

# Hệ thống quản lý nhập xuất

Môn học: Hệ điều hành



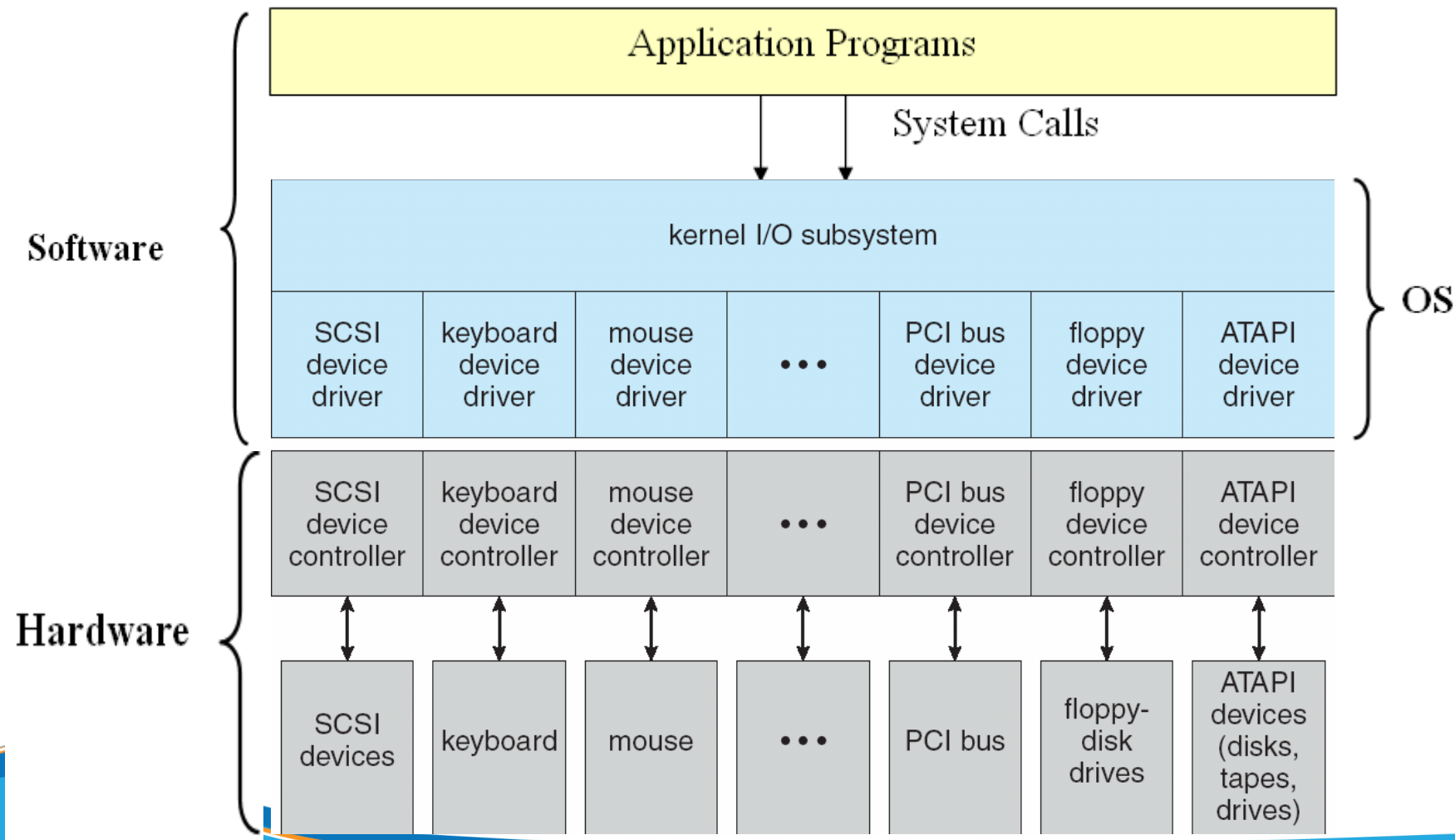
**fit@hcmus**

**KHOA CÔNG NGHỆ THÔNG TIN  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

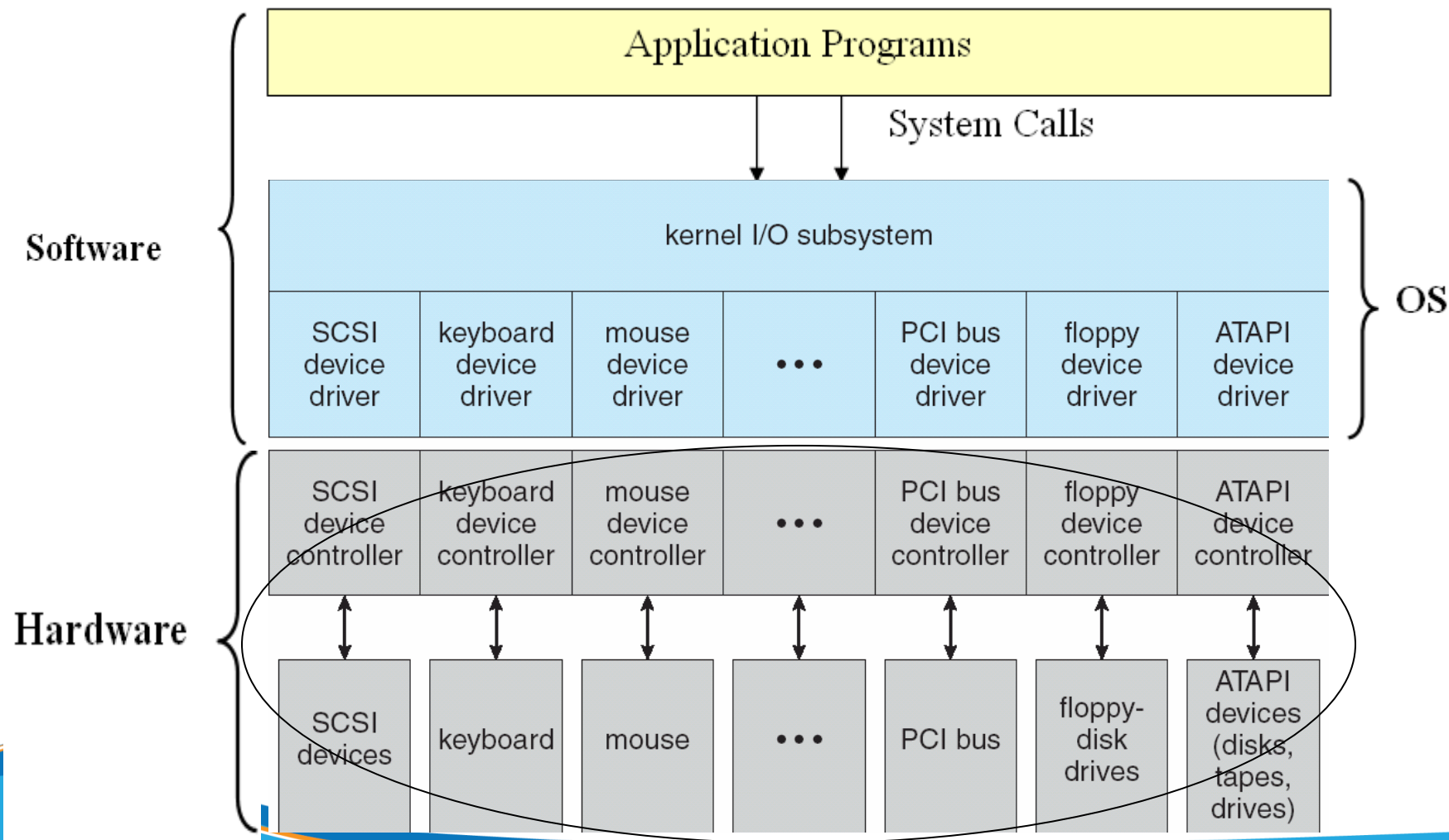
# Nội dung

- Phần cứng nhập xuất
- Một số cơ chế nhập xuất
- Giao tiếp với thiết bị nhập xuất
- I/O Subsystem

