



# **CHƯƠNG 5**

# **QUẢN LÝ NHẬP XUẤT**

# Nhiệm vụ

- Mục tiêu của bộ phận quản lý nhập xuất:
  - Tạo thành một lớp giao tiếp độc lập thiết bị
    - Che giấu các chi tiết kỹ thuật của các thiết bị phần cứng
    - Quản lý và sửa lỗi
  - Làm cho các thiết bị phần cứng đơn giản và dễ dùng
  - Cho phép chia sẻ các thiết bị phần cứng
    - Xây dựng các cơ chế bảo vệ các thiết bị được chia sẻ
    - Điều phối thiết bị để phục vụ cho nhiều nhu cầu sử dụng. cùng lúc

# Ví dụ về các thiết bị nhập xuất

- Các thiết bị giao tiếp:
  - Các thiết bị chỉ nhập : bàn phím, chuột, joystick...
  - Các thiết bị chỉ xuất : màn hình, máy in
  - Các thiết bị vừa nhập vừa xuất: card mạng.
- Các thiết bị lưu trữ:
  - Thiết bị vừa xuất, vừa nhập: đĩa (cứng/mềm), băng từ
  - Thiết bị chỉ xuất: CD-ROM.

# Ví dụ về các thiết bị nhập xuất

Device	Data rate
Keyboard	10 bytes/sec
Mouse	100 bytes/sec
56K modem	7 KB/sec
Telephone channel	8 KB/sec
Dual ISDN lines	16 KB/sec
Laser printer	100 KB/sec
Scanner	400 KB/sec
Classic Ethernet	1.25 MB/sec
USB (Universal Serial Bus)	1.5 MB/sec
Digital camcorder	4 MB/sec
IDE disk	5 MB/sec
40x CD-ROM	6 MB/sec
Fast Ethernet	12.5 MB/sec
ISA bus	16.7 MB/sec
EIDE (ATA-2) disk	16.7 MB/sec
FireWire (IEEE 1394)	50 MB/sec
XGA Monitor	60 MB/sec
SONET OC-12 network	78 MB/sec
SCSI Ultra 2 disk	80 MB/sec
Gigabit Ethernet	125 MB/sec
Ultrium tape	320 MB/sec
PCI bus	528 MB/sec
Sun Gigaplane XB backplane	20 GB/sec

# Phân loại các thiết bị nhập xuất

- Phân loại theo mục đích sử dụng:

- Các thiết bị giao tiếp:

- Các thiết bị chỉ nhập : bàn phím, chuột, ...
- Các thiết bị chỉ xuất : màn hình, máy in
- Các thiết bị vừa nhập vừa xuất: card mạng.

- Các thiết bị lưu trữ

- Thiết bị vừa xuất, vừa nhập: đĩa (cứng/mềm), băng từ
- Thiết bị chỉ xuất: CD-ROM

- Phân loại theo phương pháp truy xuất:

- Thiết bị khối:

- Tổ chức theo từng khối riêng biệt và truy xuất ngẫu nhiên (VD: CD-ROM, HDD)

- Thiết bị tuần tự

- Gửi nhận theo chuỗi bit và phải truy xuất tuần tự (VD: Card mạng, Bàn phím)

# Các đặc tính nhập xuất

- Ba đặc tính khác nhau cần xem xét khi xử lý 1 thao tác nhập xuất:
  - Blocking and Non-blocking (Chặn và không chặn)
  - Buffered and Unbuffered (Bộ đệm và không có bộ đệm)
  - Synchronous and Asynchronous (đồng bộ và không đồng bộ)

# Blocking and Non-Blocking

- **Blocking** – Tiến trình gọi sẽ treo cho đến khi hoàn tất thao tác I/ O
  - Ví dụ: Trong thiết bị mạng, nếu muốn ghi 1000 bytes, thì HĐH ghi tất cả các byte cho đến khi ghi hoàn tất.
  - Nếu thiết bị không thể thực hiện lệnh ghi được (ví dụ hỏng dây nối)?  
=>kết thúc và trả về số bytes đã ghi được.
  - ➔ Dễ hiểu và dễ sử dụng.
  - ➔ Không hiệu quả trong một số trường hợp
- **Nonblocking** – HĐH đọc và ghi các bytes khi có thể, không cần ứng dụng phải dừng lại.
  - ➔ Hiện thực bằng kỹ thuật multi-threading
  - ➔ Trả về ngay đó byte được đọc hoặc ghi

