

Chương 4: Các thiết bị vào/ra

Nội dung

1. Các hệ thống vào/ra
2. Thiết bị lưu trữ phụ

1. Các hệ thống vào/ra

1. Tổng quan
2. Các thiết bị vào/ra
3. Giao diện vào/ra ứng dụng
4. Dịch vụ vào/ra
5. Chuyển đổi yêu cầu vào/ra sang các thao tác phần cứng
6. Các luồng (stream)
7. Hiệu năng

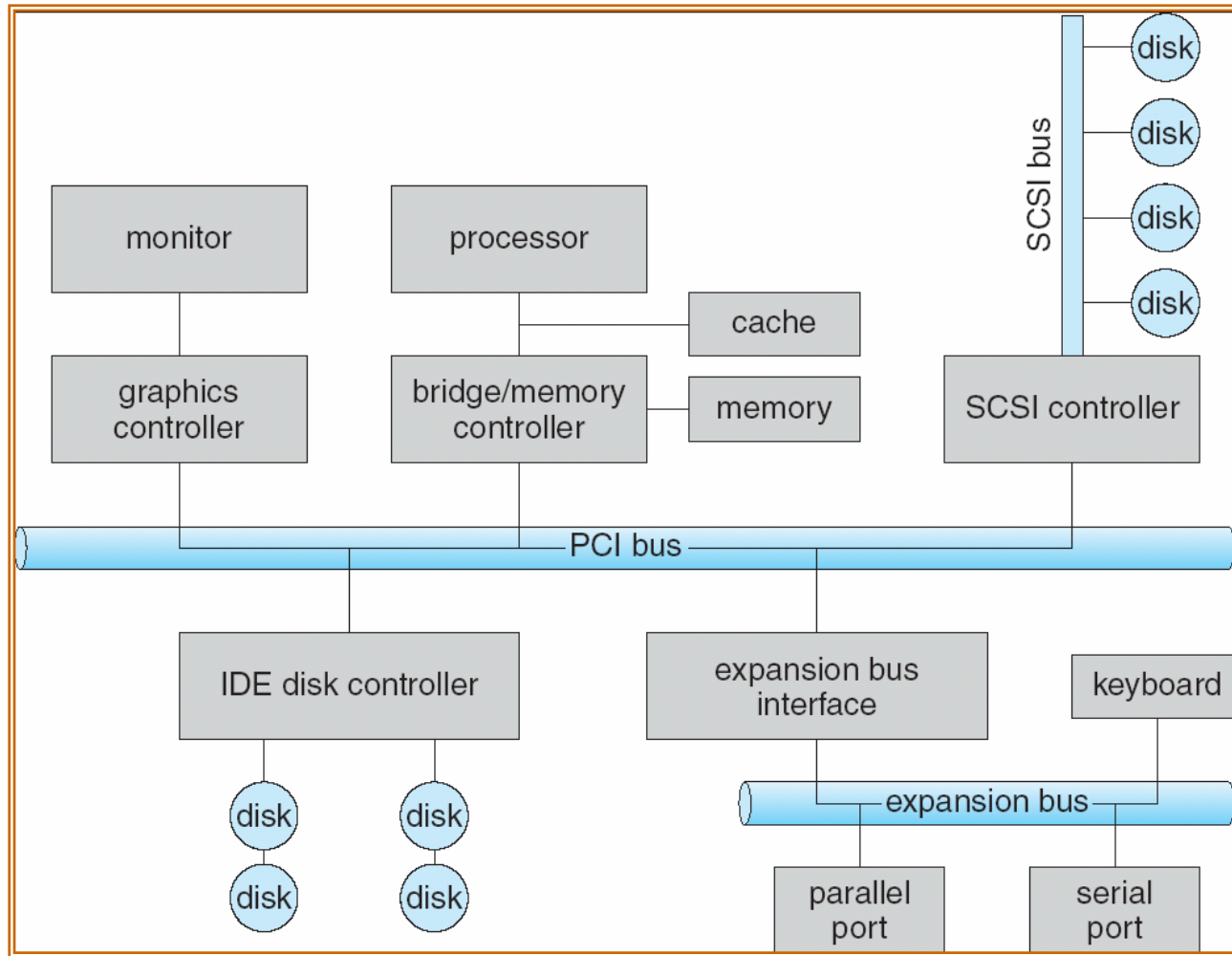
1.1. Tổng quan

- Quản lý thiết bị: thành phần cơ bản của HĐH
- Nhiều loại thiết bị
 - → nhiều phương pháp quản lý thiết bị
 - Các hệ thống con vào/ra (I/O subsystems) tách biệt sự phức tạp của các hệ vào/ra với các thành phần khác của nhân
- Hai xu hướng phát triển
 - Chuẩn hóa
 - Đa dạng loại thiết bị
- Kết hợp hai xu hướng bằng cách
 - Các thành phần phần cứng cơ bản: cổng, bus, bộ điều khiển thiết bị
 - Các trình điều khiển thiết bị

1.2. Các thiết bị vào/ra

- Các loại thiết bị vào/ra: phong phú, đa dạng
- Một số khái niệm cơ bản
 - **Port**
 - **Bus (daisy chain)**
 - **Controller (host adapter)**
- Gửi dữ liệu thông qua các lệnh vào/ra
- Gửi dữ liệu thông qua phương pháp ánh xạ bộ nhớ (memory – mapped)

Cấu trúc Bus điển hình cho PC



Địa chỉ của một số cổng vào/ra trên PC (partial)

I/O address range (hexadecimal)	device
000–00F	DMA controller
020–021	interrupt controller
040–043	timer
200–20F	game controller
2F8–2FF	serial port (secondary)
320–32F	hard-disk controller
378–37F	parallel port
3D0–3DF	graphics controller
3F0–3F7	diskette-drive controller
3F8–3FF	serial port (primary)

Thanh ghi của thiết bị vào ra

- Thông thường, một cổng Vào/Ra bao gồm 4 thanh ghi
- Thanh ghi trạng thái: hoàn thành, số byte sẵn sàng
- Thanh ghi điều khiển: thiết lập bởi (host) và đọc bởi bộ điều khiển
 - Xác định công việc bộ điều khiển phải làm
- Thanh ghi dữ liệu vào: bộ điều khiển ghi, host đọc
- Thanh ghi dữ liệu ra: host ghi và bộ điều khiển đưa ra thiết bị

Các giao thức Vào/Ra

- **Thăm dò (Polling)**
 - Host kiểm tra đều đặn để xem liệu thiết bị có sẵn sàng
- **Điều khiển bởi ngắt (Interrupt-driven)**
 - Bộ điều khiển ngắt host khi nó sẵn sàng
- **Truy cập bộ nhớ trực tiếp-Direct Memory Access (DMA)**
 - Host thiết lập một khối chuyển (large transfer), bộ điều khiển ngắt khi nó hoàn thành

Thực thi Thăm dò

- Cần các bit bắt tay (handshaking bits)
- Bộ điều khiển thiết lập **BUSY** bit
 - **BUSY==0** sẵn sàng nhận lệnh mới
 - **BUSY==1** đang bận
- Host thiết lập **CMD-RDY** bit – **CMD-RDY==0** không có việc cho bộ điều khiển làm
 - **CMD-RDY==1** có việc cho bộ điều khiển làm

Thực thi Thăm dò

Host

```
while (BUSY==1) ;  
if (want to write) {  
    WRITE      = 1;  
    DATA-OUT  = <output byte>;  
    CMD-RDY    = 1;  
} else {  
    ...  
}
```

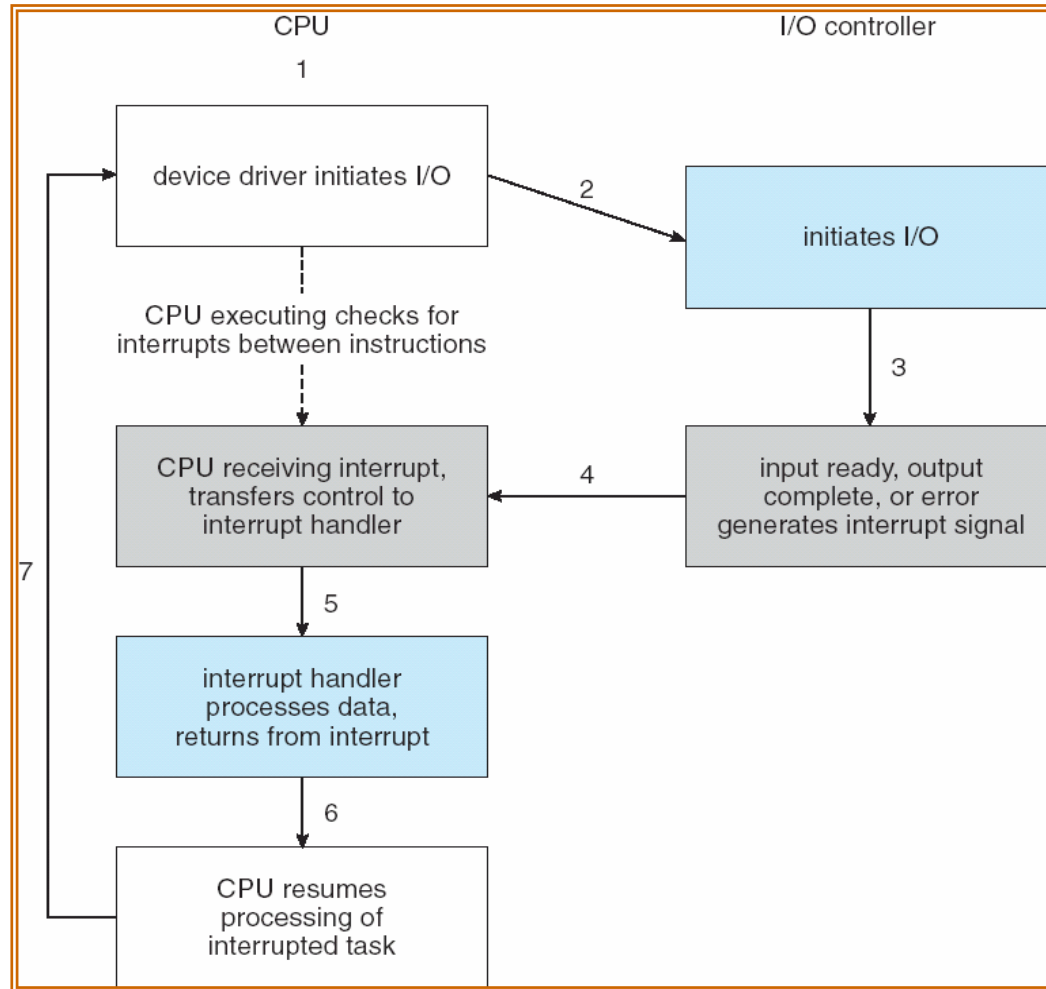
Controller

```
while (CMD-RDY==0) ;  
BUSY = 1;  
if (WRITE == 1) {  
    write DATA-OUT to device  
} else {  
    ...  
}  
CMD-RDY = 0;  
ERROR   = 0;  
BUSY    = 0;
```

Thăm dò

- Nếu thiết bị quá chậm, bên chủ sẽ tốn nhiều thời gian để kiểm tra **BUSY** bit để xem nó có thể làm việc hay không
- Nếu thiết bị quá nhanh, bộ điều khiển có thể tràn bộ đệm **DATA_IN** trước khi bên chủ có cơ hội kiểm tra và lấy dữ liệu
- **Giải pháp:** Vào/Ra được điều khiển bởi ngắt (interrupt-driven I/O)

Vào/Ra điều khiển bởi ngắt



Bảng vector sự kiện của Intel Pentium

vector number	description
0	divide error
1	debug exception
2	null interrupt
3	breakpoint
4	INTO-detected overflow
5	bound range exception
6	invalid opcode
7	device not available
8	double fault
9	coprocessor segment overrun (reserved)
10	invalid task state segment
11	segment not present
12	stack fault
13	general protection
14	page fault
15	(Intel reserved, do not use)
16	floating-point error
17	alignment check
18	machine check
19–31	(Intel reserved, do not use)
32–255	maskable interrupts

Vào/Ra được điều khiển bởi ngắt

- Bus có một dây yêu cầu ngắt (interrupt request-IRQ)
- CPU lưu ý các dây IRQ sau mỗi lệnh
- Nếu IRQ được đưa ra:
 - Trạng thái của chương trình đang chạy được lưu lại
 - Nhảy tới dịch vụ xử lý ngắt (ISR)
 - Dịch vụ ngắt xử lý ngắt
 - Dịch vụ ngắt ISR hoàn thành
 - Trạng thái của chương trình bị ngắt được phục hồi

Điều khiển bởi ngắt

- Khái niệm ngắt được sử dụng cho:
 - Các thiết bị vào ra
 - Các ngoại lệ (chia 0, ..)
 - Lỗi trang trong hệ thống bộ nhớ ảo
 - Lời gọi hệ thống

Khuyết điểm của Vào/Ra điều khiển bởi ngắt và thăm dò

■ Vào:

- CPU yêu cầu dữ liệu
- CPU chuyển dữ liệu từ thiết bị vào bộ nhớ

■ Ra:

- CPU chuyển dữ liệu từ bộ nhớ tới thiết bị
- CPU thăm dò việc hoàn thành hoặc chờ ngắt hoàn thành

Vì sao cần sử dụng CPU để chuyển dữ liệu giữa bộ nhớ và các thiết bị?