# Cấu trúc tổng quát hệ thống nhập xuất



# Phần cứng nhập xuất

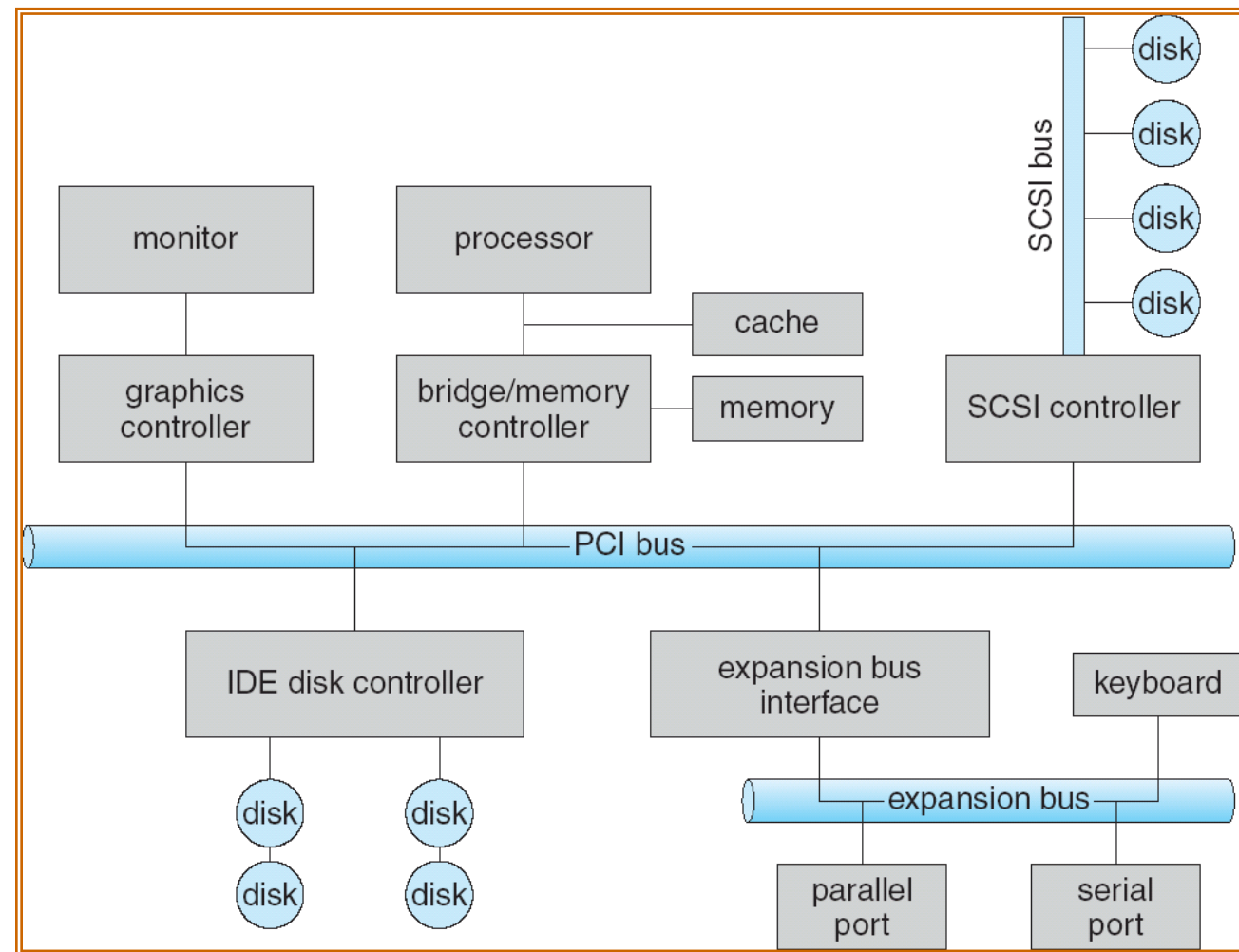


# Phần cứng nhập xuất (tt)

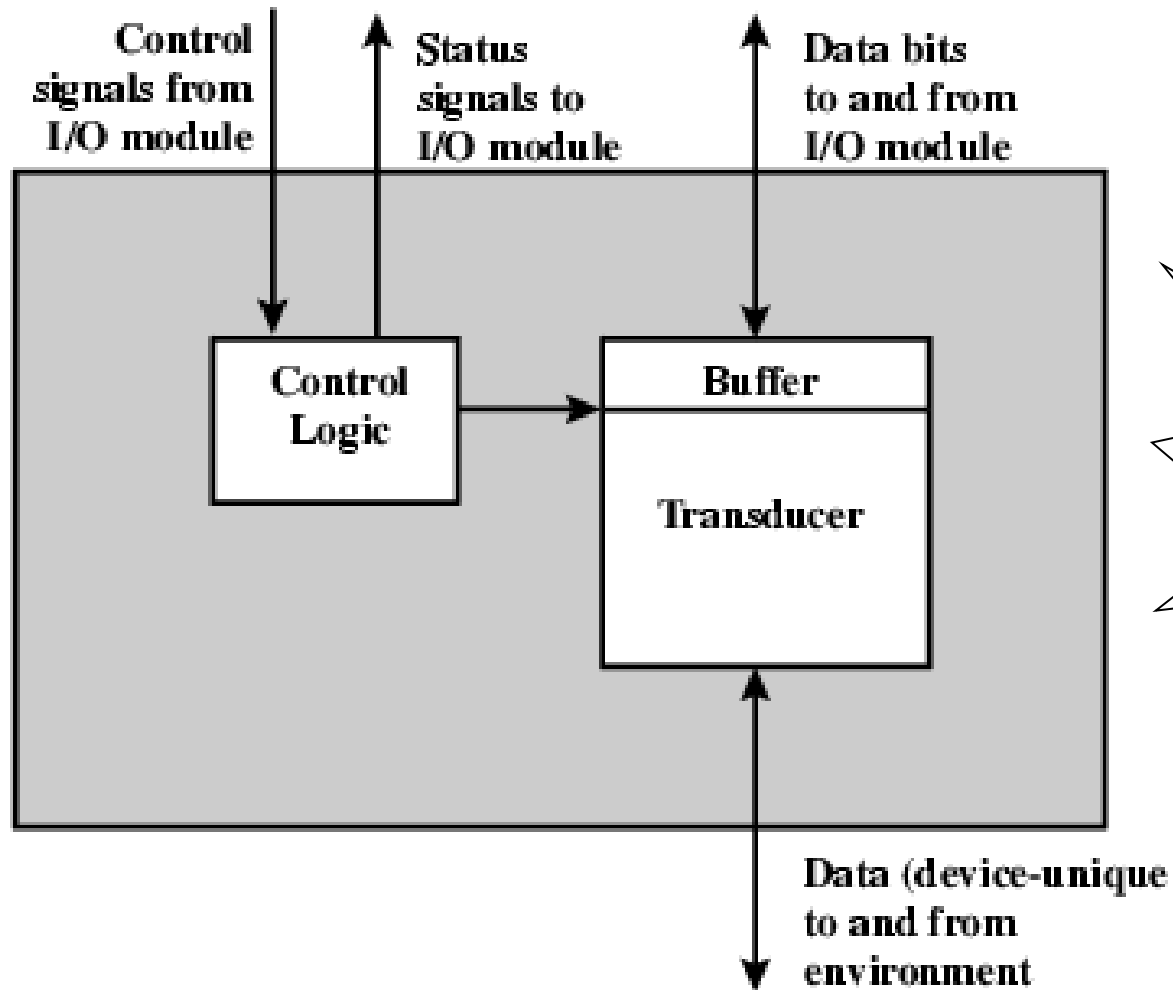
- Rất nhiều loại thiết bị nhập xuất:
  - Thiết bị giao tiếp người dùng: màn hình, bàn phím.
  - Thiết bị giao tiếp hệ thống: đĩa, băng từ.
  - Thiết bị giao tiếp hệ thống khác : cạc mạng, modem.