# Buffered và Unbuffered

- **Buffered:**
  - Trong trường hợp buffer dữ liệu của thiết bị quá nhỏ, để không phải chờ quá lâu khi thực hiện IO
    - buffered I/O cho phép kernel copy lại dữ liệu
    - Bên write(): cho phép ứng dụng tiếp tục ghi dữ liệu
    - Bên read(): khi thiết bị báo có dữ liệu đến, kernel chép dữ liệu vào buffer. Khi tiến trình gọi read(), kernel chỉ việc copy từ buffer.
  - Khuyết điểm buffered I/O?
    - Thêm chi phí để thực hiện copy
    - Chậm trễ việc gửi dữ liệu
- **Unbuffered:** Không chấp nhận ghi dữ liệu vào kernel



# Synchronous và Asynchronous

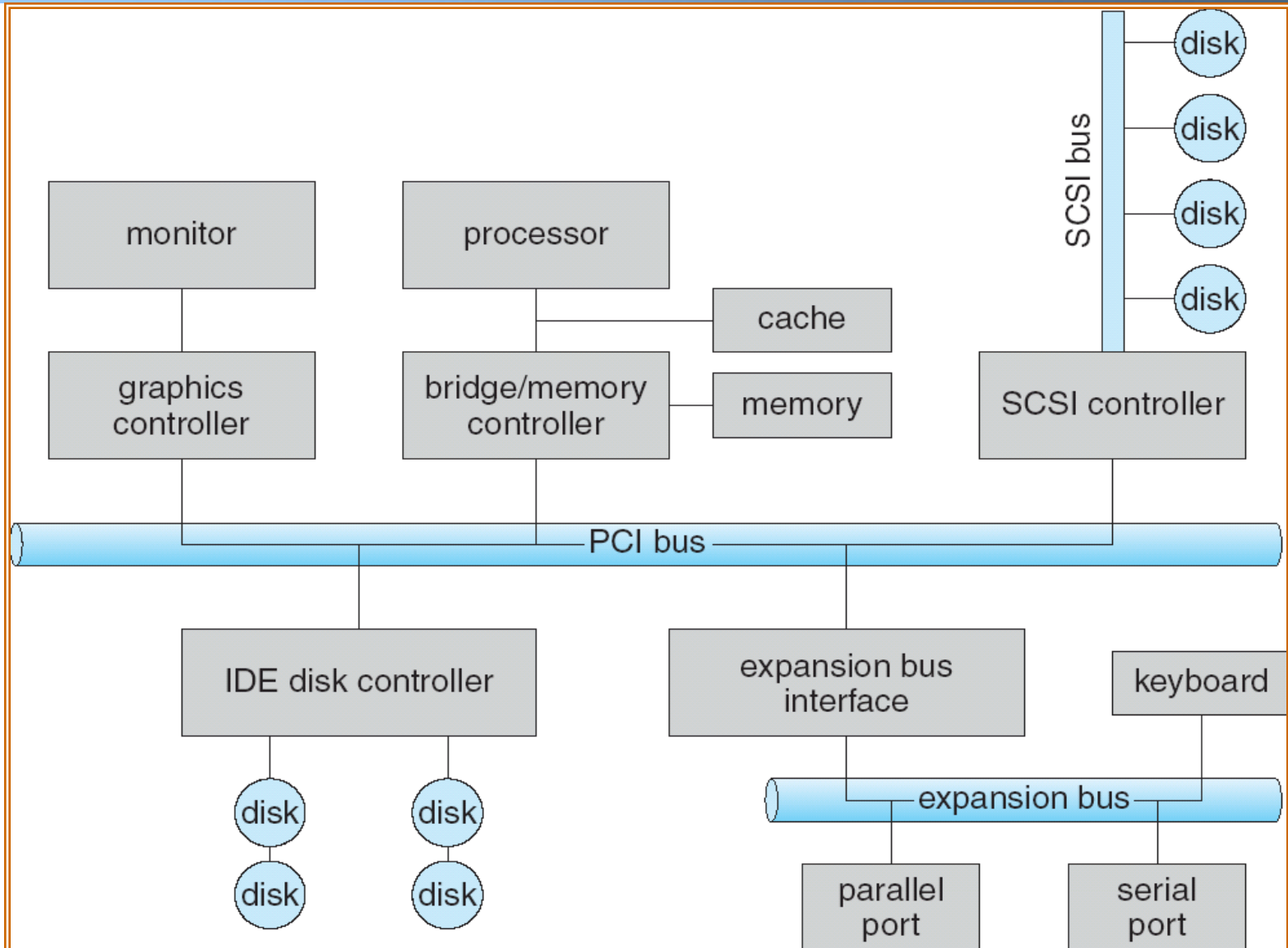
- **Synchronous:** các xử lý khác thuộc ứng dụng của người dùng cuối sẽ phải tạm dừng lại (paused) để chờ các thao tác nhập xuất của nó hoàn tất
- **Asynchronous:** các xử lý khác của ứng dụng có thể thực thi song song với các thao tác nhập xuất