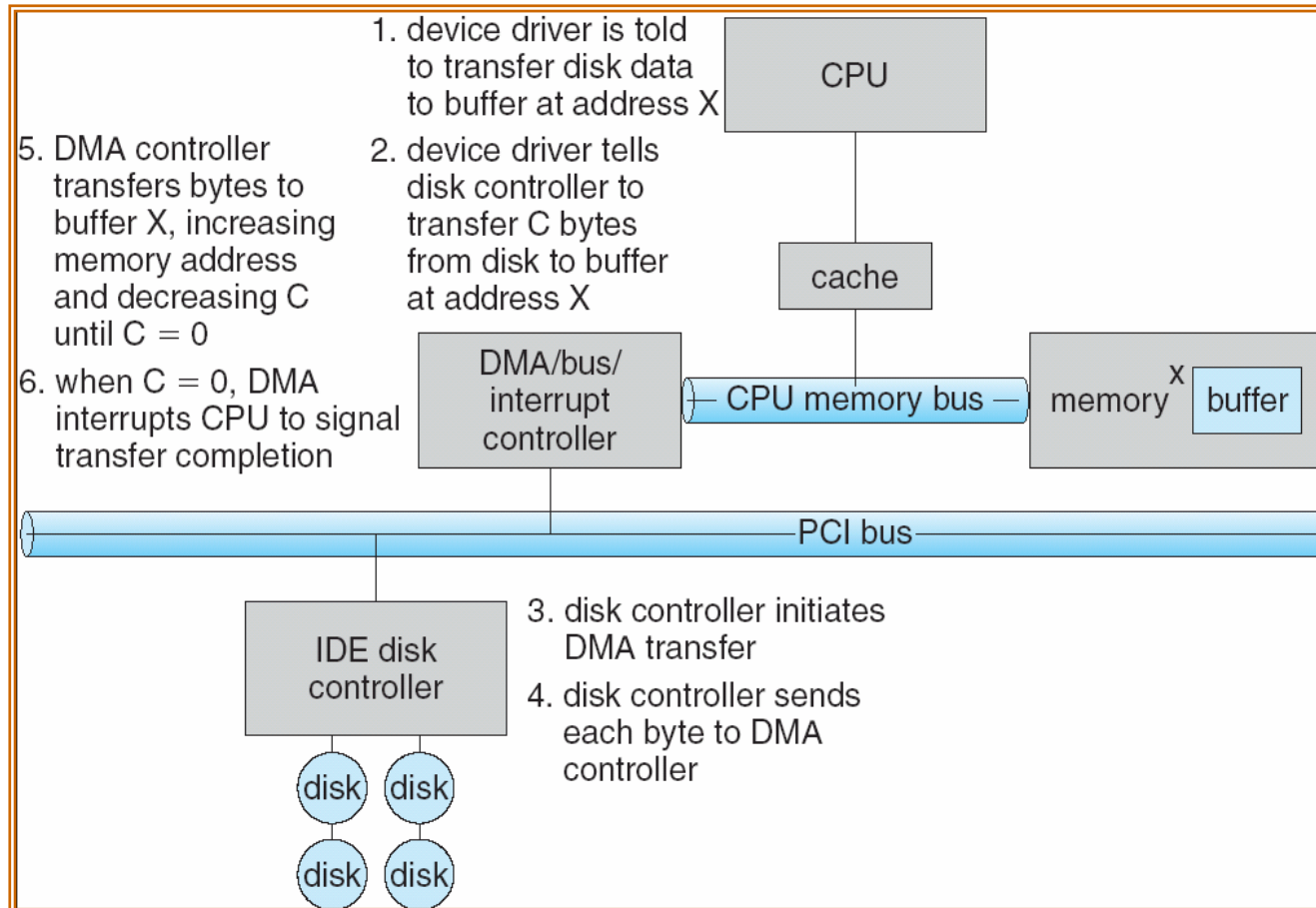
Không cần nếu sử dụng DMA!

Truy cập bộ nhớ trực tiếp

Direct Memory Access (DMA)

- Được sử dụng để tránh **dùng CPU** trong khi chuyển dữ liệu lớn
- Yêu cầu có bộ điều khiển **DMA**
- Việc chuyển dữ liệu trực tiếp giữa thiết bị Vào/Ra và bộ nhớ, không cần thông qua CPU

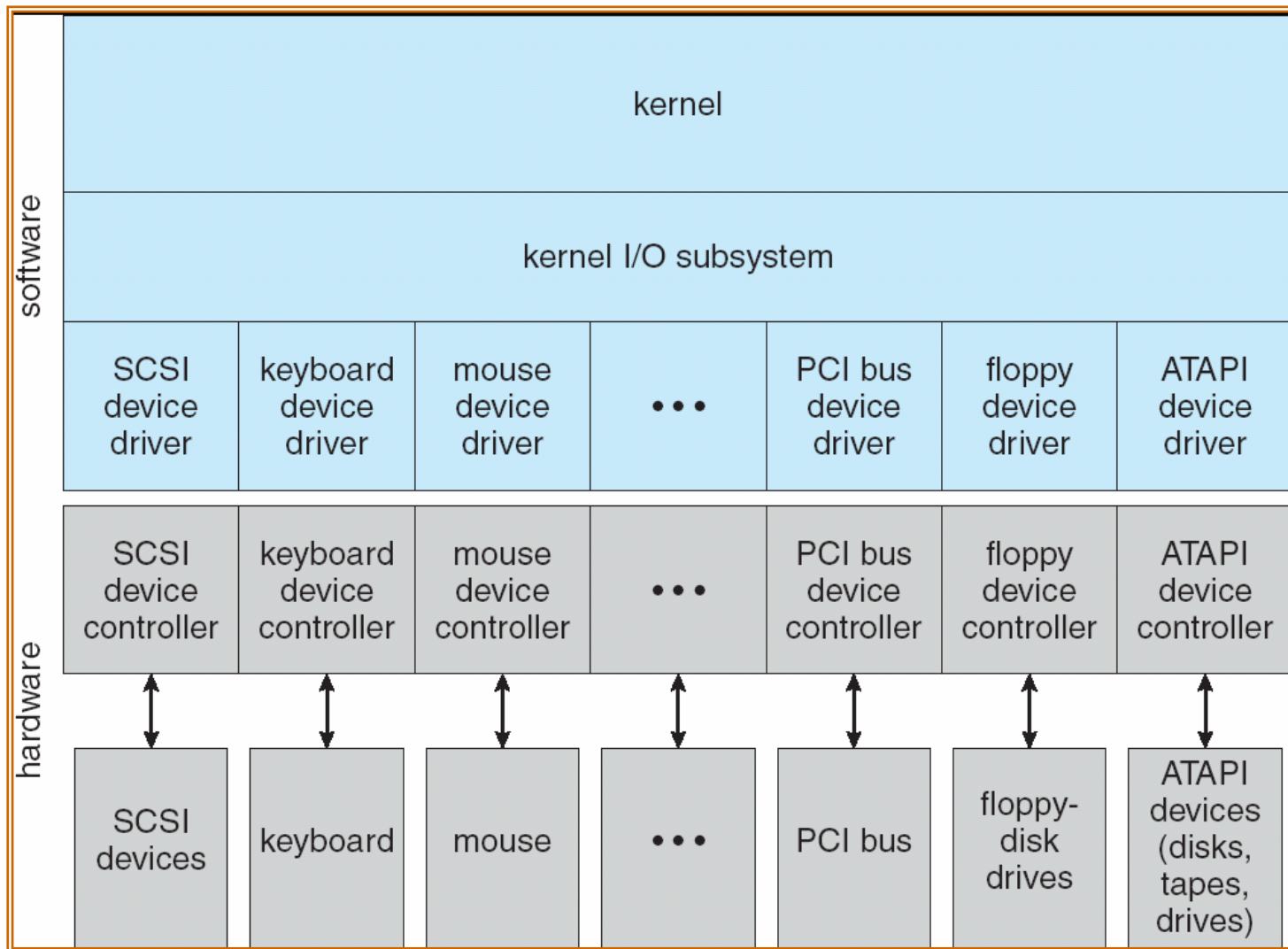
6 bước thực hiện trong chuyển theo DMA²



1.3. Giao diện vào/ra ứng dụng

- Các lời gọi hệ thống đóng gói hoạt động thiết bị trong các lớp tổng quát
- Trình điều khiển thiết bị che dấu sự khác biệt giữa các bộ điều khiển thiết bị (nhân HĐH không thấy được sự khác biệt đó)
- Một số loại thiết bị
 - **Character-stream** hay **block**
 - **Sequential** hay **random-access**
 - **Sharable** hay **dedicated**
 - **Speed of operation**
 - **read-write, read only, or write only**

Cấu trúc vào/ra trong nhân



Đặc điểm của các thiết bị vào/ra

aspect	variation	example
data-transfer mode	character block	terminal disk
access method	sequential random	modem CD-ROM
transfer schedule	synchronous asynchronous	tape keyboard
sharing	dedicated sharable	tape keyboard
device speed	latency seek time transfer rate delay between operations	
I/O direction	read only write only read-write	CD-ROM graphics controller disk

Các thiết bị vào/ra theo kí tự hoặc theo khối

- Các thiết bị vào/ra theo khối như các đĩa cứng
 - Các lệnh vào/ra gồm read, write hay seek
 - Truy nhập theo dạng lệnh vào/ra
 - Truy nhập file ánh xạ bộ nhớ
- Các thiết bị vào/ra theo kí tự
 - Các câu lệnh bao gồm put, get
 - Các thư viện hỗ trợ dịch vụ lưu bộ đệm và chế độ chỉnh sửa (ví dụ: nhập backspace → xóa một kí tự khỏi luồng dữ liệu vào)

