



TRƯỜNG ĐẠI HỌC THỦY LỢI  
KHOA CÔNG NGHỆ THÔNG TIN



# KIẾN TRÚC MÁY TÍNH

Bộ môn Mạng và an toàn thông tin

Giảng viên: TS. Đoàn Thị Quế

# Chương 3

## Tổng quan về máy tính và hệ thống kết nối trong máy tính

# Nội dung chính

## Phần I. Tổng quan về máy tính

3.1 Các thành phần của máy tính

3.2 Hoạt động của máy tính

## Phần II. Hệ thống kết nối

3.3 Cấu trúc kết nối

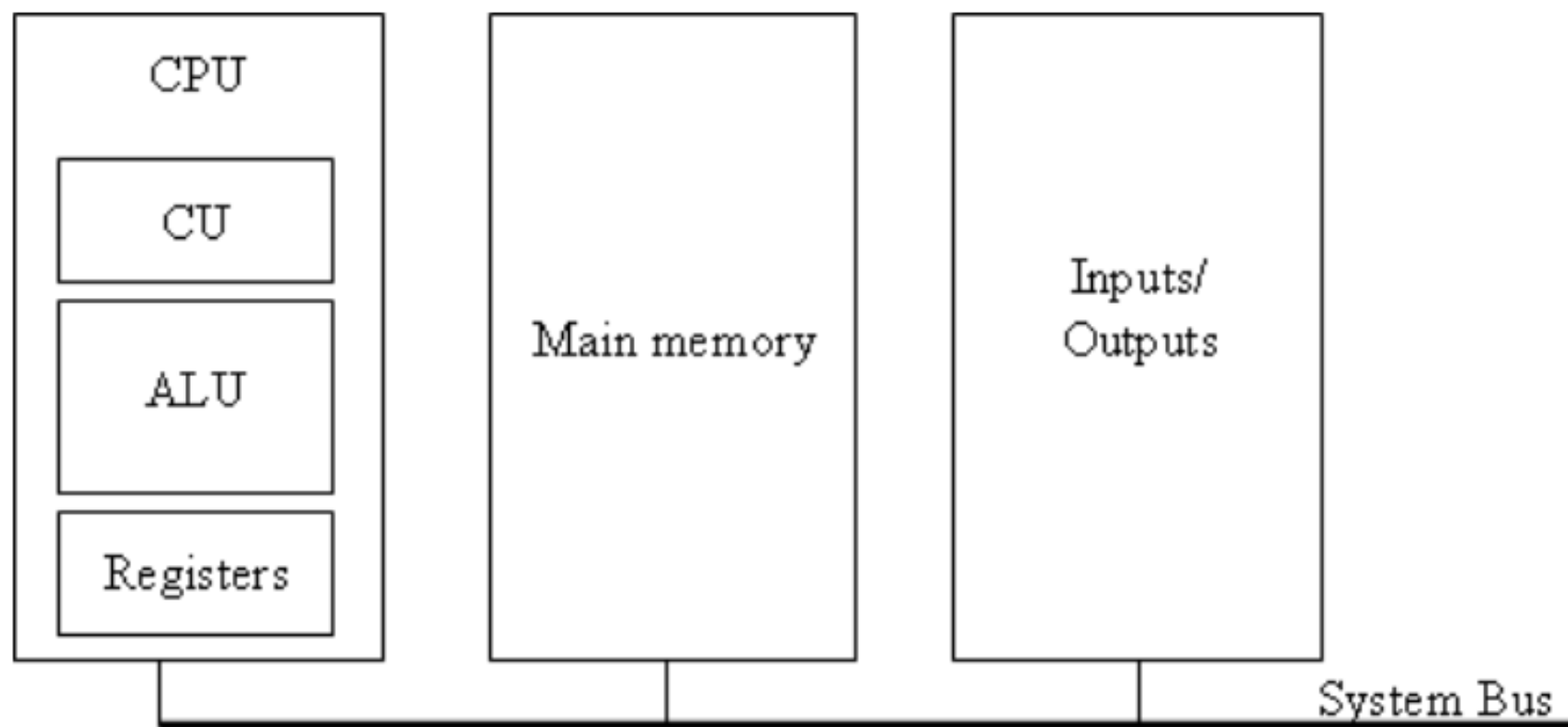
3.4 Hệ thống bus

3.5 Kết nối điểm-điểm

3.6 PCI Express

# 3.1. Các thành phần của máy tính

- Máy tính hiện đại ngày nay được thiết kế dựa trên **kiến trúc von Neumann**
- Kiến trúc Von Neumann có 3 điểm chính:
  - Dữ liệu và lệnh được lưu trữ trên cùng một bộ nhớ đọc-ghi (RAM)
  - Nội dung của dữ liệu được định vị theo vị trí (địa chỉ) mà không phụ thuộc vào kiểu dữ liệu.
  - Các lệnh được thực thi một cách tuần tự (trừ trong một số trường hợp yêu cầu gọi đến câu lệnh khác).



Kiến trúc von-Neumann hiện đại

# Các thành phần của máy tính (tiếp)

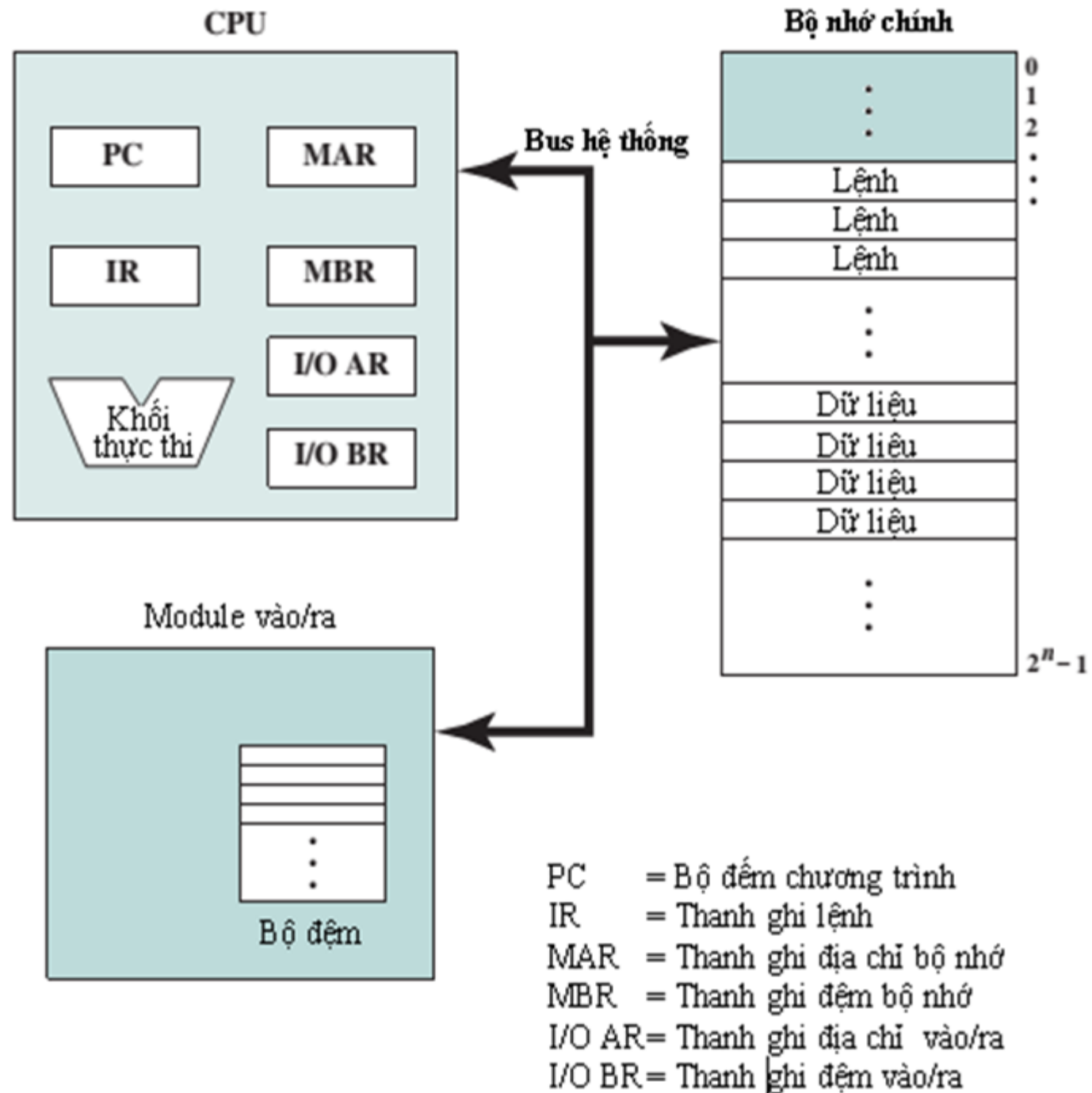
## ▪ Phần cứng

- CPU
  - Khối điều khiển (CU): Thực hiện biên dịch từng lệnh và tạo tín hiệu điều khiển
  - Khối tính toán số học và logic (ALU): Thực hiện các phép toán số học và logic
- Các Module vào/ra (I/O module)
  - Module vào: bao gồm các thành phần cơ bản cho việc nhận vào dữ liệu và lệnh; chuyển đổi chúng thành dạng tín hiệu sử dụng bên trong hệ thống
  - Module ra: công cụ để hiển thị kết quả
- Bộ nhớ trong:
  - Bộ nhớ chính (ROM, RAM): lưu trữ lệnh, dữ liệu
  - Bộ nhớ Cache: cải thiện hiệu suất của hệ thống
- Bus hệ thống (System bus)
  - Vận chuyển thông tin giữa CPU, I/O module và bộ nhớ trong

## ▪ Phần mềm

- Là các chương trình
- Mỗi chương trình gồm một chuỗi các lệnh
- Quá trình thực hiện chương trình là truy xuất lệnh từ bộ nhớ và thực thi lệnh của CPU

# Các thành phần của máy tính



## Giải thích một số thanh ghi trong hình:

- Thanh ghi **MAR (Memory Address Register)** chứa địa chỉ trong bộ nhớ cho lần đọc hoặc ghi tiếp theo
- Thanh ghi **MBR (Memory Buffer Register)** dữ liệu được ghi vào bộ nhớ hoặc nhận dữ liệu được đọc từ bộ nhớ.
- Thanh ghi **I/O AR (I/O Address Register)** xác định một thiết bị I/O cụ thể.
- Thanh ghi **I/O BR (I/O Buffer Register)** được sử dụng để trao đổi dữ liệu giữa một mô-đun I/O và CPU.
- Thanh ghi **PC (Program counter Register)** chứa địa chỉ lệnh tiếp theo
- Thanh ghi **IR (Instruction Register)** chứa lệnh đang được thực thi



# Nội dung chính

## Phần I. Tổng quan về máy tính

3.1 Các thành phần của máy tính

**3.2 Hoạt động của máy tính**

## Phần II. Hệ thống kết nối

3.3 Cấu trúc kết nối

3.4 Hệ thống bus

3.5 Kết nối điểm-điểm

3.6 PCI Express

## 3.2. Hoạt động của máy tính

- Thực hiện chương trình
- Thực hiện xử lý ngắt
- Thực hiện vào/ra

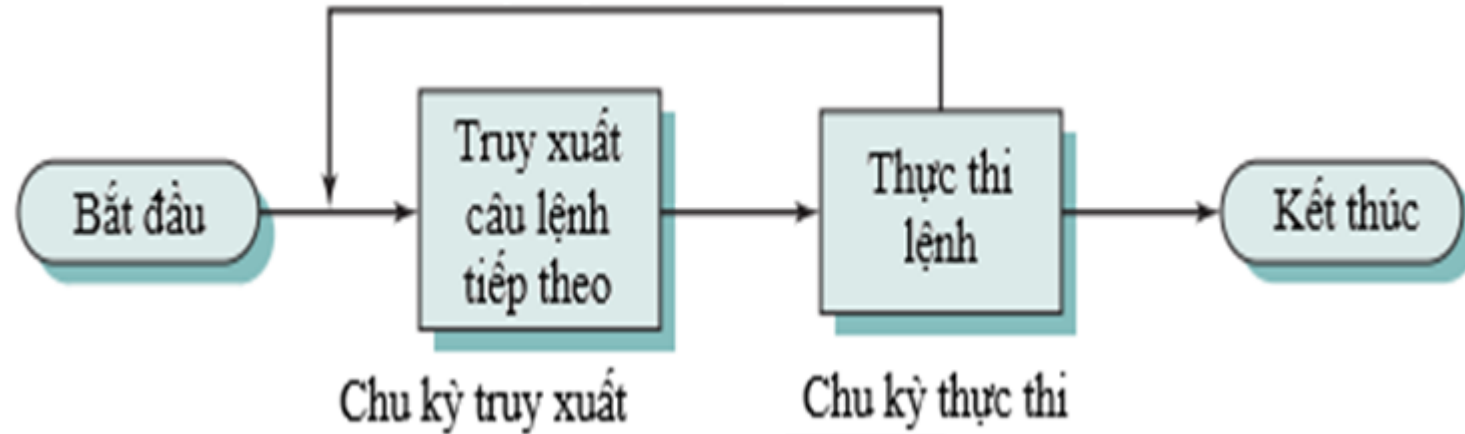
## 3.2. Hoạt động của máy tính

### a. Thực hiện chương trình

- Chương trình gồm một tập các lệnh lưu trữ trong bộ nhớ
- CPU phải thực hiện lần lượt các lệnh:
  - Quá trình CPU thực hiện 1 lệnh gồm 2 bước:
    - Bước 1: Lấy (truy xuất) lệnh từ bộ nhớ
    - Bước 2: Thực thi lệnh.
  - Việc thực thi một chương trình là quá trình lặp đi lặp lại việc truy xuất và thực thi lệnh
- Quá trình thực hiện một lệnh được gọi là **chu kỳ lệnh (instruction cycle)**
  - Quá trình truy xuất lệnh từ bộ nhớ được gọi là **chu kỳ truy xuất (fetch cycle)**
  - Quá trình thực thi lệnh được gọi là **chu kỳ thực thi (execute cycle)**

