

NGUYÊN LÝ HỆ ĐIỀU HÀNH

Giảng viên: TS. Đoàn Thị Quế

Bộ môn: Mạng và an toàn thông tin

Giới thiệu môn học

- Tên môn học: Nguyên lý hệ điều hành
- Mã môn học: CSE 353
- Phân loại môn học: Môn bắt buộc.
- Số tín chỉ: 3 (LT: 45; BT: 0; TN: 0)
- Mục tiêu môn học cung cấp kiến thức:
 - Các khái niệm cơ bản về hệ điều hành
 - Nguyên lý chung về hệ điều hành

Giới thiệu môn học (tiếp)

■ Tài liệu môn học:

- Giáo trình

- [1] Bộ môn Kỹ thuật máy tính và mạng (dịch). *Các hệ điều hành hiện đại – Tập 1*. Đại học Thủy Lợi, 2015.

- Các tài liệu tham khảo

- [2] A.S. Tanenbaum and H. Bos. *Modern Operating Systems*. 4th Edition, Pearson, 2015.

- [3] A. Silberschatz, P.B. Galvin and G. Gagne. *Operating System Concepts*. 9th Edition, Wiley, 2012.

- Slide bài giảng

Giới thiệu môn học (tiếp)

■ Kiểm tra và đánh giá

- Điểm quá trình: 50%
 - ◆ Điểm chuyên cần: 10%
 - Nghỉ 1 buổi trừ 1 điểm, đi muộn 1 buổi trừ 0.5 điểm
 - ◆ Điểm đánh giá giữa kỳ: 40%
 - ◆ Điểm thưởng cho chữa bài, ... (cộng vào điểm bài kiểm tra)
- Điểm thi kết thúc học phần: 50%
 - ◆ Thi viết, 60 phút
- Điều kiện dự thi kết thúc học phần: Theo Quy chế đào tạo của trường
 - ◆ Có mặt ở lớp từ 80% trở lên thời gian qui định cho học phần
 - ◆ Dự ít nhất 50% số lần kiểm tra theo qui định và đạt trung bình các bài kiểm tra từ 4 trở lên

Nội dung môn học



Chương 1: Giới thiệu chung

Chương 2: Tiến trình và luồng

Chương 3: Sự bế tắc

Chương 4: Quản lý bộ nhớ

Chương 5: Quản lý vào/ra

Chương 6: Hệ thống file

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Các thành phần của hệ thống máy tính