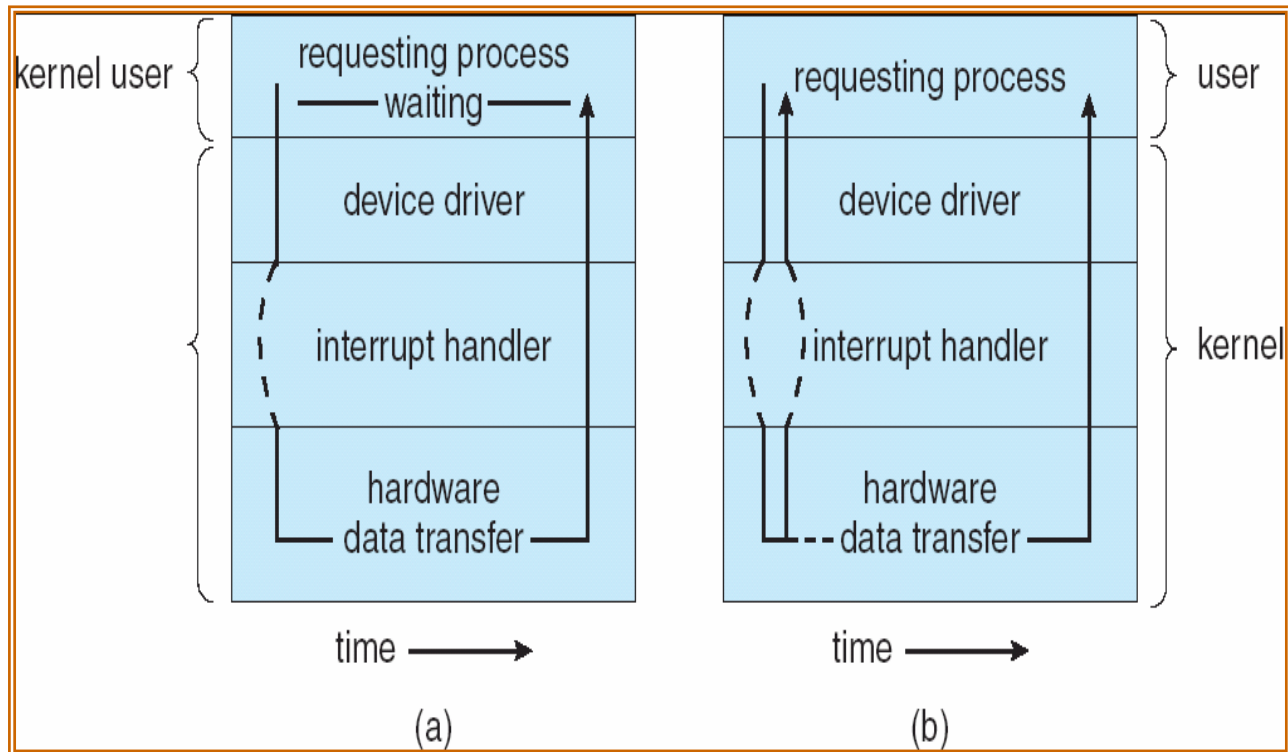
Các đồng hồ và thời gian

- Cho biết thời gian hiện tại, bộ đếm thời gian
- **Bộ đếm lập trình** được sử dụng cho đánh giá thời gian, ngắt định kì
- `ioctl` (on UNIX) thực hiện đồng hồ hoặc bộ đếm thời gian

Vào/ra phong tỏa và không phong tỏa

- **Phong tỏa** – tiến trình bị treo cho đến khi hoàn tất vào ra
 - Dễ hiểu và sử dụng
 - Không hiệu quả với một số nhu cầu người dùng
- **Không phong tỏa**
 - Thực hiện thông qua đa luồng
 - Trả về nhanh với tham số trả về là số bytes đọc hoặc đã ghi
- **Không đồng bộ** - Tiến trình thực hiện khi I/O đang thực hiện
 - Khó dùng
 - Các hệ thống con vào/ra thông báo với tiến trình khi thực hiện vào/ra xong

Hai phương pháp vào/ra



1.4. Hệ thống con vào/ra

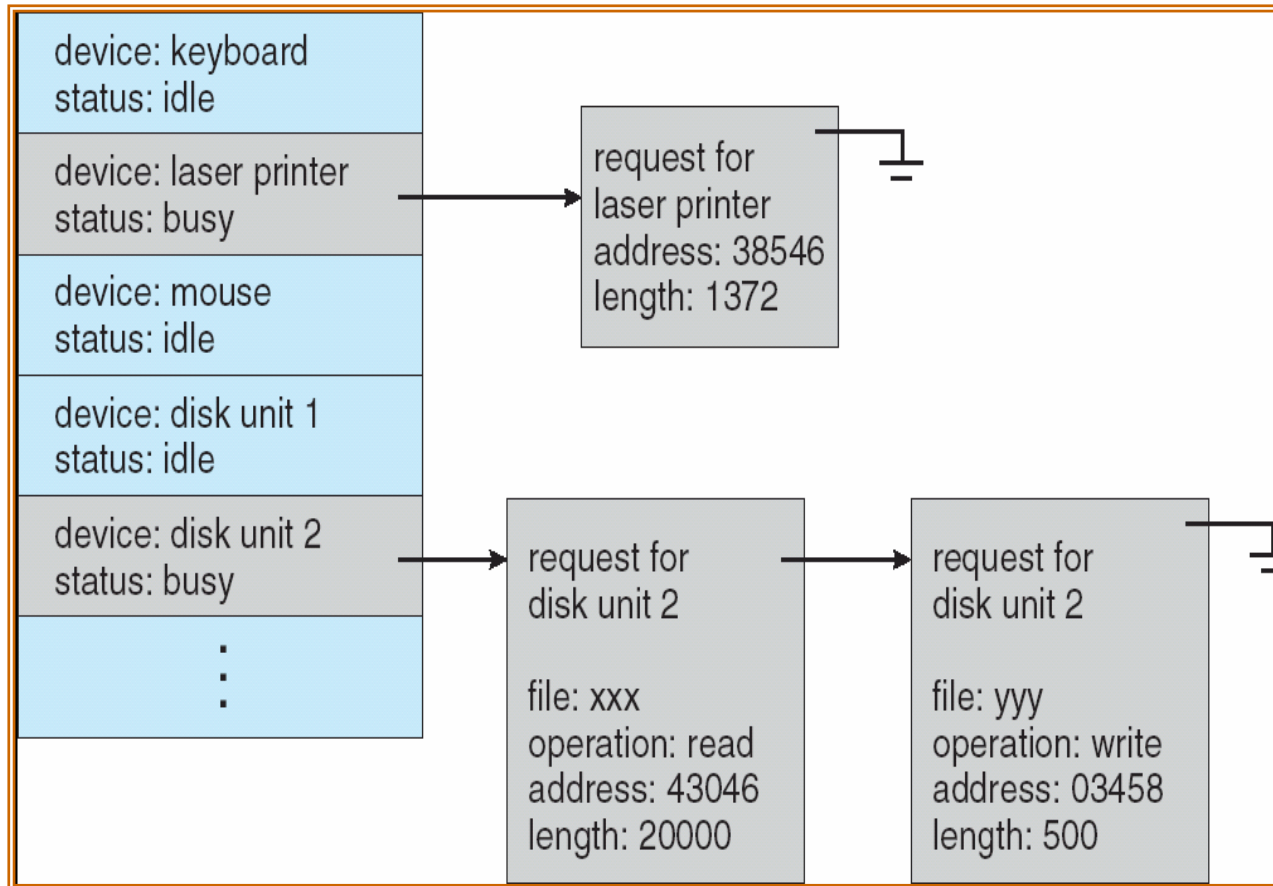
■ Lập lịch

- ❑ Sắp xếp thứ tự vào ra thông qua hàng đợi thiết bị
- ❑ Cố gắng sắp xếp phục vụ các yêu cầu vào/ra một cách công bằng

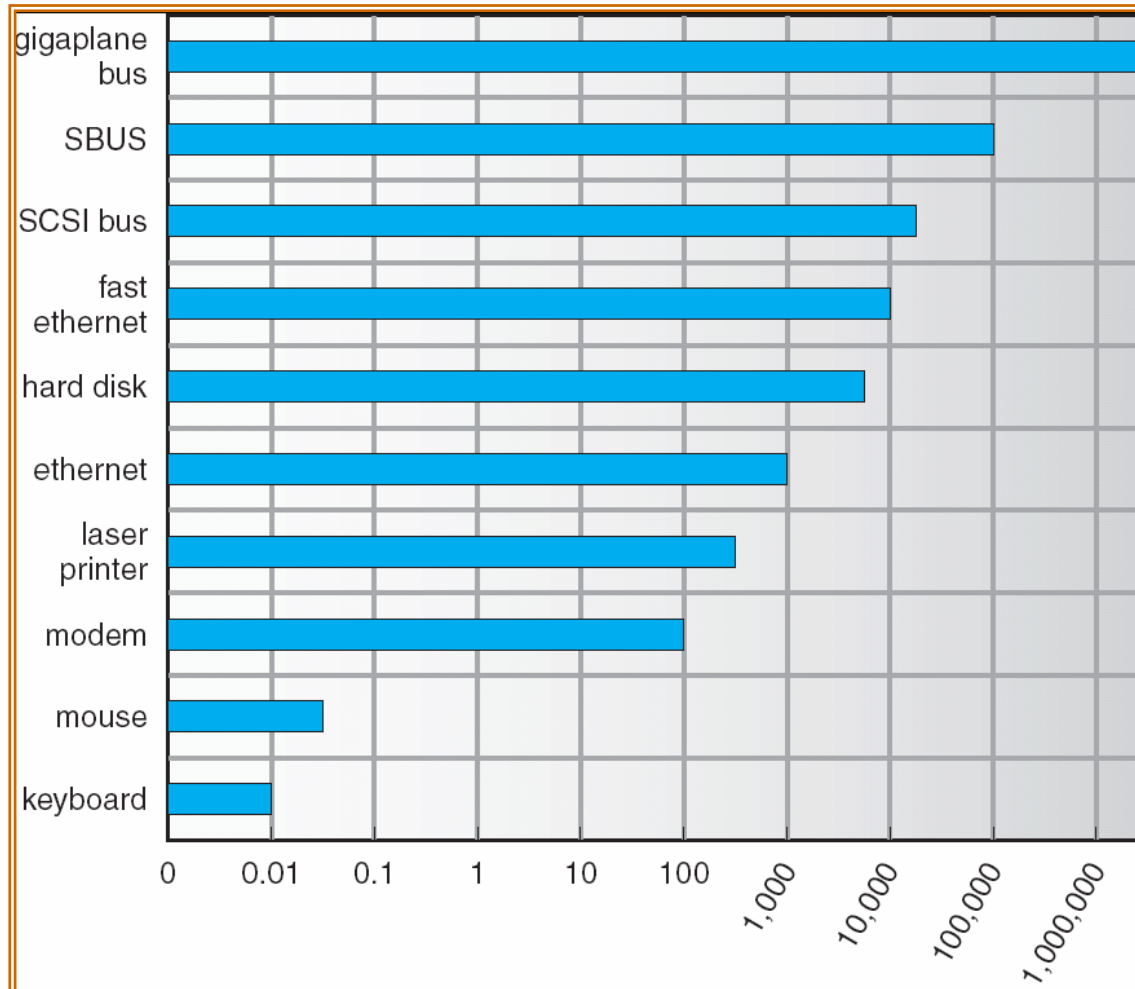
■ Lưu trữ bộ đệm – lưu dữ liệu trong bộ nhớ trong khi đang chuyển dữ liệu giữa các thiết bị trong hệ thống, nhằm

- ❑ Khắc phục sự khác biệt về tốc độ
- ❑ Khắc phục sự khác biệt về cách thức truyền file

Bảng trạng thái thiết bị



Hệ thống Sun Enterprise 6000 – Tốc độ truyền



Hệ thống con vào/ra (tt)