# Phần cứng nhập xuất (tt)

- Một số khái niệm chung
  - Cổng (Port)
  - Đường/ kênh truyền (Bus)
  - Mạch/ bộ điều khiển (Controller)

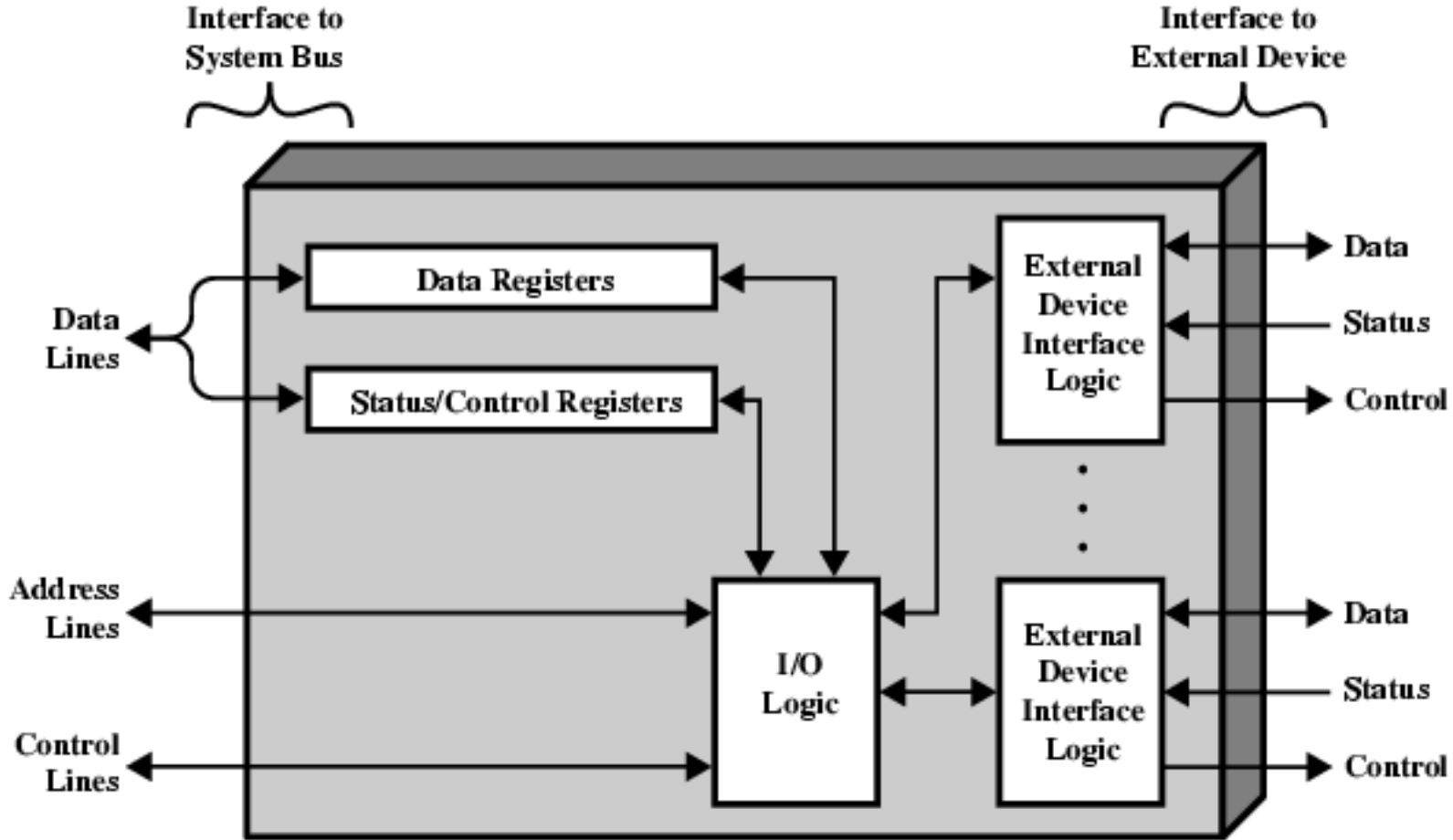


# Cấu trúc thiết bị



Consider the input from keyboard and output to screen

# Cấu trúc bộ điều khiển





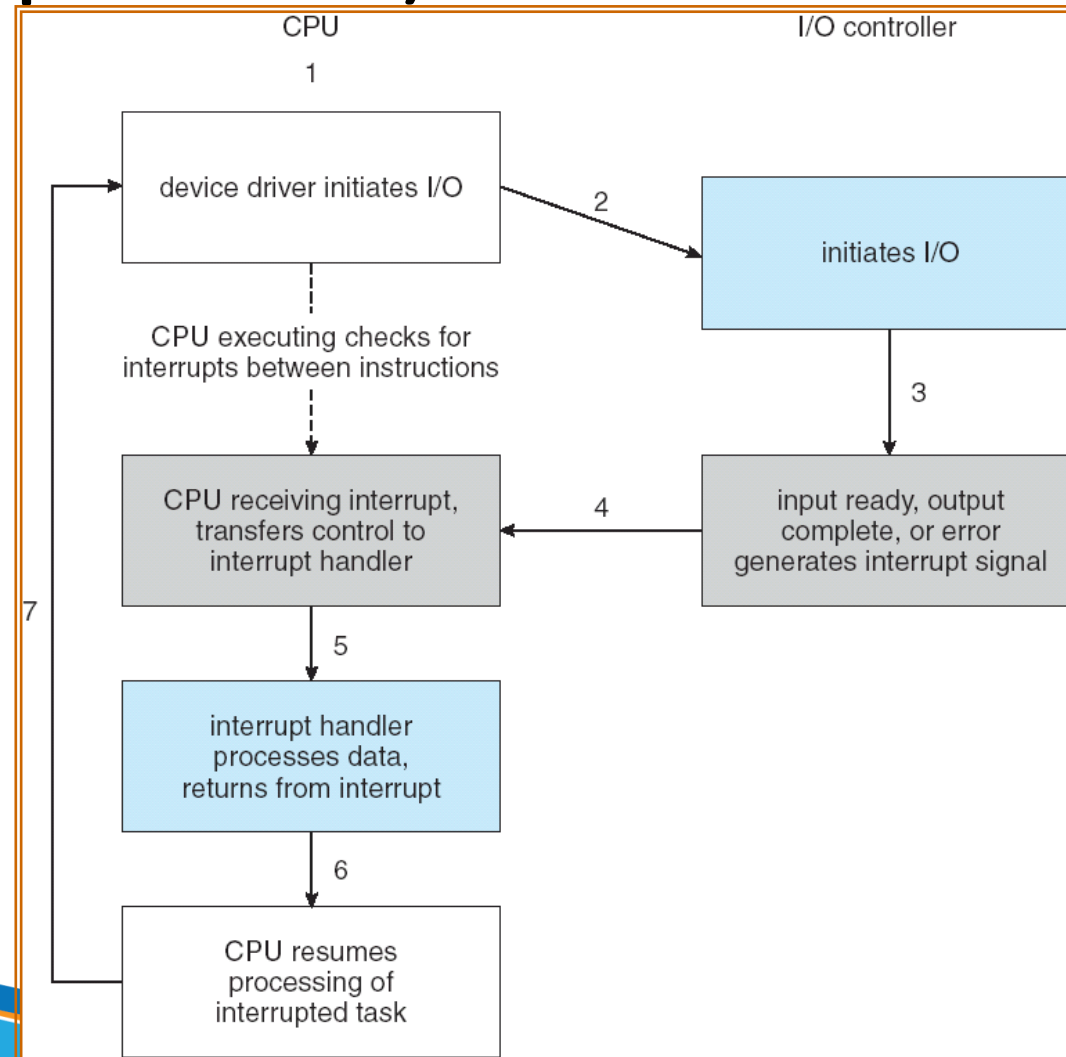
# Một số cơ chế nhập xuất (tt)