# Chu kỳ lệnh

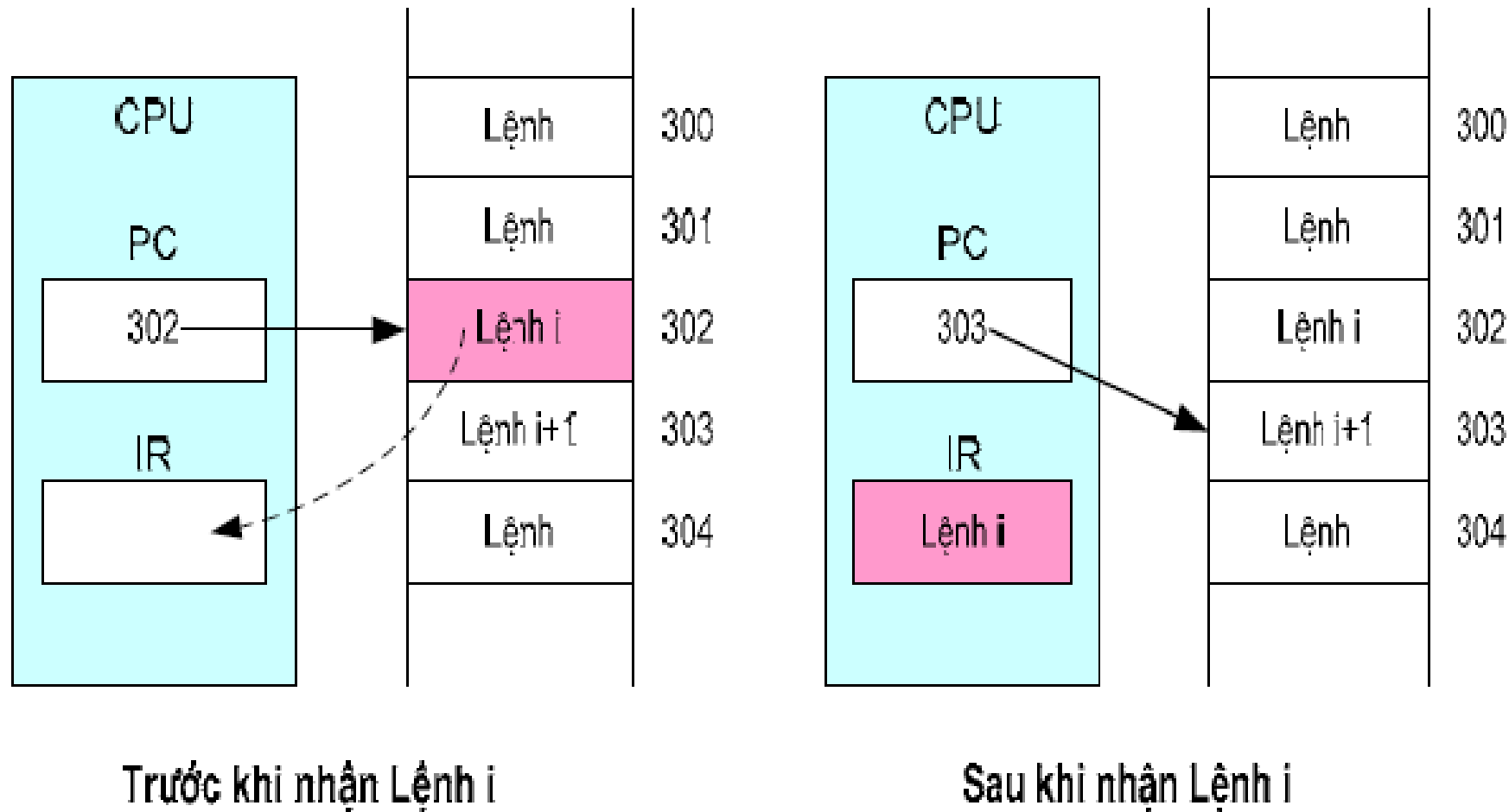
## Chu kỳ truy xuất



### Chu kỳ lệnh cơ bản

- Vào đầu mỗi chu kỳ lệnh, bộ xử lý truy xuất một lệnh từ bộ nhớ:
  - Thanh ghi PC (Program Counter) giữ địa chỉ của lệnh được truy xuất tiếp theo
  - Lệnh vừa được truy xuất được tải vào thanh ghi lệnh IR (Instruction Register)
  - Bộ xử lý tăng PC sau mỗi lần truy xuất lệnh, do đó nó sẽ truy xuất được lệnh tiếp theo vào lần sau.

# Minh họa chu kỳ truy xuất



# Chu kỳ lệnh

## Chu kỳ thực thi

- CPU giải mã lệnh và thực hiện các hoạt động (action) tương ứng được chỉ ra trong mã lệnh (Opcode)
- Có 4 nhóm hoạt động chính của một CPU:

### **Bộ xử lý – bộ nhớ**

- Dữ liệu truyền từ bộ xử lý đến bộ nhớ hoặc ngược lại

### **Bộ xử lý – I/O**

- Dữ liệu truyền đến/đi từ thiết bị ngoại vi bằng cách truyền thông tin giữa bộ xử lý và module vào/ra

### **Xử lý dữ liệu**

- Bộ xử lý thực hiện một số phép toán số học hoặc logic trên dữ liệu

### **Điều khiển**

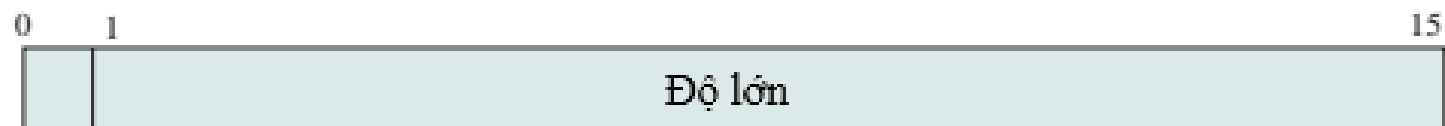
- Đưa ra lệnh, chỉ rõ thứ tự thực hiện các lệnh bị thay đổi

# Ví dụ 3.5: thực hiện lệnh

Máy giả thiết gồm một số thông tin cấu hình như sau:



(a) Định dạng lệnh



(b) Định dạng số nguyên

Thanh ghi PC = Địa chỉ của lệnh

Thanh ghi IR = Lệnh sẽ được thực hiện

Thanh ghi AC = Thanh ghi lưu trữ dữ liệu tạm thời

(c) Các thanh ghi trong CPU

0001 = Đọc dữ liệu từ bộ nhớ vào thanh ghi AC

0010 = Lưu trữ dữ liệu từ thanh ghi AC vào bộ nhớ

0101 = Đọc dữ liệu từ bộ nhớ và cộng vào thanh ghi AC

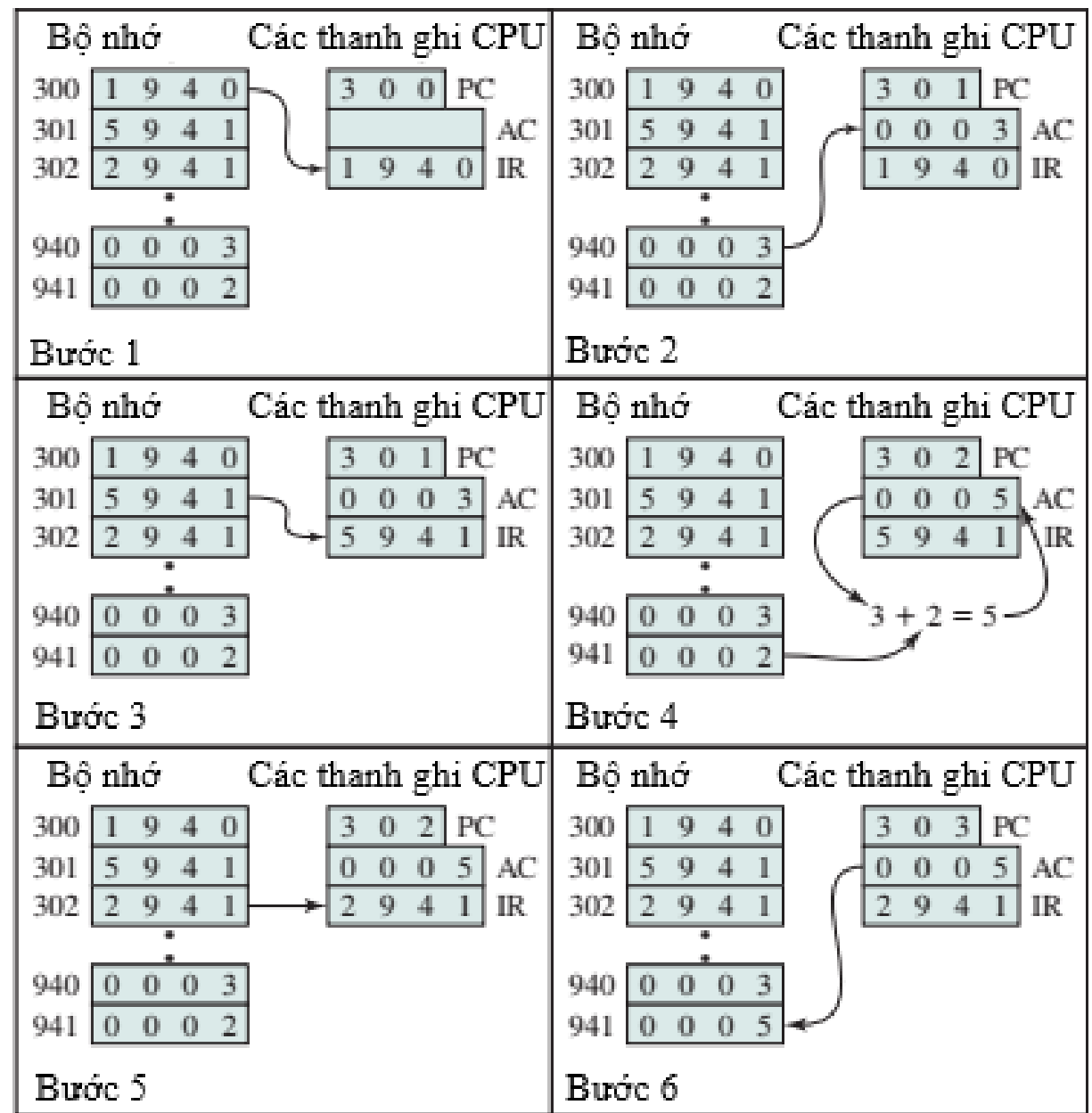
(d) Các mã lệnh được sử dụng

Hình 3.4 Đặc tính của máy giả thiết

## Ví dụ 3.5

### Thực hiện lệnh

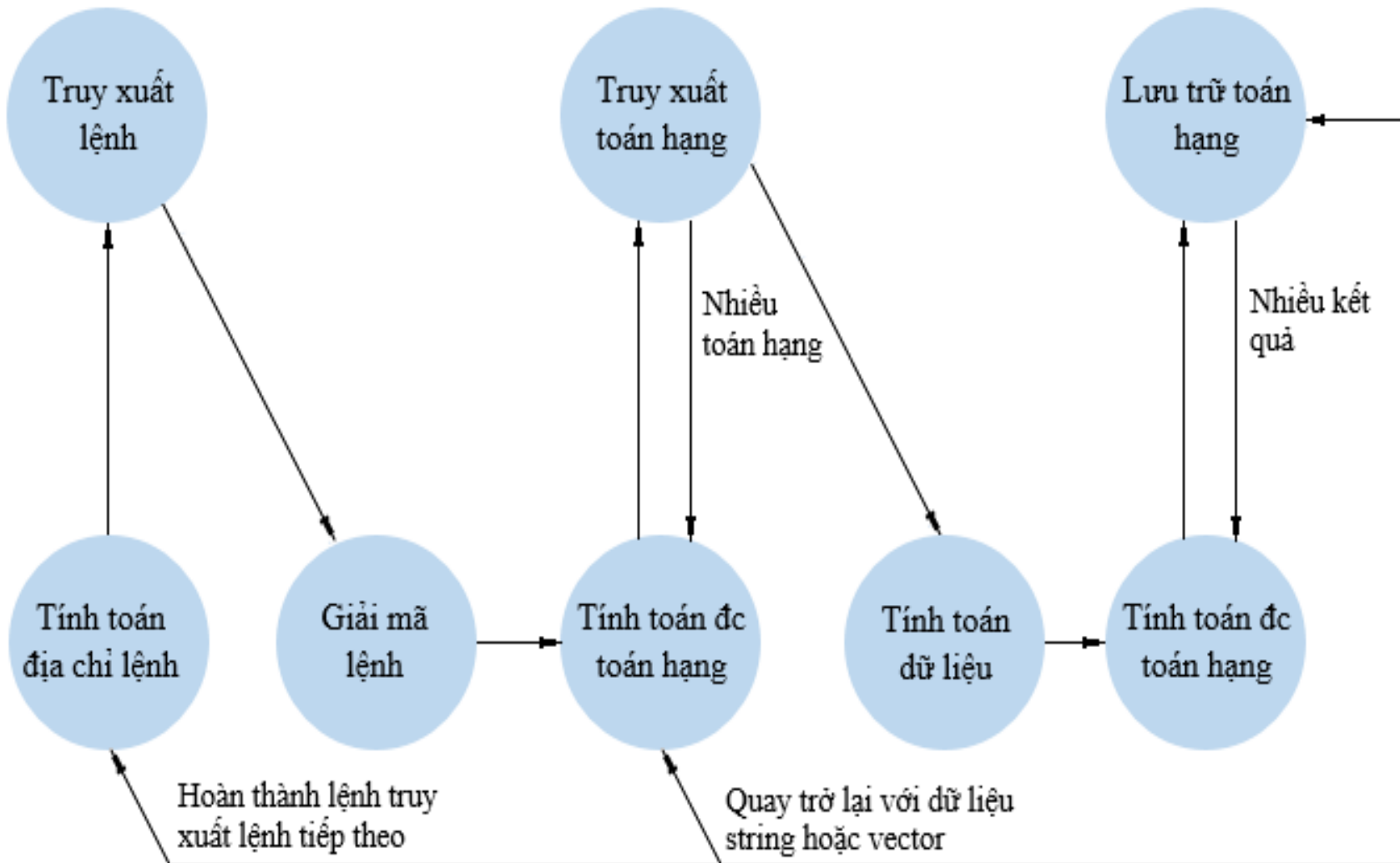
0001 = Đọc dữ liệu từ bộ nhớ vào thanh ghi AC  
 0010 = Lưu trữ dữ liệu từ thanh ghi AC vào bộ nhớ  
 0101 = Đọc dữ liệu từ bộ nhớ và cộng vào thanh ghi AC



Dữ liệu và lệnh được biểu diễn ở hệ thập lục phân



# Sơ đồ trạng thái chu kỳ lệnh



# Bài tập áp dụng

Máy giả thiết trong ví dụ 3.5 hai lệnh vào/ra sau:

0011 = Đọc dữ liệu từ module I/O vào thanh ghi AC

0111 = Ghi dữ liệu từ AC ra module I/O

Biết các thiết bị ngoại vi được đánh địa chỉ 12 bit. Mô tả hoạt động của chương trình sau (tương tự như Ví dụ 3.5):

1. Đọc dữ liệu từ thiết bị 5 vào thanh ghi AC.
2. Đọc dữ liệu từ ô nhớ có địa chỉ 940 và cộng với AC.
3. Ghi AC ra thiết bị 6.

Giả sử giá trị được lấy từ thiết bị 5 là 8 và ngăn nhớ có địa chỉ 940 chứa giá trị 7.  
(Dữ liệu, lệnh và địa chỉ được biểu diễn ở hệ thập lục phân).

