



ĐẠI HỌC ĐÀ NẴNG  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG VIỆT - HÀN  
Vietnam - Korea University of Information and Communication Technology

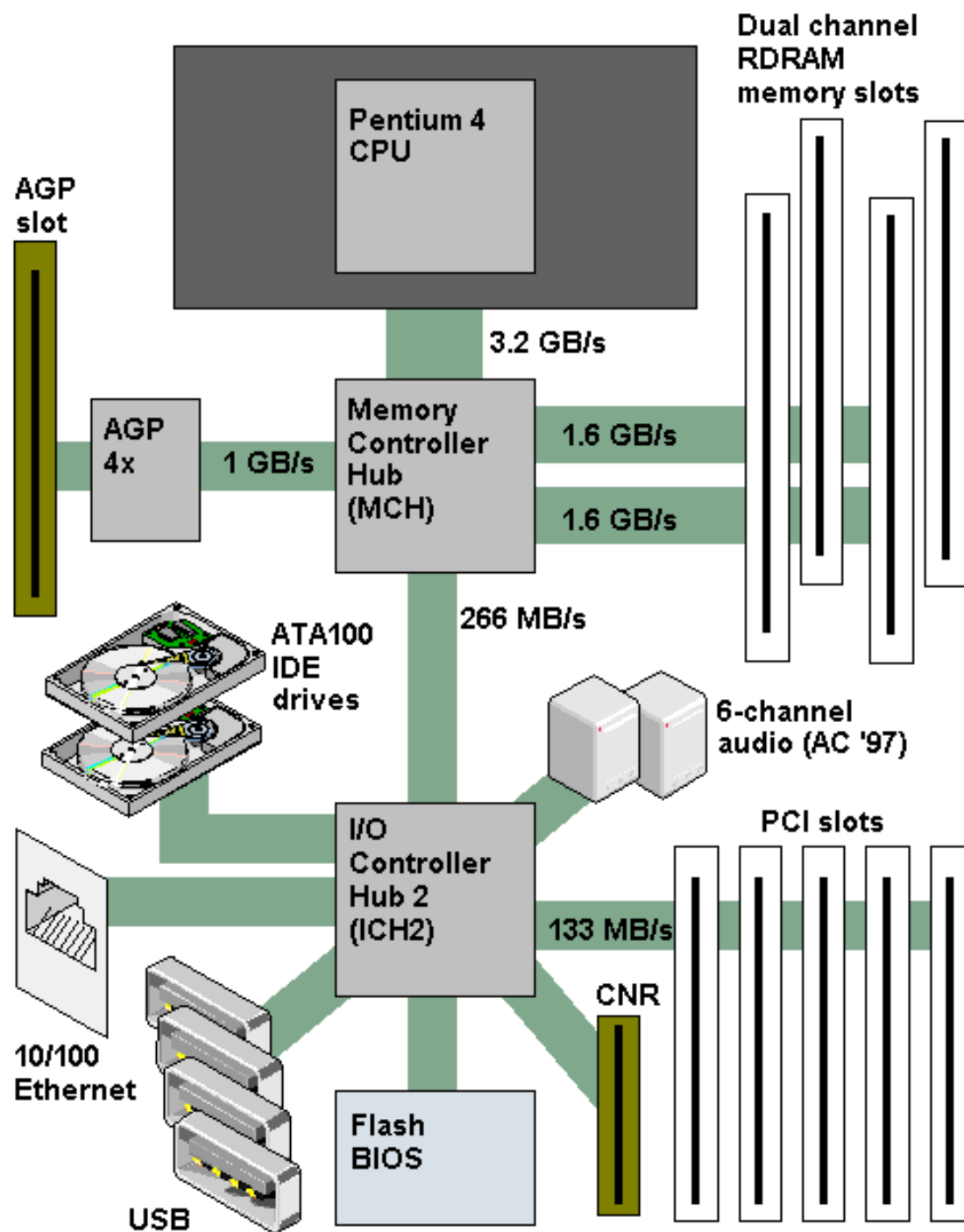
# CHƯƠNG 4

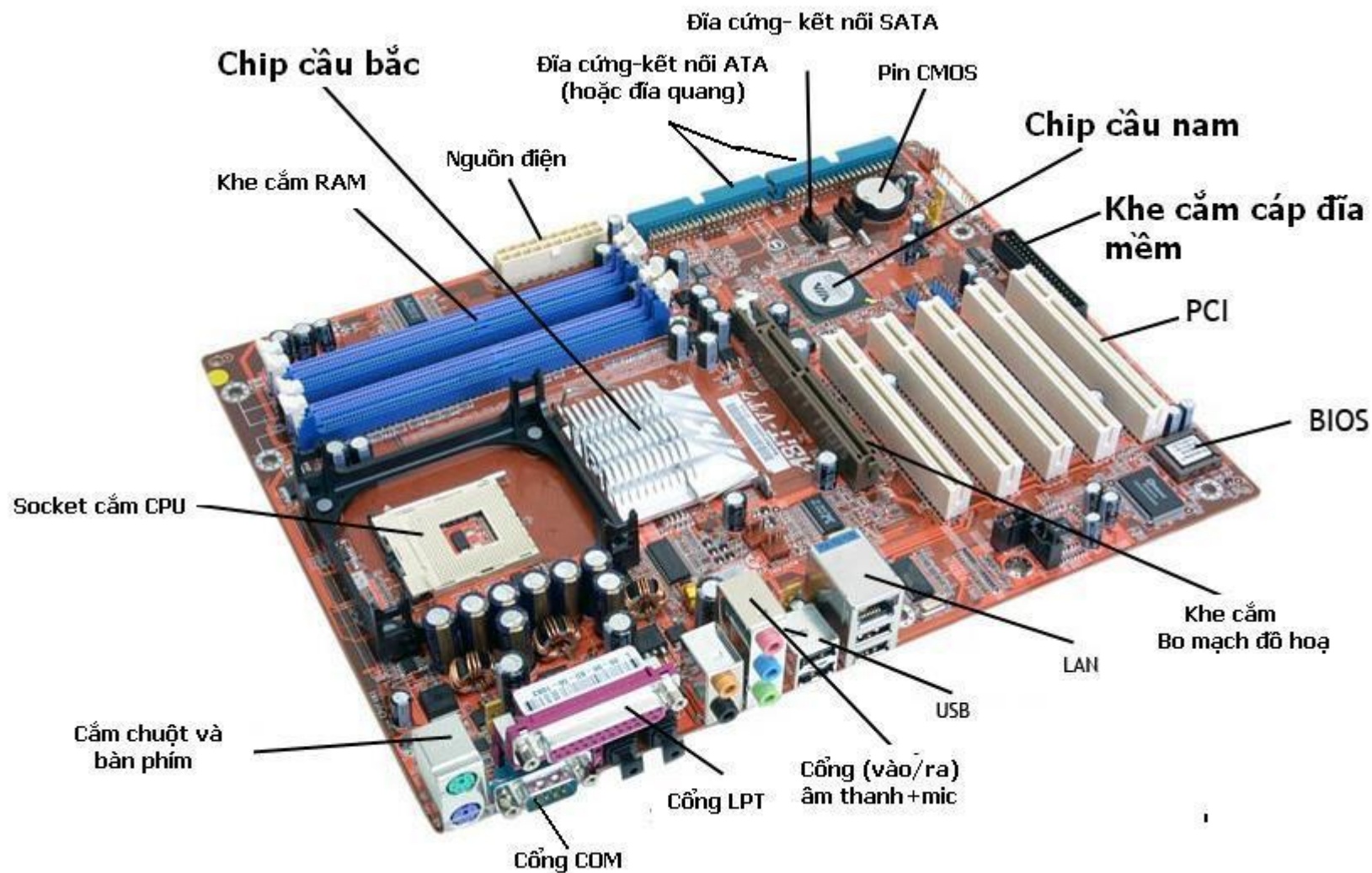
# CÁC THIẾT BỊ VÀO RA



# NỘI DUNG

1. Các thiết bị ngoại vi
2. Các mô đun nhập xuất (I/O)
3. Truy cập bộ nhớ trực tiếp (DMA)
4. Kênh nhập xuất và vi xử lý

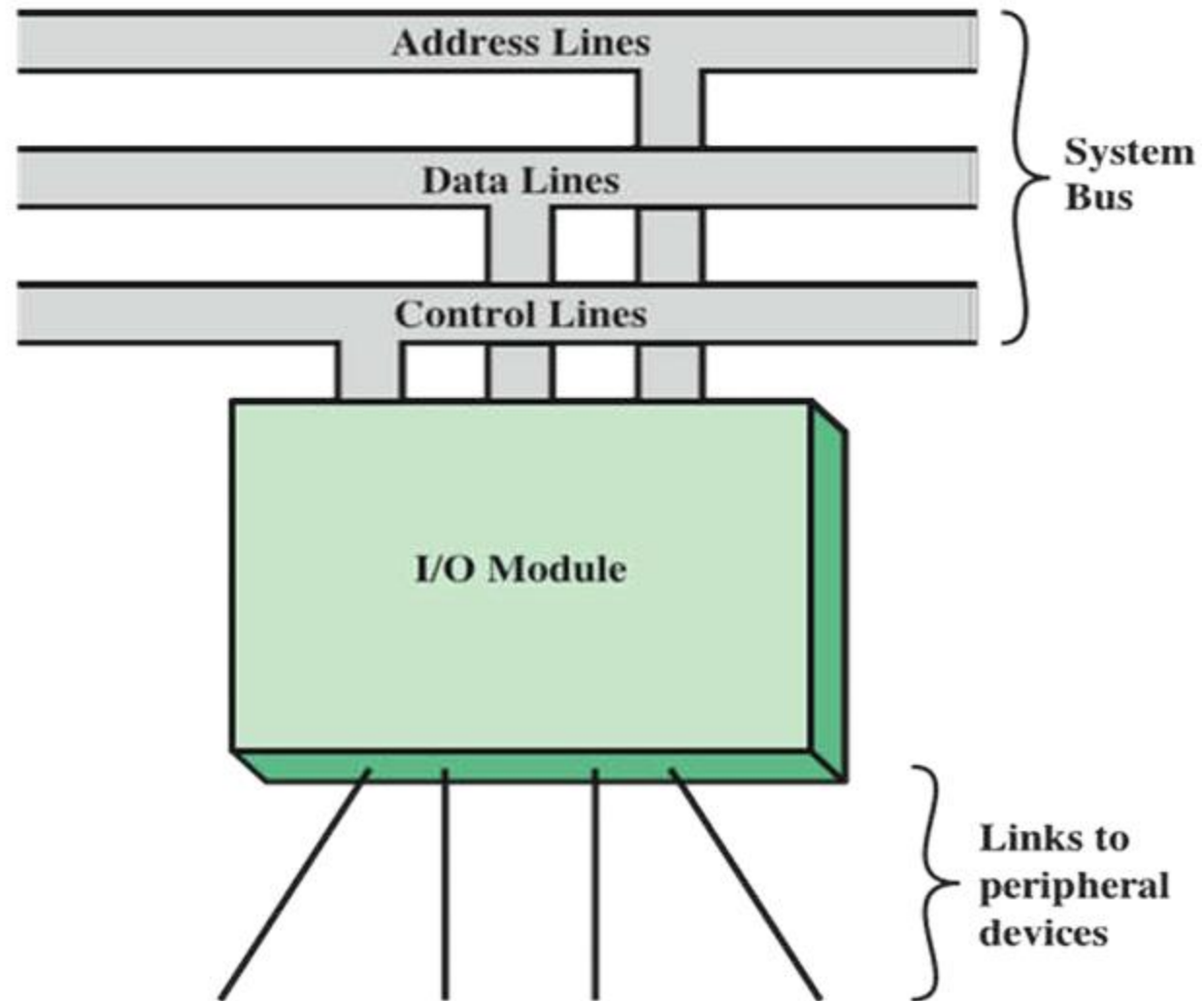






# Tổng quan về hệ thống Vào/ra

- **Chức năng:** Trao đổi thông tin giữa máy tính với thế giới bên ngoài
- **Thành phần chính:**
  - Các thiết bị ngoại vi.
  - Các mô-đun I/O.
- **Tại sao không kết nối thiết bị ngoại vi trực tiếp với hệ thống bus?**
  - Có rất nhiều thiết bị ngoại vi khác nhau.
    - Phương thức làm việc khác nhau.
    - Tốc độ truyền dữ liệu khác nhau.
    - Định dạng dữ liệu khác nhau.
  - Tốc độ truyền của thiết bị ngoại vi phần lớn chậm hơn tốc độ của CPU và RAM
- **Cần các mô-đun I/O**
  - Kết nối các thiết bị ngoại vi vào bus hệ thống.
  - Chứa mạch logic để thực hiện truyền thông giữa thiết bị ngoại vi.