■ Caching

- Lưu lặp tại thiết bị lưu trữ nhanh hơn nhằm tăng hiệu năng truy cập dữ liệu

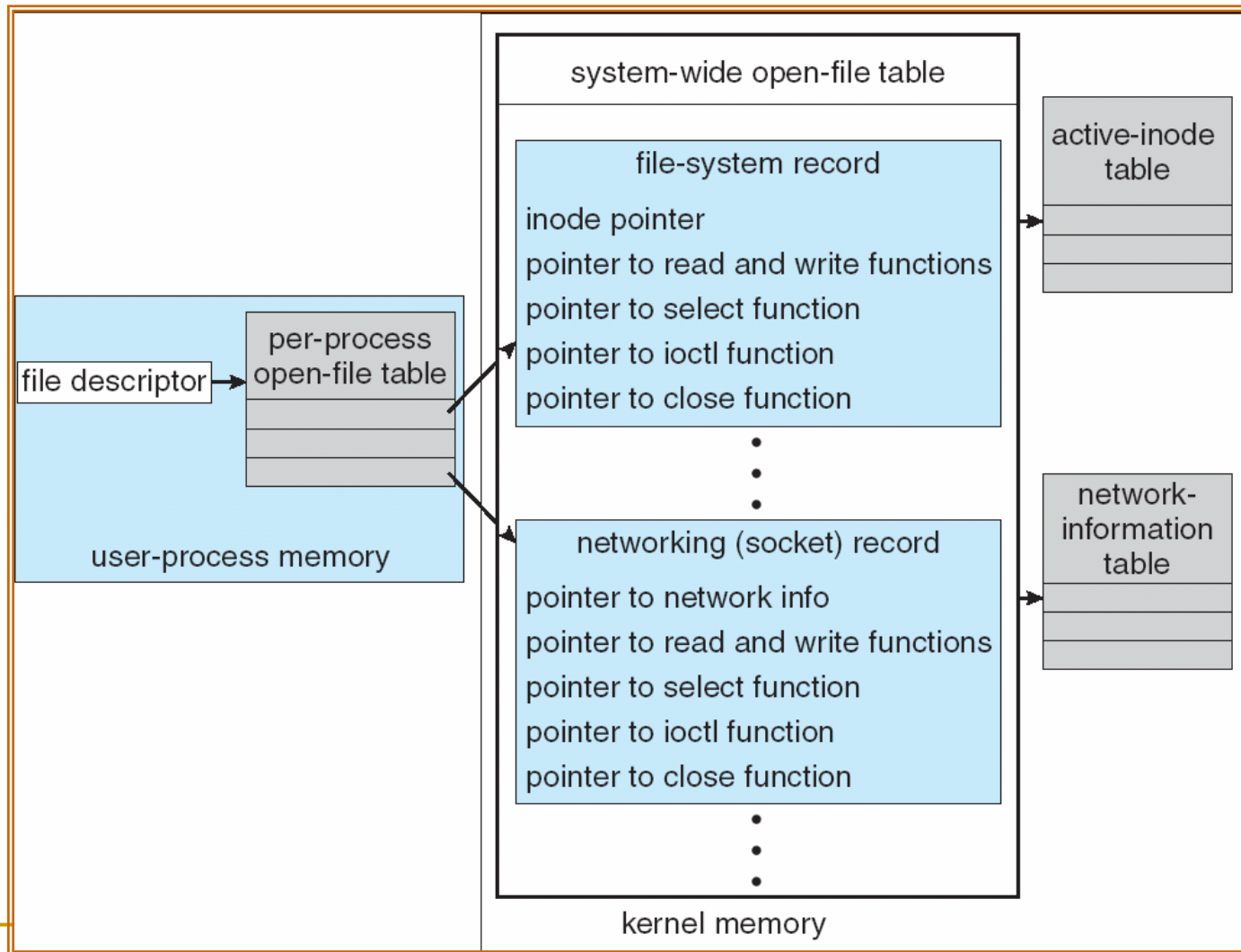
■ Spooling

- Bộ đệm lưu trữ output cho thiết bị, không cho phép đan xen (**interleaved**)
 - Ví dụ: máy in

Các cấu trúc dữ liệu mức nhân

- Nhân lưu giữ thông tin trạng thái cho các thành phần vào/ra
 - Bảng file mở, các kết nối mạng, trạng thái thiết bị vào/ra theo kí tự
- Nhiều cấu trúc dữ liệu phức tạp để lưu vết các bộ đệm, phân phối bộ nhớ, các khối “dirty” (đang sửa đổi)
- Các phương pháp hướng đối tượng và truyền thông báo để cài đặt vào/ra

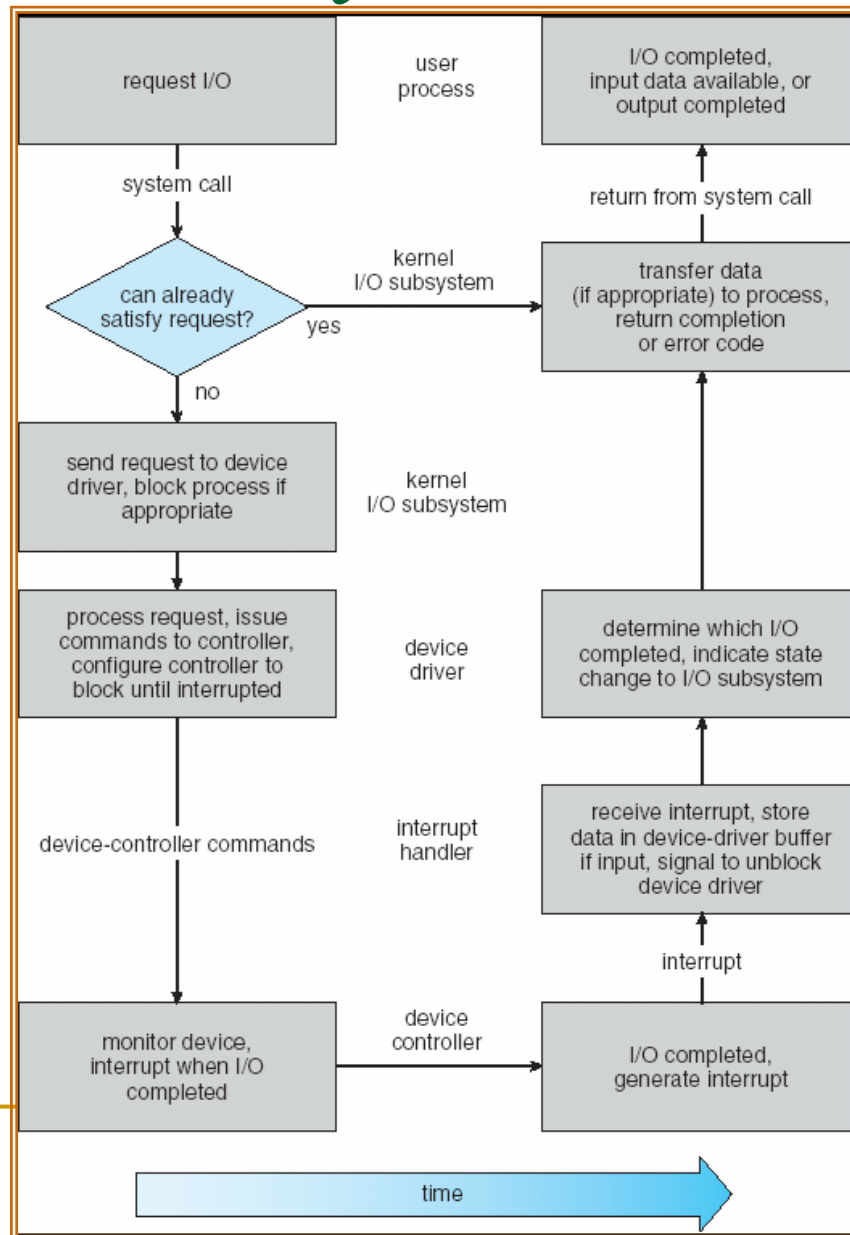
Cấu trúc vào/ra mức nhân trong UNIX I/O



1.5. Chuyển đổi yêu cầu vào/ra sang các thao tác phần cứng

- Xem xét việc đọc một file từ đĩa (của một tiến trình nào đó):
 - Xác định thiết bị lưu trữ file
 - Dịch tên sang dạng biểu diễn trên thiết bị
 - Đọc dữ liệu từ đĩa vào bộ đệm
 - Đánh dấu dữ liệu sẵn sàng đối với tiến trình
 - Chuyển điều khiển đến tiến trình

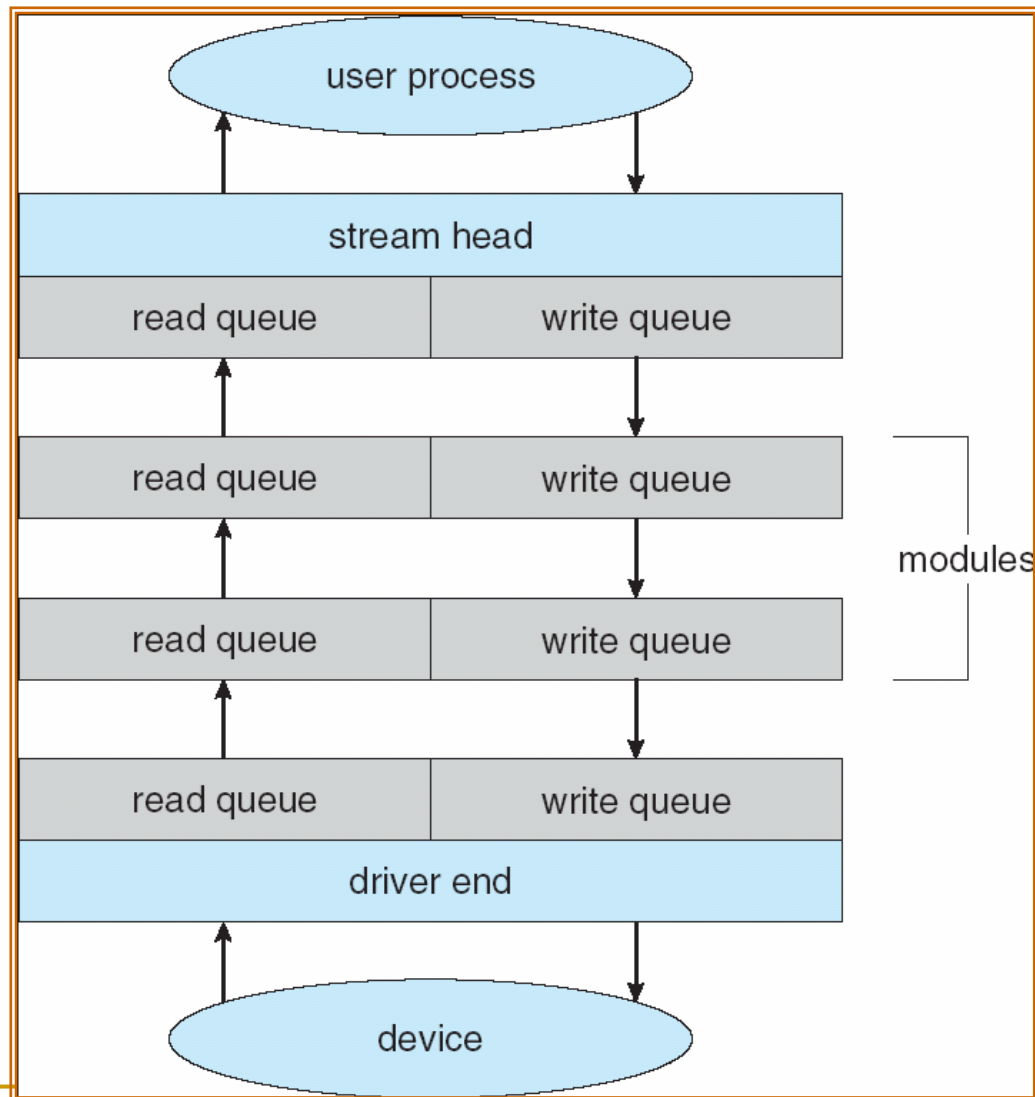
Chu kì của một yêu cầu vào/ra



1.6. Các luồng

- **STREAM (luồng)** – kênh giao tiếp full-duplex giữa một tiến trình mức người dùng và một thiết bị trong hệ thống Unix
- Một LUỒNG bao gồm:
 - Giao diện luồng ở đầu tiến trình người dùng
 - Các giao diện đầu thiết bị
 - Không hay nhiều modules luồng giữa chúng.
- Mỗi module bao gồm một hàng đợi đọc và một hàng đợi ghi
- Sử dụng truyền thông báo để giao tiếp giữa các hàng đợi