1.2. Tổ chức của máy tính

1.3. Khái niệm hệ điều hành

1.4. Lịch sử phát triển hệ điều hành

1.5. Phân loại hệ điều hành

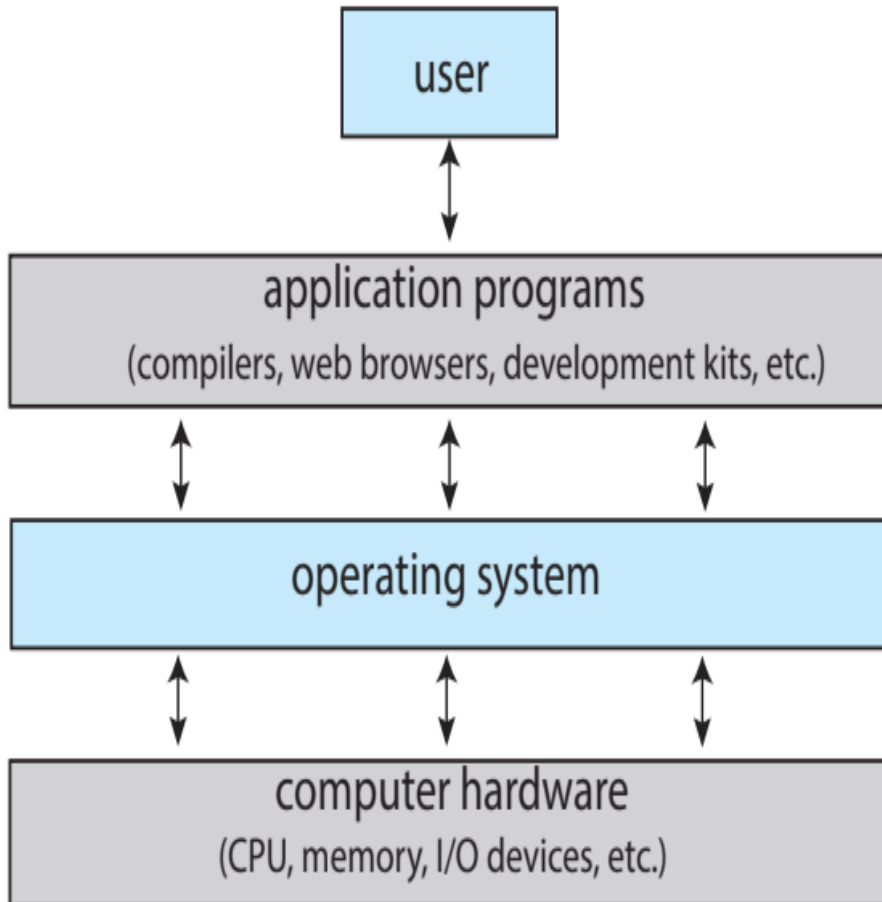
1.6 Hoạt động của hệ điều hành

1.7 Các dịch vụ của hệ điều hành

1.8 Các khái niệm trong hệ điều hành

1.9 Cấu trúc của hệ điều hành

1.1 – Các thành phần của hệ thống máy tính



- **Phần cứng (Hardware):** Cung cấp tài nguyên tính toán và lưu trữ (CPU, bộ nhớ, thiết bị vào/ra).
- **Hệ điều hành (Operating system):** điều khiển và phối hợp việc sử dụng phần cứng cho những ứng dụng khác nhau của người sử dụng.
- **Chương trình ứng dụng (Application programs):** sử dụng tài nguyên của máy tính để giải quyết các yêu cầu của người sử dụng (chương trình dịch, trình duyệt web, hệ cơ sở dữ liệu, chương trình soạn thảo, game, ...).
- **Người dùng (Users):** Người sử dụng máy tính (người sử dụng thuần túy và người viết các chương trình ứng dụng) hay máy tính khác.

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Các thành phần của hệ thống máy tính