# Bộ điều khiển thiết bị - Device Controller

- **Mỗi thiết bị nhập xuất:**
  - Thành phần cơ: Bản thân thiết bị
  - Thành phần điện: bộ điều khiển cho thiết bị đó (device controller)
- **Bộ điều khiển:**
  - Trung gian giao tiếp giữa thiết bị và HĐH
  - Thông qua bus - hệ thống mạch truyền dẫn
  - **Công việc:**
    - Nhận lệnh từ HĐH
    - Chuyển đổi dãy bit thành các byte và đặt chúng vào trong bộ đệm (buffer) của bộ điều khiển.
    - Báo hiệu cho HĐH khi tác vụ hoàn tất.



# Các thiết bị nhập xuất và bus hệ thống



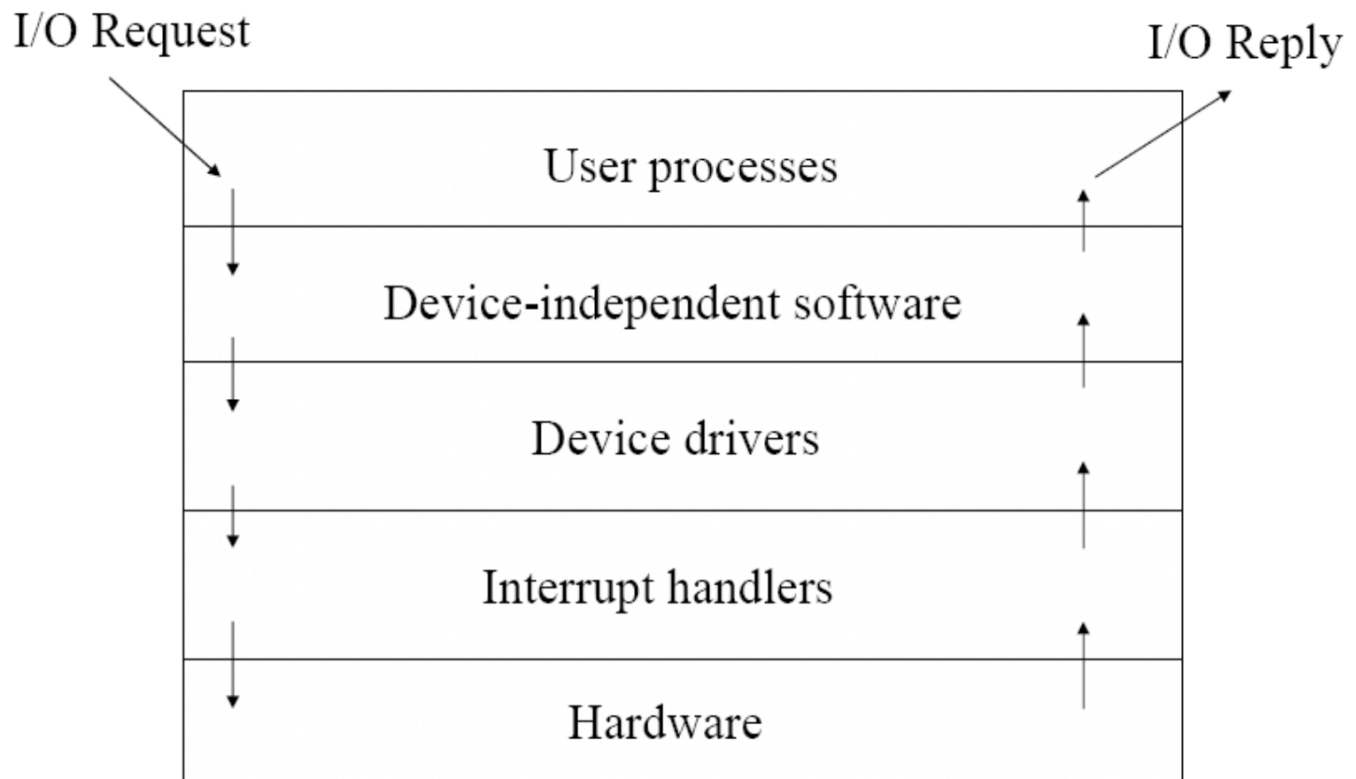
# Địa chỉ giao tiếp thiết bị

- HĐH giao tiếp với thiết bị thông qua **địa chỉ nhập xuất của bộ điều khiển**:

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)