## 3.2. Hoạt động của máy tính

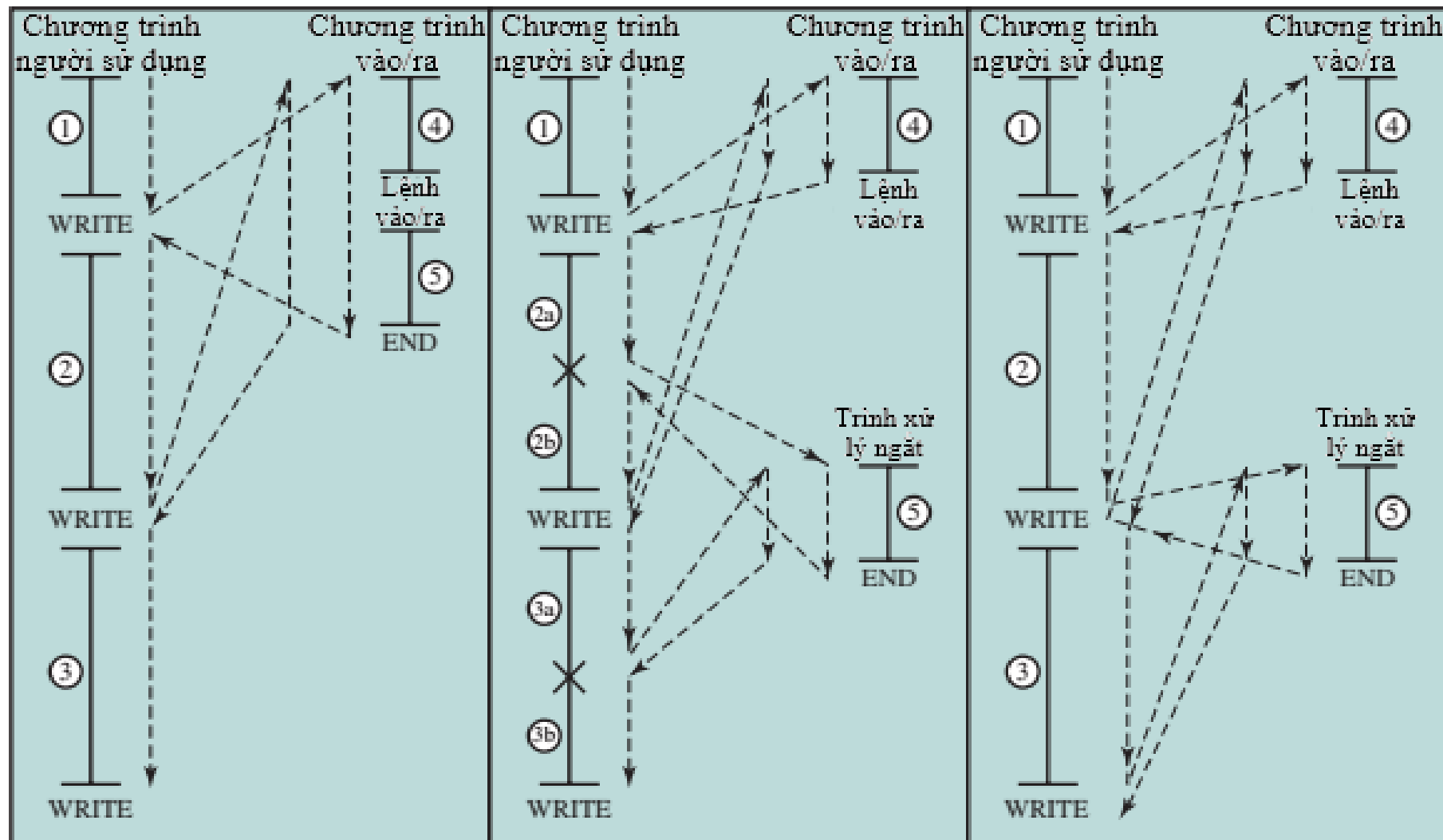
### b. Thực hiện xử lý ngắt

- **Ngắt** là một cơ chế máy tính cho phép các module khác (I/O, bộ nhớ) có thể ngắt quá trình xử lý thông thường của BXL.
- Một số ngắt:
  - **Ngắt chương trình:** Sinh ra bởi lỗi thi hành lệnh, ví dụ như tràn số học, lỗi chia cho 0, cố tình thực hiện các lệnh máy không hợp lệ, hoặc tham chiếu ngoài phạm vi bộ nhớ mà người sử dụng được phép
  - **Ngắt định thời:** Sinh ra bởi đồng hồ nằm trong bộ xử lý. Nó cho phép hệ điều hành thực hiện các chức năng cơ bản nhất định.
  - **Ngắt I/O:** Sinh ra bởi bộ điều khiển I/O, để báo hiệu hoàn thành một thao tác, yêu cầu dịch vụ từ bộ xử lý, hoặc báo hiệu các trường hợp lỗi
  - **Ngắt do lỗi phần cứng:** Gây ra bởi một số lỗi, như lỗi nguồn hay lỗi bộ nhớ

# Quá trình xử lý ngắt

- Ngắt được đưa vào chủ yếu như là 1 để để cải thiện hiệu quả xử lý:
  - Trong trường hợp VXL thực hiện chương trình có trao đổi dữ liệu với I/O
  - Do tốc độ của I/O chậm hơn rất nhiều so với VXL → VXL phải đợi I/O
  - Giải pháp: trong lúc chờ đợi I/O, VXL thực hiện tiếp các phần công việc khác → đến khi I/O xong, nó sẽ gửi tín hiệu đến VXL (tín hiệu y/c ngắt) → VXL dừng công việc đang làm (ngắt), phục vụ I/O → VXL tiếp tục công việc đang thực hiện