1.2. Tổ chức của máy tính

1.3. Khái niệm hệ điều hành

1.4. Lịch sử phát triển hệ điều hành

1.5. Phân loại hệ điều hành

1.6 Hoạt động của hệ điều hành

1.7 Các dịch vụ của hệ điều hành

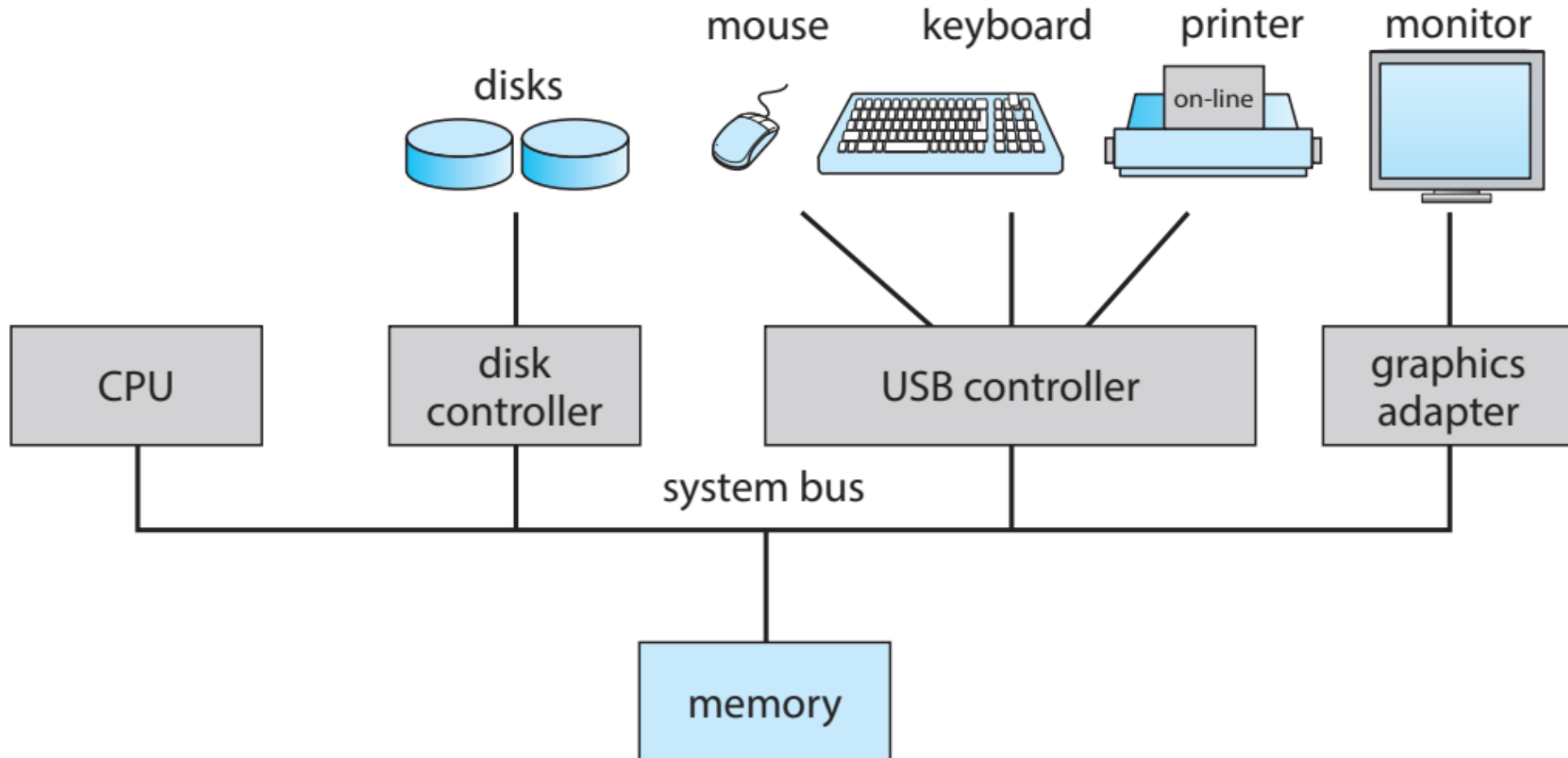
1.8 Các khái niệm trong hệ điều hành

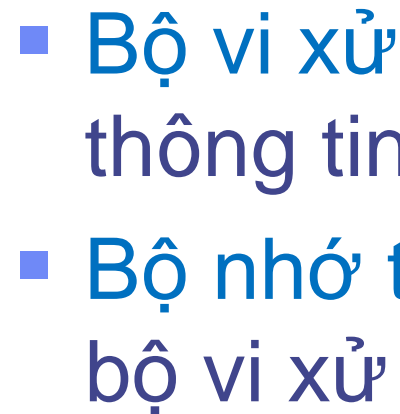
1.9 Cấu trúc của hệ điều hành

1.2-Tổ chức của máy tính

- Cấu trúc chung của hệ thống máy tính
- CPU và chu trình thực hiện lệnh
- Ngắt
- Bộ nhớ
- Vào/ra dữ liệu

1. Cấu trúc chung của hệ thống máy tính



- 
- **Bộ vi xử lý – CPU (Central Processing Unit):** xử lý các thông tin và điều khiển phần lớn hoạt động của máy tính.
 - **Bộ nhớ trong:** Là bộ nhớ có khả năng liên lạc trực tiếp với bộ vi xử lý, là nơi lưu trữ dữ liệu phục vụ cho quá trình xử lý.
 - **Bộ điều khiển vào ra:** Để điều khiển việc giao tiếp với thiết bị ngoại vi.
 - **Bus hệ thống:** vận chuyển thông tin của CPU, bộ nhớ trong và bộ điều khiển vào ra