# Mô hình phân lớp trong quản lý nhập xuất

- Hệ thống nhập xuất được tổ chức theo từng lớp, mỗi lớp có 1 chức năng nhất định và có sự hỗ trợ liên hoàn lẫn nhau



# Phần mềm độc lập thiết bị (Device- independent software)

- Chức năng:
  - Độc lập với thiết bị: tạo ra giao tiếp chung cho tất cả các thiết bị
  - Bảo vệ thiết bị
  - Cung cấp bộ đệm (buffer) để hỗ trợ cho quá trình đồng bộ hoá hoạt động của hệ thống
  - Cấp phát và giải phóng thiết bị
  - Thông báo lỗi cho người dùng (nếu có)

# Trình điều khiển thiết bị (Device driver)

- Chức năng:

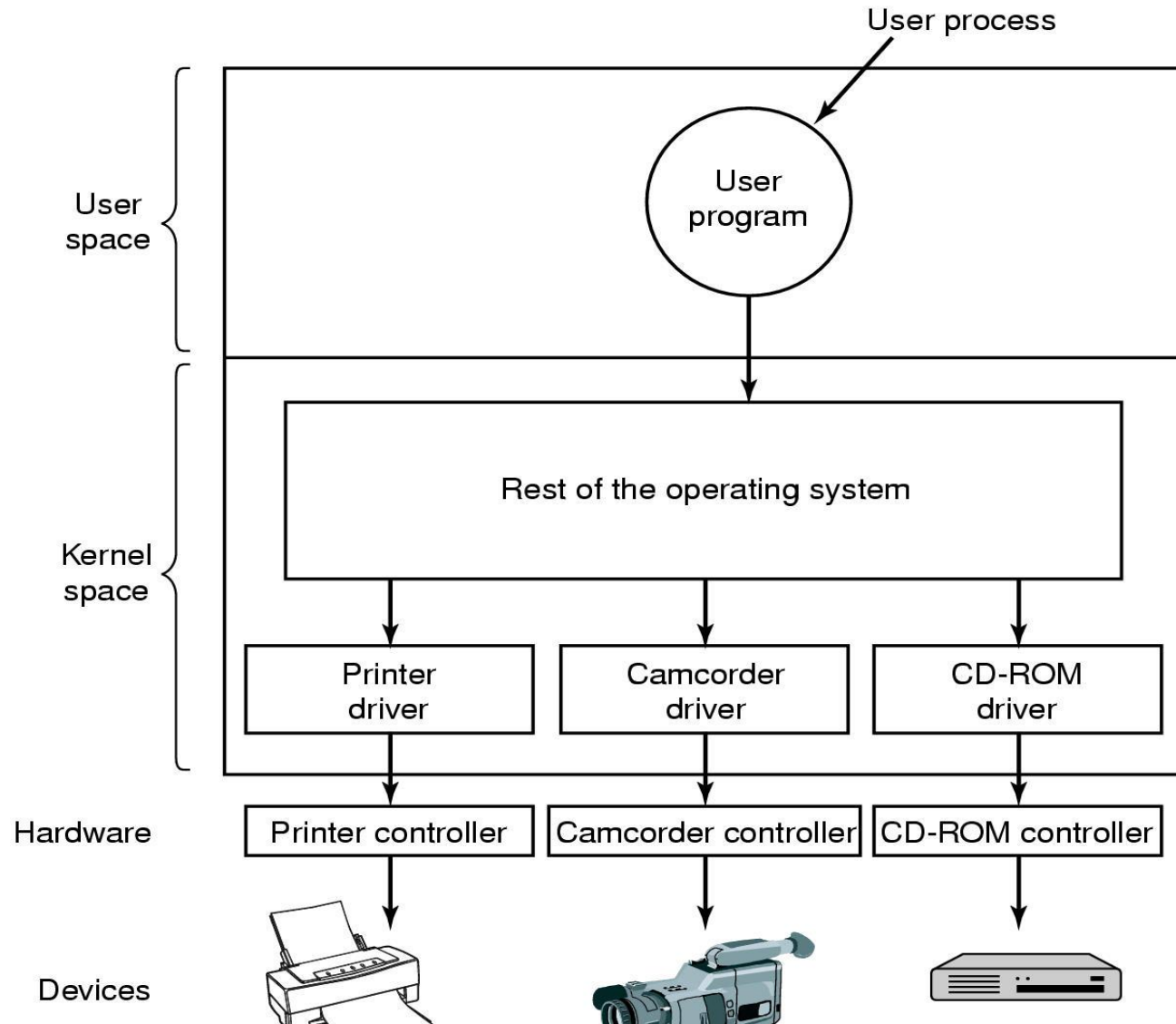
- Nhận yêu cầu từ phía lớp phần mềm độc lập thiết bị.
- Chuyển đổi yêu cầu trừu tượng này thành cụ thể.
- Điều phối yêu cầu này cho bộ điều khiển thiết bị (device controller).
- Giám sát thực hiện yêu cầu.