# Điều khiển dòng chương trình

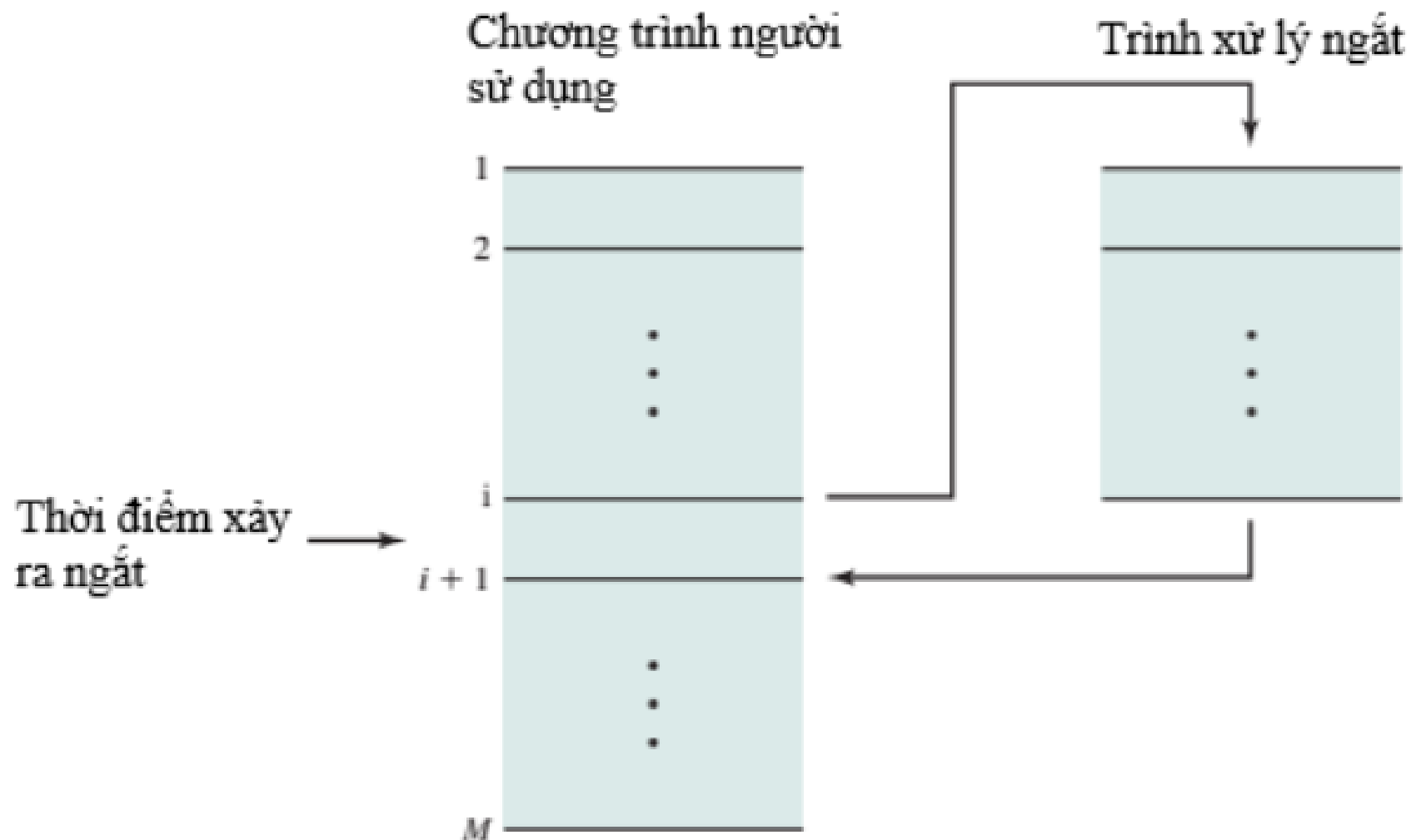


a) Không có ngắt

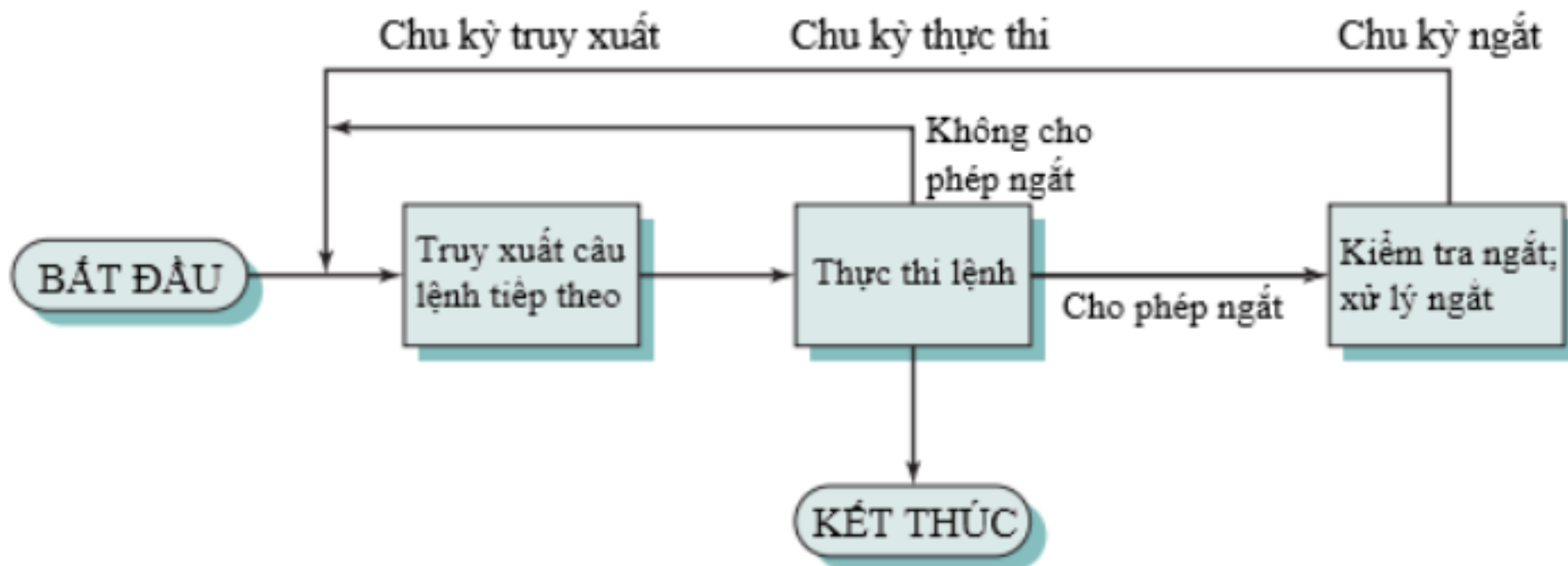
b) Sử dụng ngắt, đợi vào/ra ngắn

c) Sử dụng ngắt, đợi vào/ra dài

# Điều khiển ngắt

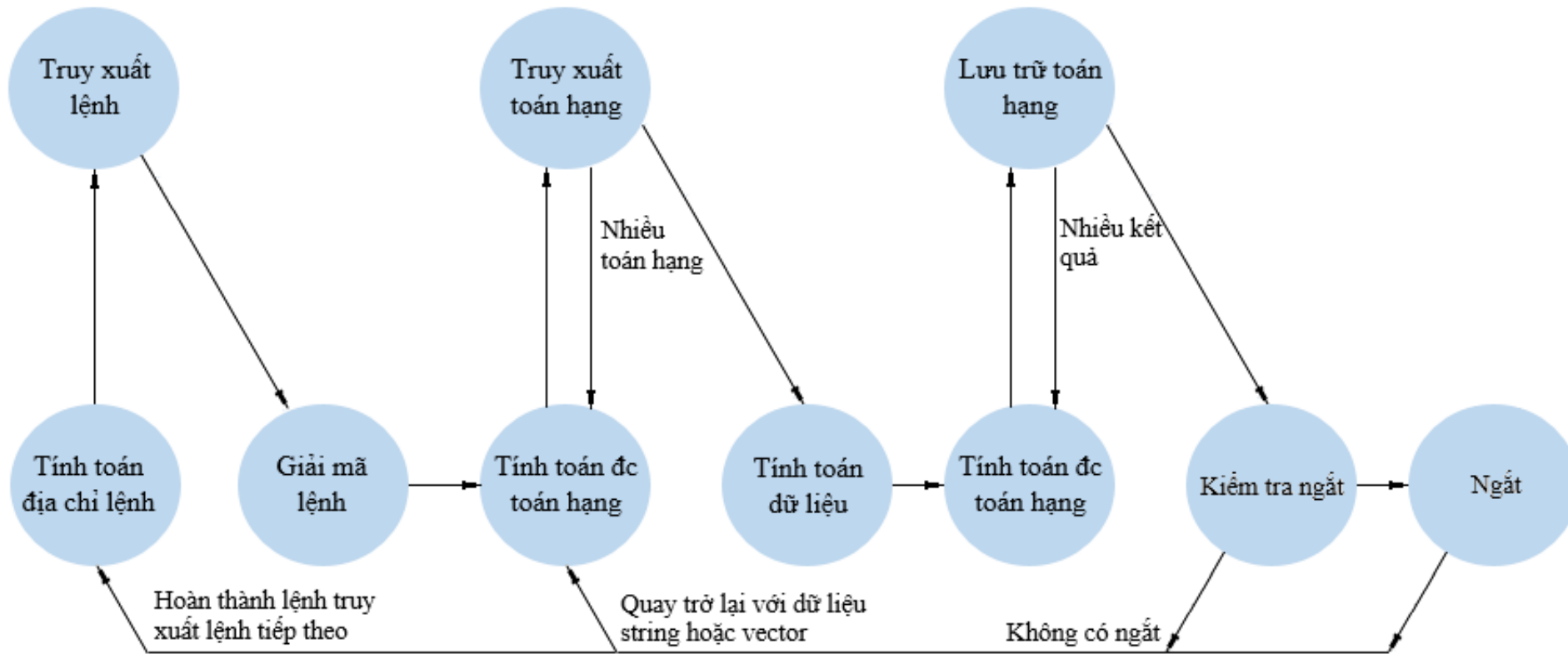


# Chu kỳ lệnh có ngắt



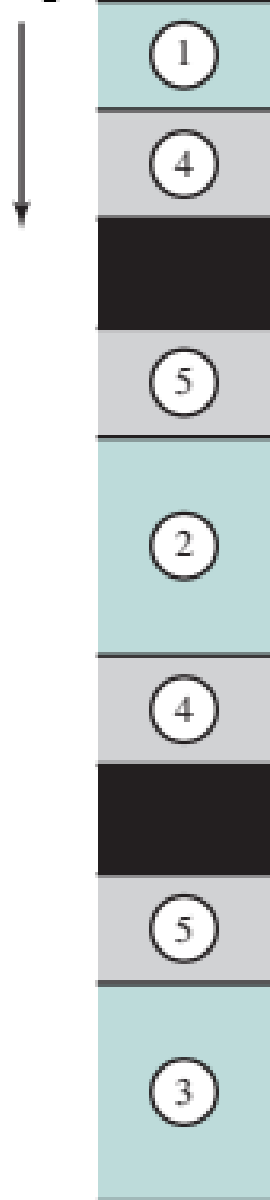
Hình 3.9 Chu kỳ lệnh có ngắt

# Sơ đồ trạng thái chu kỳ lệnh Có ngắt





Thời gian



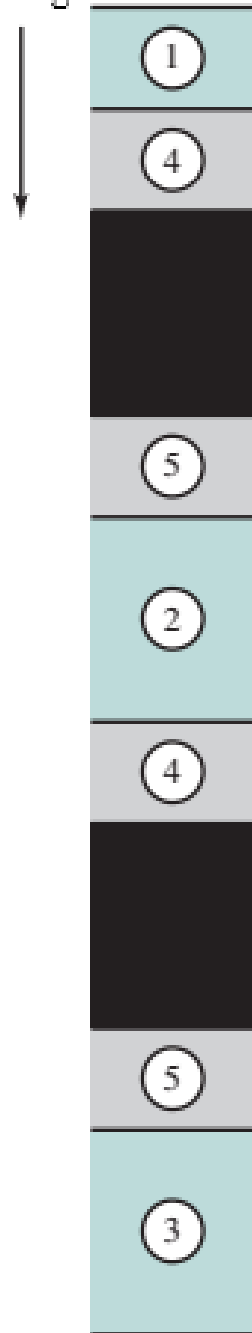
(a) Không có ngắt



(b) Có ngắt

**Minh họa  
thời gian  
thực hiện  
chương  
trình**

Thời gian



(a) Không có ngắt

Hoạt động vào/ra;  
bộ xử lý đợi

Hoạt động vào/ra;  
bộ xử lý đợi



(b) Có ngắt

Hoạt động vào ra  
cùng lúc với bxl  
thực thi; sau đó  
bxl đợi

Hoạt động vào ra  
cùng lúc với bxl  
thực thi; sau đó  
bxl đợi

**Định thời  
chương  
trình:  
Đợi I/O dài**

# Xử lý nhiều ngắt

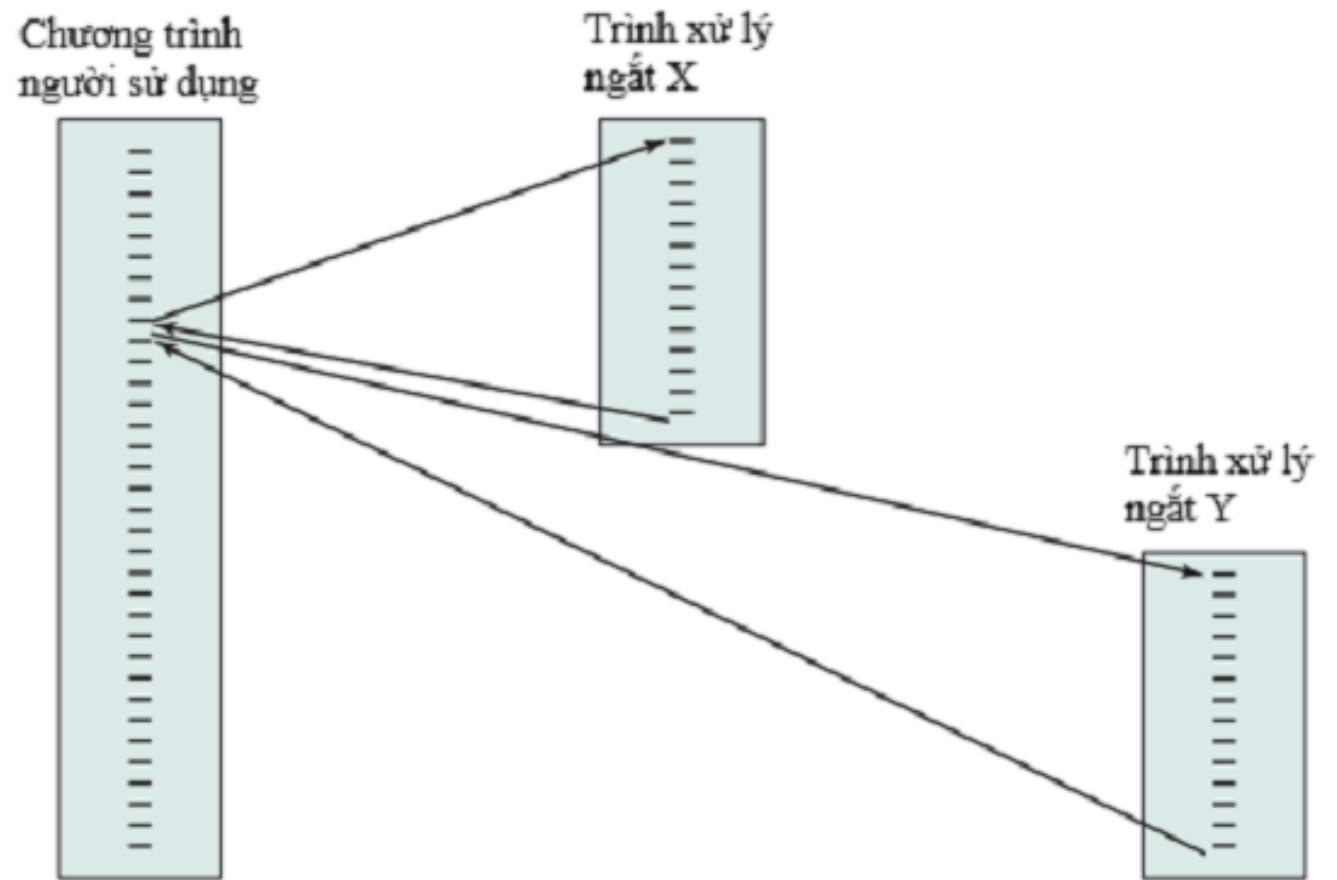
Trong trường hợp có nhiều ngắt, hai phương pháp để xử lý:

- Xử lý ngắt tuần tự
- Xử lý ngắt ưu tiên

# Xử lý nhiều ngắt

## ■ Xử lý ngắt tuần tự:

- Bộ vi xử lý sẽ bỏ qua các tín hiệu ngắt khác trong khi xử lý một yêu cầu ngắt
- Các yêu cầu ngắt đó sẽ phải chờ đến khi bộ xử lý xử lý xong ngắt hiện tại
- Sau khi thực thi xong một ngắt, bộ xử lý sẽ kiểm tra xem có ngắt nào đang chờ không. Các ngắt sẽ được xử lý lần lượt

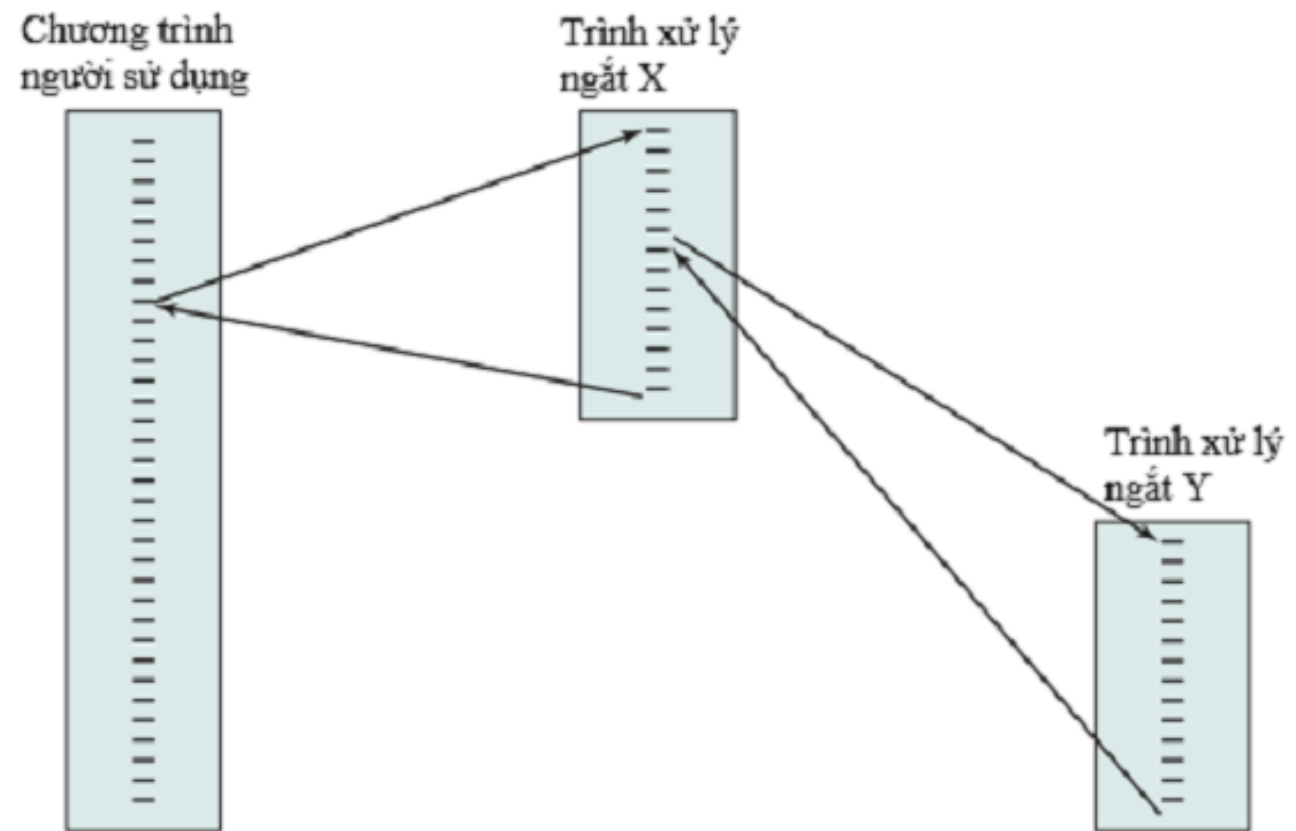


(a) Xử lý ngắt tuần tự

# Xử lý nhiều ngắt

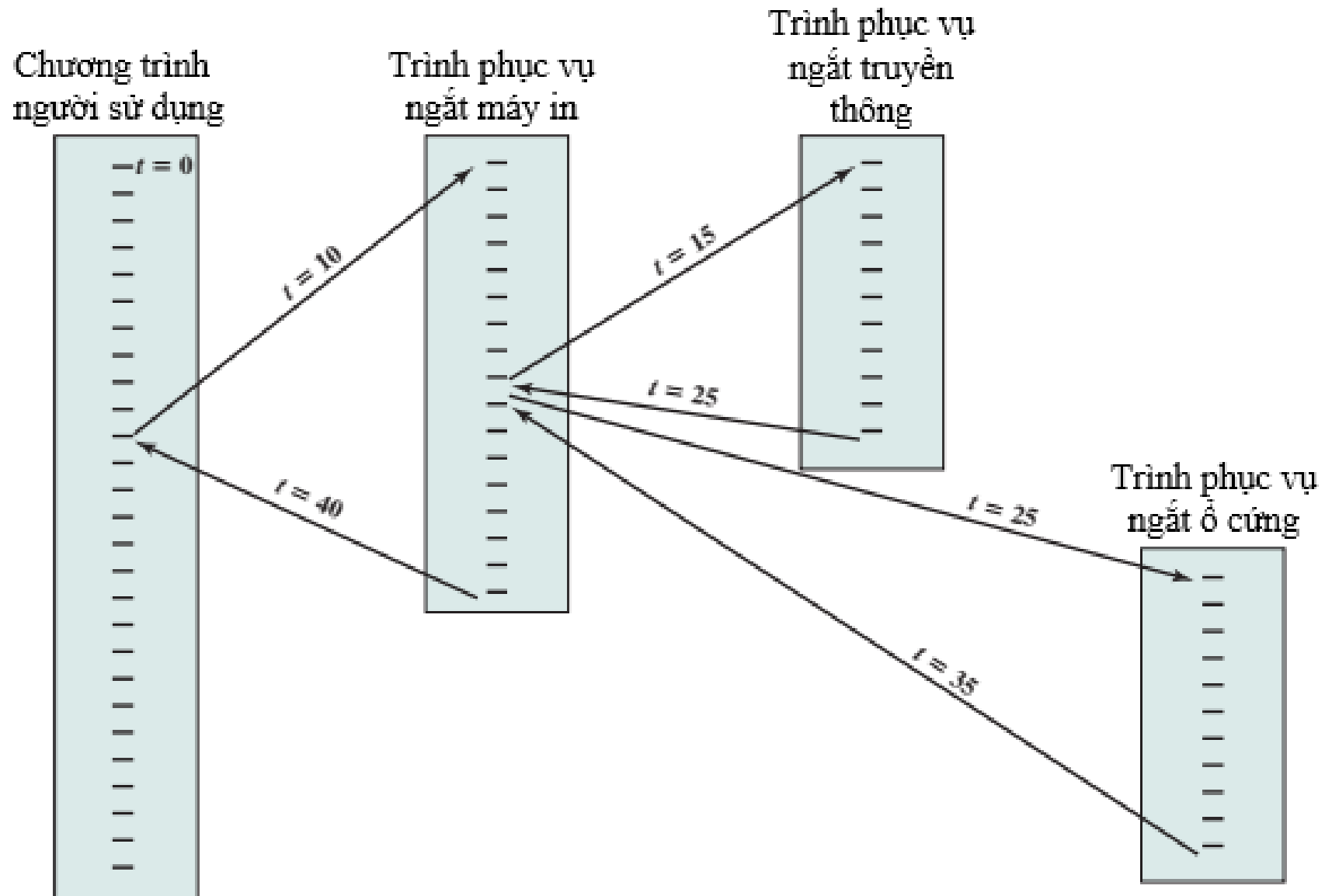
## ■ Xử lý ngắt ưu tiên:

- Cho phép các ngắt có mức độ ưu tiên cao hơn được ngắt các ngắt có mức độ ưu tiên thấp hơn



(b) Xử lý ngắt ưu tiên

# Ví dụ quá trình xử lý đa ngắt ưu tiên



## c. Thực hiện vào/ra

- Module I/O có thể trao đổi dữ liệu trực tiếp với bộ xử lý (tương tự như cách bộ xử lý đọc hoặc ghi dữ liệu với bộ nhớ):
  - Bộ xử lý đọc hoặc ghi dữ liệu với module I/O thông qua việc định một địa chỉ của một thiết bị cụ thể.
  - Trình tự các lệnh thực hiện trao đổi dữ liệu với I/O giống như ví dụ 3.5 với các lệnh vào/ra thay vì các lệnh tham chiếu bộ nhớ.
- Cơ chế truy cập bộ nhớ trực tiếp (Direct Memory Access - DMA): cho phép I/O trao đổi dữ liệu trực tiếp với bộ nhớ
  - Bộ xử lý cấp cho module I/O quyền đọc/ghi vào bộ nhớ do đó việc truyền tin giữa module I/O và bộ nhớ có thể diễn ra trực tiếp mà không cần thông qua bộ xử lý
  - Giải phóng bộ XL khỏi nhiệm vụ điều khiển việc chuyển dữ liệu