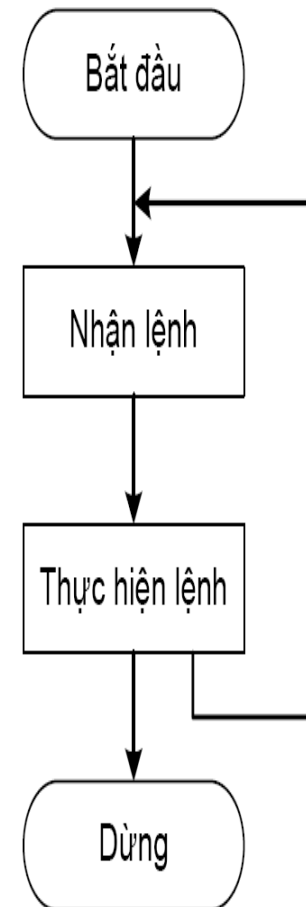
2. CPU và chu trình thực hiện lệnh

Các thành phần của CPU

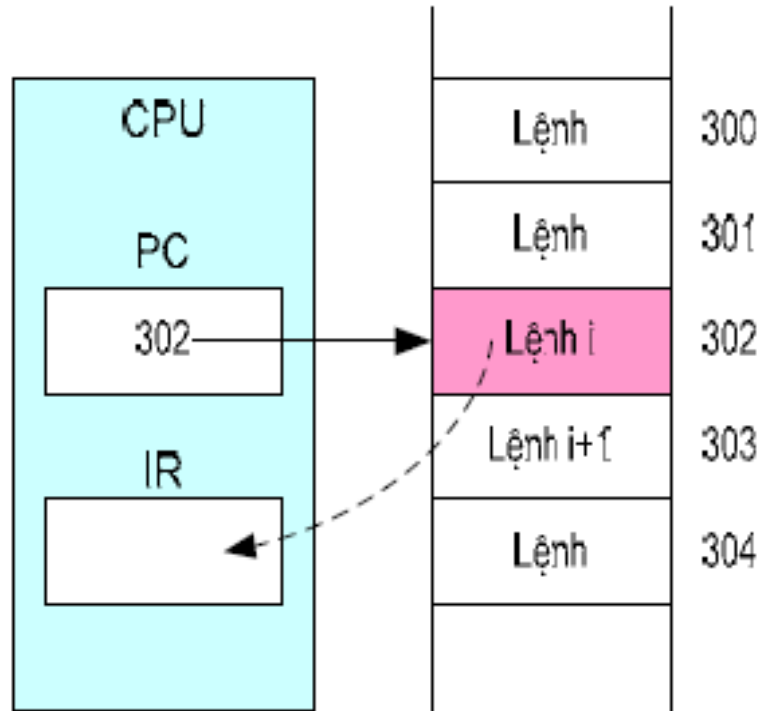
- Bộ điều khiển (Control Unit - CU): Điều khiển hoạt động của CPU và các hoạt động khác máy tính
- ALU (Arithmetic & Logic Unit): Khối số học và logic. Đây là nơi thực hiện các phép tính số học (cộng, trừ, nhân, chia) và các phép logic (Not, And, Or...).
- Các thanh ghi: Cung cấp khả năng nhớ bên trong CPU. Mỗi thanh ghi có khả năng chứa được một dãy các bit dữ liệu (độ dài còn phụ thuộc vào từng loại CPU).
- Hệ thống nối ghép bên trong CPU (Bus nội bộ): Cho phép liên lạc giữa các bộ phận bên trong CPU.

Chu trình thực hiện lệnh

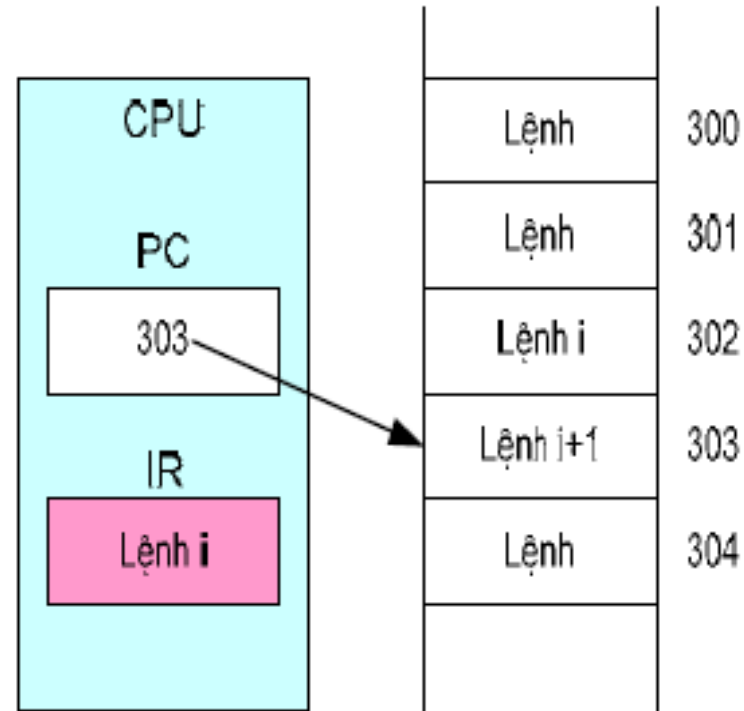
- Thực hiện chương trình gồm các chu trình lệnh:
 - Nhận lệnh (*Fetch Instruction – FI*)
 - Thực hiện lệnh (*Execute Instruction – EI*)
- Thực hiện chương trình bị dừng nếu thực hiện lệnh bị lỗi hoặc gặp lệnh dừng.