- Kiểm soát vòng (Polling/ Programed I/O).
  - CPU kiểm soát quá trình nhập xuất.
  - CPU tốn thời gian chờ bộ điều khiển thiết bị để truyền từng đơn vị dữ liệu.
- Ngắt (Interrupt Driven).
  - CPU không phải chờ, bộ điều khiển thiết bị sẽ báo (ngắt) CPU khi hoàn thành công việc theo từng đơn vị dữ liệu.
- Truy cập trực tiếp bộ nhớ (Direct Memory Access – DMA).
  - Hạn chế sự can thiệp của CPU trong quá trình truyền dữ liệu.
  - Cần một bộ điều khiển DMA.
  - Bộ điều khiển DMA báo (ngắt) CPU khi hoàn thành công việc theo từng block dữ liệu.

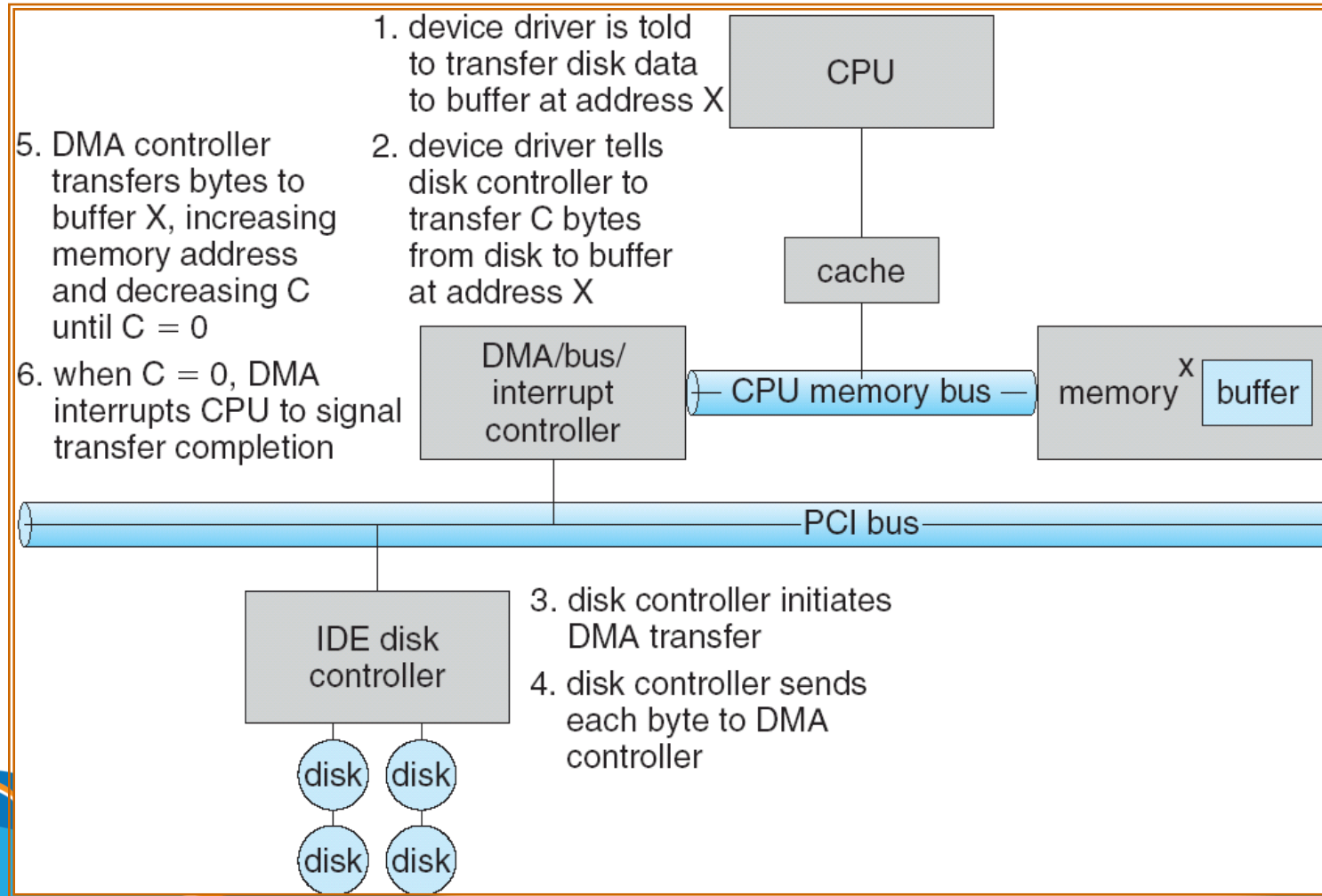
# Kiểm soát vòng (Polling/ Programed I/O)

1. Bộ xử lý liên tục kiểm tra bit *busy* cho đến khi bit này bị xóa, gọi là quá trình **Waiting** hay **Polling**.
2. Bộ xử lý bật bit *write* trong thanh ghi điều khiển và ghi một byte vào thanh ghi dữ liệu ra.
3. Bộ xử lý bật bit *command-ready*.
4. Khi bộ điều khiển nhận ra bit *command-ready* được bật, nó bật bit *busy*.
5. Bộ điều khiển đọc yêu cầu trong thanh ghi điều khiển, dữ liệu trong thanh ghi dữ liệu ra và thực thi yêu cầu nhập xuất.
6. Bộ điều khiển xóa bit *command-ready*, xóa bit *error* trong thanh ghi trạng thái (thực thi thành công), và xóa bit *busy* để kết thúc một yêu cầu nhập xuất.