- Ví dụ:

- HĐH muốn đọc tập tin “io.sys” trên đĩa ở thư mục C:\.
- Trình điều khiển đĩa phải hiểu là cần đọc khối nào.
- Trình điều khiển đĩa chuyển yêu cầu này cho bộ điều khiển đĩa.
- Bộ điều khiển đĩa phải kiểm tra hoạt động của motor đĩa, xác định đầu đọc đã đúng vị trí chưa.



# Device driver và Device controller ?





# Bộ kiểm soát ngắt - Interrupt handler

- Tương tác giữa HĐH và các thiết bị phần cứng đều được thực hiện thông qua cơ chế ngắt (interrupt).
- Bộ kiểm soát ngắt sẽ tiếp nhận các ngắt từ HĐH và ứng dụng của người dùng cuối.
- Dựa trên bảng “Interrupt vector” để phân phối các ngắt đến các bộ điều khiển thiết bị tương ứng.
- Quản lý và giám sát quá trình thực hiện ngắt.
- Nhận ngắt thông báo quá trình nhập xuất hoàn tất hoặc có lỗi xảy ra trong quá trình nhập xuất từ bộ điều khiển thiết bị để chuyển lên cho HĐH.

# TRUY CẬP I/O BẰNG PHƯƠNG PHÁP POLLING

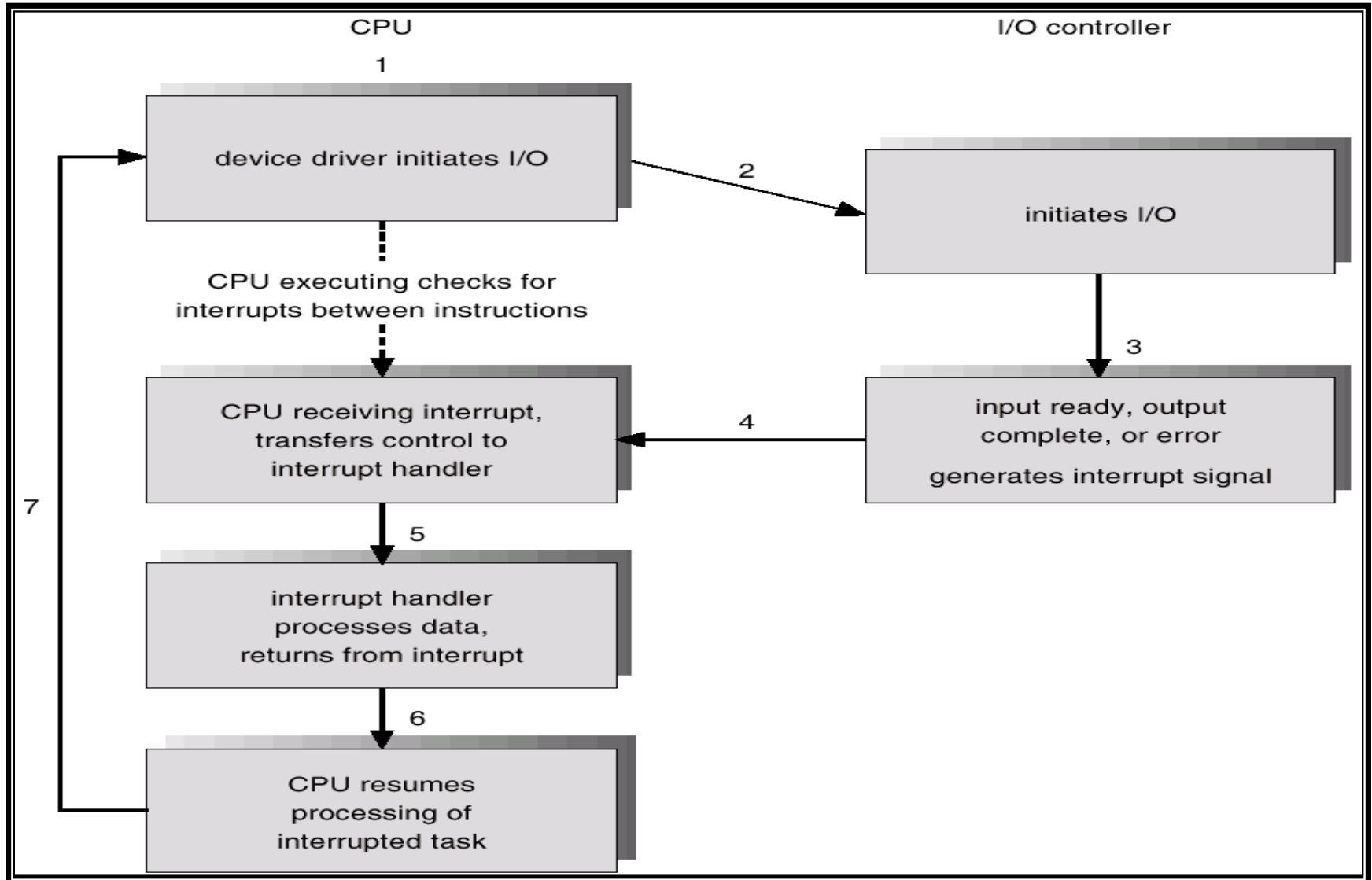
- Hiện thực:
  - Trạng thái thiết bị: busy, error hay command-ready
  - Khi thiết bị rảnh, CPU ra lệnh truy cập I/O
  - Controller đọc lệnh và thực thi tác vụ.
  - Khi thực thi xong, controller đặt lại trạng thái của thiết bị: idle hoặc error.
  - CPU liên tục kiểm tra trạng thái thiết bị để đọc, ghi dữ liệu nếu cần thiết
- Nhận xét

# TRUY CẬP I/O BẰNG CÁCH SỬ DỤNG NGẮT QUÃNG

- Thiết bị I/O tạo ngắt quãng khi hoàn tất tác vụ
- Khi có ngắt quãng
  - Trình xử lý ngắt quãng nhận interrupt
  - Xác định thiết bị gây ngắt quãng
  - Lấy dữ liệu từ device register (lệnh trước là lệnh đọc)
  - Khởi động lệnh tiếp theo cho thiết bị đó
- Xử lý ngắt quãng theo độ ưu tiên, có thể hoãn hoặc hủy một số ngắt quãng
- Ngắt quãng cũng được dùng khi xảy ra biến cố
- Nhận xét?