Nhận lệnh



Trước khi nhận Lệnh i



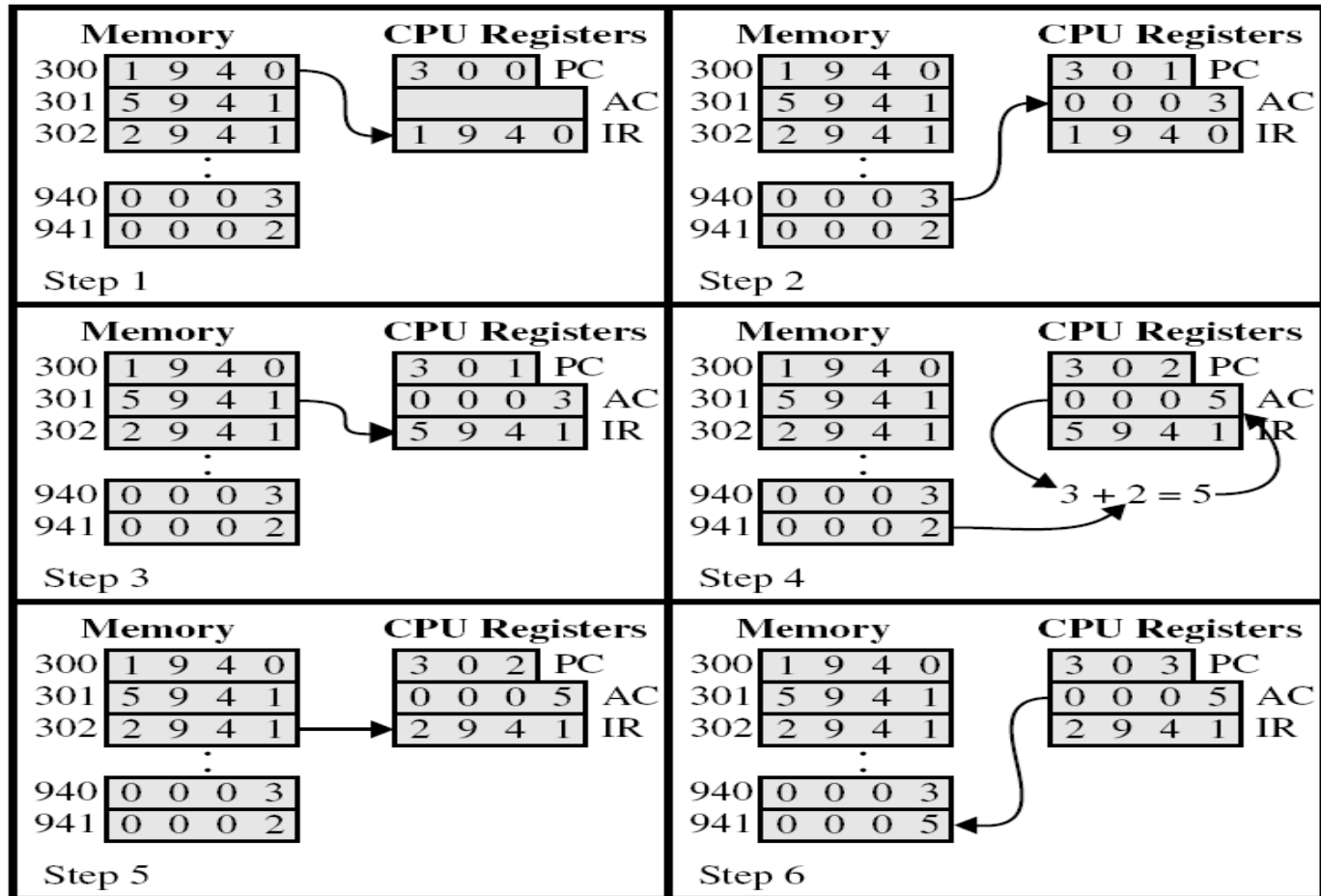
Sau khi nhận Lệnh i

PC-Program Counter
IR-Instruction Register

Thực hiện lệnh

- CPU giải mã lệnh đã được nhận
 - Các kiểu thao tác của lệnh:
 - ◆ Trao đổi dữ liệu giữa CPU và bộ nhớ chính
 - ◆ Trao đổi dữ liệu giữa CPU và module vào/ra
 - ◆ Xử lý dữ liệu: thực hiện các phép toán số học hoặc phép toán logic với các dữ liệu
 - ◆ Điều khiển
- Phát tín hiệu điều khiển thực hiện thao tác mà lệnh yêu cầu

