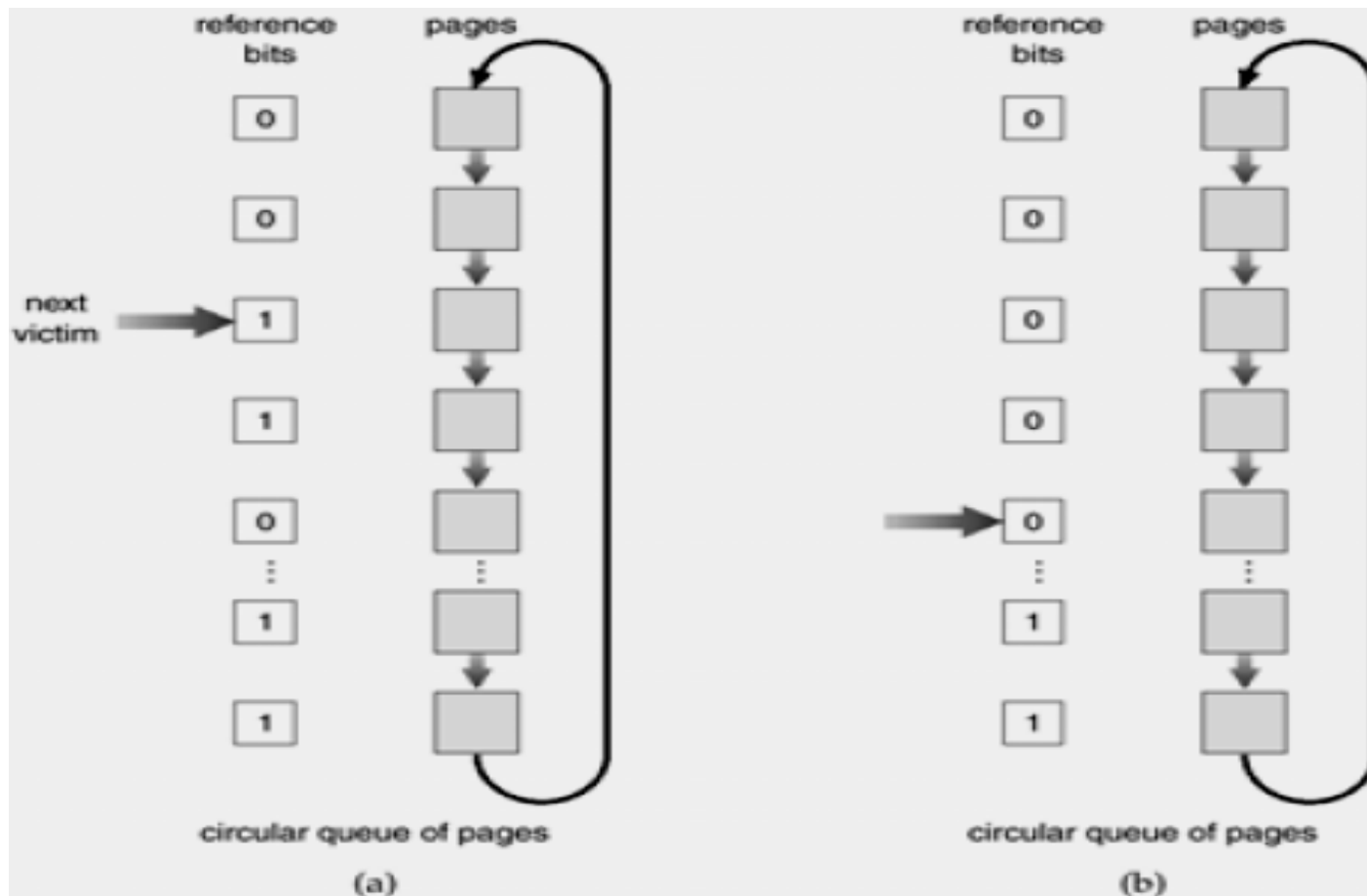
Thiết lập bit tham chiếu = 0.

Đề trang lại trong bộ nhớ.

Thay trang kế tiếp (theo FIFO) với luật tương tự.

Một cách thực hiện giải thuật là sử dụng queue vòng tròn.

Thay trang: Giải thuật thay trang "cơ hội thứ hai"



❖ Bộ nhớ ảo dạng phân đoạn

▪ Tổ chức phân đoạn

- Bộ nhớ ảo bao gồm các đoạn (segment) có kích thước không cố định
- Khi nạp đoạn vào bộ nhớ thì hệ điều hành tìm khoảng trống đủ để nạp đoạn
- Có bảng đoạn quản lý các đoạn

❖ Nhận xét về phân trang, phân đoạn

- Trang trong suốt đối với người lập trình
- Phân trang tránh được phân mảnh bên ngoài
- Người lập trình sử dụng được đoạn
- Phân đoạn phù hợp với lập trình theo khối, cấu trúc dữ liệu thay đổi, dùng chung và bảo vệ bộ nhớ

6. Bộ nhớ ảo (Virtual Memory)

- ❖ Bộ nhớ ảo dạng phân đoạn có phân trang
 - Kết hợp các ưu điểm của phân đoạn và phân trang
 - **Tổ chức:**
 - Bộ nhớ ảo bao gồm các đoạn
 - Trong mỗi đoạn thực hiện phân trang

Hệ thống nhớ trên PC hiện nay

- ❖ Bộ nhớ cache: tích hợp trên chip vi xử lý
- ❖ Bộ nhớ chính: Tồn tại dưới dạng các mô-đun nhớ RAM
 - SIMM – Single Inline Memory Module
 - 30 chân: 8 đường dữ liệu
 - 72 chân: 32 đường dữ liệu
 - DIMM – Dual Inline Memory Module
 - 64 đường dữ liệu
 - RIMM – Rambus DRAM

❖ ROM BIOS chứa các chương trình sau:

- Chương trình POST (Power On Self Test)
- Chương trình CMOS Setup
- Chương trình Bootstrap loader
- Các trình điều khiển vào-ra cơ bản (BIOS)

❖ CMOS RAM:

- Chứa thông tin cấu hình hệ thống
- Đồng hồ hệ thống
- Có pin nuôi riêng

❖ Video RAM: quản lý thông tin của màn hình

❖ Các loại bộ nhớ ngoài

❖ Tìm hiểu khái niệm về các hệ thống:

- NAS (Network Attached Storage)
- SAN (Storage Area Network)

1. Các bộ phận lưu trữ thông tin Các giao diện với các bộ phận vào ra.
2. Mô tả vận hành của ổ đĩa cứng. Cách lưu trữ thông tin trong ổ đĩa cứng
3. Mô tả các biện pháp an toàn trong việc lưu trữ thông tin trong đĩa cứng.
4. Nguyên tắc vận hành của đĩa quang. Ưu khuyết điểm của các loại đĩa quang