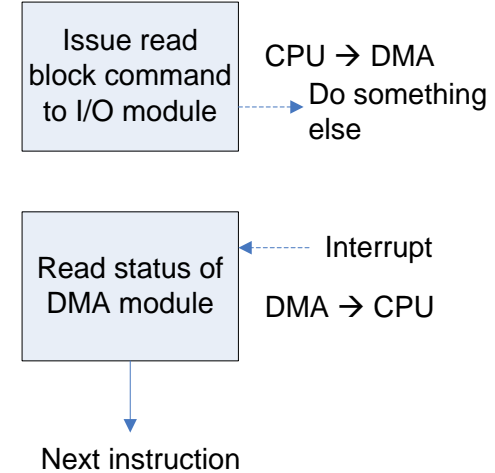
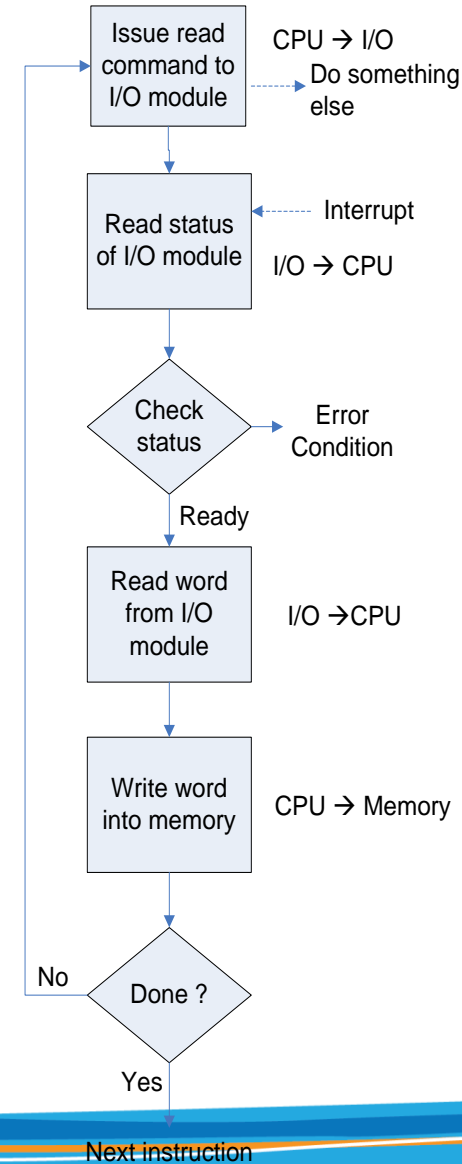
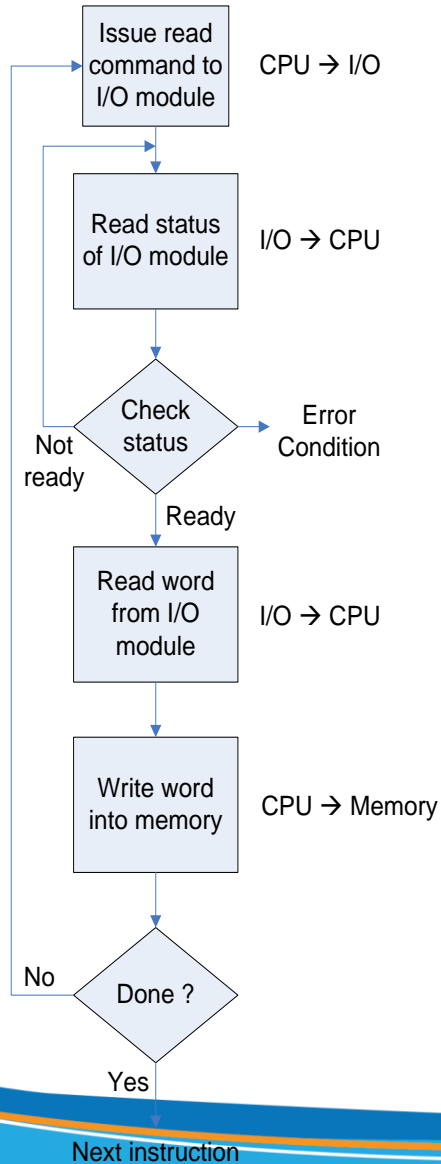
# Ngắt (Interrupt Driven)



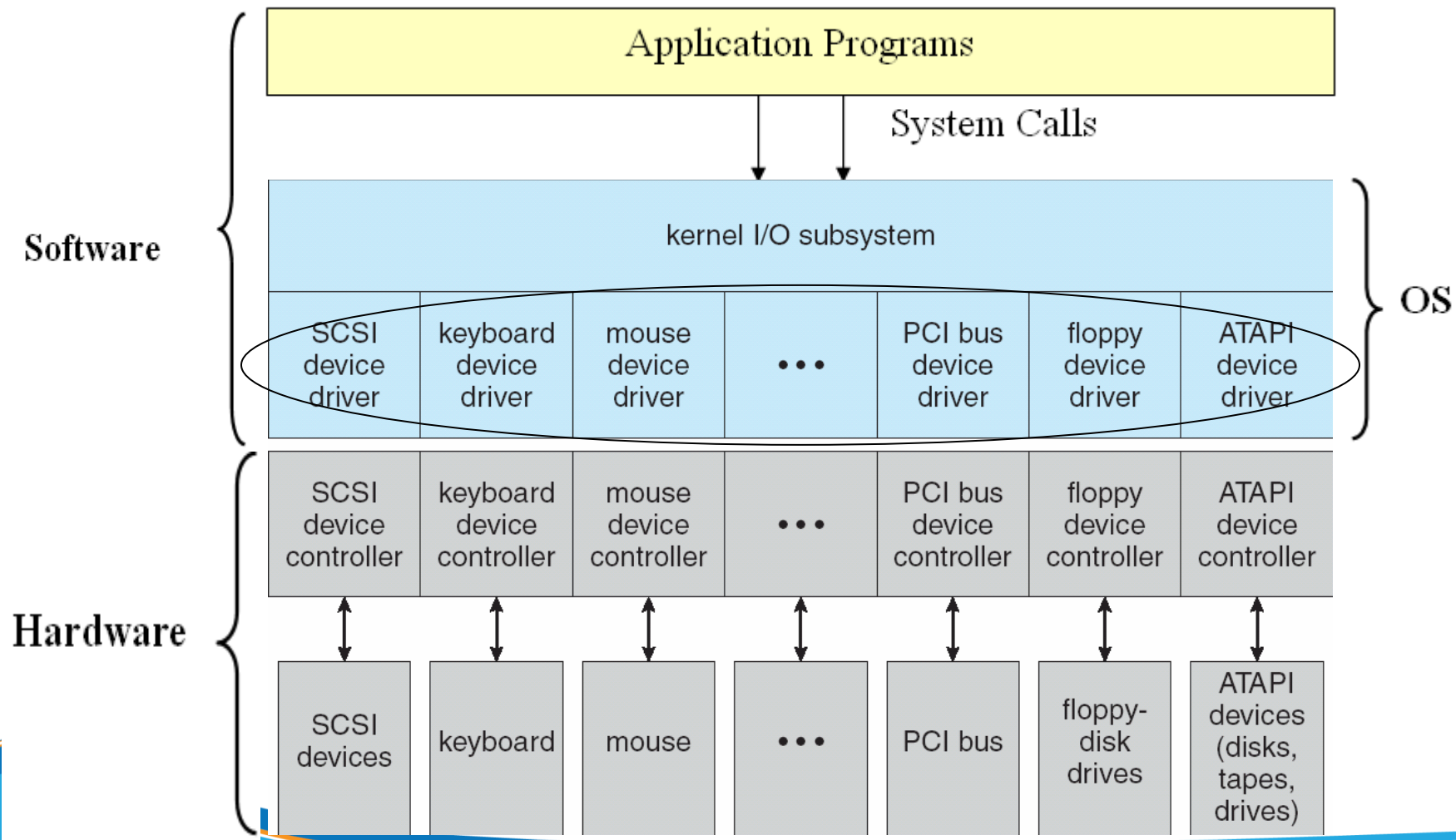
# Truy cập trực tiếp bộ nhớ (DMA)