**Generic Model of an I/O Module**



# 1. Các thiết bị ngoại vi

## ✓ Khái niệm

- Một thiết bị bên ngoài kết nối tới mô-đun I/O
- Cung cấp phương tiện trao đổi dữ liệu giữa môi trường bên ngoài và máy tính
- Nối tới máy tính bằng một liên kết đến mô-đun I/O
- Liên kết được sử dụng để trao đổi thông tin điều khiển, trạng thái và dữ liệu giữa mô-đun I/O và thiết bị ngoại vi



# 1. Các thiết bị ngoại vi

## Thiết bị nhập

- Bàn phím
- Thiết bị vào quang
  - Đầu đọc thẻ
  - Đầu đọc mã vạch
  - ...
- Các thiết bị vào từ tính
  - Đầu đọc băng từ, đĩa từ
- Thiết bị vào màn hình
  - Màn hình cảm ứng
  - Chuột

## Thiết bị xuất

- Máy đọc lỗ thẻ
- Màn hình
- Máy in
- Loa

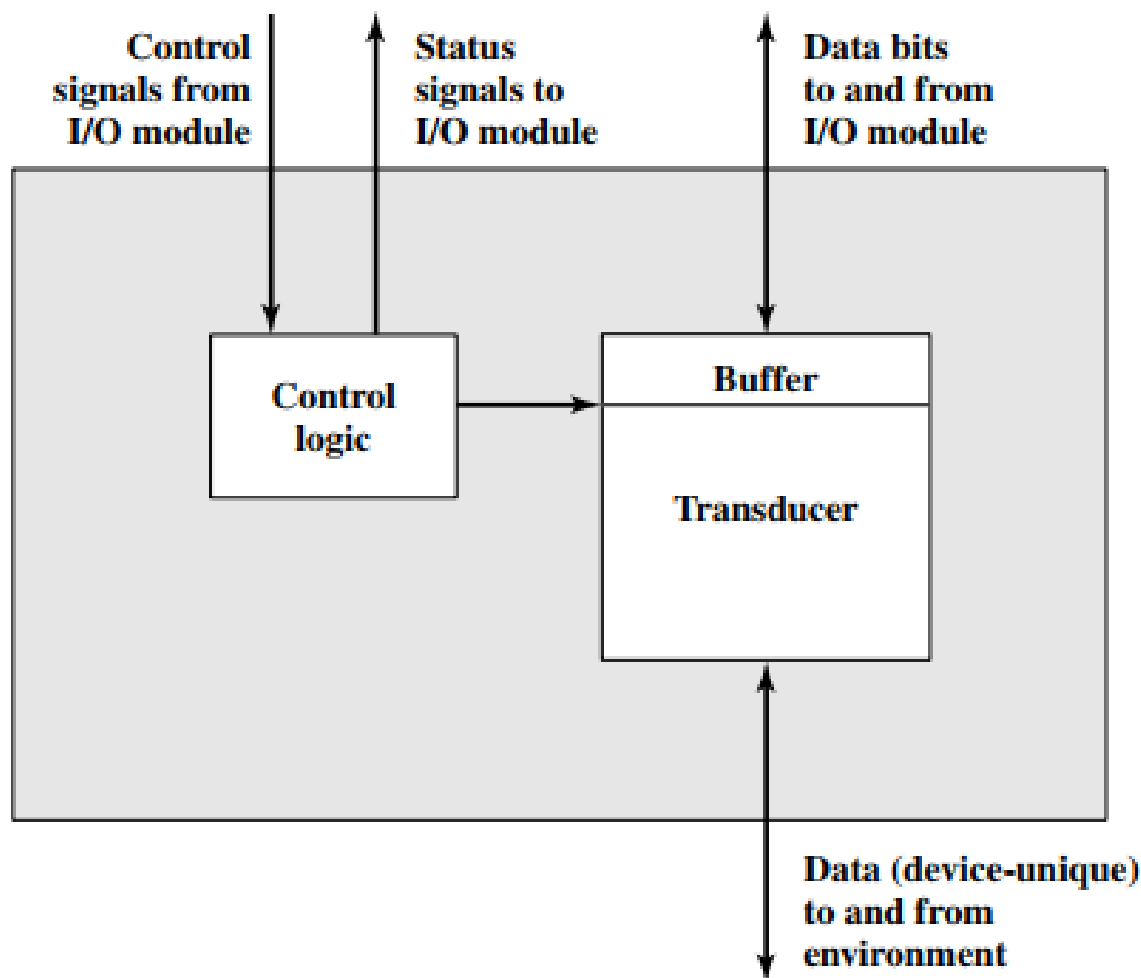


# 1. Thiết bị ngoại vi

Phân thành 3 loại chính:

- Giao tiếp người-máy: Người đọc được.
  - Thích hợp cho người dùng máy tính.
  - Màn hình, bàn phím, máy in,...
- Giao tiếp máy - máy: Máy đọc được.
  - Thích hợp cho giao tiếp với thiết bị.
  - Hệ thống đĩa và băng từ, cảm biến và bộ truyền động.
- Truyền thông.
  - Thích hợp để liên lạc với các thiết bị từ xa.
  - Modem, Card mạng (NIC),...

# Sơ đồ khối thiết bị ngoại vi



- Giao diện mô-đun I/O gồm tín hiệu điều khiển, dữ liệu và trạng thái.
- Bộ chuyển đổi tín hiệu: chuyển đổi dữ liệu giữa dạng điện và dạng năng lượng khác
- Bộ đệm dữ liệu: đệm dữ liệu khi truyền giữa mô-đun I/O và thiết bị ngoại vi.
- Khối logic điều khiển: điều khiển hoạt động của thiết bị ngoại vi đáp ứng theo yêu cầu từ mô-đun I/O.



# 1. Thiết bị ngoại vi

## Ví dụ:

- Khi người dùng ấn một phím trên bàn phím, một tín hiệu điện được tạo ra bởi bộ chuyển đổi trong bàn phím và được dịch sang mẫu bit của mã IRA tương ứng.
- Mẫu bit này được truyền đến mô đun I/O trong máy tính.
- Trên đầu ra, các ký tự mã IRA được truyền đến một thiết bị ngoại vi từ mô đun I/O.
- Bộ biến đổi giải mã và gửi các tín hiệu điện cần thiết đến thiết bị đầu ra để hiển thị ký tự được chỉ định hoặc thực hiện chức năng điều khiển theo yêu cầu.



# 1. Thiết bị ngoại vi

## ✓ Mã IRA

- Đơn vị trao đổi cơ bản là ký tự.
  - + Mỗi ký tự được gán cho một mã.
  - + Mã nhị phân 7-bit, có thể biểu diễn 128 ký tự.
- Hai loại ký tự:
  - + In được: Các ký tự chữ cái, số và ký tự đặc biệt có thể được in trên giấy hoặc hiển thị trên màn hình
  - + Điều khiển
    - Điều khiển việc hiển thị các ký tự.
    - Các ký tự điều khiển khác liên quan đến các thủ tục truyền tin.



## 2. Module I/O

### ✓ Chức năng của Mô đun I/O:

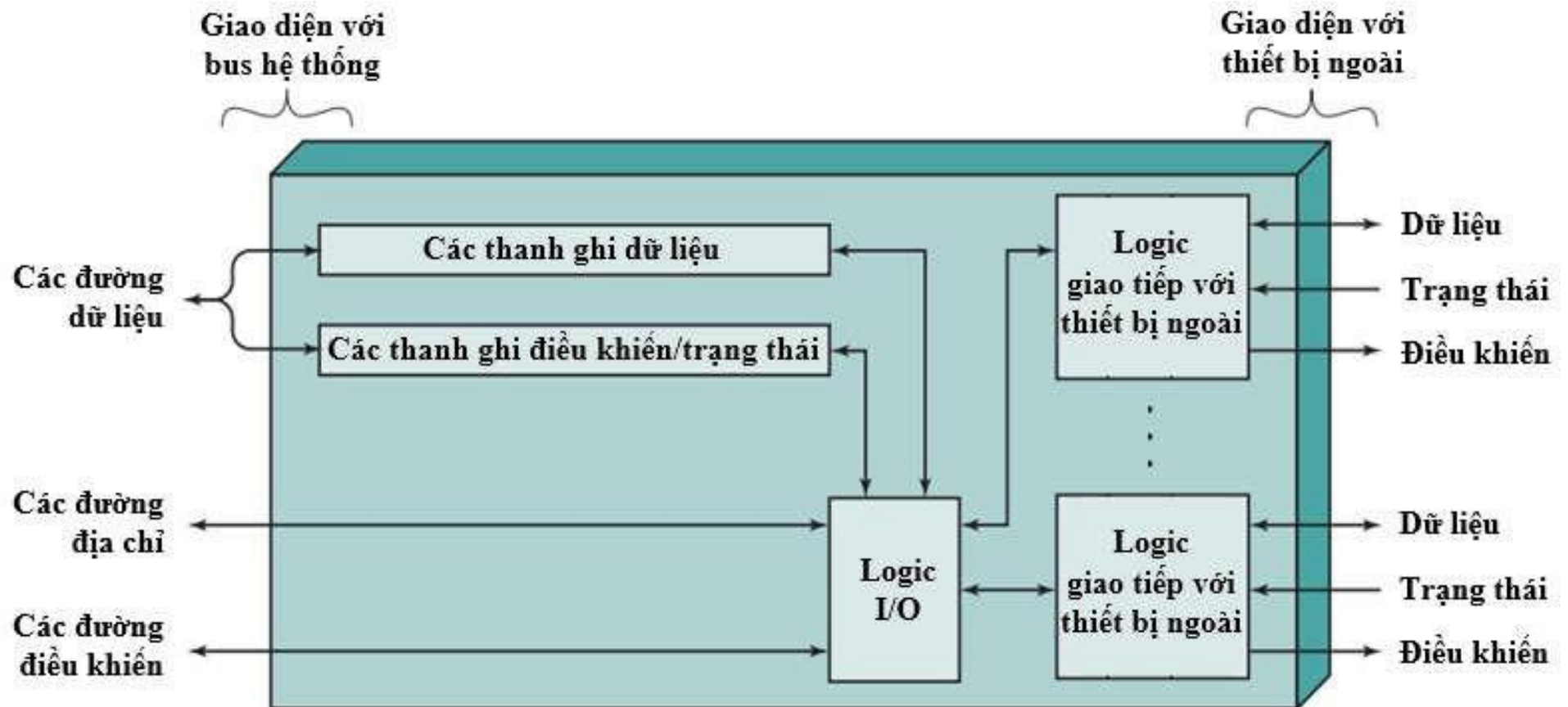
- Điều khiển và định thời: Phối hợp luồng lưu lượng truy cập giữa tài nguyên bên trong và thiết bị ngoại vi.
- Trao đổi thông tin với bộ xử lý: Gồm giải mã lệnh, dữ liệu, báo cáo trạng thái, nhận dạng địa chỉ.
- Trao đổi thông tin với thiết bị: Gồm các lệnh, thông tin trạng thái và dữ liệu.
- Đệm dữ liệu: Thực hiện các hoạt động đệm cần thiết để cân bằng tốc độ thiết bị và bộ nhớ.
- Phát hiện lỗi: Phát hiện và báo cáo lỗi.



## Các bước truyền dữ liệu I/O

- CPU hỏi mô đun I/O để kiểm tra trạng thái thiết bị I/O được gắn vào.
- Mô đun I/O trả về trạng thái thiết bị.
- Nếu thiết bị đã sẵn sàng, CPU yêu cầu truyền dữ liệu bằng cách gửi một lệnh tới mô đun I / O.
- Mô đun I/O nhận dữ liệu từ thiết bị.
- Mô đun I/O truyền dữ liệu đến CPU.

# Sơ đồ khối mô đun I/O





# Các thành phần của mô-đun I/O

- Thanh ghi đệm dữ liệu: đệm dữ liệu trong quá trình trao đổi.
- Các cổng vào-ra (I/O Port): kết nối với thiết bị ngoại vi, mỗi cổng có một địa chỉ xác định.
- Thanh ghi trạng thái/điều khiển: lưu giữ thông tin trạng thái/điều khiển cho các cổng vào-ra.
- Khối logic điều khiển: điều khiển mô-đun vào-ra.