# Nội dung chính

## Phần I. Tổng quan về máy tính

3.1 Các thành phần của máy tính

3.2 Hoạt động của máy tính

## Phần II. Hệ thống kết nối

**3.3 Cấu trúc kết nối**

3.4 Hệ thống bus

3.5 Kết nối điểm-điểm

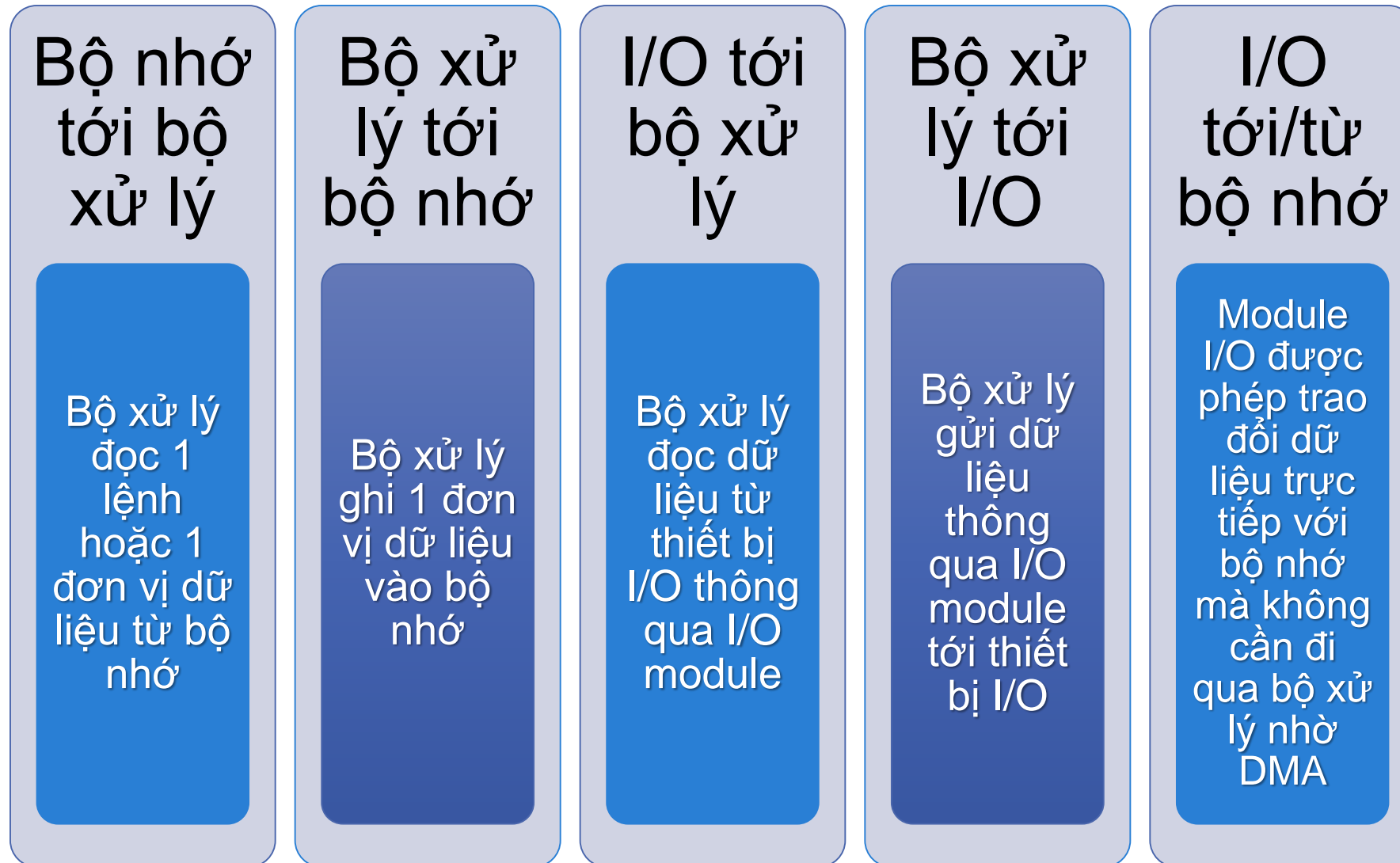
3.6 PCI Express



## 3.3 Cấu trúc kết nối

- Các thành phần chính (bộ vi xử lý, bộ nhớ, I /O) của máy tính cần được kết nối để trao đổi dữ liệu với nhau
- Một tập các đường kết nối tạo thành ***cấu trúc kết nối*** (Interconnection Structures)
- Cấu trúc này được thiết kế phụ thuộc vào cơ chế trao đổi dữ liệu giữa các thành phần máy tính.

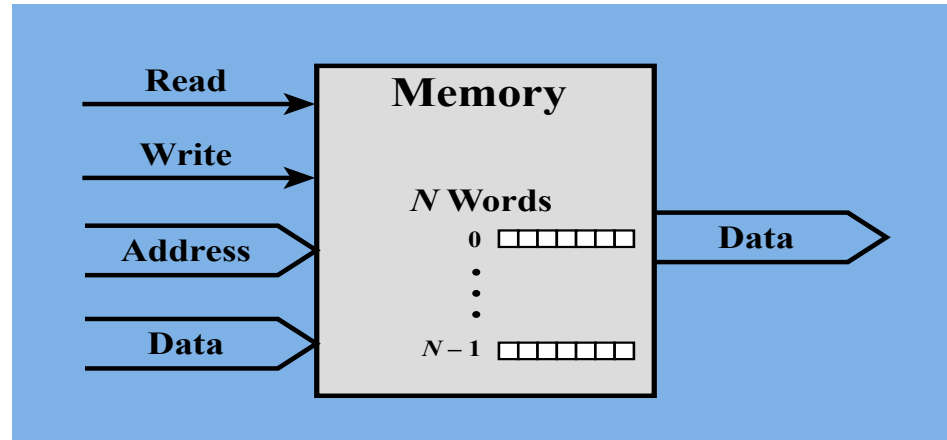
# Cấu trúc kết nối hỗ trợ các hình thức truyền sau:



# Các dạng dữ liệu đến/đi từ các thành phần máy tính

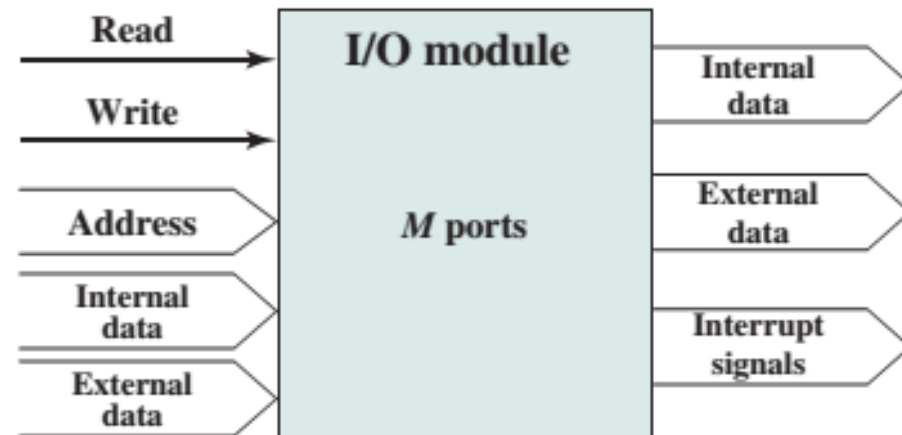
## ▪ Bộ nhớ:

- T/h điều khiển đọc/ghi
- T/h địa chỉ
- Dữ liệu đi vào/ra



## ▪ Module I/O

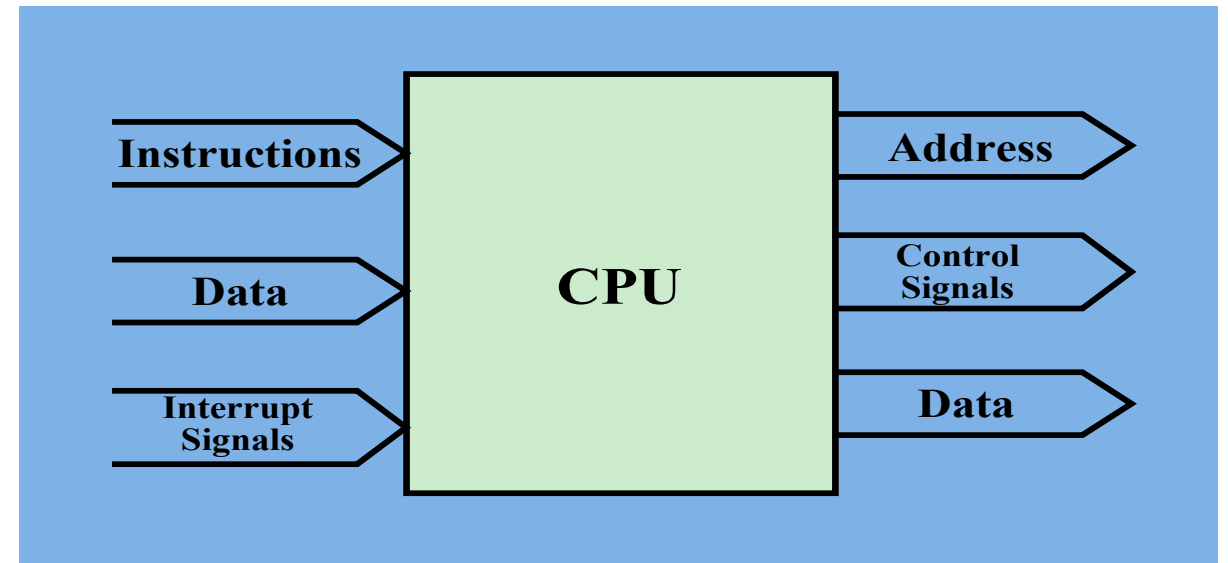
- T/h điều khiển đọc/ghi
- T/h địa chỉ
- Dữ liệu bên trong (ngoài): dữ liệu đến từ CPU(thiết bị ngoại vi)
- Tín hiệu ngắt: phát ra từ module I/O gửi đến CPU



# Các dạng dữ liệu đến/đi từ các thành phần máy tính

## ■ CPU

- Lệnh: truy xuất từ bộ nhớ
- Dữ liệu đến hoặc đi từ bộ nhớ hoặc I/O
- Tín hiệu ngắt: do I/O module gửi tới
- Tín hiệu địa chỉ: định vị một ô nhớ hoặc một tb ngoại vi
- Tín hiệu điều khiển



# Một số cấu trúc kết nối

- Cấu trúc kết nối phổ biến nhất:
  - Cấu trúc bus (cấu trúc đơn bus và cấu trúc đa bus)
- Cấu trúc kết nối điểm – điểm: QPI và PCI Express

# Nội dung chính

## Phần I. Tổng quan về máy tính

3.1 Các thành phần của máy tính

3.2 Hoạt động của máy tính

## Phần II. Hệ thống kết nối

3.3 Cấu trúc kết nối

**3.4 Hệ thống bus**

3.5 Kết nối điểm-điểm

3.6 PCI Express

## 3.4 Hệ thống Bus

- Giới thiệu sơ lược về bus
- Bus hệ thống
- Các yếu tố trong thiết kế bus

# 1. Giới thiệu sơ lược về bus

## ■ Bus là gì?

- là tập hợp các đường kết nối dùng để vận chuyển thông tin giữa các module của máy tính
- Là đường truyền chia sẻ: Tín hiệu truyền bởi 1 thiết bị bất kì có thể được nhận bởi tất cả các thiết bị khác kết nối với bus đó
  - Nếu 2 thiết bị cùng truyền 1 lúc, tín hiệu của chúng sẽ bị chồng nhau và bị méo

## ■ **Độ rộng bus:** là số đường dây của bus có thể truyền các bit thông tin đồng thời

## ■ Hệ thống máy tính có một số loại bus khác nhau:

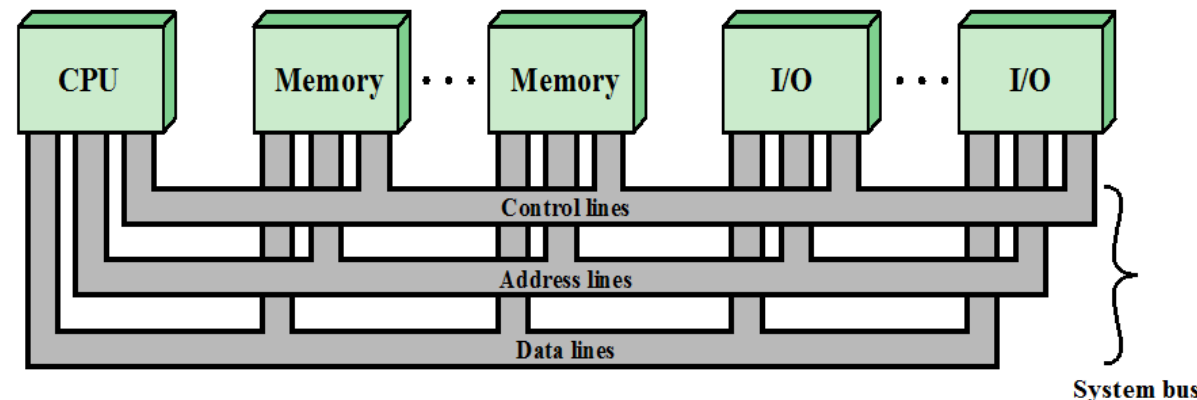
- Theo tín hiệu truyền trên bus:
  - Bus địa chỉ (Address bus)
  - Bus dữ liệu (Data bus)
  - Bus điều khiển (Control bus)
- Theo nghi thức truyền thông:
  - Bus đồng bộ (Synchronous bus)
  - Bus không đồng bộ (Asynchronous bus)
- Theo vị trí của bus:
  - Bus kết nối các thành phần bên trong bộ xử lý: Internal bus
  - Bus kết nối các thành phần chính của máy tính (bộ xử lý, bộ nhớ, I/O): system bus



## 2. Bus hệ thống

### ■ Bus hệ thống (System bus):

- Kết nối các thành phần chính của máy tính (bộ xử lý, bộ nhớ, module I/O)
- Gồm 50 đến 100 đường:
  - *Data bus (bus dữ liệu):* gồm các đường truyền dữ liệu
  - *Address bus (bus địa chỉ):* gồm các đường địa chỉ
  - *Control bus (bus điều khiển):* các đường truyền tín hiệu điều khiển

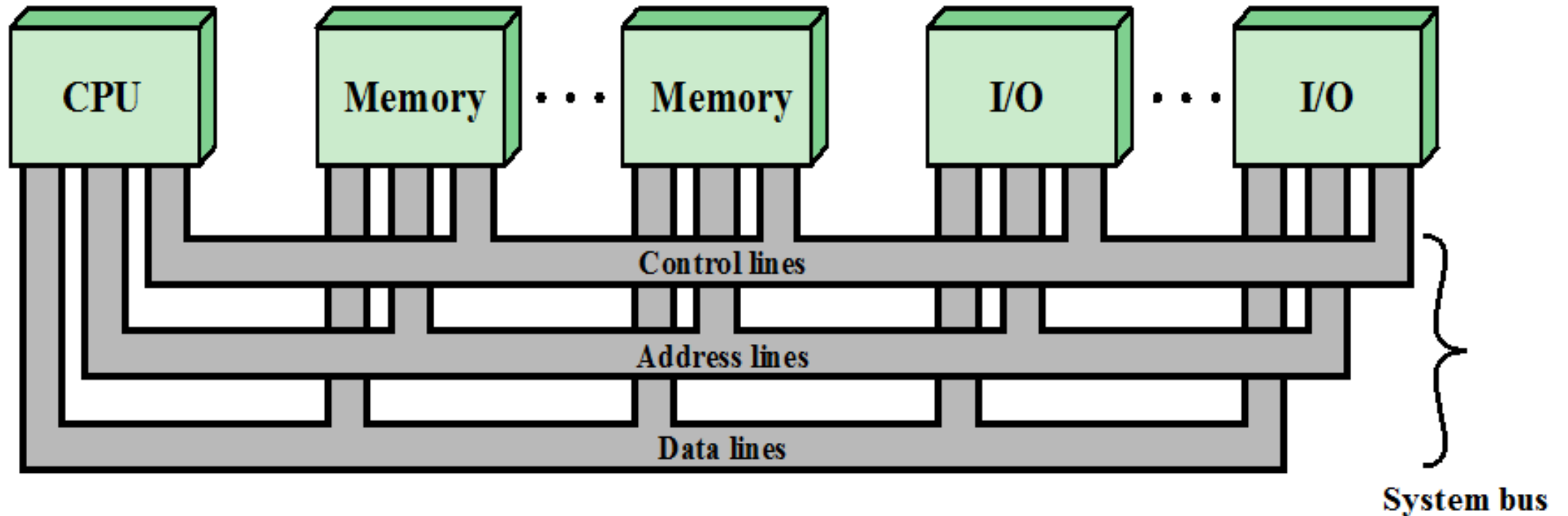


# Bus hệ thống (System bus)

- Cấu trúc bus là cách mà CPU và các bộ phận của máy tính liên kết với nhau qua bus.
- Hai kiểu cấu trúc bus:
  - Cấu trúc đơn bus
  - Cấu trúc đa bus