# So sánh việc truyền một block dữ liệu



# Giao tiếp với thiết bị nhập xuất



# Giao tiếp với thiết bị nhập xuất (tt)

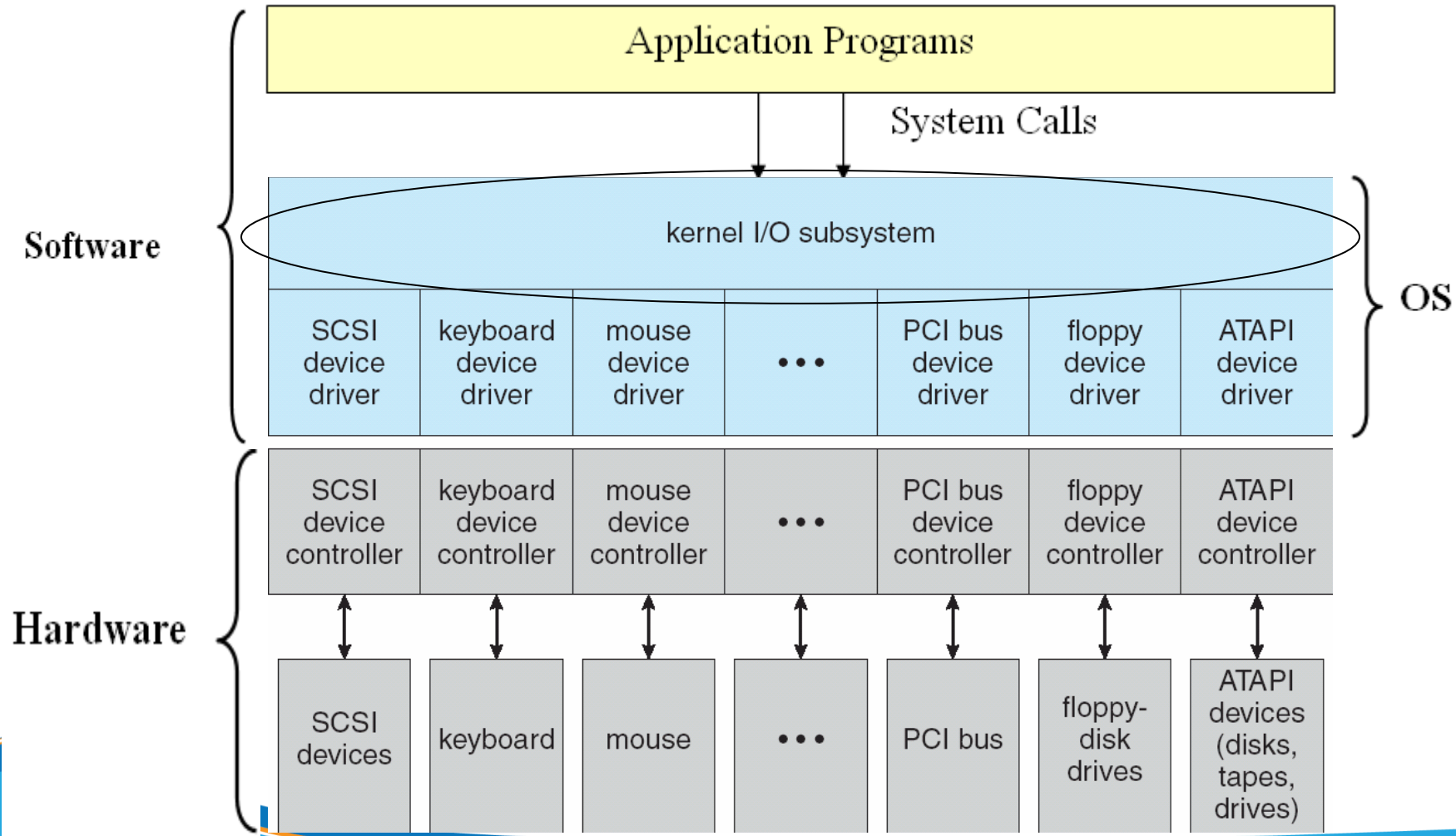
- Tầng trình điều khiển thiết bị (Device driver):
  - Đơn giản hóa việc giao tiếp với các thiết bị nhập xuất:
    - Phía nhà phát triển hệ điều hành.
    - Phía nhà sản xuất thiết bị nhập xuất.

# Các đặc điểm của thiết bị nhập xuất

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



# I/O Subsystem



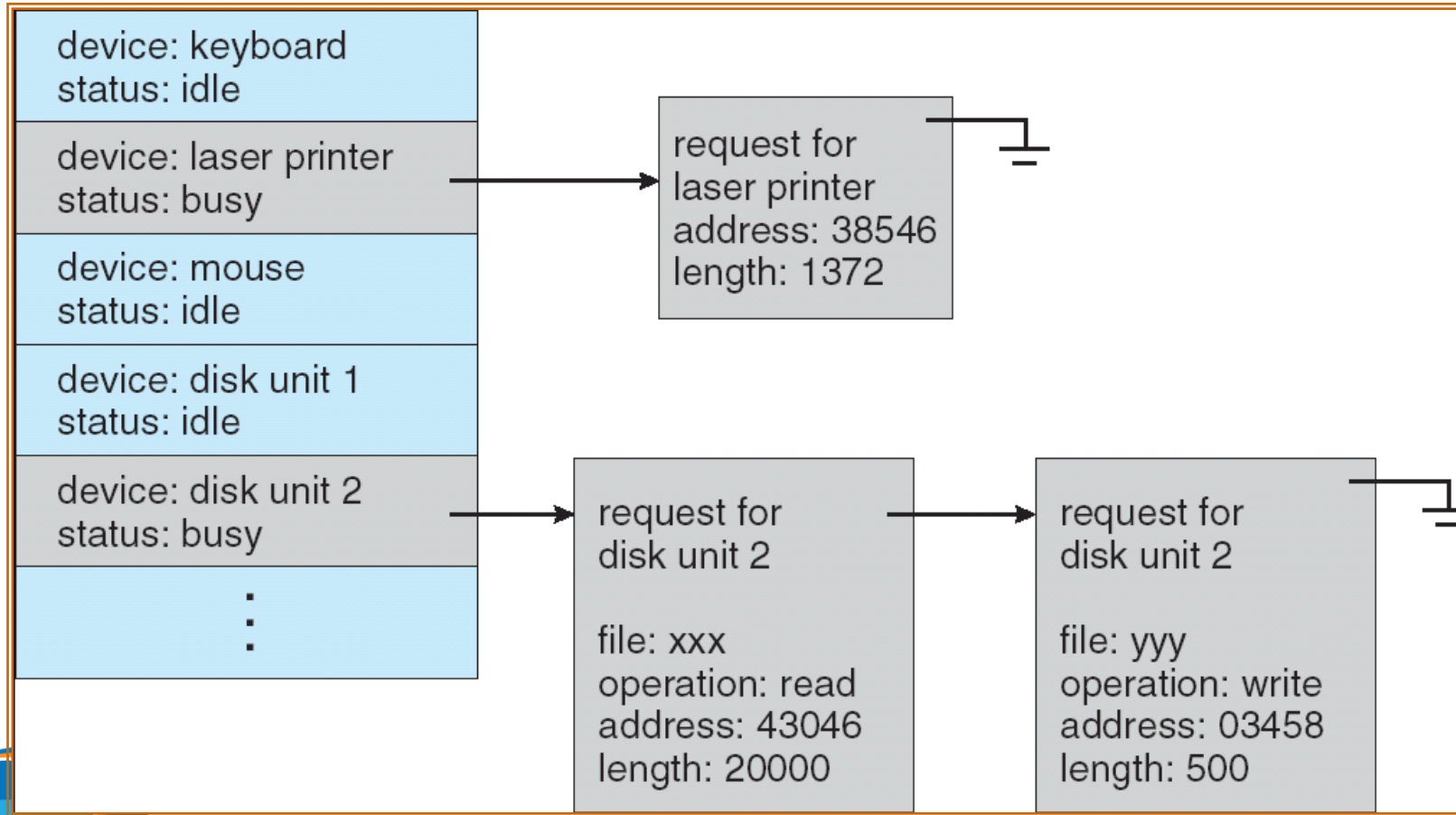
# I/O Subsystem (tt)

- Các thiết bị rất khác nhau về kiểu, chức năng, và tốc độ.
- Tiếp nhận yêu cầu nhập xuất từ lời gọi hệ thống và thực hiện một số xử lý ban đầu, rồi chuyển điều khiển cho trình điều khiển của thiết bị tương ứng.
- Cung cấp một số dịch vụ sau:
  - Lập lịch (Scheduling).
  - Đệm dữ liệu (Buffering).
  - Đệm dữ liệu truy xuất nhanh (Caching).
  - Spooling.
  - Quản lý lỗi (Error Handling).