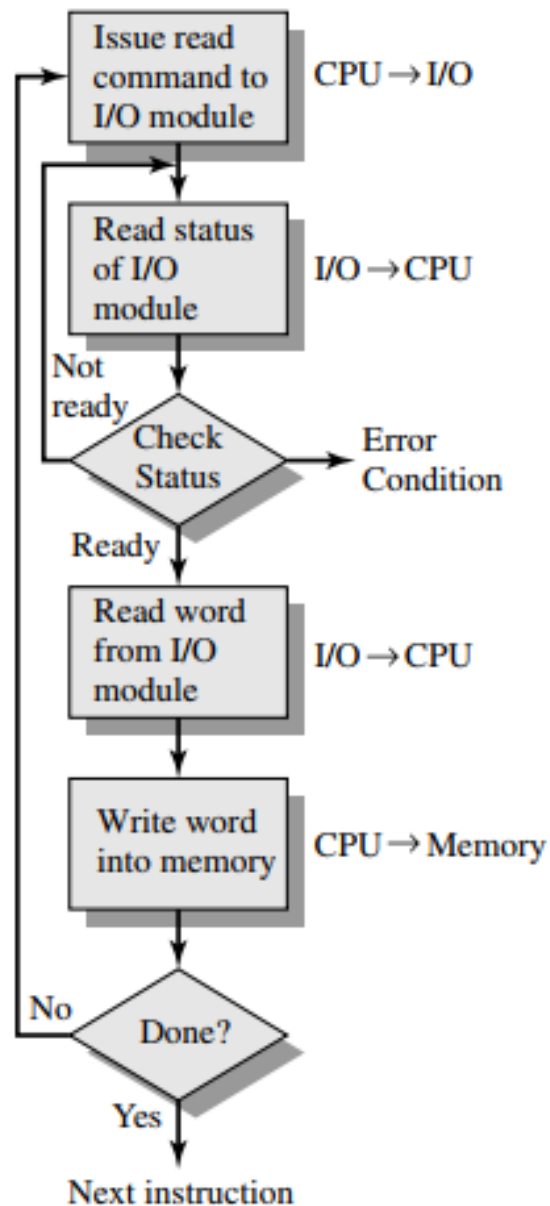
# Các kỹ thuật I/O

## ✓ Có 3 kỹ thuật chính

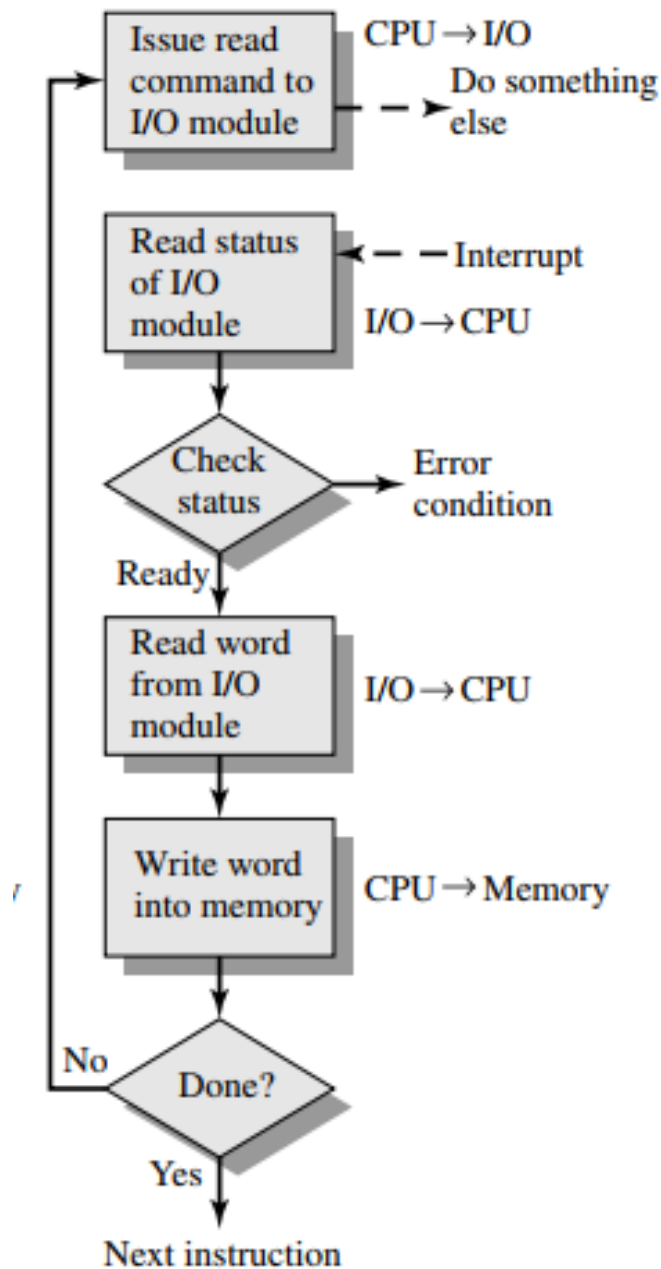
- I/O bằng chương trình (Programmed IO)
- I/O điều khiển bằng ngắt (Interrupt Driven IO)
- Truy cập bộ nhớ trực tiếp (DMA)

	Không có ngắt	Sử dụng ngắt
Truyền dữ liệu giữa bộ nhớ và thiết bị ngoại vi thông qua CPU	I/O chương trình	I/O điều khiển ngắt
Truyền dữ liệu trực tiếp giữa bộ nhớ và thiết bị ngoại vi		Truy cập bộ nhớ trực tiếp (DMA)

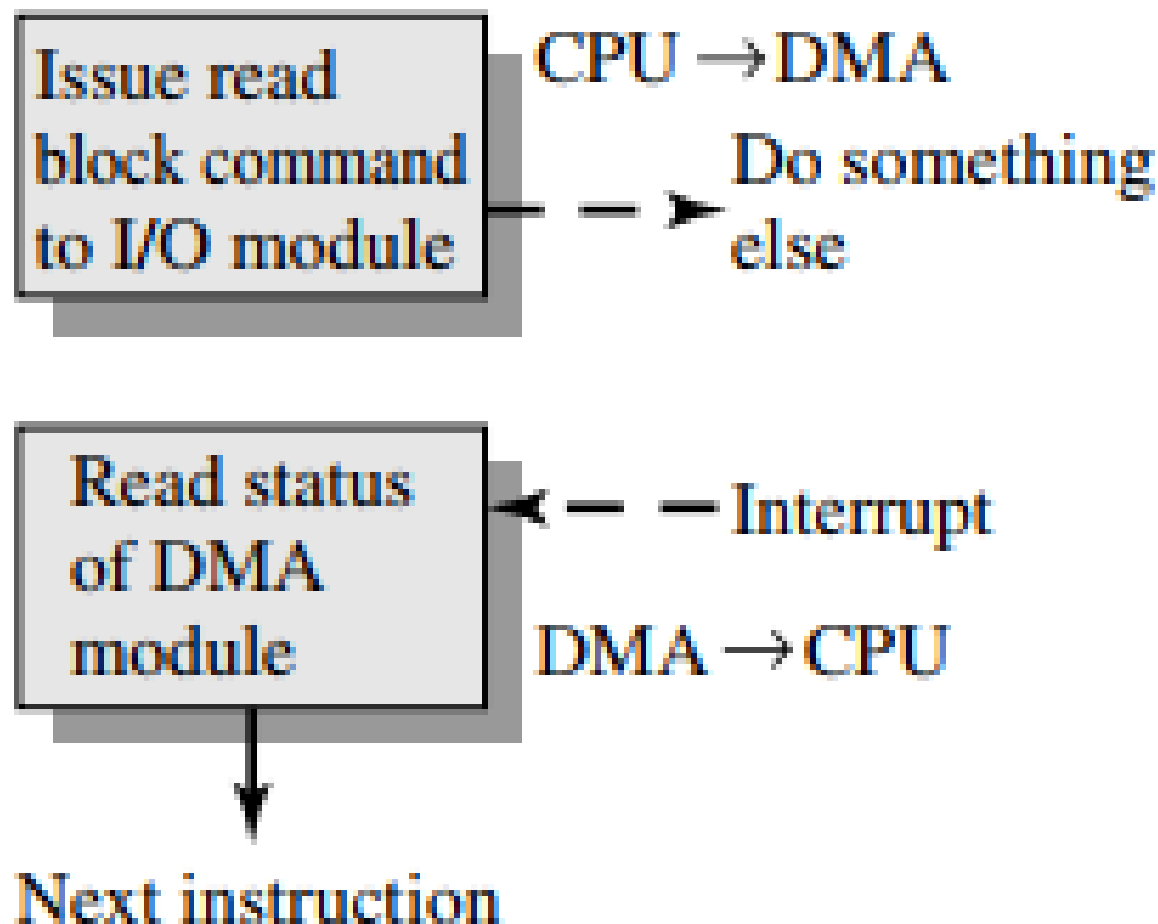
# I/O bằng chương trình



# I/O điều khiển bằng ngắt



# I/O bằng truy cập bộ nhớ trực tiếp (DMA)





## I/O bằng chương trình

- **Nguyên tắc chung:** CPU điều khiển trực tiếp hoạt động vào-ra bằng chương trình (cần phải lập trình vào-ra).
- CPU thi hành một chương trình cho phép nó trực tiếp điều hành các hoạt động I/O
- Khi **CPU** ra lệnh cho mô đun I/O, nó phải **đợi** cho đến khi hoạt động I/O hoàn tất
- CPU hoạt động nhanh hơn mô đun I/O sẽ gây **lãng phí** thời gian xử lý



# Lệnh I/O (Command)

- Để thực hiện 1 lệnh liên quan đến vào/ra, CPU đưa ra
  - + Địa chỉ: xác định mô đun I/O và thiết bị.
  - + Mệnh lệnh: 4 loại mệnh lệnh I/O.



## Lệnh I/O (Command)

- **Control** - Dùng để kích hoạt một thiết bị ngoại vi và chỉ định nó phải làm gì. (VD: quay đĩa).
- **Test** - kiểm tra các điều kiện trạng thái liên quan đến mô-đun I/O và các thiết bị ngoại vi (VD: nguồn? Lỗi?).
- **Read** - Cho phép mô đun I/O lấy dữ liệu từ thiết bị ngoại vi và đặt nó vào bộ đệm bên trong.
- **Write** - Cho phép mô đun I/O lấy dữ liệu từ bus dữ liệu rồi chuyển dữ liệu đó đến thiết bị ngoại vi.



## Hoạt động I/O bằng chương trình

- CPU yêu cầu hoạt động vào/ra: ra mệnh lệnh cho mô đun I/O mong muốn.
- Mô đun I/O thực hiện yêu cầu.
- Mô đun I/O đặt bit trạng thái.
- Mô đun I/O không thông báo trực tiếp cho CPU, không ngắt CPU.
- CPU kiểm tra định kỳ bit trạng thái.
  - + Nếu chưa sẵn sàng thì CPU có thể đợi hoặc quay lại sau
  - + Nếu sẵn sàng thì chuyển sang trao đổi dữ liệu với mô đun vào-ra



# Chỉ lệnh I/O

- Phân biệt:
  - + CPU truy xuất **chỉ lệnh (instruction)** từ bộ nhớ.
  - + CPU đưa ra **mệnh lệnh (command)** cho một mô-đun I/O để thi hành chỉ lệnh trên.

# Địa chỉ hoá cổng vào-ra

Không gian địa chỉ bộ nhớ	N bit	Không gian địa chỉ vào-ra	$N_1$ bit
	000...000		00...00
	000...001		00...01
	000...010		00...10
	000...011		00...11
	000...100		
	000...101		
.		.	
.		.	
.		.	
		.	
		.	
		.	
.			
.			
.			
			11...11
	111...111		

Không gian địa chỉ của bộ xử lý



# Các phương pháp ánh xạ địa chỉ vào-ra

## ■ I/O ánh xạ bộ nhớ

- + Thiết bị ngoại vi và bộ nhớ chia sẻ một không gian địa chỉ chung.
- + Cổng vào-ra đánh địa chỉ theo không gian địa chỉ bộ nhớ.
- + Không có mệnh lệnh riêng cho I/O.

## ■ CPU dùng chung lệnh để truy cập cả bộ nhớ và thiết bị I/O.



# Các phương pháp ánh xạ địa chỉ vào-ra

## ■ I/O riêng biệt

- + Không gian địa chỉ riêng biệt.
- + Cổng vào-ra được đánh địa chỉ theo không gian địa chỉ vào-ra.
- + Có mệnh lệnh riêng cho I/O.



## I/O điều khiển bằng ngắt

- Với I/O chương trình, **CPU phải đợi** một thời gian dài để mô đun I/O sẵn sàng cho việc nhận hoặc truyền dữ liệu.
- Giải pháp: I/O điều khiển bằng ngắt
  - + CPU ra lệnh I/O cho mô đun, sau đó thực hiện các công việc hữu ích khác, **không phải đợi**.
  - + Khi nào mô đun I/O đã sẵn sàng để trao đổi dữ liệu nó sẽ ngắt bộ xử lý để yêu cầu phục vụ. Bộ xử lý thực thi việc truyền dữ liệu rồi quay trở lại công việc của nó.