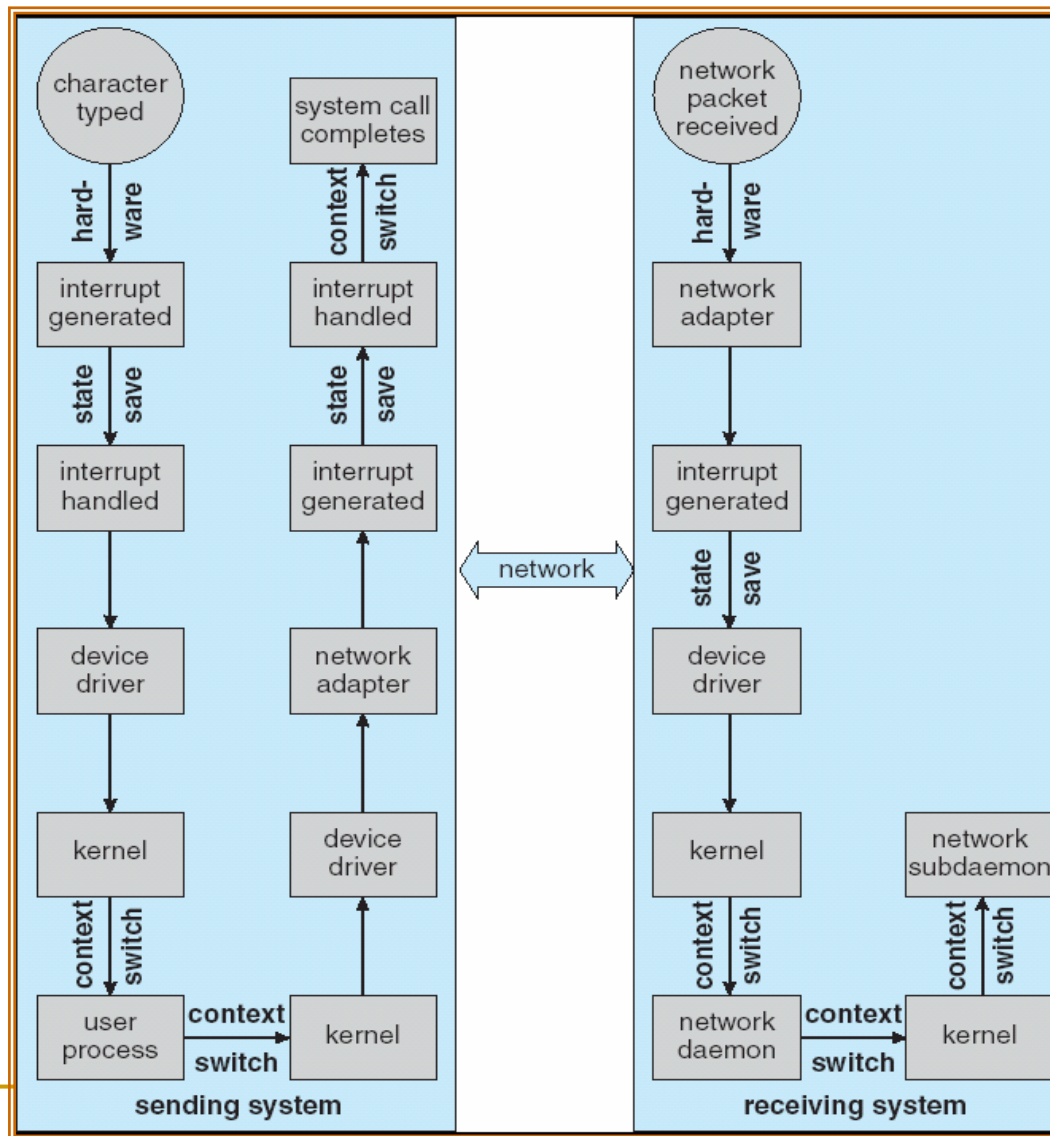
Cấu trúc các luồng (streams)



1.7. Hiệu năng

- Vào/ra là nhân tố quan trọng ảnh hưởng đến hiệu năng của hệ thống:
 - ❑ Đòi hỏi CPU thực thi trình điều khiển thiết bị, các đoạn mã thực thi vào/ra trong nhân
 - ❑ Chuyển đổi ngữ cảnh do ngắt
 - ❑ Sao chép dữ liệu
 - ❑ Tắc nghẽn trên mạng đối với các giao tiếp trên mạng

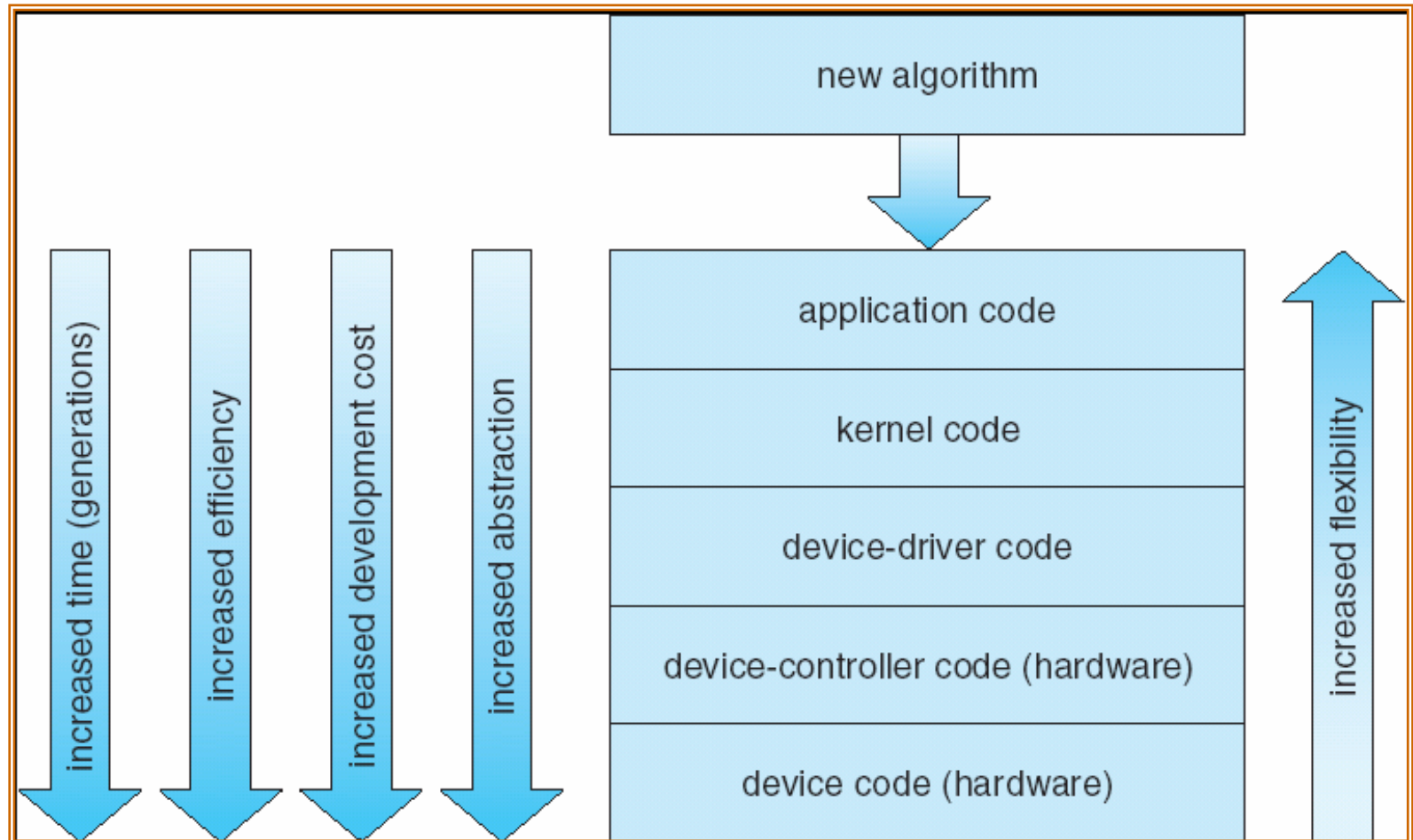
Các giao tiếp computer-computer



Tăng hiệu năng

- Giảm số lượng chuyển đổi ngữ cảnh
- Giảm dung lượng sao chép
- Giảm ngắt bằng cách truyền theo các khối lớn, các bộ điều khiển thông minh
- Sử dụng DMA
- Cân bằng giữa CPU, bộ nhớ, bus và hiệu năng vào ra để cho thông lượng lớn nhất

Device-Functionality Progression



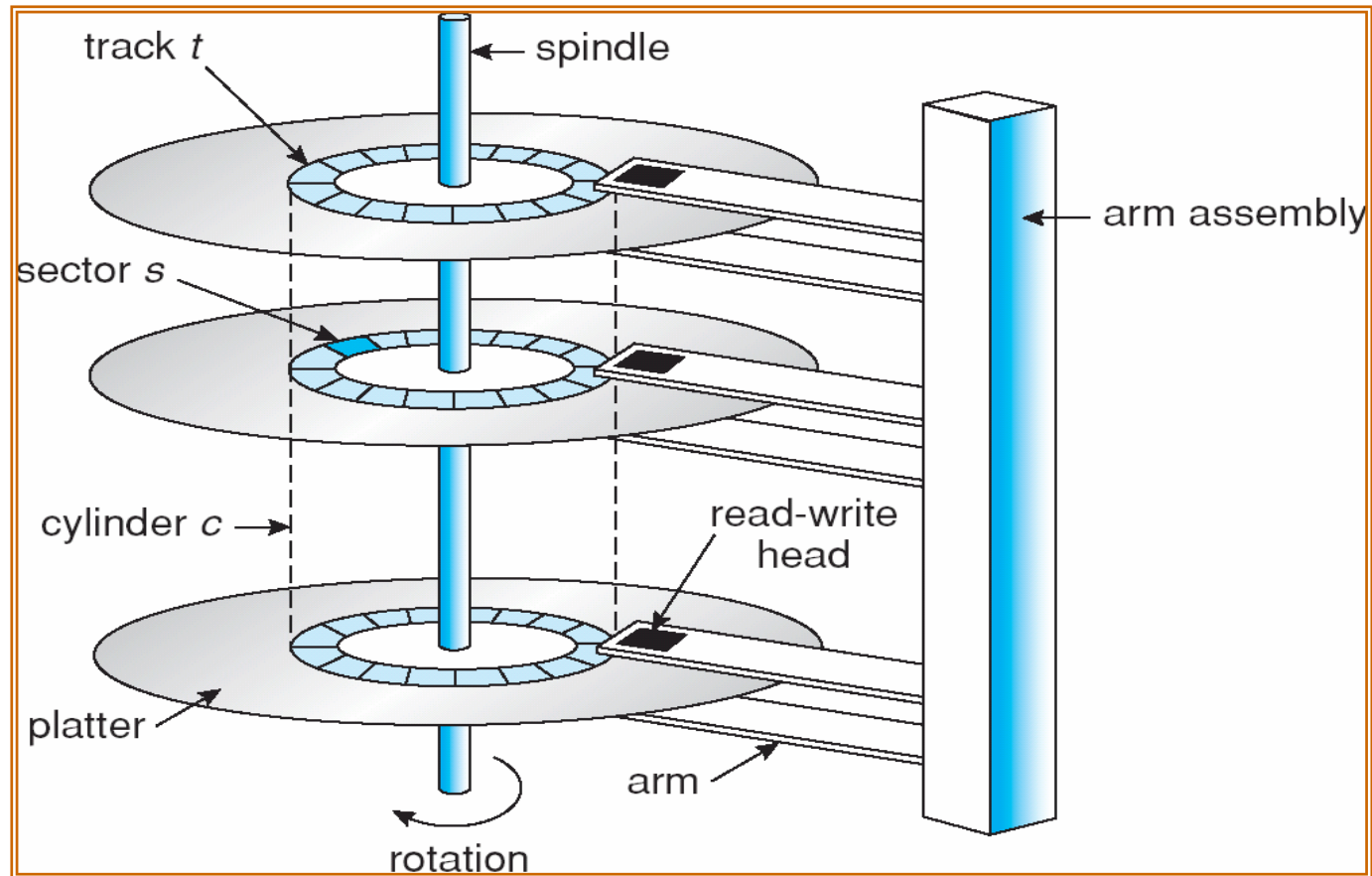
2. Thiết bị lưu trữ phụ

- Thiết bị lưu trữ
- Lập lịch vào ra với đĩa
- Lưu trữ tin cậy

Đĩa từ (Magnetic disks)

- Được dùng để cung cấp lưu trữ thứ 2 cho các máy tính hiện đại
- **Tốc độ truyền (Transfer rate)** là tốc độ truyền dữ liệu giữa thiết bị và máy tính
- **Thời gian xác định vị trí Positioning time (random-access time)** là thời gian để di chuyển disk arm tới cylinder yêu cầu (**seek time**) và thời gian để sector yêu cầu quay tới đầu đọc gọi là (**rotational latency**)
- **Host controller** trong máy tính sử dụng đường bus để giao tiếp với **disk controller** được thiết lập trong thiết bị

Dịch chuyển đầu đĩa



Đĩa từ

- Các đĩa có thể di dời được (removable)
- Đĩa gắn với máy tính thông qua bus Vào/Ra (I/O bus)
 - Ví dụ: EIDE, ATA, SATA, USB, Fibre Channel, SCSI