# Cấu trúc đơn bus

- Tất cả các module kết nối chung vào system bus



# Bus dữ liệu (*data bus*)

- Chức năng: truyền dữ liệu giữa các module hệ thống
- Độ rộng bus dữ liệu: Xác định số bit dữ liệu có thể được trao đổi đồng thời
- Data bus bao gồm 8, 16, 32, 64, 128 đường hay nhiều hơn
- Độ rộng bus dữ liệu là yếu tố chính quyết định hiệu suất toàn hệ thống

# Bus địa chỉ (*Address bus*)

- Được sử dụng để xác định địa chỉ của ngăn nhớ hoặc cổng vào/ra (I/O port)
  - Ví dụ: Nếu bộ xử lý muốn đọc 1 từ ở bộ nhớ, nó sẽ đặt địa chỉ của từ đó lên đường bus địa chỉ.
- Độ rộng bus địa chỉ xác định dung lượng nhớ tối đa của bộ nhớ:
  - $n$  bit:  $A_{n-1}, A_{n-2}, \dots, A_2, A_1, A_0$   
→ có thể đánh địa chỉ tối đa  $2^n$  ngăn nhớ

Ví dụ: Bus địa chỉ 20 bit, ngăn nhớ tổ chức theo byte → số địa chỉ ngăn nhớ tối đa  $2^{20}$   
→ Dung lượng nhớ tối đa mà bộ xử lý quản lý được: 1MB
- Các bit cao được sử dụng để lựa chọn module cụ thể trên bus còn bit thấp dùng để chọn vị trí của ngăn nhớ trong bộ nhớ hoặc cổng vào/ra trong I/O module.

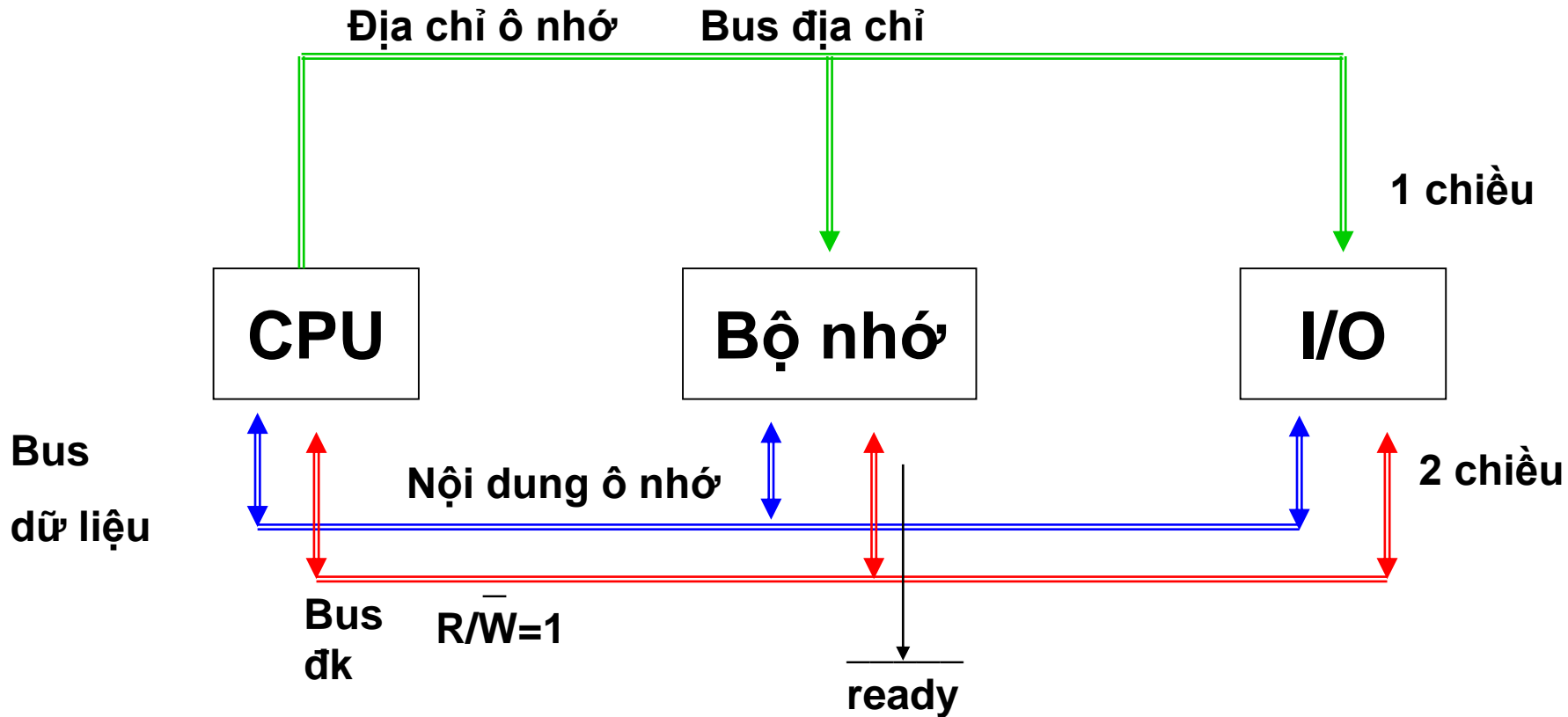
# Bus điều khiển (*Control bus*)

- Được sử dụng để vận chuyển các tín hiệu điều khiển.
  - Các tín hiệu điều khiển đọc/ghi
  - Các tín hiệu điều khiển ngắt
  - Các tín hiệu điều khiển bus

# Hoạt động của bus hệ thống

- 4 hoạt động (tác vụ) trên bus:
  - Đọc bộ nhớ (memory Read)
  - Ghi bộ nhớ (memory Write)
  - Đọc I/O (I/O Read)
  - Ghi I/O (I/O Write)
- Mỗi tác vụ trên bus diễn ra trong một số nguyên lần chu kỳ xung nhịp.
- Khoảng thời gian ngắn nhất để lặp lại tác vụ gọi là chu kỳ bus.
- Các sự kiện trong một chu kỳ bus:
  - Bước 1: chiếm quyền sử dụng bus
  - Bước 2: gửi địa chỉ qua bus địa chỉ, (gửi dữ liệu lên bus dữ liệu) và yêu cầu qua bus điều khiển
  - Bước 3 : truyền dữ liệu từ địa chỉ đã được định vị

# Ví dụ: đọc dữ liệu từ Bộ nhớ vào VXL





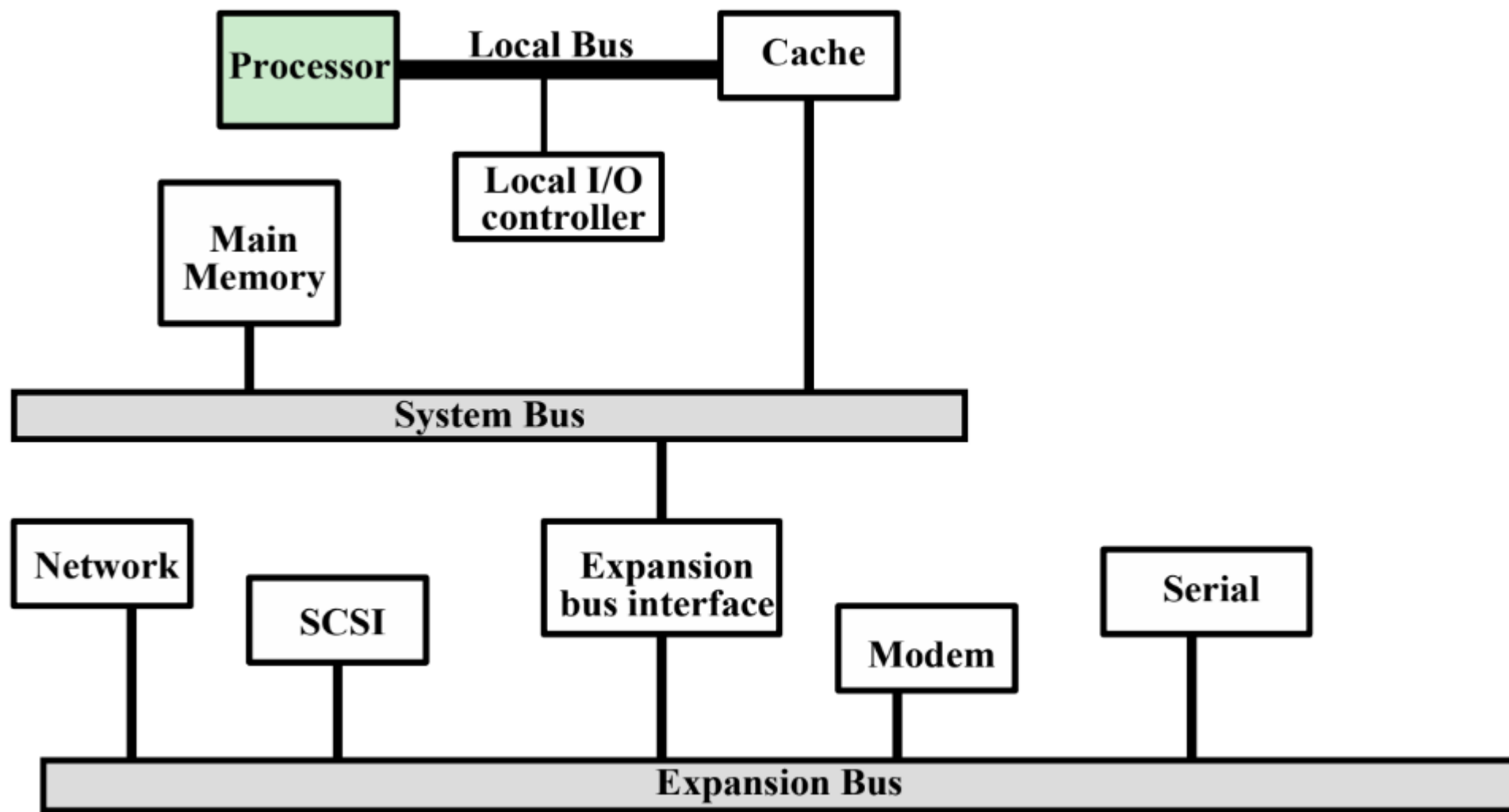
## **b. Cấu trúc đa bus**

- Các vấn đề xảy ra với cấu trúc đơn bus:
  - Nếu một số lượng lớn các thiết bị được kết nối với bus sẽ xảy ra vấn đề sau:
    1. Nhiều thiết bị gắn vào bus, chiều dài bus càng dài và do đó trễ truyền càng lớn.
    2. Hiện tượng nút cổ chai: lượng dữ liệu cần truyền quá lớn so với khả năng của bus.=> hiệu suất hệ thống sẽ giảm
- Khắc phục nhược điểm trên: Cấu trúc đa bus

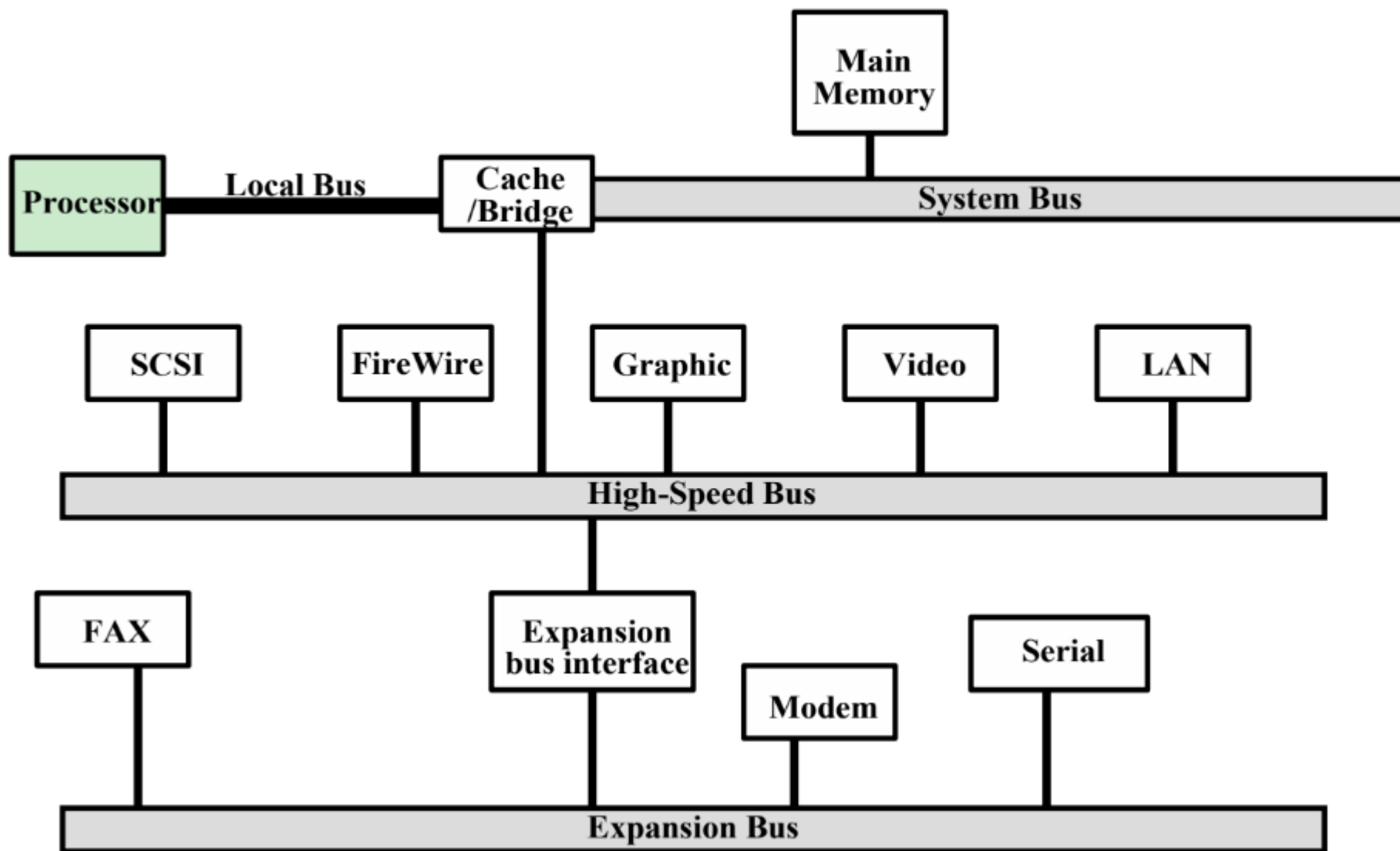
## **b. Cấu trúc đa bus**

- Phân cấp bus: Tổ chức thành nhiều bus trong hệ thống máy tính
  - Cho các thành phần khác nhau:
    - Bus của bộ xử lý
    - Bus của bộ nhớ chính
    - Các bus vào-ra
  - Các bus khác nhau về tốc độ
- Bus bộ nhớ chính và các bus vào-ra không phụ thuộc vào bộ xử lý.