# Hoạt động vào dữ liệu:

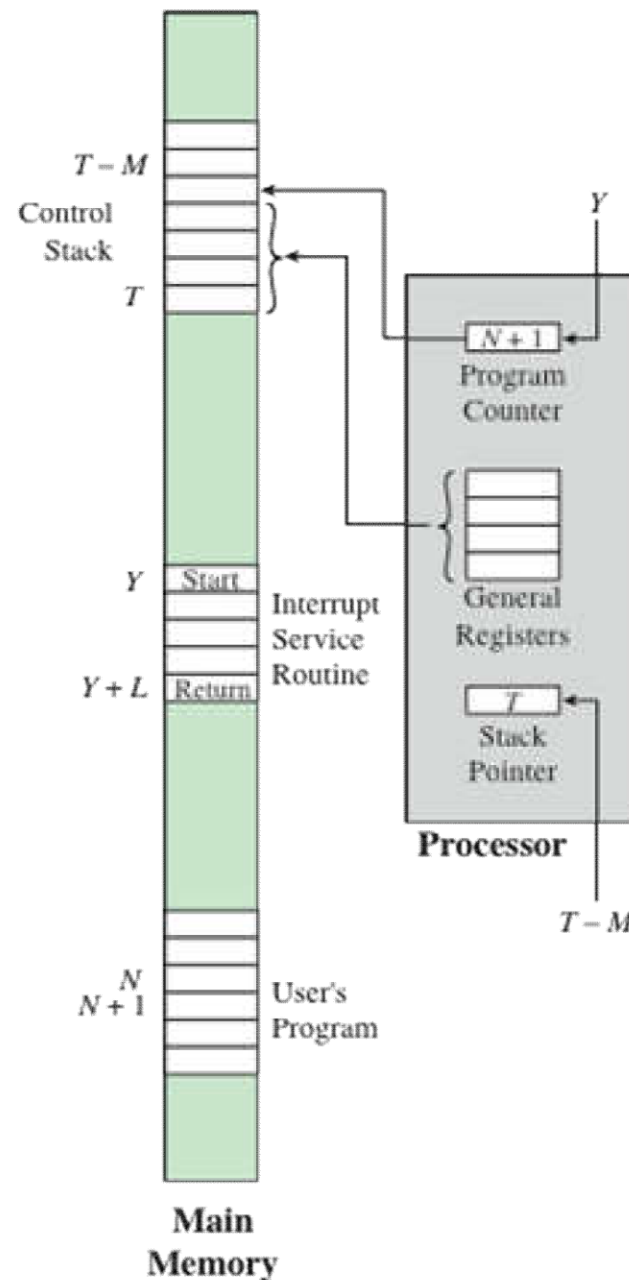
## Nhìn từ mô đun I/O

- Mô đun I/O nhận mệnh lệnh READ từ CPU
- Mô đun I/O nhận dữ liệu từ thiết bị ngoại vi, trong khi đó CPU làm việc khác
- Khi đã có dữ liệu mô đun I/O phát tín hiệu ngắt CPU
- CPU yêu cầu dữ liệu
- Mô đun I/O chuyển dữ liệu đến CPU

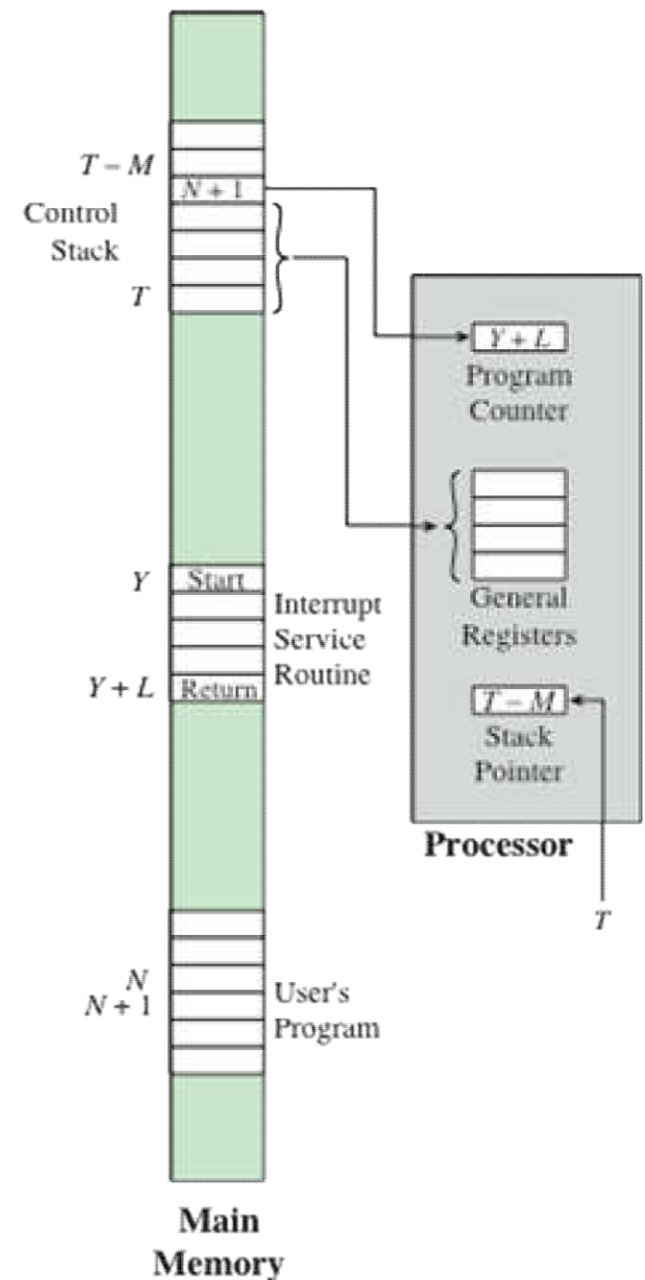
## Nhìn từ CPU

- Phát mệnh lệnh READ
- Làm việc khác
- Cuối mỗi chu kỳ lệnh, kiểm tra tín hiệu ngắt
- Nếu bị ngắt :
  - Lưu ngữ cảnh (nội dung các thanh ghi)
  - Thực hiện chương trình con ngắt
  - Khôi phục ngữ cảnh của chương trình đang thực hiện

Thay đổi  
trong bộ  
nhớ và  
thanh ghi  
khi có một  
ngắt



(a) Interrupt occurs after instruction at location  $N$



(b) Return from interrupt



# Vấn đề nảy sinh khi thiết kế

*Hai vấn đề thiết kế phát sinh khi ngắt I/O:*

- + Do có nhiều mô-đun I/O, làm thế nào để **xác định được mô-đun I/O nào** phát tín hiệu ngắt ?
- + Nếu cùng lúc xảy ra nhiều ngắt, bộ vi xử lý **lựa chọn 1 ngắt để xử lý** như thế nào?

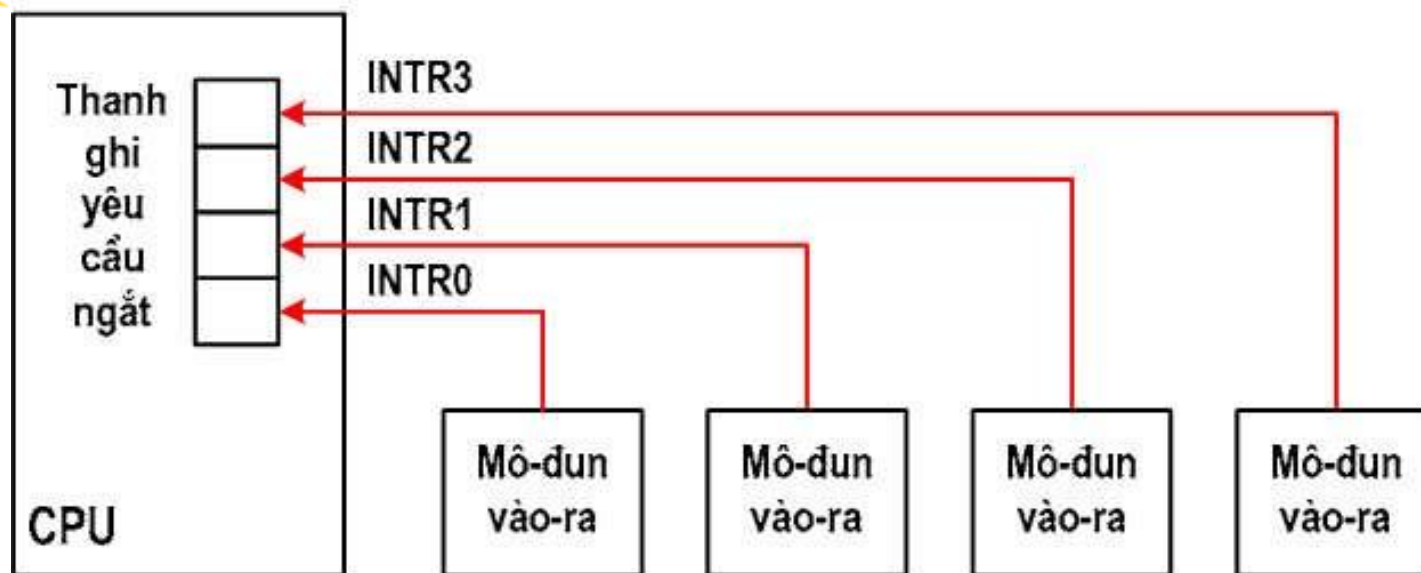


# Nhận diện thiết bị

Bốn loại kỹ thuật chung được sử dụng phổ biến:

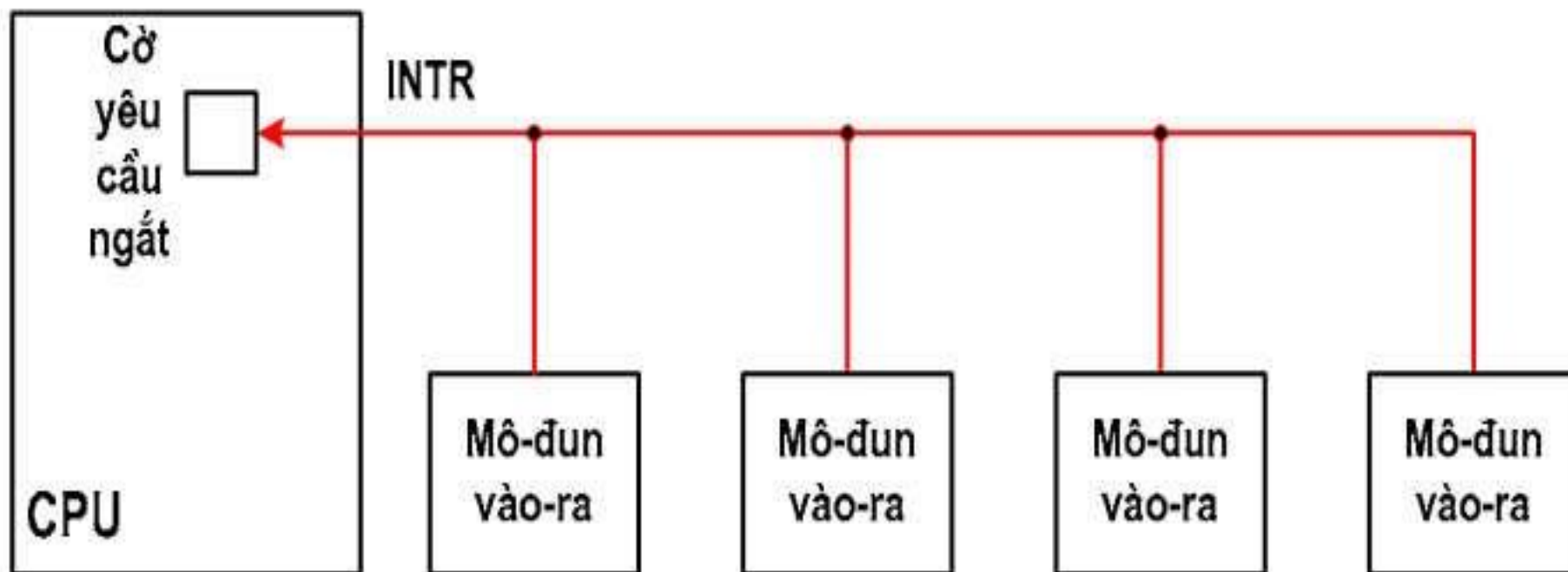
- Dùng nhiều đường ngắt.
- Thăm dò bằng phần mềm.
- Chuỗi Daisy (thăm dò bằng phần cứng).
- Phân xử bus.

# Nhiều đường ngắt



- Nhiều đường ngắt giữa bộ xử lý và mô-đun I/O.
- Cách tiếp cận đơn giản nhất.
- Hạn chế số lượng mô-đun I/O.
- Các đường ngắt được quy định mức ưu tiên.

# Thăm dò bằng phần mềm

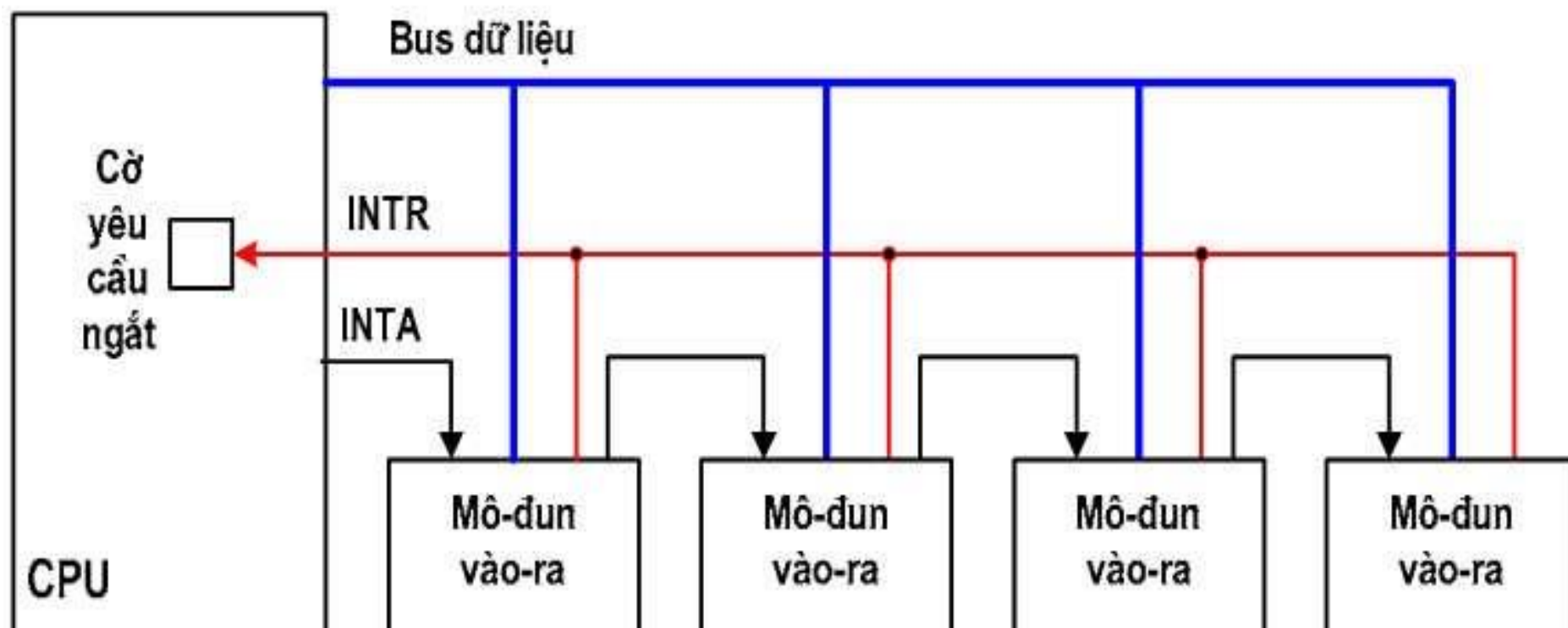




# Thăm dò bằng phần mềm

- Khi bộ xử lý phát hiện ra một ngắt, nó rẽ nhánh tới một trình dịch vụ ngắt (phần mềm) chuyên thăm dò từng mô đun I/O để xác định mô đun nào gây ra ngắt.
- **Tốn thời gian:** Thứ tự thăm dò mô đun phụ thuộc thứ tự ưu tiên.