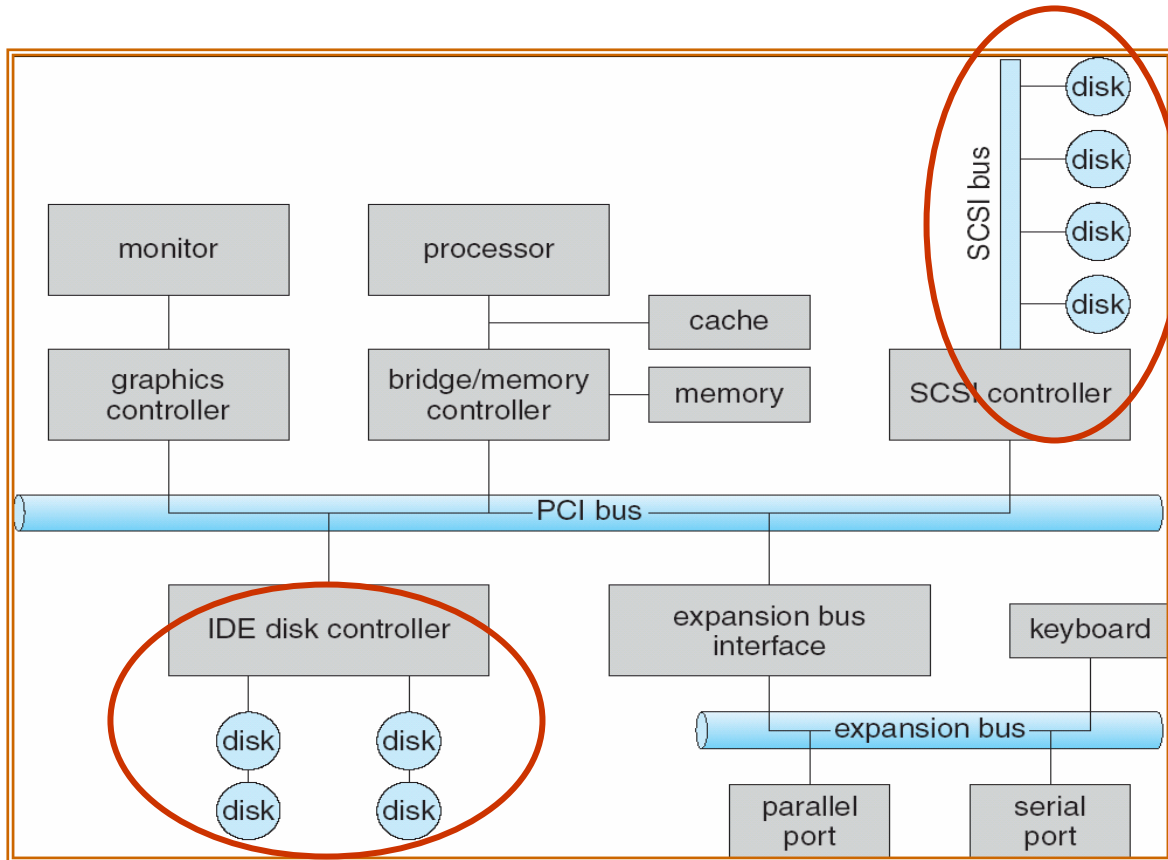
Băng từ (Magnetic tape)

- ❑ Là môi trường lưu trữ thường dùng trước đây
- ❑ Có thể dùng tương đối lâu dài, và lưu trữ lượng lớn thông tin
- ❑ Thời gian truy cập chậm
- ❑ Chậm hơn khoảng 1000 lần so với đĩa
- ❑ Được sử dụng chủ yếu cho sao lưu và lưu dữ liệu không được sử dụng thường xuyên

Cấu trúc đĩa

- Ổ đĩa được coi như một mảng lớn 1 chiều chứa các khối logic (*logical blocks*). Khối logic là đơn vị truyền nhỏ nhất
- Mảng 1 chiều này được ánh xạ vào các sector của đĩa tuần tự
 - Sector 0 là sector đầu tiên của track đầu tiên trên cylinder ngoài cùng.
 - Ánh xạ tiếp tục theo track đó, sau đó là các track còn lại của cylinder đó, và sau đó là các cylinder còn lại từ ngoài vào trong.

Ví dụ: gắn đĩa



SCSI

- SCSI sử dụng phổ biến cho đĩa cứng, nhưng có thể nối tới các thiết bị khác như scanners, printers
- SCSI sử dụng phổ biến trong các trạm làm việc hiệu năng cao, servers,
- Desktop computers notebooks thường sử dụng ATA/IDE

Gắn đĩa (Disk Attachment)

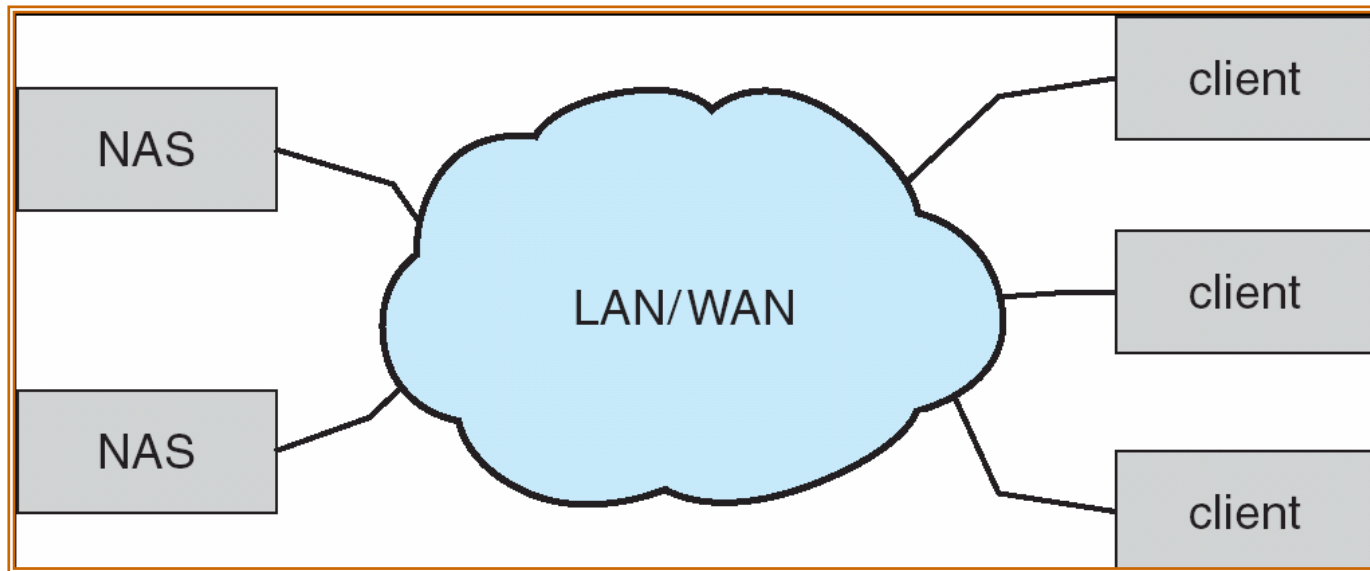
- SCSI bản thân cũng là 1 bus. Có thể tới 16 thiết bị trên cùng 1 sợi cáp
- FC là kiến trúc tuần tự (serial) tốc độ cao
 - Có thể là bộ chuyển mạch tốc độ cao với 24-bit không gian địa chỉ – là cơ bản của mạng lưu trữ (**storage area networks-SANs**) ở đó nhiều đơn vị chủ gắn với nhiều đơn vị lưu trữ

SCSI



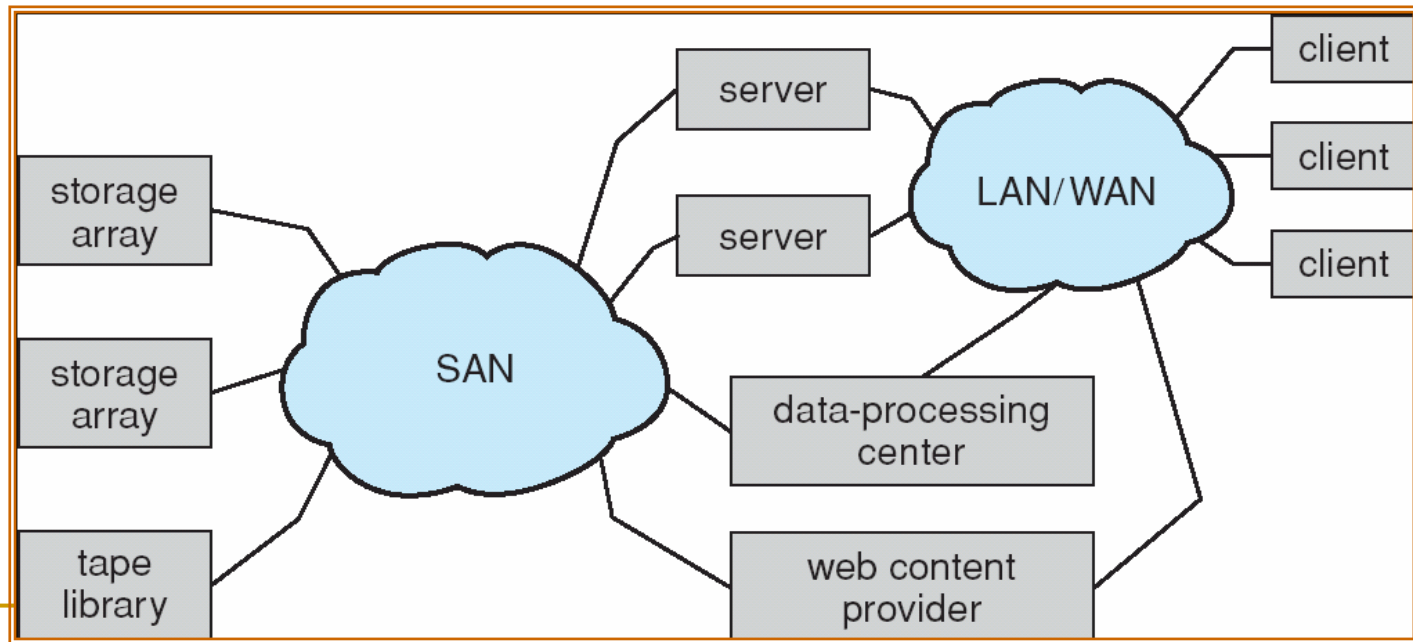
Lưu trữ gắn với mạng (Network-Attached Storage-NAS)

- Lưu trữ gắn với mạng (**NAS**) được truy cập thông qua mạng thay vì kết nối cục bộ (ví dụ bus)
- Các giao thức phổ biến: NFS, CIFS và iSCSI. Được cài đặt thông qua RPC giữa client và thiết bị



Mạng lưu trữ (SAN)

- Sử dụng phổ biến trong môi trường lưu trữ lớn (đang trở nên phổ biến)



Lập lịch cho đĩa (Disk Scheduling)

- HĐH có trách nhiệm sử dụng phần cứng hiệu quả
- Thời gian truy cập gồm 2 thành phần:
 - *Thời gian tìm kiếm (Seek time)* là thời gian để đĩa di chuyển các đầu đĩa tới cylinder chứa sector yêu cầu
 - *Độ trễ quay (Rotational latency)* là thời gian thêm để đợi đĩa quay tới sector yêu cầu tới đầu đĩa
- Tối thiểu thời gian tìm kiếm
- Thời gian tìm kiếm \approx khoảng cách (distance)