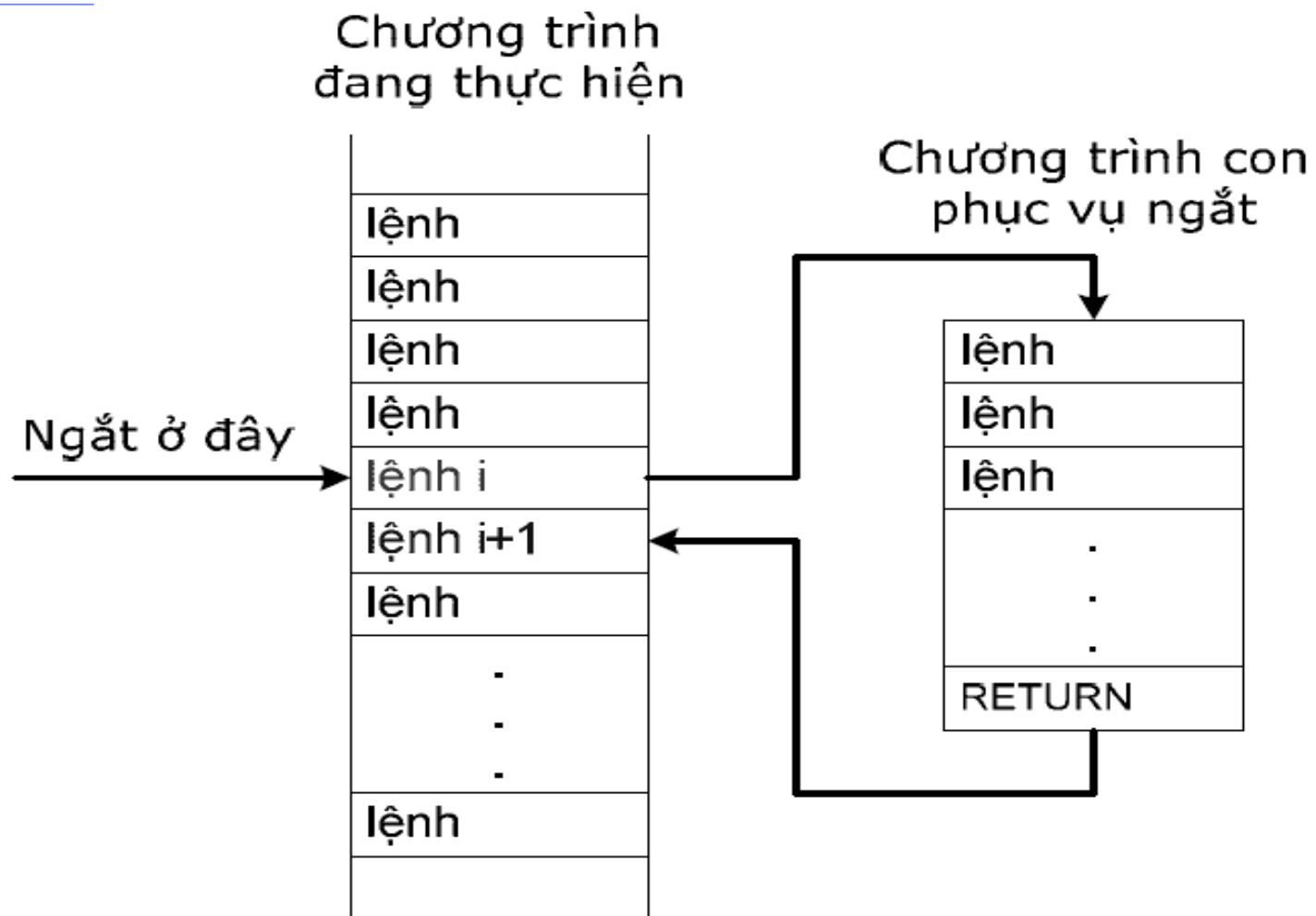
Ví dụ về thực hiện chương trình



3. Ngắt (Interrupt)

- **Khái niệm:** ngắt là cơ chế cho phép CPU tạm dừng chương trình đang thực hiện để chuyển sang thực hiện một chương trình khác, gọi là *chương trình con phục vụ ngắt*
- Các loại ngắt:
 - Ngắt cứng: Thiết bị phần cứng yêu cầu thực hiện ngắt bằng cách gửi tín hiệu qua bus.
 - Ngắt mềm: chương trình đang thực hiện, có yêu cầu ngắt bằng cách sử dụng lời gọi hệ thống (system call).
 - ◆ Ví dụ: khi cần ghi ra file, chương trình có thể gửi yêu cầu ngắt dưới dạng lời gọi hệ thống ghi ra file.