# Thăm dò bằng phần cứng (chuỗi Daisy)





# Thăm dò bằng phần cứng (chuỗi Daisy)

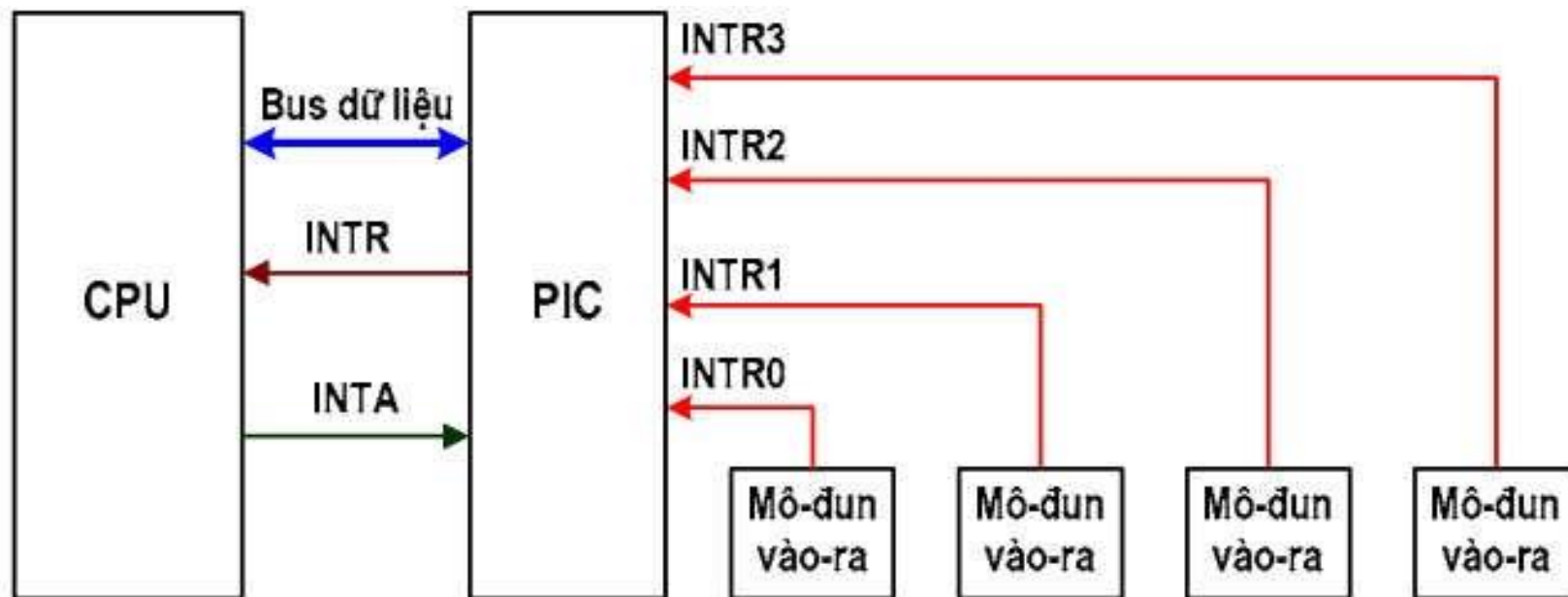
- Dòng thông báo ngắt (INTA) được nối chuỗi qua các mô đun.
- CPU phát tín hiệu INTA đến mô-đun vào-ra đầu tiên.
- Nếu mô đun vào - ra đó không gây ra ngắt thì nó gửi tín hiệu đến mô đun tiếp cho đến khi xác định được mô đun gây ngắt.
- Thứ tự các mô đun vào – ra kết nối trong chuỗi xác định thứ tự ưu tiên.



## Phân xử bus (vector)

- Vector - địa chỉ của mô đun I/O hoặc mã nhận dạng khác của mô đun I/O.
- Ngắt vector - bộ xử lý sử dụng vector như một con trỏ tới trình phục vụ ngắt thích hợp, tránh phải thực hiện trình phục vụ ngắt chung.
- Mô đun I/O phải kiểm soát được của bus trước khi nó bật dòng yêu cầu ngắt.
- Khi bộ xử lý phát hiện ra ngắt, nó phản ứng trên dòng thông báo ngắt.
- Mô đun yêu cầu sẽ đặt các vector của nó trên các đường dữ liệu.

# Bộ điều khiển ngắt lập trình được



- PIC –Programmable Interrupt Controller.
- Có nhiều đường vào yêu cầu ngắt có qui định mức ưu tiên.
- PIC chọn một yêu cầu ngắt không bị cấm có mức ưu tiên cao nhất gửi tới CPU.



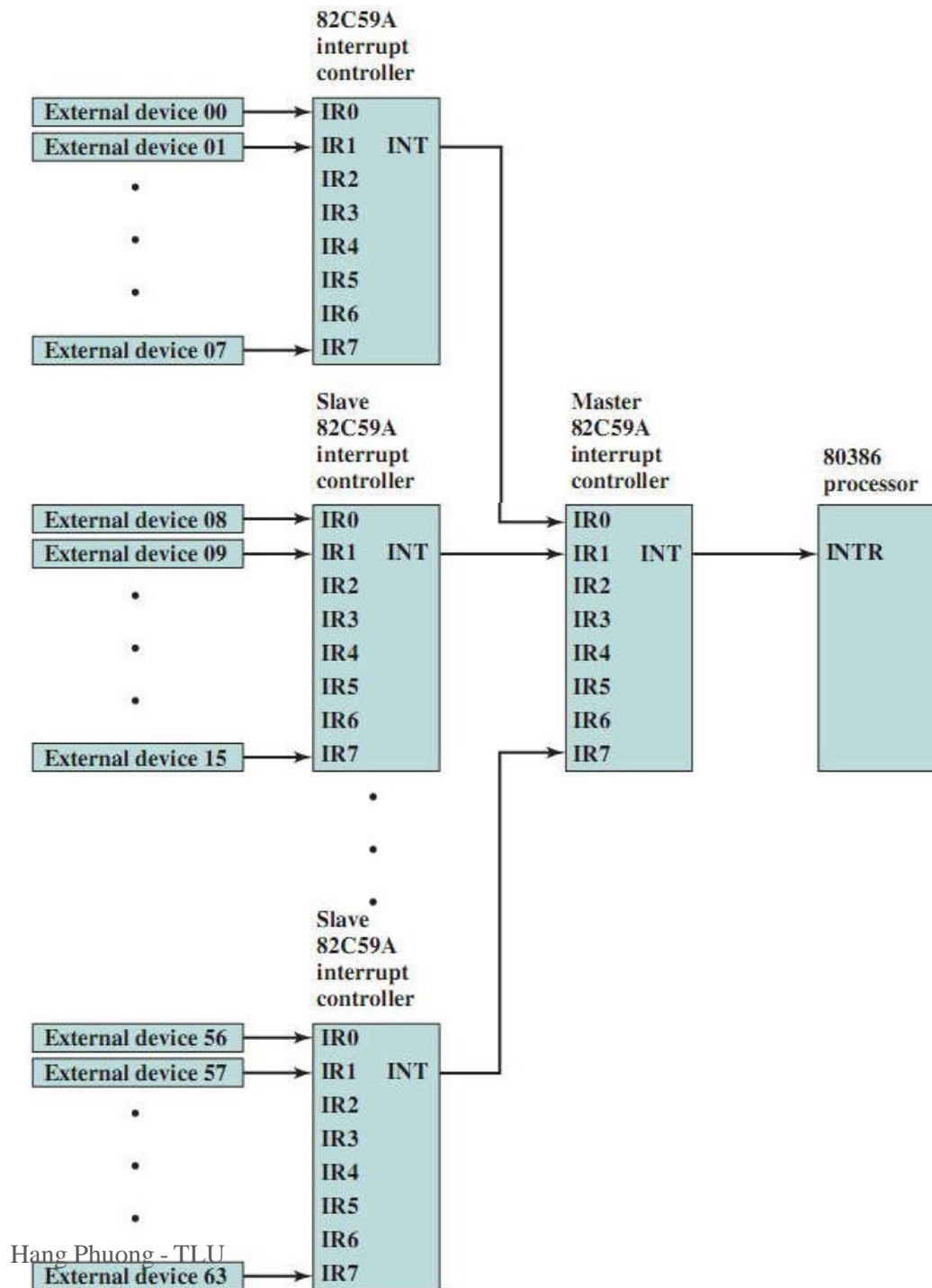
## Ví dụ - Bộ điều khiển ngắt Intel 82C59A

- 80x86 có một đường ngắt.
- Hệ thống 8086 sử dụng bộ điều khiển ngắt 8259A.
- 8259A có 8 đường ngắt.
- 82C59A lập trình được, được sử dụng để điều khiển giao tiếp với một số thiết bị ngoại vi.

# Bộ điều khiển ngắt Intel 82C59A

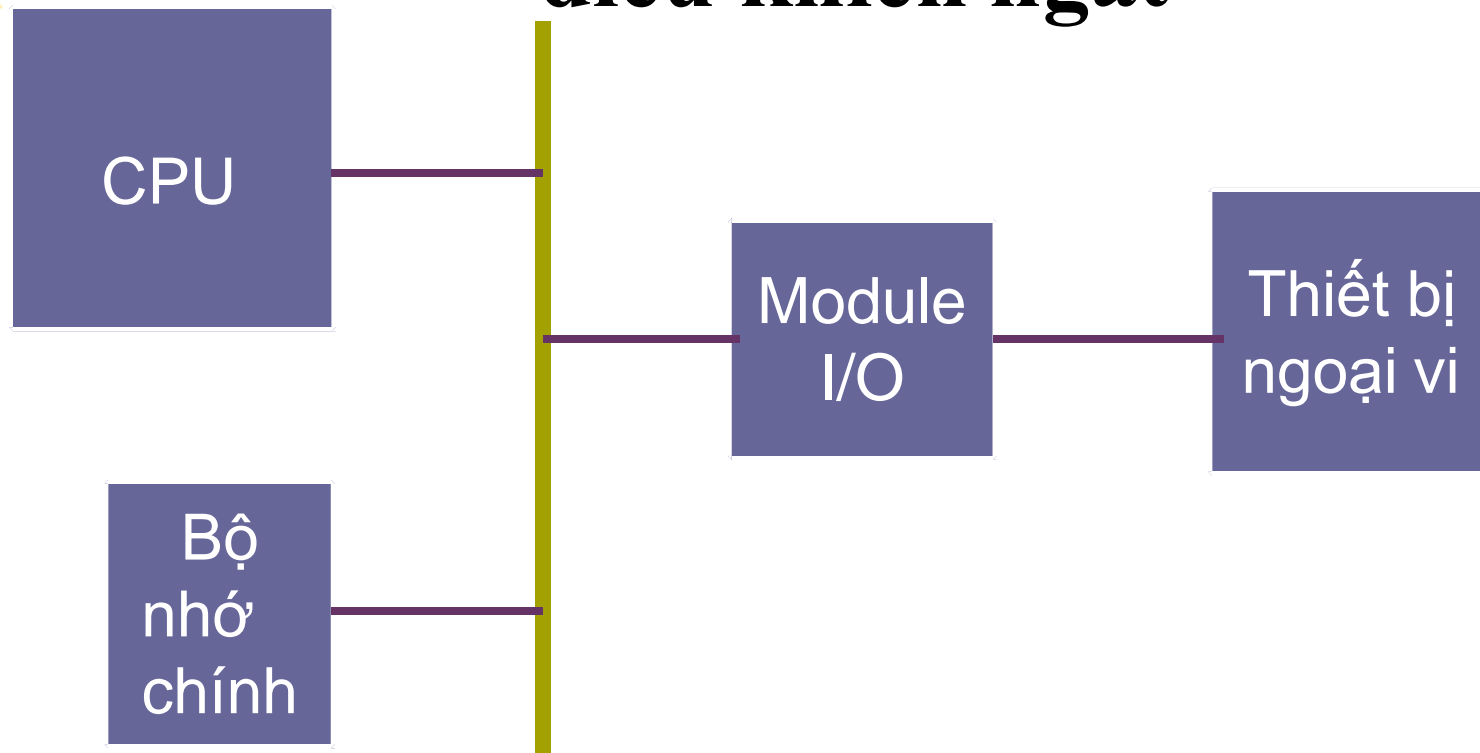
Chuỗi sự kiện:

- 8259A nhận ngắt
- 8259A xác định ưu tiên
- 8259A báo hiệu 8086 (nâng INTR)
- CPU phản hồi
- 8259A đặt véc tơ trên bus dữ liệu
- CPU xử lý ngắt



Hang Phuong - TLU

# Nhược điểm của I/O chương trình và điều khiển ngắt



- + Tốc độ truyền I/O bị giới hạn bởi tốc độ kiểm tra và phục vụ thiết bị của bộ xử lý.
- + Bộ xử lý gắn với việc quản lý truyền I/O. Một số lệnh phải được thực hiện cho mỗi lần truyền I/O.