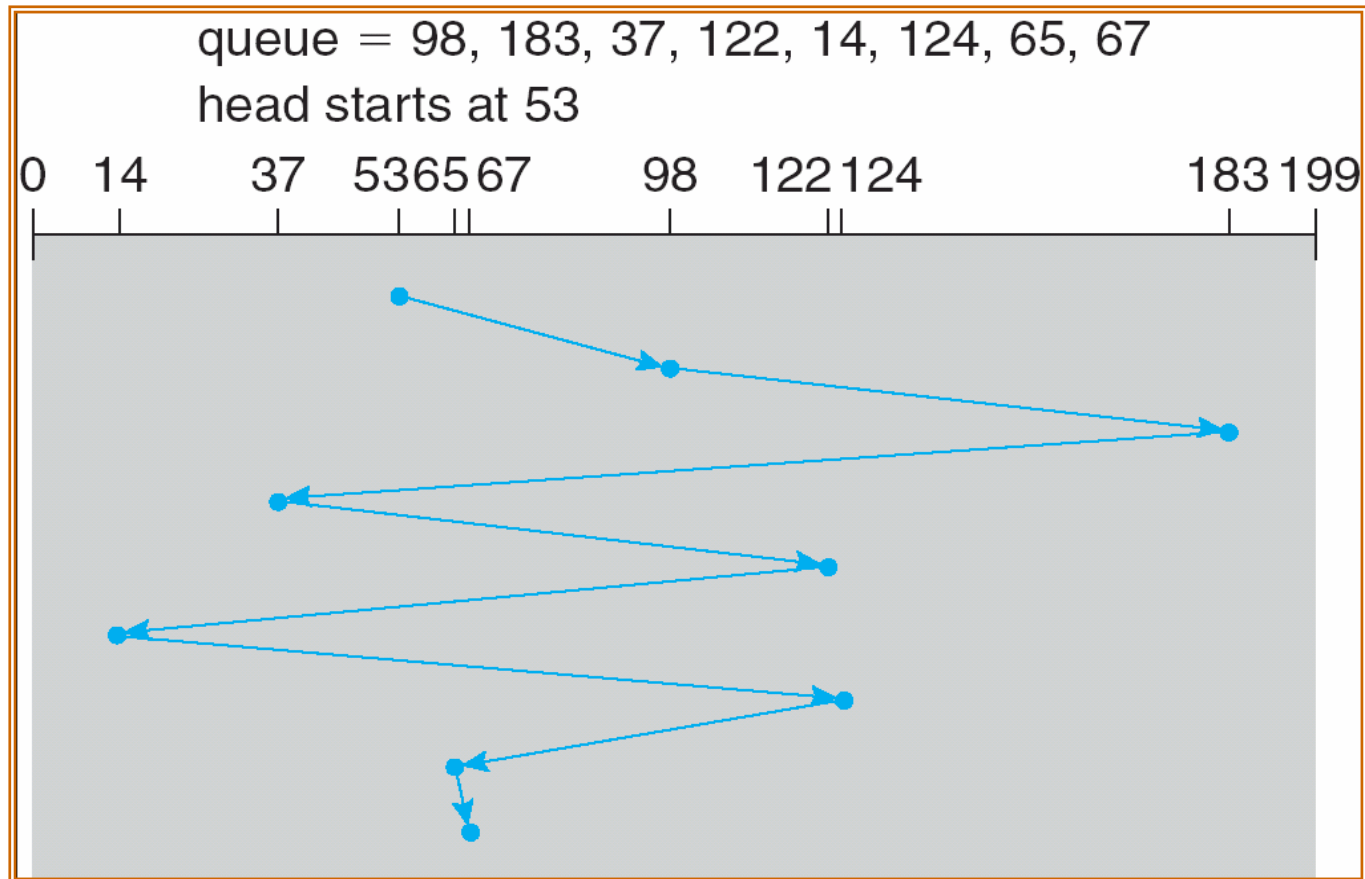
Lập lịch cho đĩa

- Tồn tại vài thuật toán để lập lịch cho các yêu cầu Vào/Ra với đĩa
- Sử dụng ví dụ sau để đánh giá các thuật toán:
 - Hàng đợi yêu cầu:

98, 183, 37, 122, 14, 124, 65, 67

- Đầu đọc: 53

FCFS

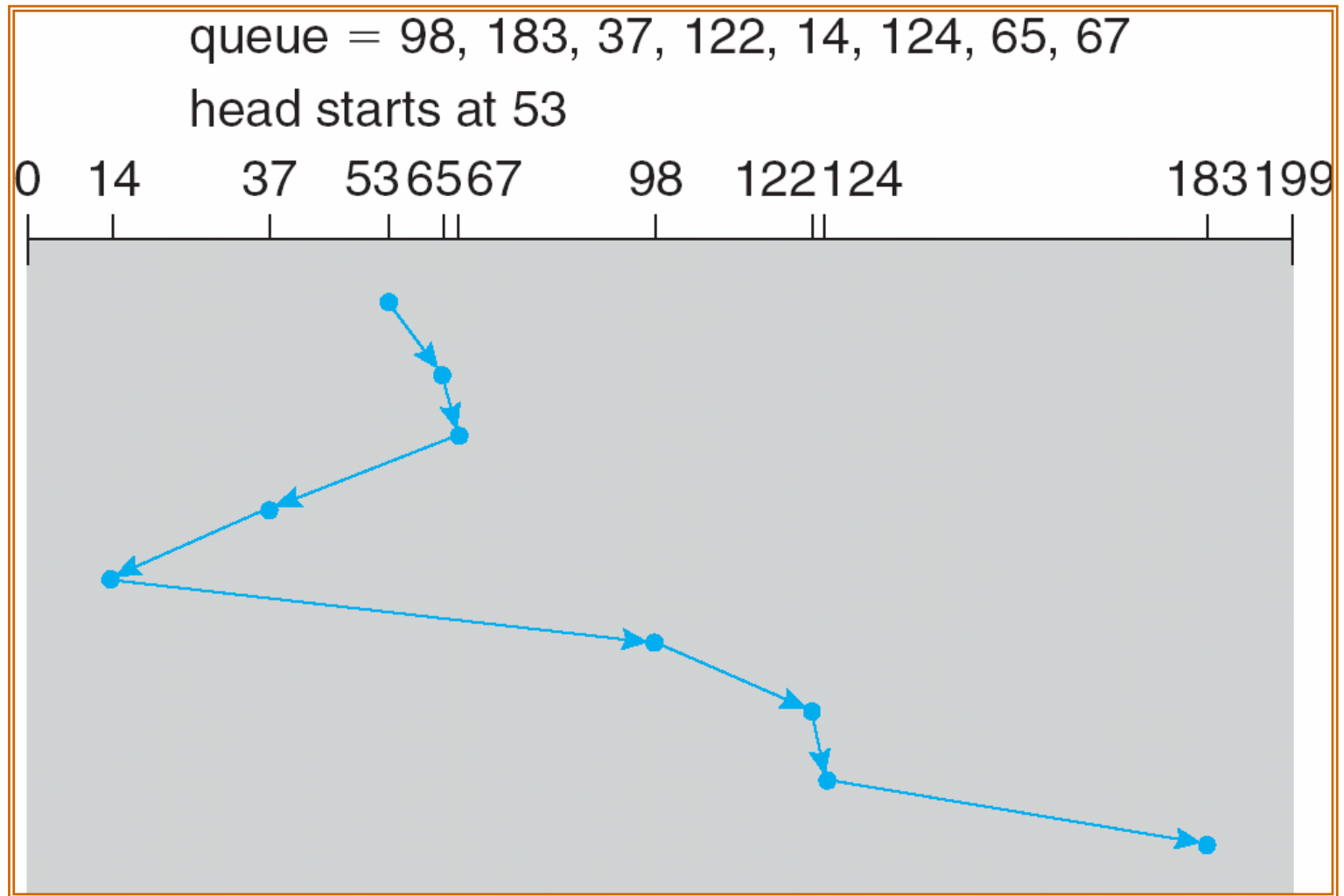


Tổng số di chuyển đầu đọc: 640 cylinder

SSTF (Shortest Seek Time First)

- Chọn yêu cầu với khoảng cách tối thiểu với vị trí đầu đọc hiện tại
- Có thể bắt một số yêu cầu phải chờ đợi quá lâu (giống như lập lịch SJF)
- Minh họa cần tổng số di chuyển đầu đọc:
236 cylinders.

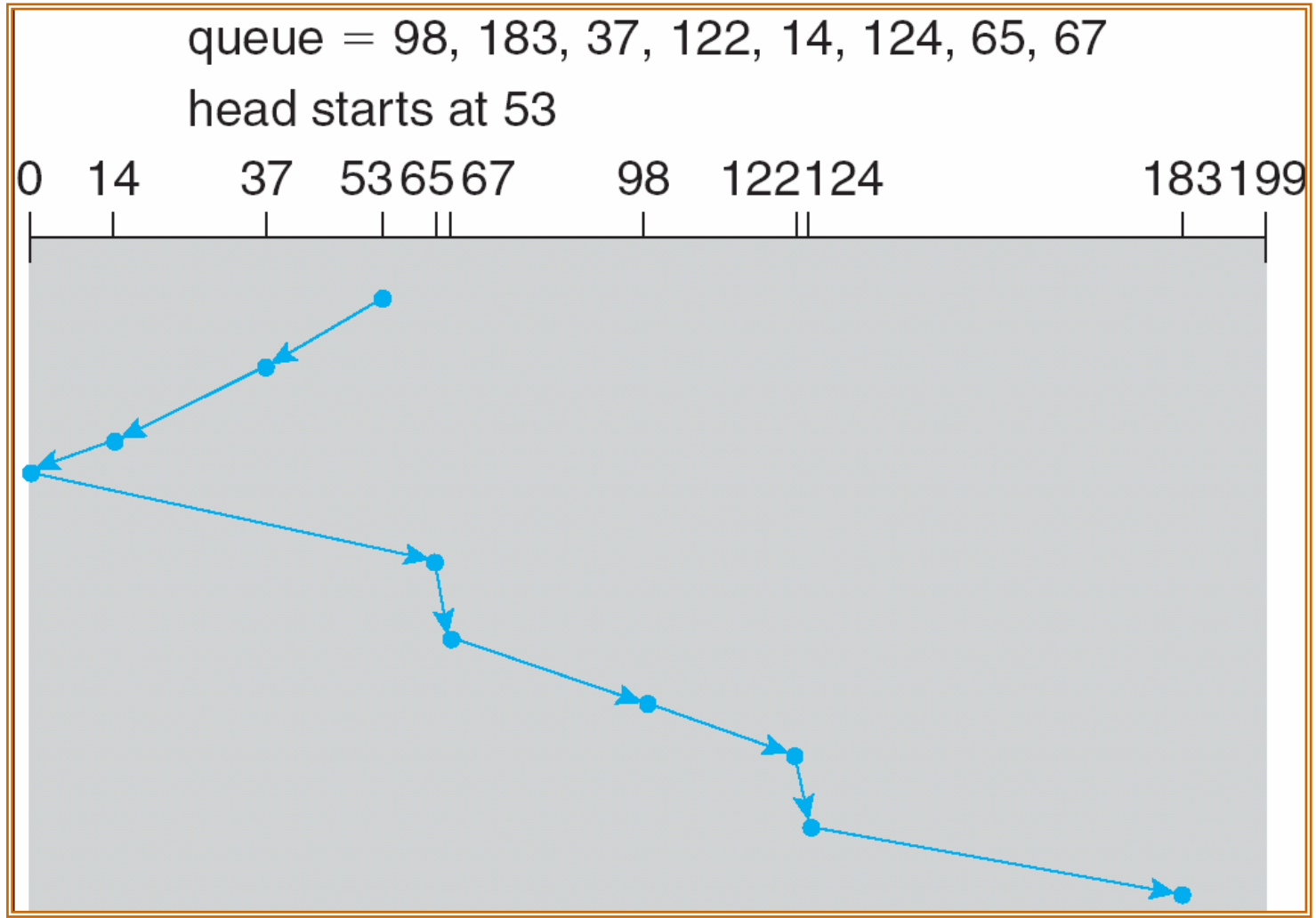
SSTF



SCAN

- Tay đĩa bắt đầu tại một điểm kết thúc của đĩa và di chuyển hướng tới điểm kết thúc kia. Khi đến điểm kết thúc, đầu đọc di chuyển ngược lại
- Còn gọi là thuật toán thang máy (*elevator algorithm*)
- Minh họa cần tổng số di chuyển đầu đọc 208 cylinder.