# Cấu trúc đa bus truyền thống (Có cache)



# Cấu trúc đa bus hiệu suất cao



## c. Các yếu tố trong thiết kế Bus

### 1. Loại bus

Chuyên dụng

Ghép kênh

### 2. Phương pháp trọng tài

Tập trung

Phân tán

### 3. Định thời

Đồng bộ

Bất đồng bộ

### 4. Độ rộng bus

Địa chỉ

Dữ liệu

### 5. Loại truyền dữ liệu

Đọc

Ghi

Đọc thay đổi ghi

Đọc sau khi ghi

Khởi

# 1. Các loại bus: chuyên dụng và ghép kênh.

- Bus chuyên dụng sử dụng cho một chức năng cụ thể: vd: bus dữ liệu, bus địa chỉ, bus điều khiển
  - Ưu điểm: nhanh hơn, ít có xung đột bus
  - Nhược điểm: tăng kích thước và chi phí
- Bus ghép kênh: các thông tin (dữ liệu, địa chỉ) được truyền trên cùng một đường.
  - Sử dụng đường điều khiển AV (Address Valid control line).
  - Khi bắt đầu, đ/c được đưa vào bus và đường AV được kích hoạt.
  - Các module nhận địa chỉ, kiểm tra xem có phải đ/c của nó không.
  - Thông tin đ/c được loại bỏ và một kênh truyền được thiết lập để truyền dữ liệu đọc hoặc ghi
  - Ưu điểm: ít đường hơn, tiết kiệm không gian và chi phí
  - Nhược điểm: mạch phức tạp hơn

## 2. Phương pháp phân xử (trọng tài)

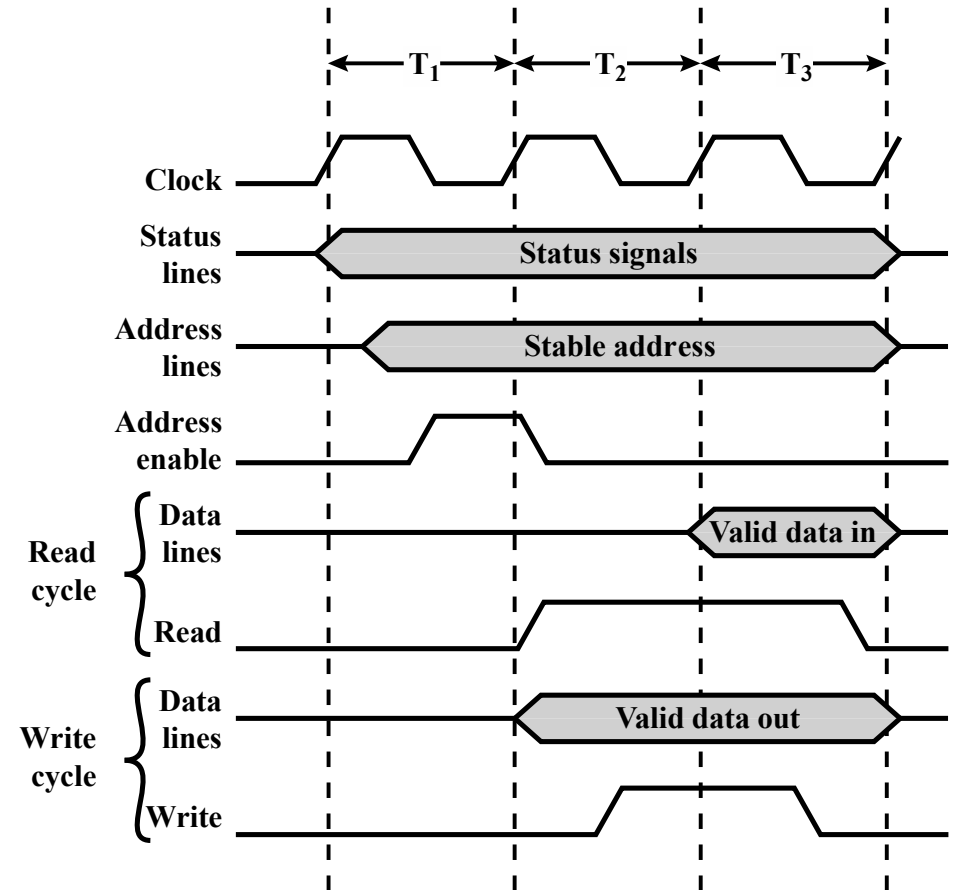
- Đôi khi, tại một thời điểm có nhiều module cần chiếm bus → cần quyết định xem module nào có quyền sử dụng bus: ***phân xử (trọng tài) bus***

→ Phương pháp phân xử bus: tập trung và phân tán

- ❑ Phân xử tập trung: bộ điều khiển (bộ phân xử) phân bổ thời gian trên bus. Bộ điều khiển này có thể là một thiết bị riêng hoặc một phần của bộ XL
- ❑ Phân xử phân tán: mỗi module chứa một access control logic và chúng làm việc cùng nhau để chia sẻ đường truyền

# 3. Định thời

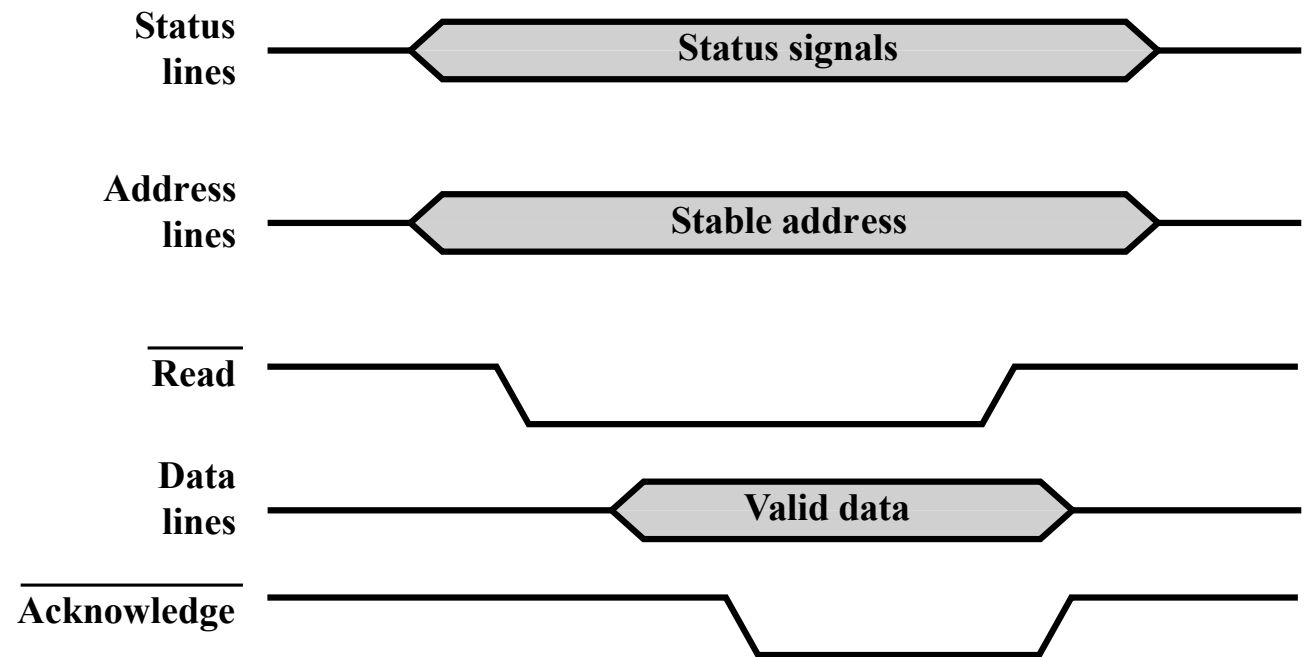
- **Định thời** là cách các sự kiện được phối hợp truyền trên bus
- Hai loại: đồng bộ và không đồng bộ
- Định thời đồng bộ:
  - Mỗi hoạt động truyền trên bus được thực hiện theo các xung đồng hồ
  - Bus chứa một đường xung đồng hồ (clock line) truyền liên tiếp một chuỗi các bit 0, 1
  - Khoảng thời gian T được gọi là chu kỳ đồng hồ
  - Tất cả các thiết bị trong máy tính đều đọc được và đồng bộ các hoạt động truyền theo xung này



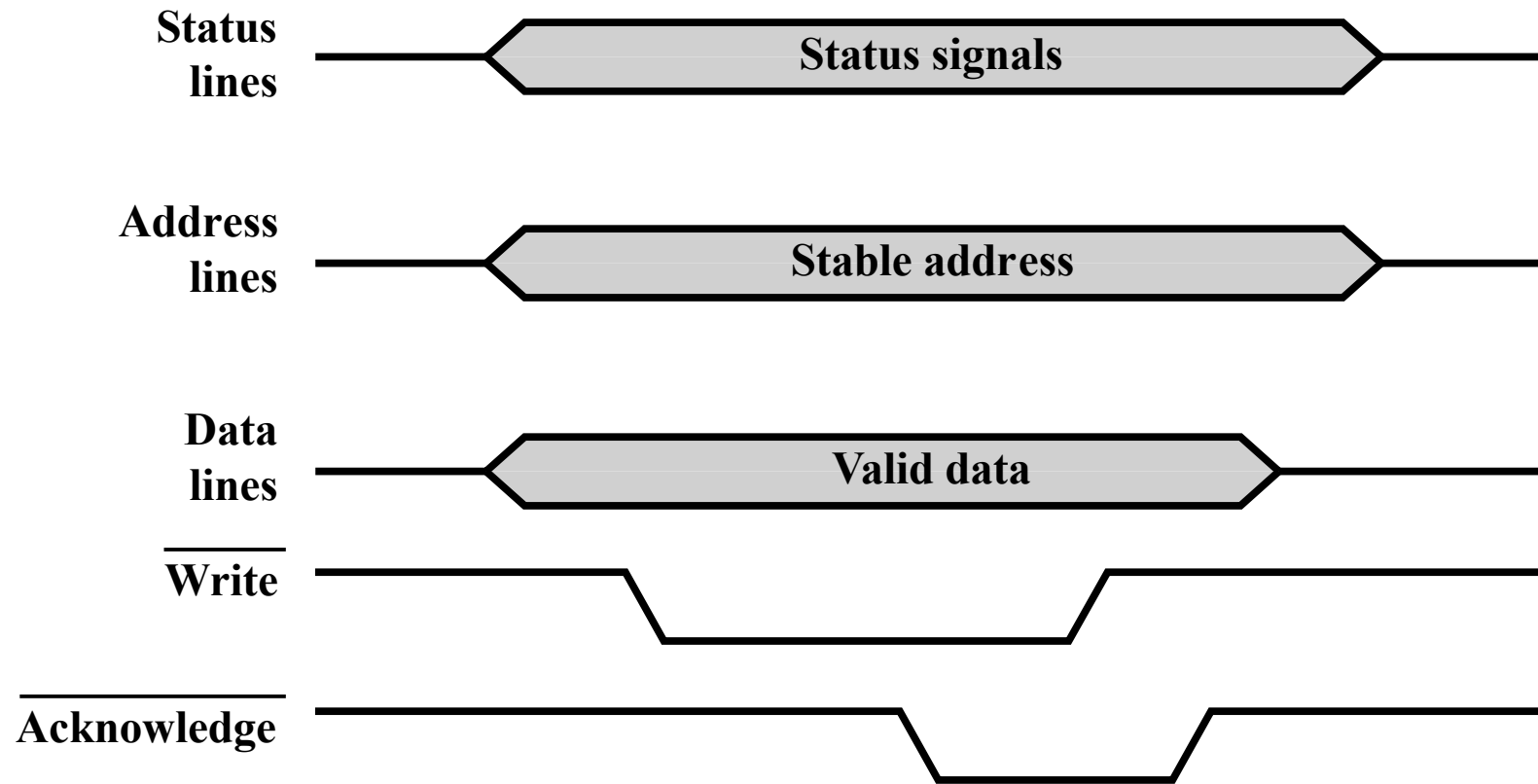


# 3. Định thời (tiếp)

- Định thời không đồng bộ
  - Không sử dụng tín hiệu đồng hồ.
  - Sau khi dữ liệu được đưa vào bus, bộ nhớ gửi một tín hiệu ACK để báo cho VXL biết việc đọc hoặc ghi dữ liệu
- Truyền đồng bộ: thực hiện đơn giản tuy nhiên ít linh hoạt hơn truyền không đồng bộ
- Việc truyền theo xung đồng hồ đôi khi làm giảm hiệu suất hệ thống
- Truyền không đồng bộ: hiệu quả hơn trong trường hợp nhiều thiết bị có tốc độ xử lý khác nhau chia sẻ chung bus



(a) System bus read cycle



(b) System bus write cycle

# Nội dung chính

## Phần I. Tổng quan về máy tính

3.1 Các thành phần của máy tính

3.2 Hoạt động của máy tính

## Phần II. Hệ thống kết nối

3.3 Cấu trúc kết nối

3.4 Hệ thống bus

**3.5 Kết nối điểm-điểm**

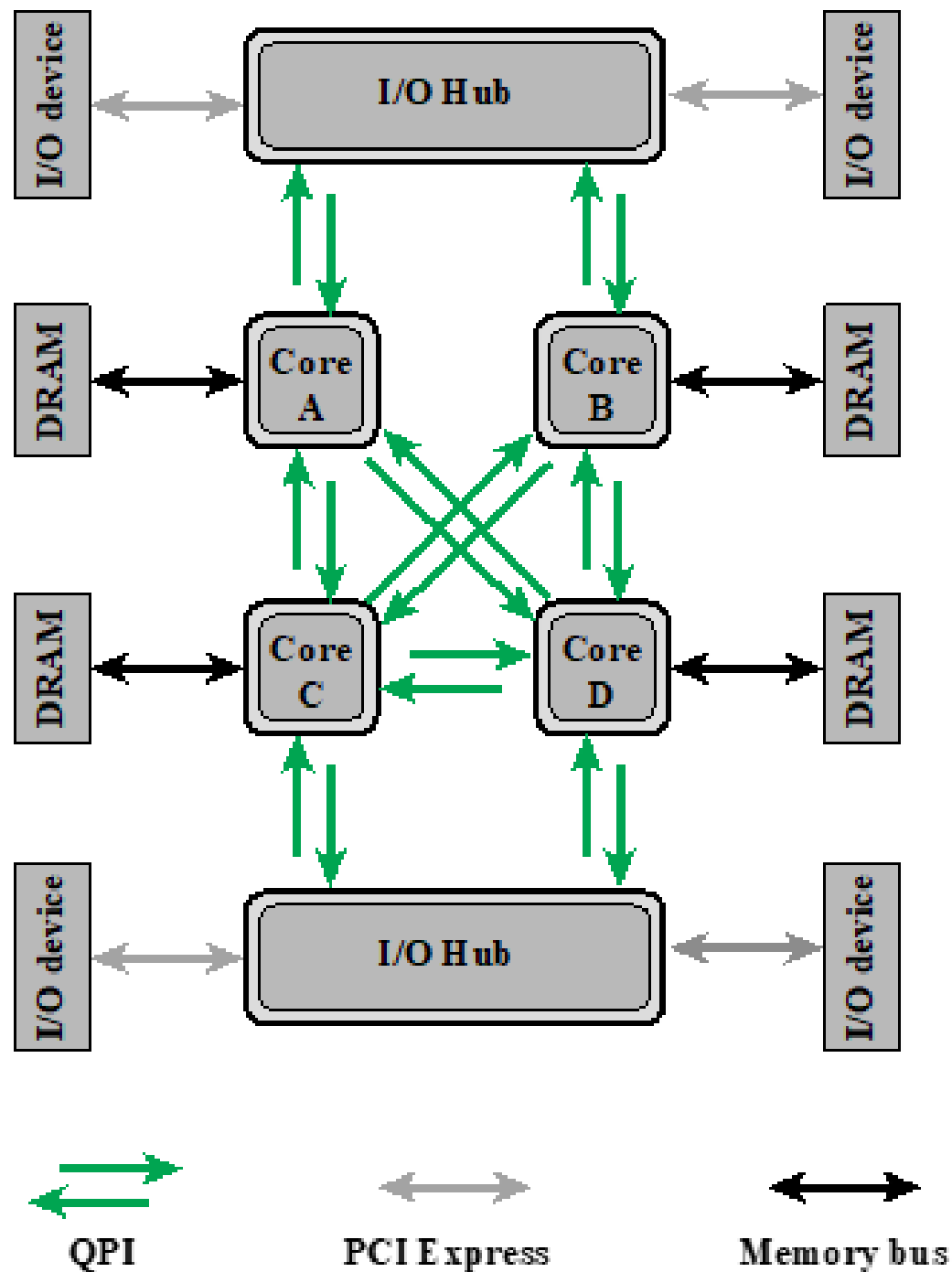
3.6 PCI Express

## 3.5. Kết nối điểm - điểm

- **Nhược điểm của hệ thống kết nối bus:**
  - Tốc độ của bus đồng bộ không cao do khó khăn trong việc tăng tần số tín hiệu đồng hồ.
  - Khi tốc độ dữ liệu cao, việc thực hiện các chức năng đồng bộ và phân xử bus một cách kịp thời trở nên khó khăn hơn
  - Với chip đa nhân, nếu sử dụng bus để kết nối, trao đổi dữ liệu giữa các nhân sẽ không đáp ứng được tốc độ vi xử lý
- Giải pháp: **Kết nối điểm-điểm**: có độ trễ thấp, tốc độ dữ liệu cao, và khả năng mở rộng tốt hơn.
- 2 loại kết nối điểm – điểm: QPI và PCIe