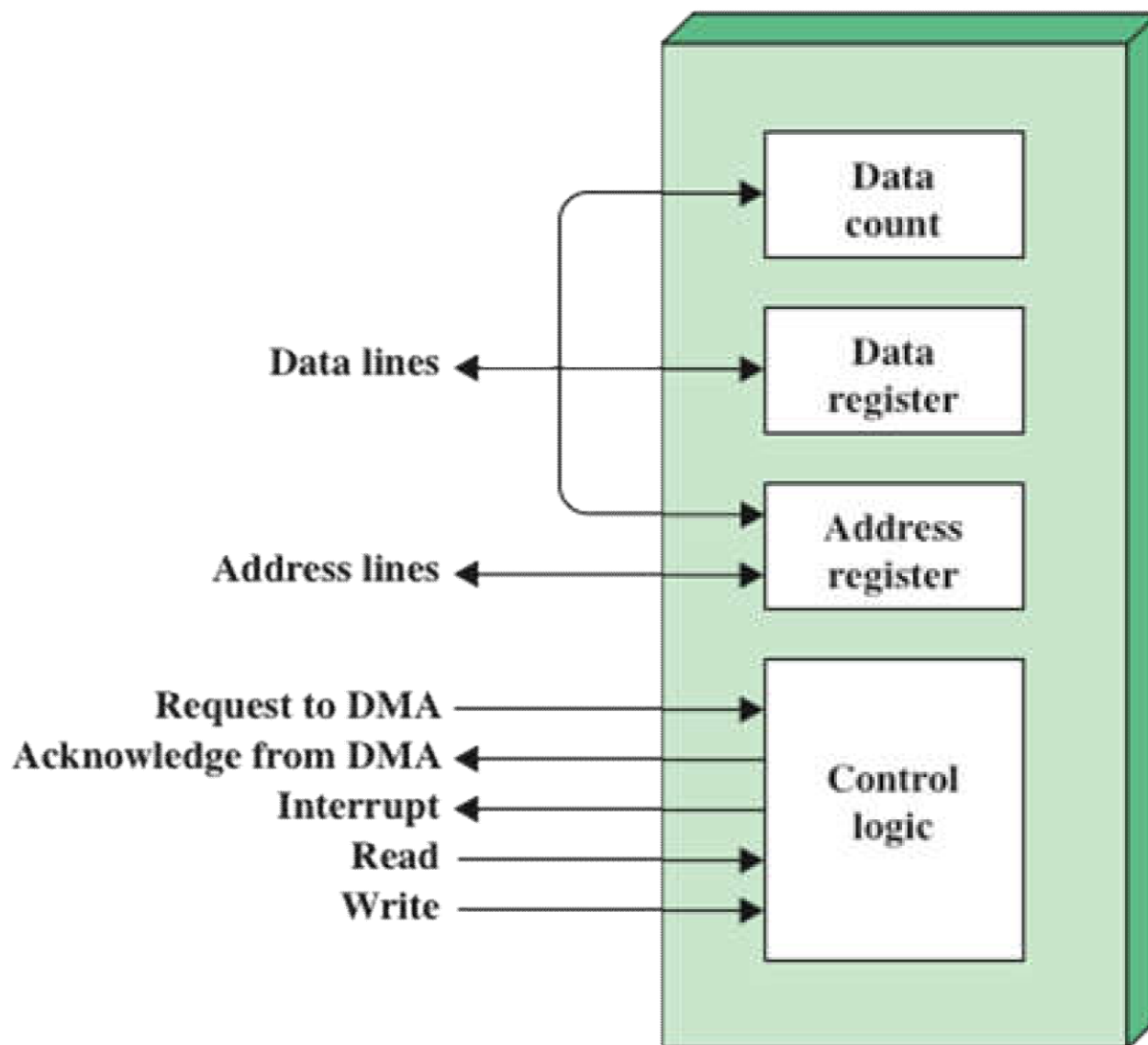
### 3. Truy cập bộ nhớ trực tiếp (DMA)

*DMA là giải pháp hiệu quả hơn khi cần di chuyển lượng dữ liệu lớn*

- **DMA:** Mô đun I/O và bộ nhớ chính trực tiếp trao đổi dữ liệu mà không có sự tham gia của CPU.
- **Chức năng DMA:**
  - + Là 1 mô đun bổ sung trên hệ thống bus.
  - + Bộ điều khiển DMA (DMAC) chiếm quyền điều khiển I/O của CPU.

# Sơ đồ cấu trúc mô-đun DMA

- Thanh ghi dữ liệu: chứa dữ liệu trao đổi.
- Thanh ghi địa chỉ: chứa địa chỉ ngăn nhớ dữ liệu.
- Bộ đếm dữ liệu: chứa số lượng từ dữ liệu cần trao đổi.
- Logic điều khiển: điều khiển hoạt động.





# Hoạt động DMA

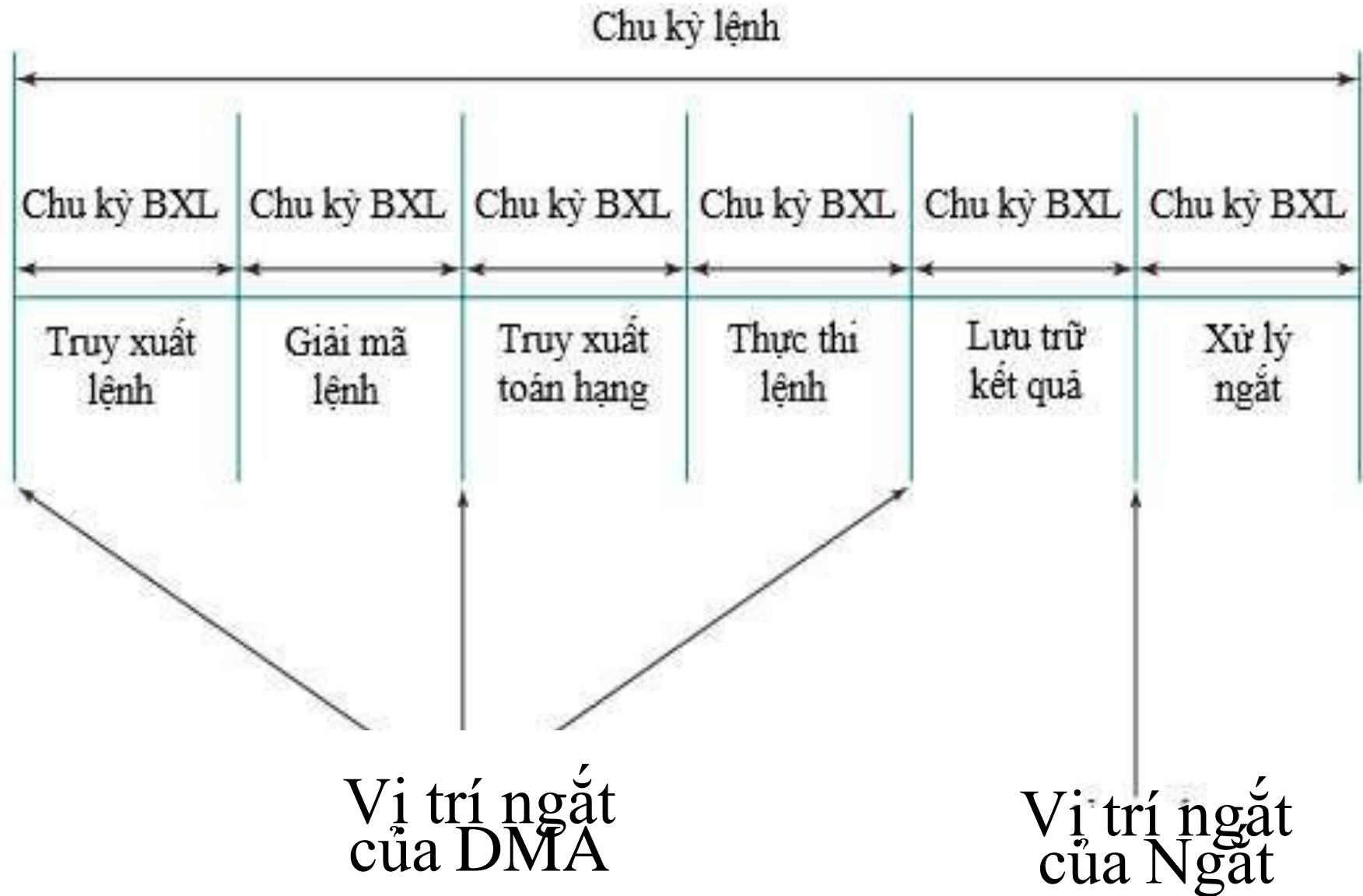
- CPU “nói” cho bộ điều khiển DMA.
  - + Yêu cầu đọc/ghi dữ liệu.
  - + Địa chỉ thiết bị I/O liên quan.
  - + Địa chỉ bắt đầu của khối dữ liệu cần đọc/ghi trong bộ nhớ.
  - + Số lượng từ dữ liệu cần truyền.
- CPU tiếp tục thực hiện công việc khác.
- Bộ điều khiển DMA điều khiển truyền dữ liệu.
- Sau khi truyền được một từ dữ liệu thì.
  - + Nội dung thanh ghi địa chỉ tăng.
  - + Nội dung bộ đếm dữ liệu giảm.
- Khi bộ đếm dữ liệu = 0 Bộ điều khiển DMA gửi ngắt tới CPU để báo kết thúc DMA.



## Các kiểu thực hiện DMA

- **DMA truyền theo khối (Block-transfer DMA):** Bộ điều khiển DMA sử dụng bus để truyền xong cả khối dữ liệu.
- **DMA đánh cắp chu kỳ (Cycle Stealing DMA):** Bộ điều khiển DMA cưỡng bức CPU treo tạm thời từng chu kỳ bus, DMAC chiếm bus thực hiện truyền một từ dữ liệu.
- **DMA trong suốt (Transparent DMA):** Bộ điều khiển DMA nhận biết những chu kỳ nào CPU không sử dụng bus thì chiếm bus để trao đổi một từ dữ liệu.

# DMA đánh cắp chu kỳ truyền

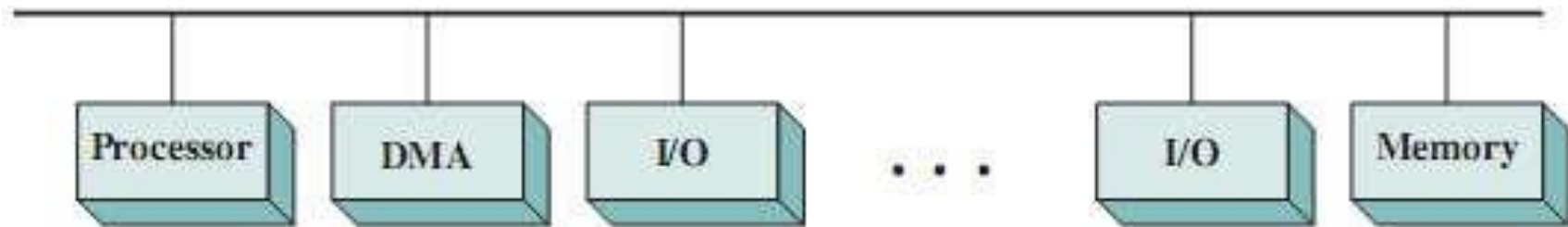




# DMA đánh cắp chu kỳ truyền

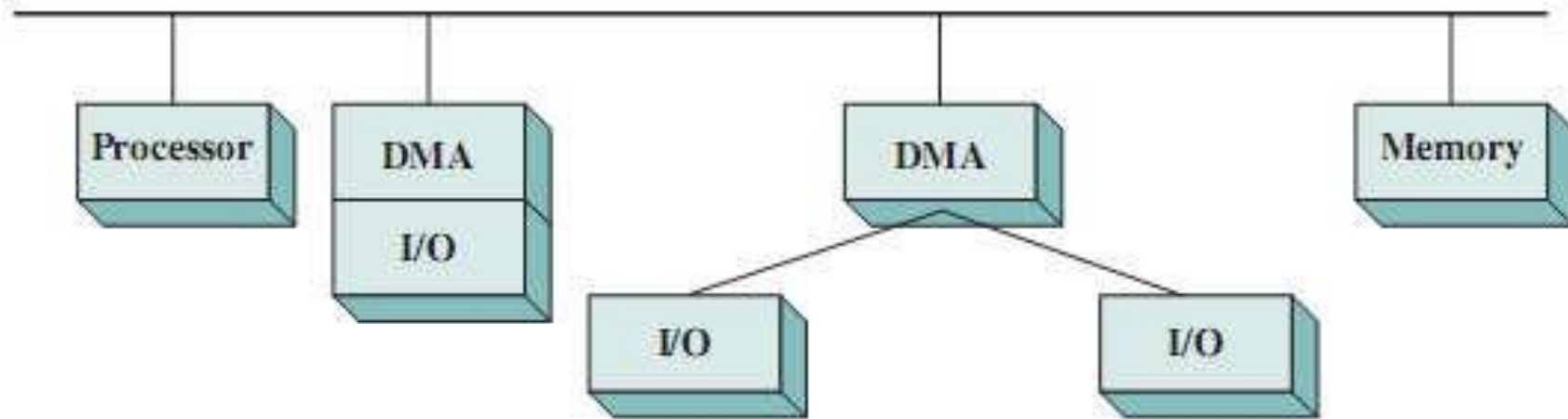
- Bộ điều khiển DMA chiếm bus trong một chu kỳ.
- Truyền một từ dữ liệu.
- Không phải là ngắt (CPU không chuyển ngữ cảnh).
- CPU bị treo ngay trước khi nó truy cập bus (tức là trước khi truy xuất toán hạng, dữ liệu hoặc ghi dữ liệu).
- Làm chậm CPU nhưng không nhiều như CPU tự thực hiện truyền.