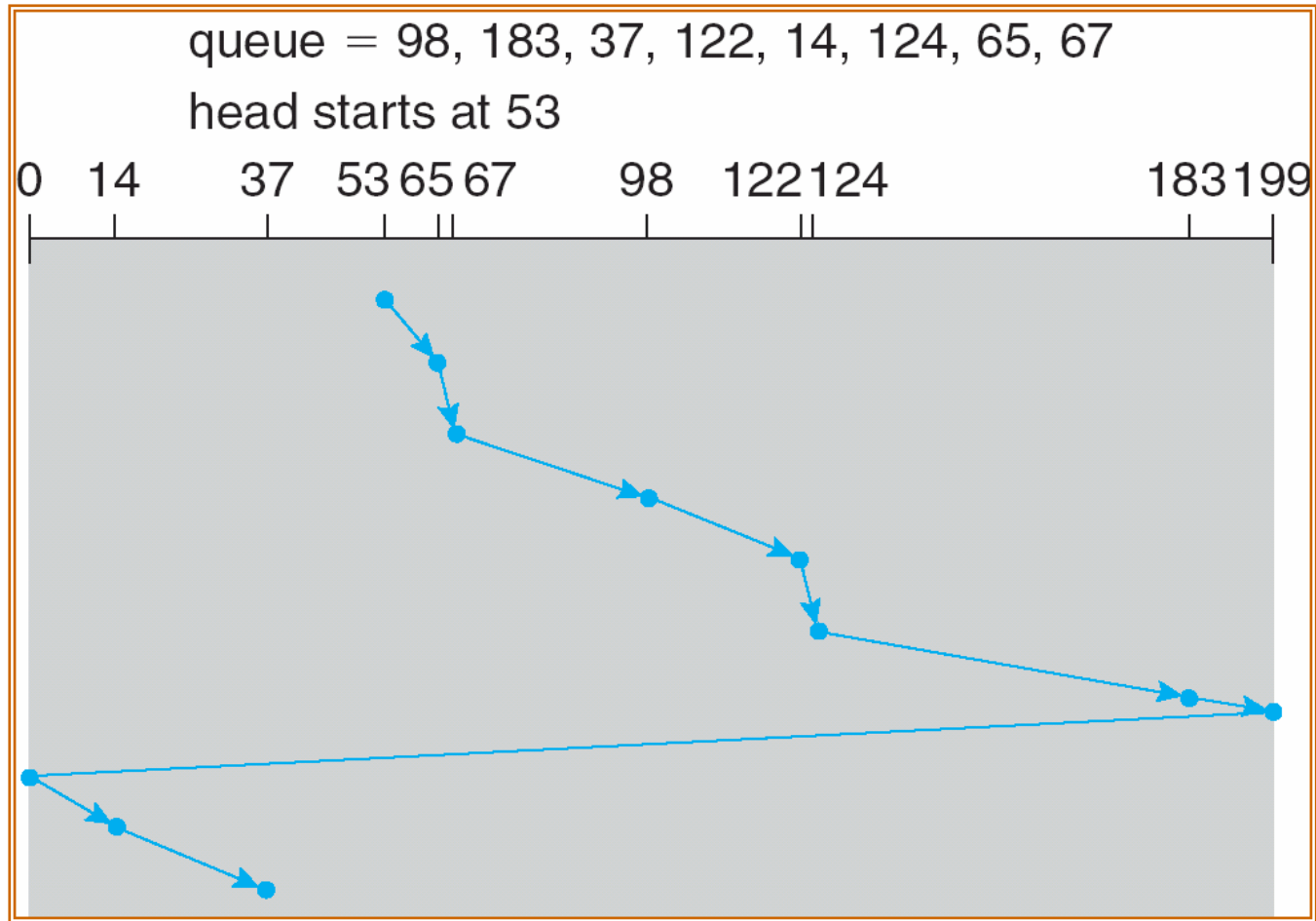
SCAN (Cont.)



C-SCAN

- Đầu đọc cũng di chuyển từ một điểm kết thúc tới điểm kia nhưng khi tới điểm kết thúc kia, nó lập tức di chuyển về điểm bắt đầu mà không phục vụ bất cứ yêu cầu nào trên đường quay về
- Cung cấp thời gian chờ giống nhau hơn so với SCAN

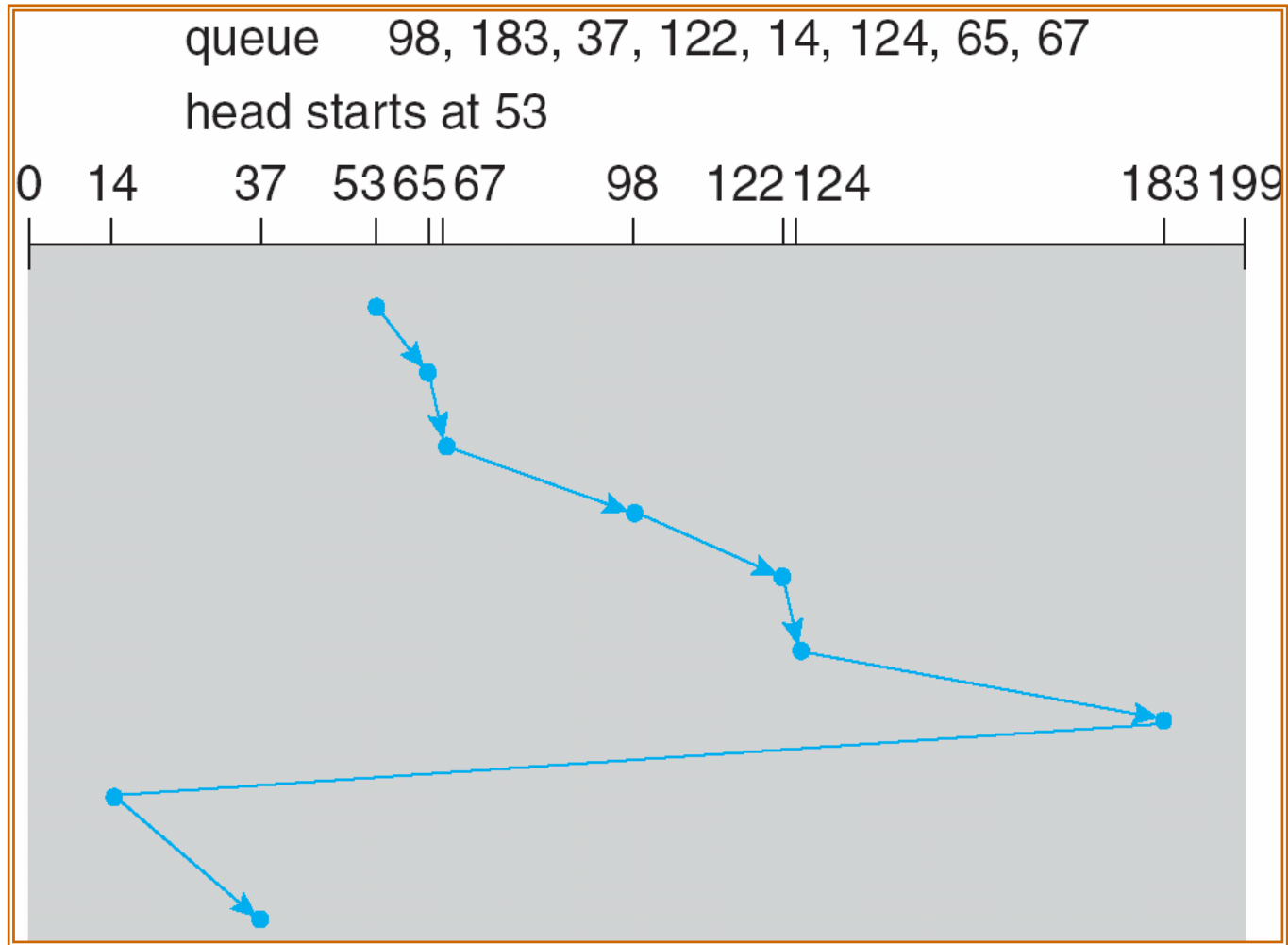
C-SCAN (Cont.)



C-LOOK

- Là phiên bản của C-SCAN
- Tay đĩa chỉ di chuyển tới yêu cầu cuối cùng trên mỗi hướng sau đó quay lại ngay, mà không đi tới điểm cuối của đĩa

C-LOOK



Lựa chọn thuật toán

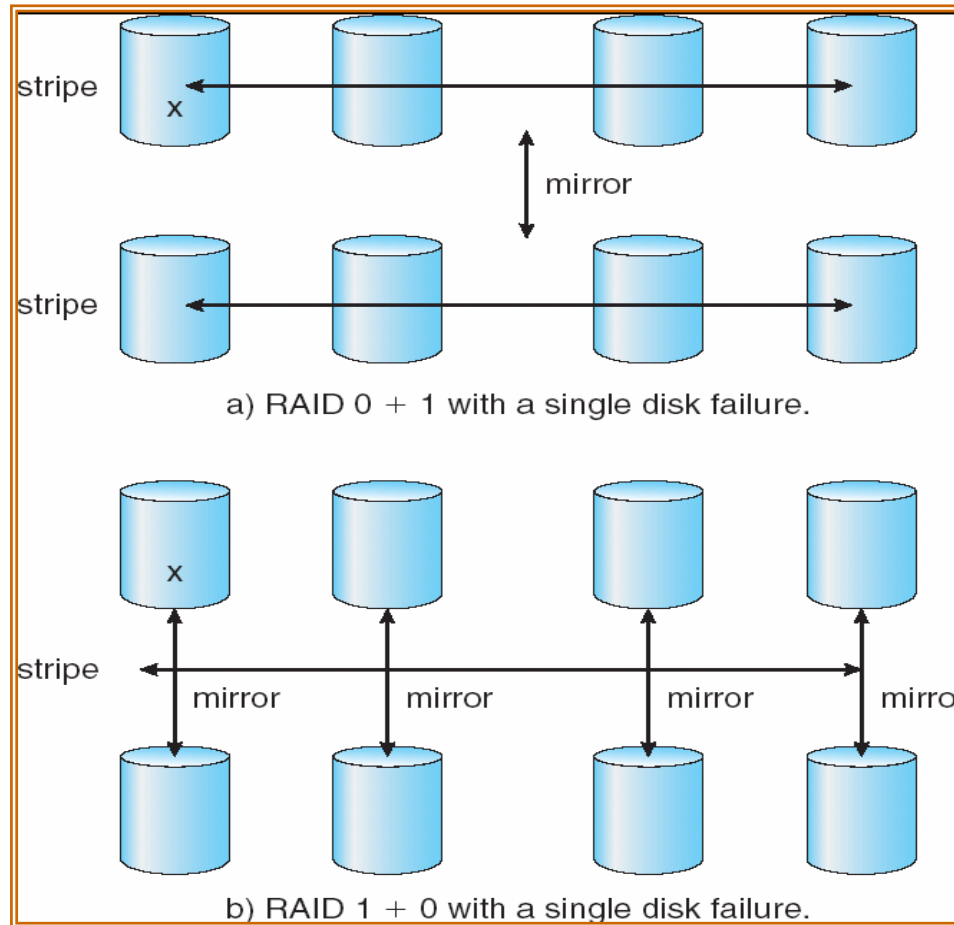
- SSTF là phổ biến với yêu cầu tự nhiên
- SCAN và C-SCAN tốt hơn cho những hệ thống có tải lớn trên đĩa
- Thuật toán lập lịch nên được viết như một module riêng rẽ của HĐH, cho phép nó thay thế các thuật toán khác nếu cần thiết

Cấu trúc RAID

(Redundant Array of Independent Disks)

- Cải tiến trong kỹ thuật sử dụng đĩa liên quan đến sử dụng nhiều đĩa làm việc cộng tác
- Sử dụng một nhóm các đĩa như là một đơn vị lưu trữ duy nhất. Một file ghi được phân bố trên nhiều đĩa. Cải thiện tốc độ truy cập đĩa
- RAID cải tiến sự tin cậy của hệ thống bằng cách lưu trữ dữ liệu dư thừa
- RAID hầu hết sử dụng đĩa cứng SCSI

RAID (0 + 1) và (1 + 0)



Lưu trữ ổn định

(Stable-Storage Implementation)

- Tạo bản sao thông tin (Replicate information) trên nhiều hơn 1 thiết bị lưu trữ độc lập
- Cập nhật thông tin được quản lý để đảm bảo rằng có thể phục hồi dữ liệu vững chắc sau bất kỳ thất bại trong chuyển dữ liệu

Thiết bị lưu trữ thứ 3

- Giá rẻ là đặc tính của thiết bị lưu trữ
- Lưu trữ đa phương tiện có thể di dời được
Ví dụ phổ biến là: CD-ROM, DVD và các loại khác