# Lập lịch (Scheduling)

- Quyền truy cập thiết bị của các tiến trình là ngang nhau.
- Có thể giảm thời gian chờ trung bình để hoàn thành các yêu cầu nhập xuất
- Duy trì một hàng đợi yêu cầu cho mỗi thiết bị.
- Khi một ứng dụng yêu cầu một lời gọi hệ thống nhập xuất, thì yêu cầu này được đưa vào hàng đợi của thiết bị tương ứng.
- Bộ lập lịch sắp xếp lại thứ tự của hàng đợi.

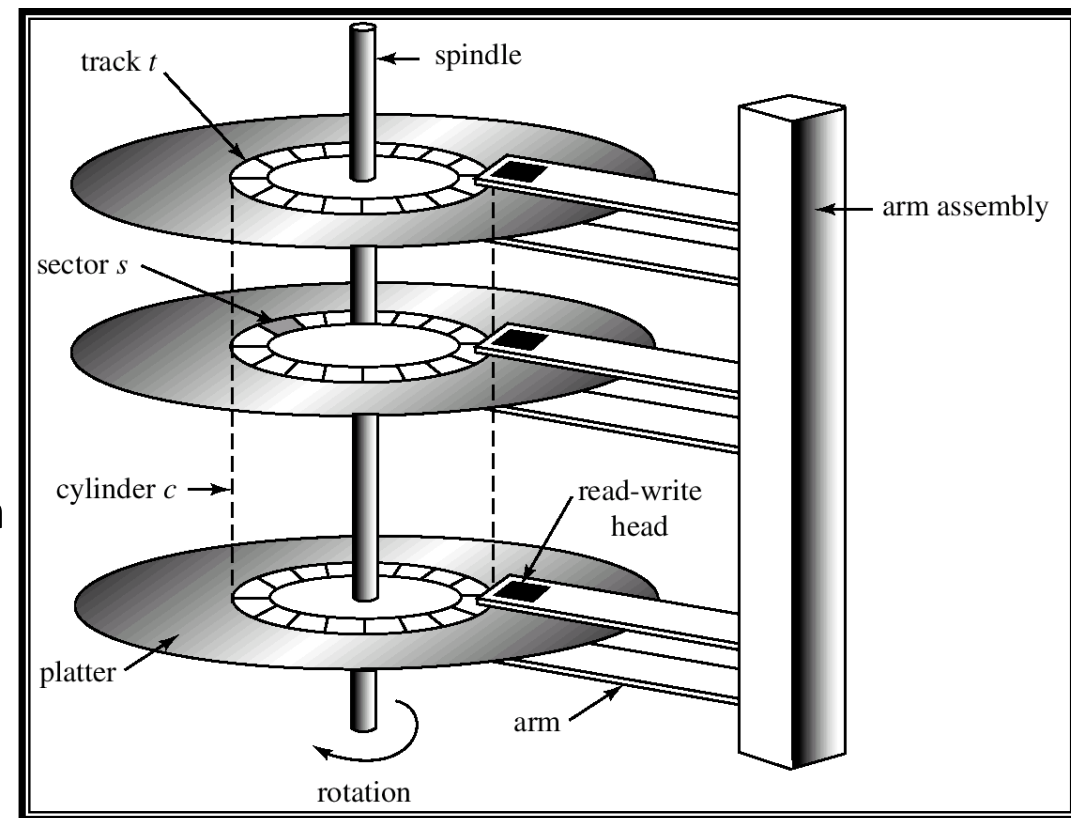
# Bảng trạng thái thiết bị



# Truy xuất đĩa từ

- Seek time: thời gian di chuyển đầu đọc tới track hay cylinder thích hợp
  - Thời gian trung bình trên các đĩa cứng hiện nay là dưới 10 ms
- Rotational delay (rotational latency): thời gian quay đĩa tới đúng sector cần đọc
  - Thời gian rotational delay trung bình là  $t_r/2$ 
    - $t_r$  là thời gian quay 1 vòng
  - Thời gian quay của các đĩa hiện nay từ 3600 rpm đến 15000 rpm (có thời gian rotational delay trung bình là 2 ms)
- Transfer time =  $b * t_r / N$ 
  - b: số byte cần đọc
  - N: tổng số byte trên 1 track
- Access time = Seek time + Rotational time + Transfer time

$$= T_{seek} + t_r/2 + b * t_r / N$$



## Ví dụ truy xuất đĩa từ

- Xét 1 đĩa cứng có 5 mặt, mỗi mặt có 200 track, mỗi track có 500 sector, seek time trung bình là 4 ms, tốc độ quay của đĩa (rotation speed) là 15000 rpm.
  - Xác định dung lượng của đĩa?
  - Giả sử có 1 tập tin có dung lượng 1.28MB, chiếm trọn 5 track (2500 sector) trên cùng một cylinder. Hãy xác định thời gian truy xuất trung bình tập tin này?
- Giải:
  - Thời gian đọc track đầu tiên:
    - Thời gian seek time trung bình: 4 ms
    - Thời gian rotational delay trung bình: 2 ms
    - Thời gian đọc 1 track (500 sectors): 4 ms
  - Do tập tin nằm trên 1 cylinder nên không tốn thời gian seek time khi đọc các track còn lại. Do đó, tổng thời gian truy xuất tập tin này là:  
$$10 + (4 \times 6) = 34 \text{ ms} = 0.034 \text{ seconds}$$