# Cấu hình DMA (1)



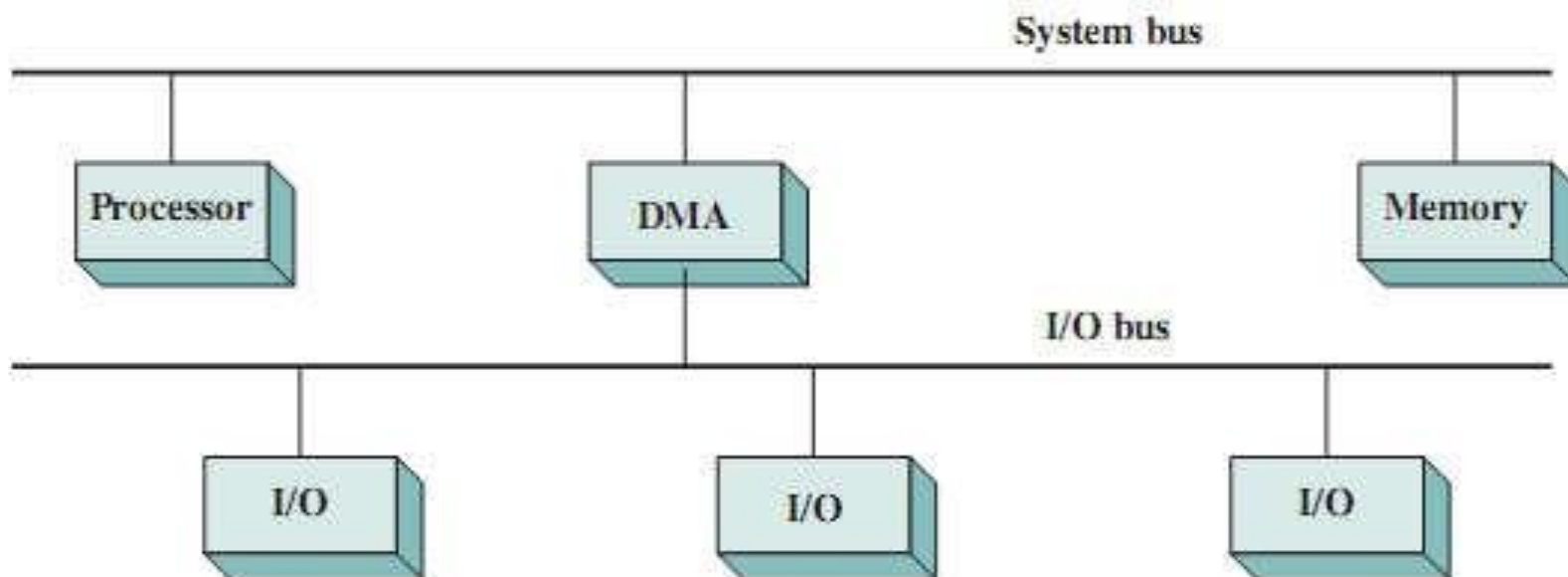
- Bus đơn, bộ điều khiển DMA tách rời.
- Mỗi lần truyền sử dụng bus 2 lần: I/O - DMA rồi sau đó DMA - bộ nhớ.
- CPU bị treo hai lần.

## Cấu hình DMA (2)



- Bus đơn, bộ điều khiển DMA tích hợp.
- Bộ điều khiển có thể hỗ trợ  $> 1$  thiết bị.
- Mỗi lần truyền sử dụng bus 1 lần: DMA - bộ nhớ.
- CPU bị treo một lần.

# Cấu hình DMA (3)



- Bus I/O riêng.
- Bus hỗ trợ tất cả các thiết bị có DMA.
- Mỗi lần truyền sử dụng bus 1 lần: DMA - bộ nhớ.
- CPU bị treo một lần.



# Bộ điều khiển DMA Intel 8237A

- Giao tiếp với họ 80x86 và DRAM
- Khi mô đun DMA cần bus nó sẽ gửi tín hiệu HOLD tới bộ vi xử lý.
- CPU phản hồi HLDA – Mô đun DMA có thể sử dụng bus.
- **Ví dụ: truyền dữ liệu từ bộ nhớ tới đĩa**
  1. Thiết bị yêu cầu DMA bằng cách nâng DREQ (yêu cầu DMA).
  2. DMA đặt HRQ (yêu cầu giữ) lên cao.
  3. CPU kết thúc chu kỳ bus hiện tại và đặt HDLA lên cao. HOLD duy trì trong suốt thời gian DMA.

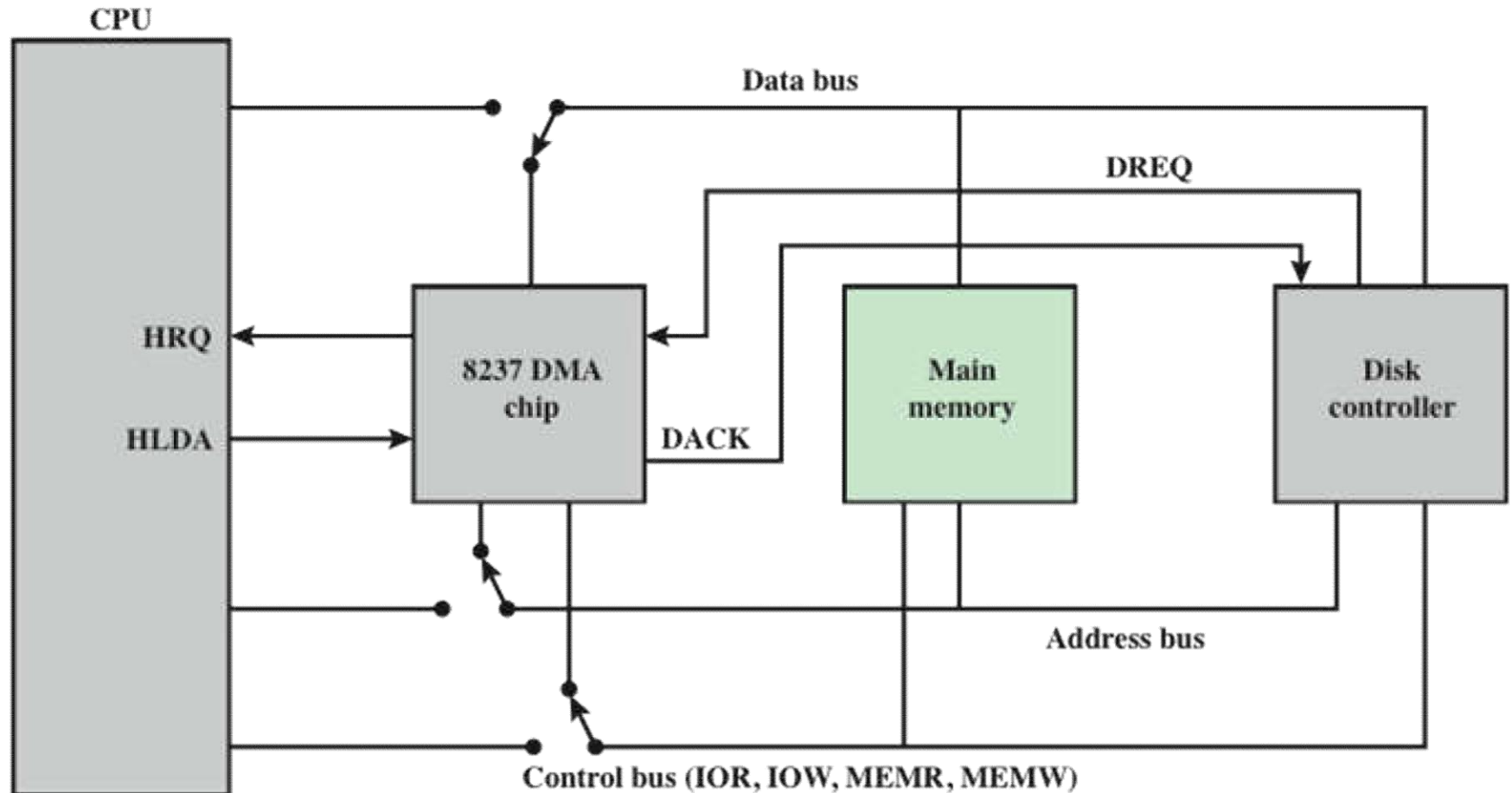


## Bộ điều khiển DMA Intel 8237A

- a) DMA kích hoạt DACK (DMA ack), bảo thiết bị bắt đầu truyền.
- b) DMA bắt đầu truyền bằng cách đặt byte đầu tiên của địa chỉ lên bus địa chỉ và kích hoạt MEMR; sau đó kích hoạt IOW để ghi vào ngoại vi. DMA giảm bộ đếm và tăng con trỏ địa chỉ. Lặp lại cho đến khi đếm về 0.
- c) DMA hủy HRQ, trả bus trở lại CPU.

# 8237 DMA

## Cách sử dụng Bus Hệ thống



DACK = DMA acknowledge  
 DREQ = DMA request  
 HLDA = HOLD acknowledge  
 HRQ = HOLD request



## 4. Kênh nhập xuất và vi xử lý

### Khái niệm

- Bộ điều khiển I/O (I/O controller) : mô đun I/O đơn giản hơn và cần có sự điều khiển chi tiết hơn từ bộ xử lý thường được sử dụng trong máy PC.
- Kênh I/O (I/O channel) hoặc bộ xử lý I/O (I/O processor): mô đun I/O có khả năng xử lý cao, hỗ trợ nhiều cho bộ xử lý thường được sử dụng trong máy mainframe.



## Đặc điểm kênh I/O

- Mở rộng khái niệm DMA.
- Có khả năng thực thi các lệnh vào ra, do đó kênh I/O kiểm soát hoàn toàn các hoạt động vào ra => không cần sự kiểm soát của CPU.
- Chương trình điều khiển vào-ra lưu trong bộ nhớ chính.
- CPU ra lệnh cho kênh I/O thực hiện chương trình trong bộ nhớ: bộ xử lý vào-ra trong kênh I/O thực hiện chương trình.



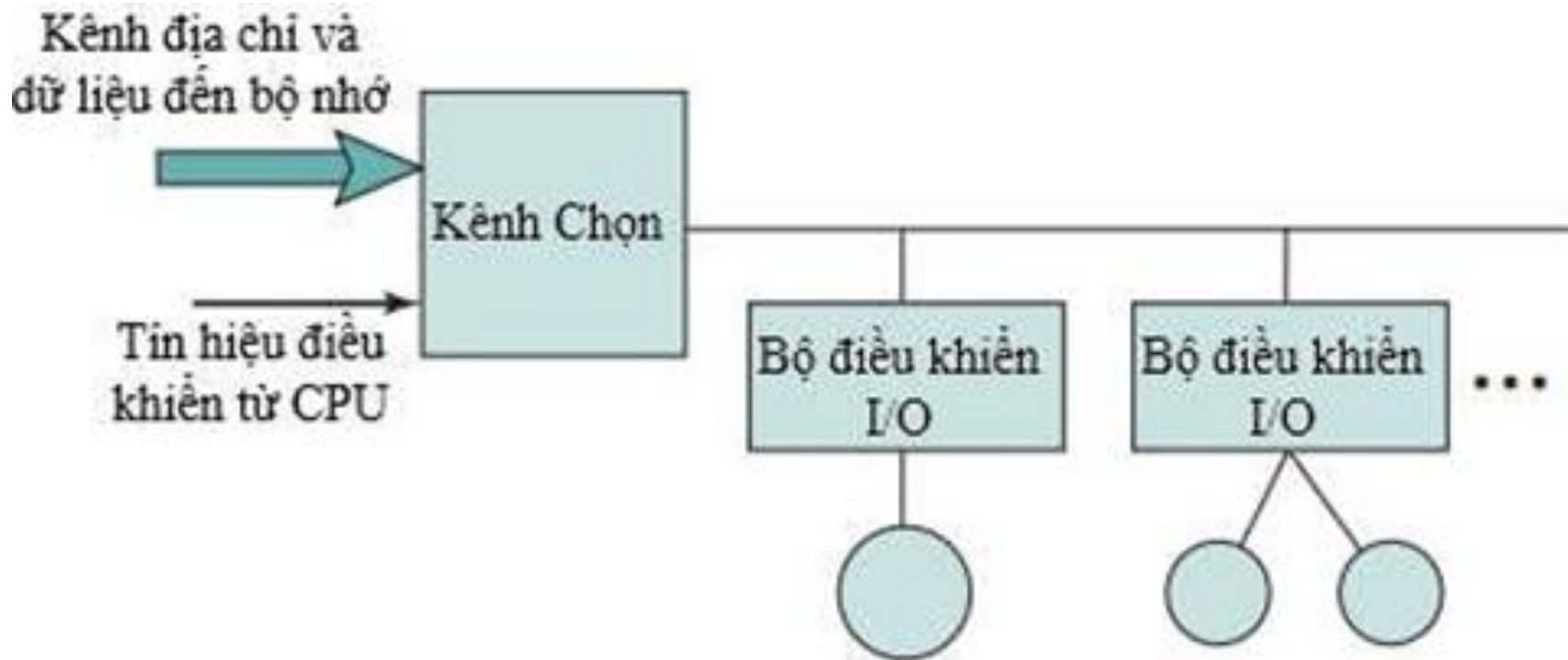
## Đặc điểm kênh I/O

- Kênh I/O lần lượt thực thi các lệnh và điều khiển việc truyền dữ liệu.
- Có 2 loại kênh I/O:
  - + Kênh chọn (selector channel).
  - + Kênh ghép (multiplexor channel).