# Đường dẫn nhanh (Quick Path Interconnect)

- Được giới thiệu vào năm 2008
- 3 đặc điểm:
  - **Nhiều kết nối trực tiếp**
    - Các thành phần được kết nối trực tiếp với nhau, giúp loại bỏ cơ chế phân xử như trong mô hình kết nối bus.
  - **Kiến trúc giao thức phân lớp**
    - Các kết nối sử dụng kiến trúc giao thức phân lớp chứ không sử dụng tín hiệu điều khiển như trong mô hình bus chia sẻ.
  - **Truyền dữ liệu gói**
    - Dữ liệu được gửi thành 1 chuỗi các gói tin chứa tiêu đề điều khiển (header) và mã kiểm soát lỗi.

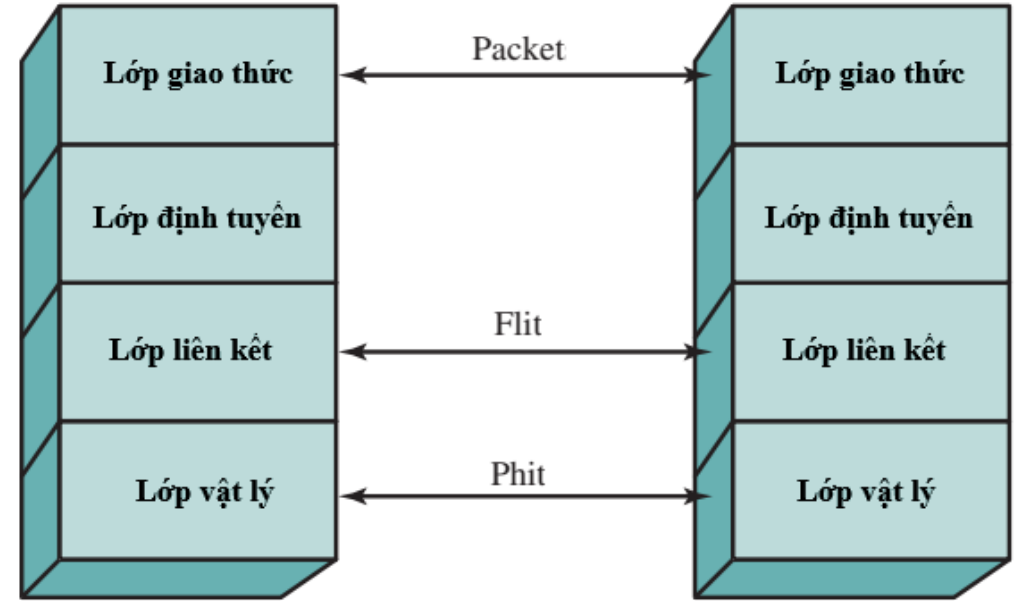


## Cấu hình chip đa nhân sử dụng QPI

# Các lớp QPI

- QPI được định nghĩa là một kiến trúc bốn lớp, bao gồm các lớp sau:

- **Vật lý:** Bao gồm dây dẫn mang tín hiệu, cũng như mạch và logic để hỗ trợ các tính năng truyền và nhận các bit 1 và 0. Đơn vị chuyển giao ở lớp này 20 bit, được gọi là Phit (physical unit).
- **Liên kết:** Chịu trách nhiệm truyền tin cậy và điều khiển luồng. Đơn vị dữ liệu của lớp Liên kết là một Flit 80-bit (flow control unit)
- **Định tuyến:** Được sử dụng để xác định đường đi mà một gói sẽ đi qua các kết nối hệ thống có sẵn
- **Giao thức:** Bộ quy tắc để trao đổi các gói tin dữ liệu (packet) giữa các thiết bị. Một gói bao gồm một số không đổi các Flit.



# Nội dung chính

## Phần I. Tổng quan về máy tính

3.1 Các thành phần của máy tính

3.2 Hoạt động của máy tính

## Phần II. Hệ thống kết nối

3.3 Cấu trúc kết nối

3.4 Hệ thống bus

3.5 Kết nối điểm-điểm

**3.6 PCI Express**

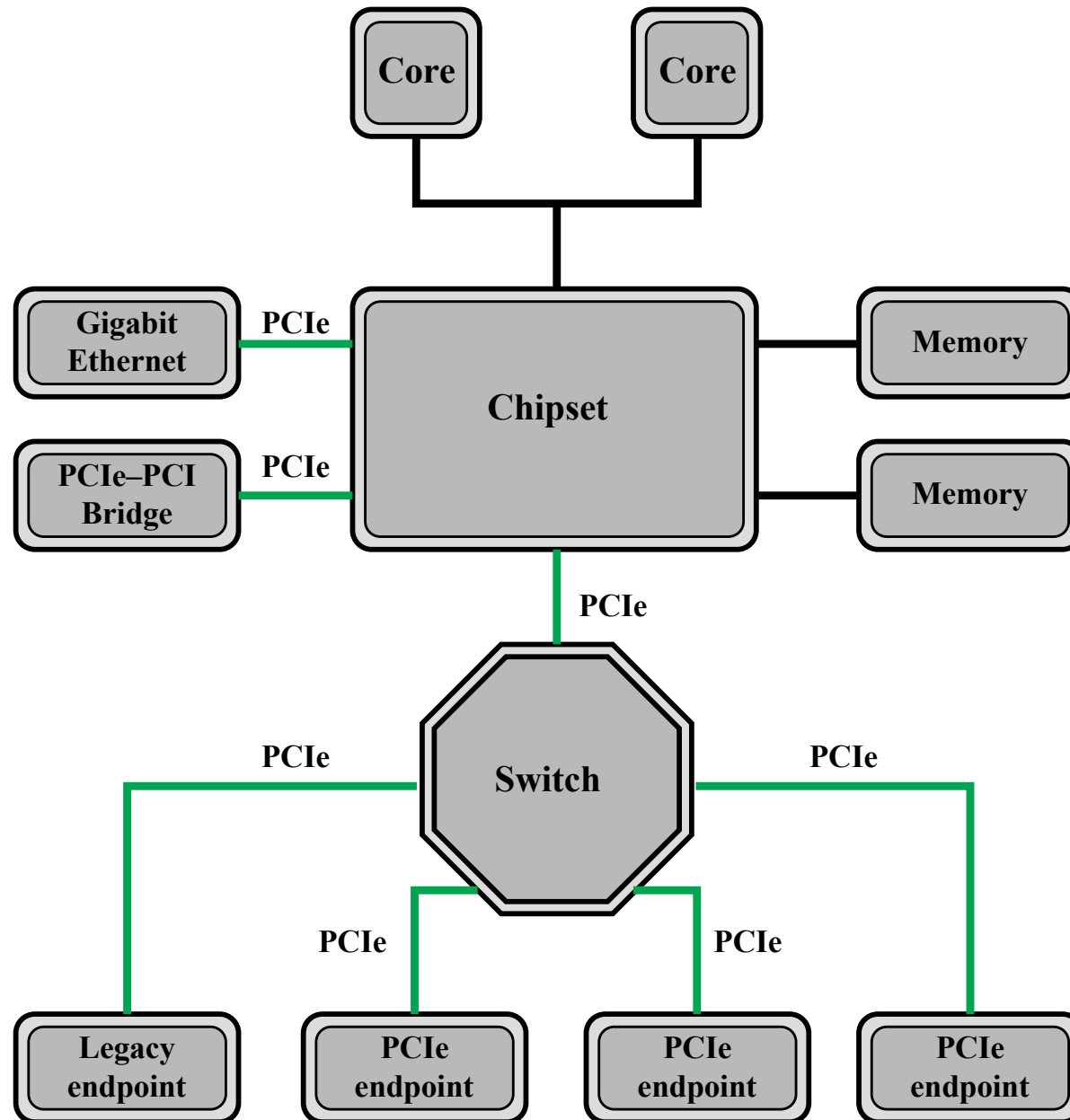


## 3.6. Kết nối thiết bị ngoại vi

### Peripheral Component Interconnect (PCI)

- Là bus tốc độ cao, độc lập với bộ xử lý, có thể hoạt động như một bus ngoại vi
- PCI cung cấp hiệu suất cao hơn cho các thiết bị I/O tốc độ cao (ví dụ: card mạng, card màn hình, bộ điều khiển ổ cứng)
- PCI Express (PCIe)
  - Cơ chế kết nối điểm-điểm nhằm thay thế cơ chế dựa trên bus như PCI
  - Yêu cầu chính với PCIe:
    - dung lượng cao để hỗ trợ nhu cầu của thiết bị I/O tốc độ dữ liệu cao hơn, như Gigabit Ethernet
    - hỗ trợ các luồng dữ liệu phụ thuộc thời gian (thời gian thực)

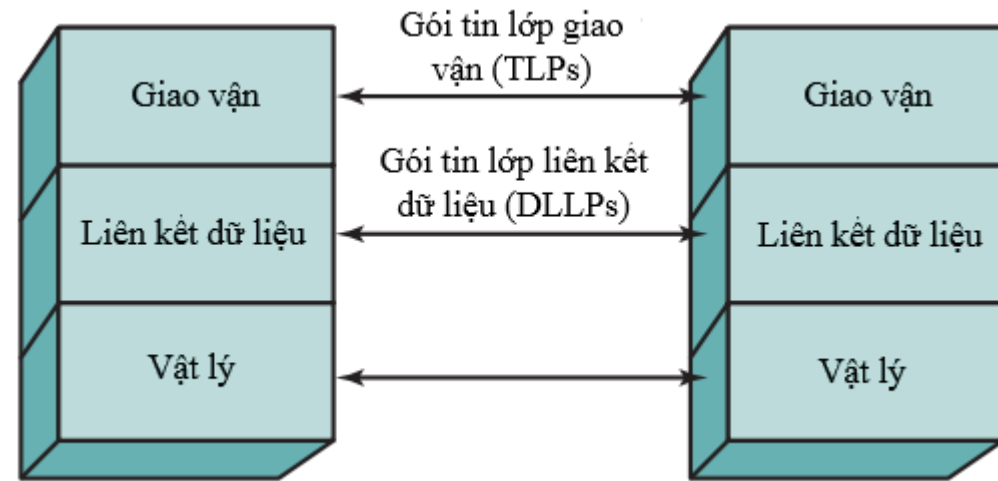
# Cấu hình PCIe



# Các lớp giao thức PCIe

## ■ Kiến trúc giao thức PCIe bao gồm các lớp sau:

- a. **Vật lý (Physical):** Bao gồm các dây dẫn thực tế mang tín hiệu, cũng như mạch và logic để hỗ trợ các tính năng cần thiết trong việc truyền và nhận các bit 1 và 0.
- b. **Liên kết dữ liệu (Data link layer – DLL):** Chịu trách nhiệm truyền tin cậy và điều khiển luồng. Các gói dữ liệu được tạo ra và được xử lý bởi DLL được gọi là gói DLLP.
- c. **Giao vận (Transaction Layer):** Tạo ra và xử lý các gói dữ liệu được sử dụng để thực hiện các cơ chế truyền dữ liệu được tải/lưu trữ và cũng quản lý điều khiển luồng của các gói tin giữa hai thiết bị. Các gói dữ liệu của lớp này được gọi là gói TLP.



# Tổng kết Chương 3

- Thành phần máy tính
- Chức năng máy tính
  - Thực hiện chương trình
  - Ngắt
  - Chức năng I/O
- Cấu trúc kết nối
- Kết nối bus
  - Cấu trúc đơn bus
  - Cấu trúc đa bus
  - Các yếu tố thiết kế bus
- Kết nối điểm-điểm
  - Lớp vật lý QPI
  - Lớp liên kết QPI
  - Lớp định tuyến QPI
  - Lớp giao thức QPI
- PCI Express
  - Kiến trúc vật lý và logic PCI
  - Lớp vật lý PCIe
  - Lớp liên kết dữ liệu PCIe
  - Lớp giao vận PCIe

# Câu hỏi ôn tập

1. Liệt kê các nhóm chức năng (function) chung được xác định bởi các lệnh máy tính.
2. Liệt kê và mô tả ngắn gọn các nhóm hoạt động (action) mà bộ xử lý có thể thực hiện.
3. Liệt kê và mô tả ngắn gọn các trạng thái có thể để thực hiện một lệnh.
4. Liệt kê và định nghĩa ngắn gọn hai phương pháp xử lý nhiễu.
5. Liệt kê các loại thông tin đầu vào và đầu ra của CPU, I/O module và bộ nhớ (memory).
6. Những dạng truyền nào phải có hỗ trợ của cấu trúc kết nối?
7. Các đặc điểm chính của kết nối bus (Bus interconnection) là gì?
8. Chức năng và đặc điểm của bus dữ liệu (Data Bus).
9. Chức năng và đặc điểm của bus địa chỉ (Address Bus).
10. Chức năng và đặc điểm của bus điều khiển (Control Bus)

# Câu hỏi ôn tập

11. Các ưu điểm của cấu trúc đa bus so với cấu trúc đơn bus là gì?
12. Các đặc điểm của kết nối điểm – điểm là gì?
13. Liệt kê các lớp của QPI.
14. Liệt kê các lớp của PCIe.

Hình ảnh và nội dung trong bài giảng này tham khảo từ cuốn sách và slide bài giảng “Computer Organization and Architecture”, 10th Edition, của tác giả William Stallings.