# CHU KỲ NGẮT QUÃNG CỦA I/O



# Cơ chế truy cập bộ nhớ trực tiếp DMA

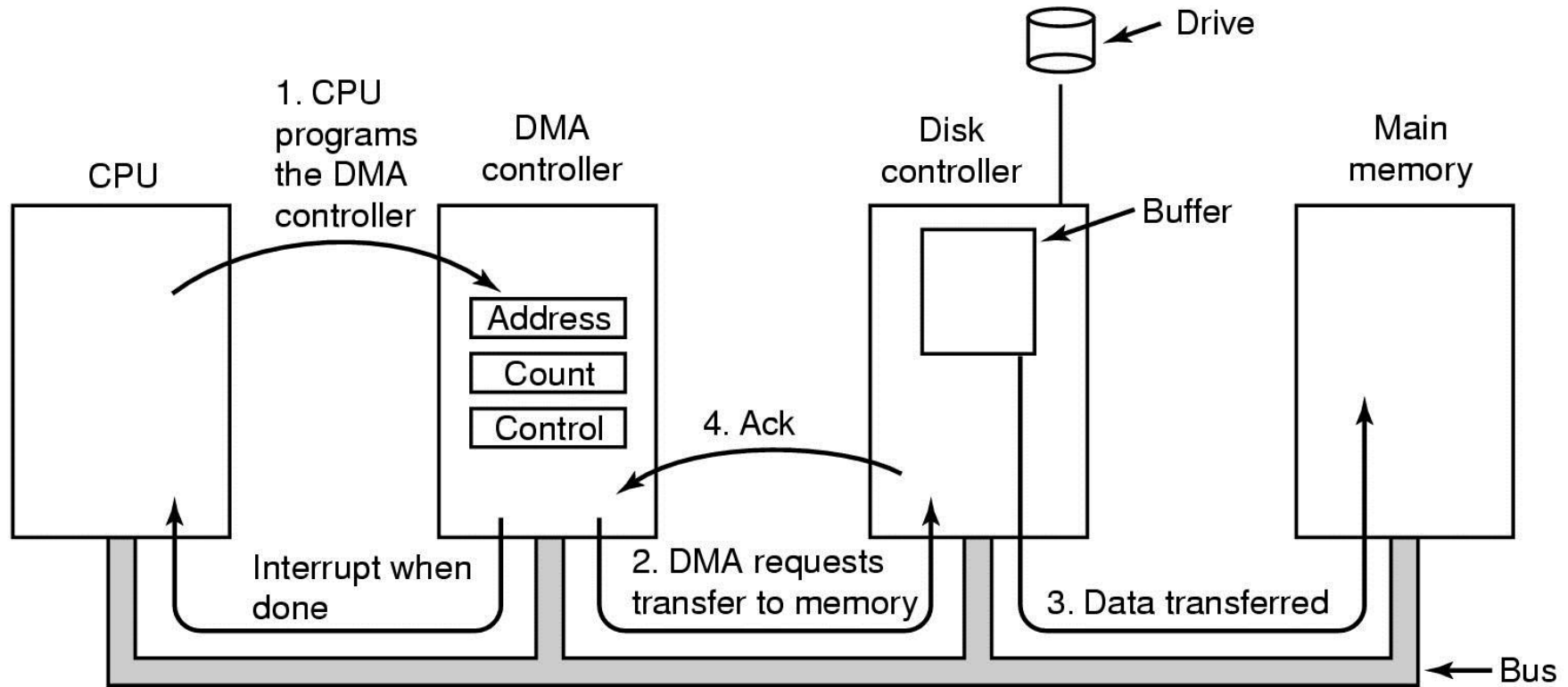
- **DMA = Direct Memory Access**
- Quá trình đọc đĩa không có DMA:
  - HĐH chuyển yêu cầu đọc đĩa cho bộ điều khiển đĩa.
  - Bộ điều khiển đọc tuần tự các khối trên đĩa đưa vào bộ đệm của bộ điều khiển đĩa.
  - Bộ điều khiển đĩa tạo ngắt để báo CPU biết quá trình đọc đĩa hoàn tất.
  - *CPU lần lượt lấy từng byte dữ liệu từ bộ đệm của bộ điều khiển đĩa để chuyển về bộ nhớ chính để thao tác.*
- Nhận xét:
  - Lãng phí thời gian xử lý của CPU để chuyển dữ liệu từ bộ đệm của bộ điều khiển đĩa về bộ nhớ chính