# Kiến trúc kênh I/O

## Kênh chọn:

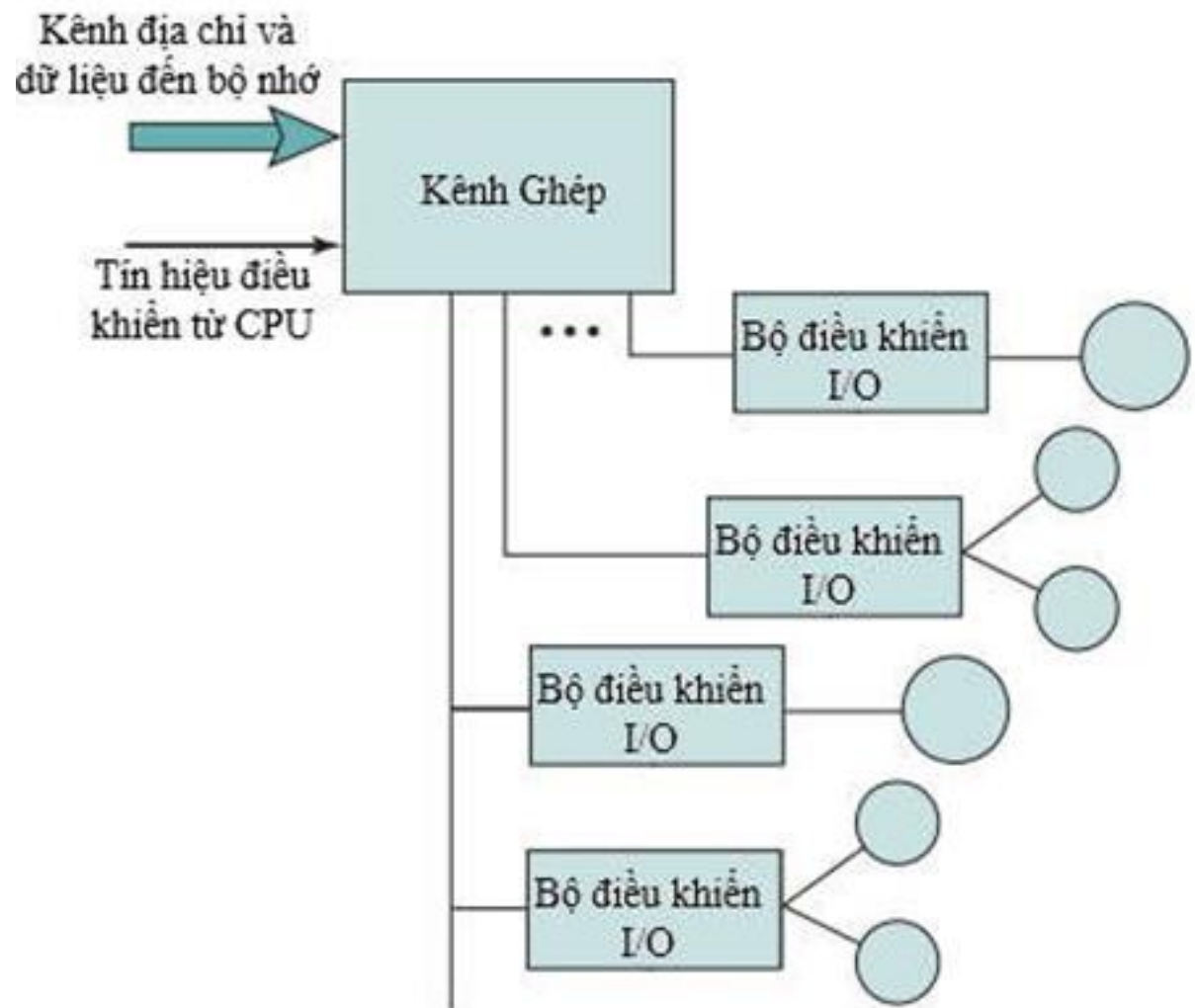
- + Điều khiển nhiều thiết bị tốc độ cao.
- + Truyền dữ liệu với từng thiết bị một.



# Kiến trúc kênh I/O

## Kênh ghép:

- + Xử lý I/O cho nhiều thiết bị tốc độ thấp cùng lúc.
- + Bộ ghép byte nhận hoặc chuyển các ký tự nhanh nhất có thể cho nhiều thiết bị.

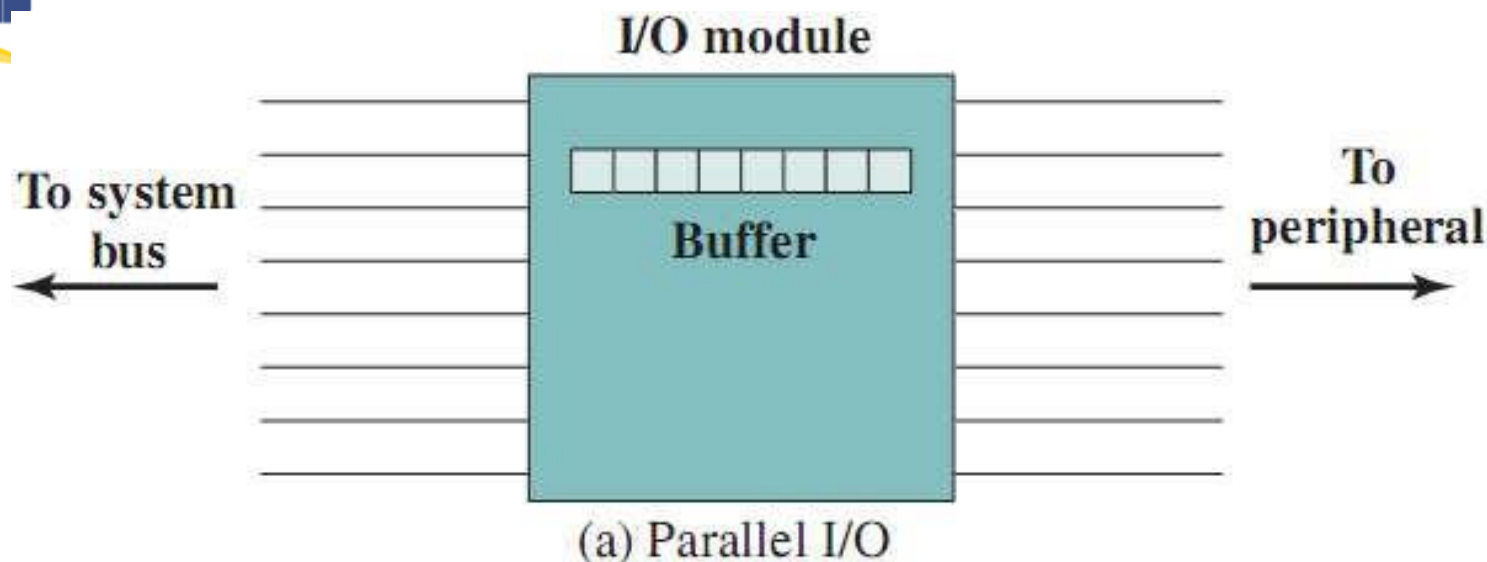




# Giao diện ngoại vi

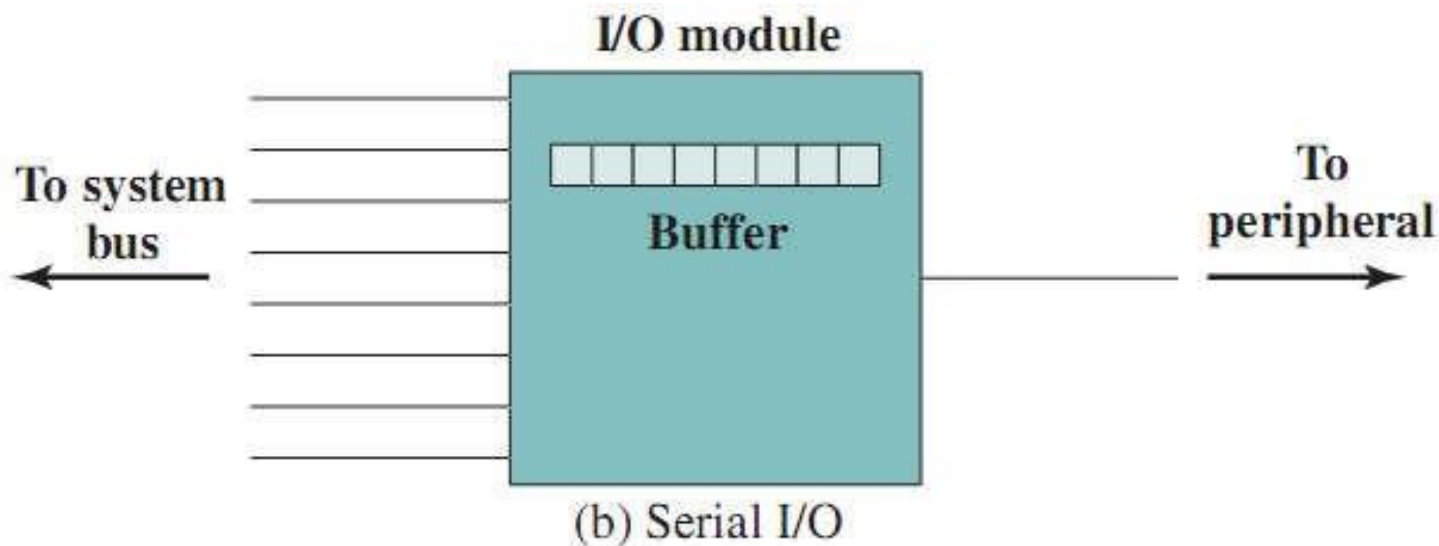
- ✓ **Các kiểu Giao diện**
  - Vào – ra song song
  - Vào – ra nối tiếp
- ✓ **Các cấu hình kết nối**
  - Điểm – điểm
  - Điểm – đa điểm

# I/O song song



- + Tốc độ nhanh.
- + Nhiều đường kết nối mô đun I/O - thiết bị ngoại vi.
- + Truyền nhiều bit đồng thời, giống như tất cả các bit của một từ được truyền đồng thời trên bus dữ liệu.
- + Sử dụng cho các thiết bị ngoại vi tốc độ cao như băng và đĩa,...

# I/O nối tiếp



- + Tốc độ chậm hơn.
- + Dùng một đường để truyền dữ liệu.
- + Truyền đi từng bit.
- + Sử dụng cho máy in và thiết bị đầu cuối...

# Các cấu hình kết nối

- Điểm - điểm (Point to Point): Thông qua một cổng vào-ra nối ghép với một thiết bị ngoại vi.
- Điểm - đa điểm (Point to Multipoint): Thông qua một cổng vào-ra cho phép nối ghép được với nhiều thiết bị ngoại vi.
- Ví dụ:
  - SCSI (Small Computer System Interface): 7 hoặc 15 thiết bị.
  - IEEE 1394 (FireWire): 63 thiết bị.





# Các cổng vào-ra thông dụng trên PC

- Các cổng PS/2: nối ghép bàn phím và chuột.
- Cổng nối ghép màn hình Displayport.
- Cổng LPT (Line Printer): nối ghép với máy in, là cổng song song (Parallel Port) – 25 chân.
- Cổng COM (Communication): nối ghép với MODEM, là cổng nối tiếp (Serial Port) - 9 hoặc 25 chân.
- Cổng USB (Universal Serial Bus): Cổng nối tiếp đa năng, cho phép nối ghép tối đa 127 thiết bị, nhờ các USB Hub.



ĐẠI HỌC ĐÀ NẴNG  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG VIỆT - HÀN  
Vietnam - Korea University of Information and Communication Technology

# THẢO LUẬN