# Các thuật toán lập lịch truy xuất đĩa

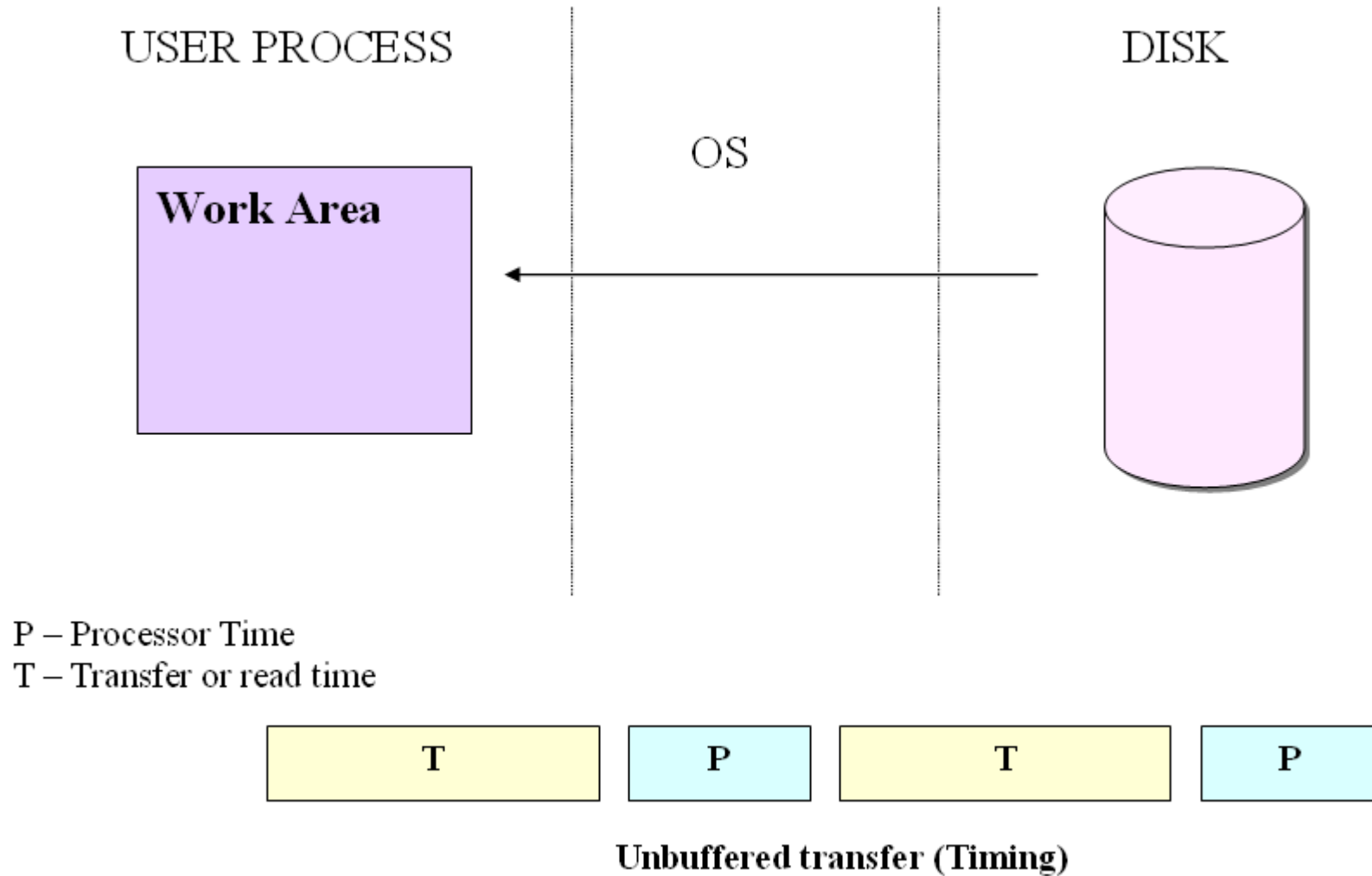
- FCFS (First Come First Serve):
  - Di chuyển đầu đọc theo thứ tự yêu cầu.
  - Không sắp xếp lại hàng đợi công việc.
  - Dễ lập trình nhưng kết quả thường không tốt.
- SSTF (Shortest Seek Time First):
  - Sau khi phục vụ 1 yêu cầu, di chuyển đầu đọc tới vị trí cần đọc gần với vị trí hiện hành của đầu đọc nhất.
  - Giảm seek time so với FCFS.
  - Vấn đề starvation.
- SCAN (elevator algorithms):
  - Đầu đọc sẽ di chuyển về một phía của đĩa và từ đó di chuyển qua phía kia.
  - Vấn đề: mật độ dày các yêu cầu chưa được phục vụ ở đầu còn lại.
- C-SCAN:
  - Tương tự SCAN, chỉ khác là khi di chuyển tới 1 đầu nào đó của đĩa, nó sẽ lập tức trở về đầu bắt đầu của đĩa.
- LOOK:
  - Tương tự SCAN nhưng đầu đọc chỉ di chuyển tới khối xa nhất ở mỗi hướng chứ không đến cuối.
- C-LOOK.

# Đệm dữ liệu (Buffering)

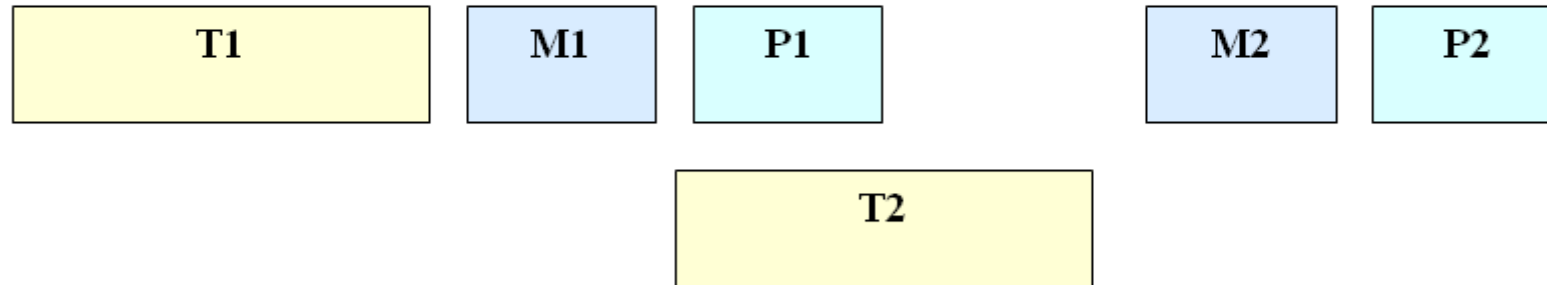
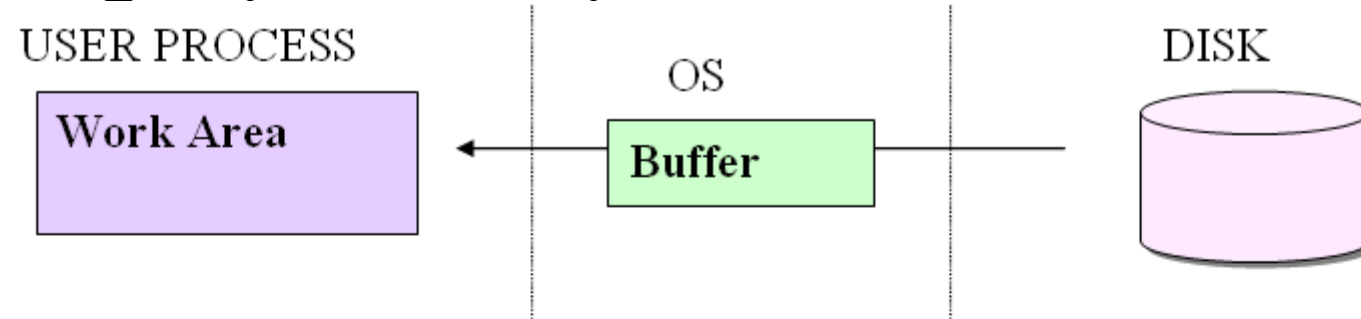
- Lưu tạm dữ liệu trong bộ nhớ khi truyền giữa các thiết bị nhằm:
  - Tốc độ truyền giữa các thiết bị
  - Kích thước dữ liệu truyền giữa các thiết bị



# Không sử dụng vùng đệm dữ liệu



# Sử dụng vùng đệm dữ liệu



Assuming M (Move time) to be less than P, this is an improvement over unbuffered transfer.

# Đệm dữ liệu truy xuất nhanh (Caching)

- Lưu trữ bản sao của dữ liệu tại một vùng nhớ truy xuất nhanh.
- Cần phân biệt cache và buffer.
  - Buffer lưu tạm bản sao của dữ liệu.
  - Cache lưu tạm bản sao của dữ liệu tại nơi có tốc độ truy xuất nhanh hơn nơi lưu dữ liệu.

# Spooling

- Đặc điểm:
  - Lưu trữ tạm dữ liệu đầu ra cho một thiết bị nếu thiết bị chỉ có thể phục vụ một yêu cầu tại một thời điểm, như máy in.
  - Chỉ chấp nhận dòng dữ liệu liên tục.
- Thao tác in:
  - Dữ liệu cần in của mỗi ứng dụng được đưa vào một tập tin riêng.
  - Hệ thống spooling sẽ chuyển lần lượt các tập tin này cho máy in.

# Quản lý lỗi (Error Handling)

- Lỗi thường ở 2 dạng:
  - Tạm thời.
  - Lâu dài.
- I/O Subsystem có thể phục hồi hiệu quả lỗi tạm thời.
- Khi yêu cầu nhập xuất xảy ra lỗi → trả về mã lỗi.

# Vòng đời của một yêu cầu nhập xuất