# Cơ chế truy cập bộ nhớ trực tiếp DMA

- Cơ chế DMA giúp CPU không bị lãng phí bằng cách:
  - HĐH gửi cho bộ điều khiển đĩa các thông số gồm:
    - các khối cần đọc
    - vị trí lưu trữ các khối này bên trong bộ nhớ chính (địa chỉ DMA)
    - số byte cần đọc.
  - Bộ điều khiển đĩa đọc các khối cần thiết lưu vào trong bộ đệm của nó.
  - ***Sau khi đọc xong, bộ điều khiển chuyển lần lượt từng byte từ bộ đệm của nó về địa chỉ DMA – nơi cần lưu trữ dữ liệu cần thiết bên trong bộ nhớ chính.***
  - Bộ điều khiển đĩa tạo 1 ngắt để thông báo cho CPU biết quá trình chuyển dữ liệu đã hoàn tất.



# Cơ chế DMA



- Không lãng phí CPU cho việc chuyển đổi dữ liệu bộ đệm sang bộ nhớ

# CÁC CHỨC NĂNG QUẢN LÝ I/O

- Do module quản lý I/O của hệ điều hành (Kernel I/O Subsystem) đảm nhận
- Các chức năng chính
  - Định thời I/O ( I/O scheduling)
  - Dành riêng thiết bị (device reservation)
  - Xử lý lỗi (error handling)
  - Buffering
  - Caching
  - Spooling



# Quản lý lỗi & Bảo vệ nhập xuất thiết bị

- Nguyên nhân:

- Người dùng hay HĐH có thể vô tình hay cố ý thực hiện các lệnh/thao tác nhập xuất bất hợp pháp gây hại cho hệ thống và thiết bị.
- VD: đọc dữ liệu trong khi đang ghi

- Khắc phục:

- Định nghĩa trước và gán đặc quyền cho các lệnh nhập xuất của hệ thống dưới dạng các lời gọi hệ thống (system call).
- Giám sát quá trình nhập xuất của người dùng cuối.
- Tất cả quá trình nhập xuất của UD phải được thực hiện thông qua các lời gọi hệ thống.

# Quản lý lỗi & bảo vệ nhập xuất thiết bị

- Khi gặp lỗi trong quá trình nhập xuất, các bộ điều khiển thiết bị sẽ trả về cho HĐH mã lỗi tương ứng
- HĐH diễn dịch mã lỗi trả về để có phương án giải quyết thích hợp.
- HĐH cũng diễn dịch và lưu vào nhật ký hệ thống (system log) các lỗi tương ứng để giúp người quản trị hệ thống giám sát lỗi và phục hồi.

# BUFFERING

- Lưu dữ liệu trong bộ nhớ thay vì chuyển trực tiếp dữ liệu giữa các thiết bị
- Có thể được hiện thực ở
  - Cấp phần cứng
  - Cấp hệ điều hành
- Mục đích
  - Xử lý vấn đề tốc độ các thiết bị khác nhau
  - Xử lý vấn đề kích thước khối dữ liệu trao đổi giữa các thiết bị khác nhau
  - Giảm thiểu thời gian quá trình bị chặn khi ghi dữ liệu

# CACHING & SPOOLING

- **Caching**

- Dùng vùng nhớ tốc độ cao để lưu bản sao của dữ liệu thường xuyên truy xuất
- Đảm bảo tính nhất quán của cache:
  - Kỹ thuật write through
  - Kỹ thuật write back

- **Spooling** (*Simultaneous Peripheral Operation On-line*)

- Dùng thiết bị lưu trữ tốc độ trung bình làm trung gian giao tiếp giữa 2 thiết bị có tốc độ chênh lệch nhau
- Ví dụ : dịch vụ in ấn



# TĂNG HIỆU SUẤT HỆ THỐNG I/O

- Giảm thiểu copy dữ liệu (caching)
- Giảm tần số interrupt (dùng kích thước khối dữ liệu truyền nhận lớn, smart controller...)
- Giảm tải cho CPU bằng DMA
- Tăng số lượng thiết bị để tránh tranh chấp
- Tăng dung lượng bộ nhớ thực
- Cân bằng hiệu suất CPU, bộ nhớ, bus và thiết bị I/O để đạt throughput cao nhất
- ...



TRƯỜNG ĐẠI HỌC  
CÔNG NGHỆ ĐÔNG Á  
EAST ASIA UNIVERSITY OF TECHNOLOGY

Thanks  
for  
listening