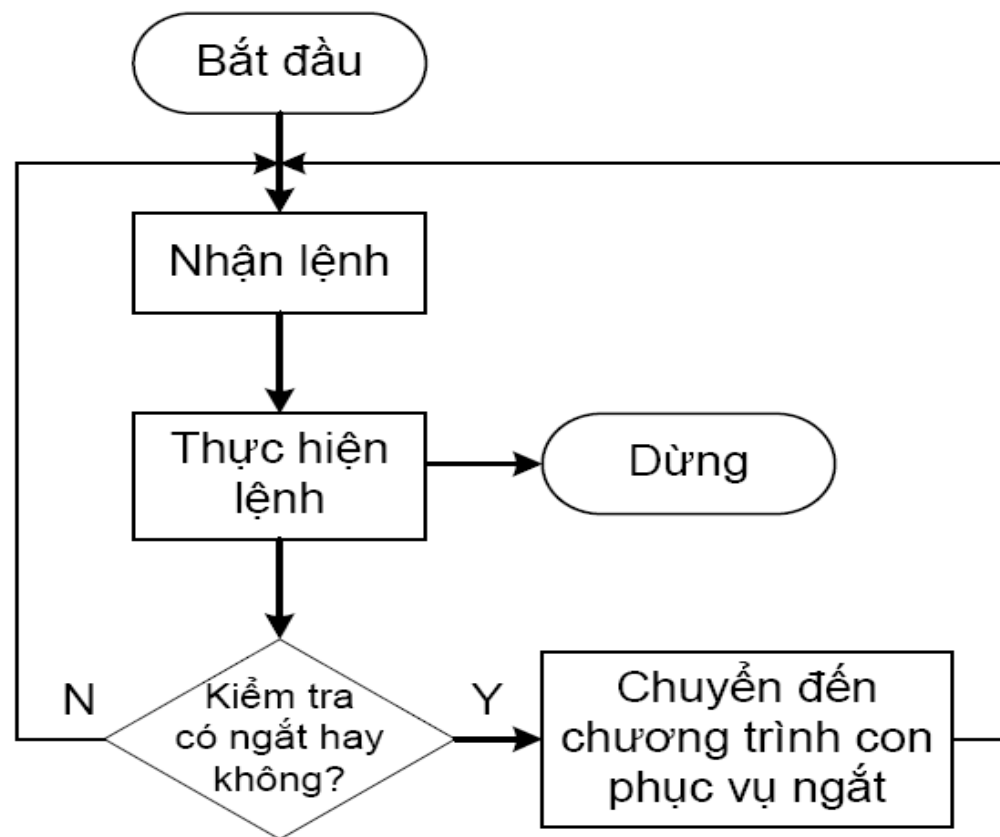
Xử lý ngắt



Xử lý ngắt (tiếp)

- Sau khi hoàn thành một lệnh, bộ xử lý kiểm tra ngắt
- Nếu không có ngắt → bộ xử lý nhận lệnh tiếp theo của chương trình hiện tại
- Nếu có tín hiệu ngắt:
 - Tạm dừng chương trình đang thực hiện
 - Cắt ngữ cảnh (các thông tin liên quan đến chương trình bị ngắt)
 - Thiết lập PC trở đến chương trình con phục vụ ngắt
 - Chuyển sang thực hiện chương trình con phục vụ ngắt
 - Cuối chương trình con phục vụ ngắt, khôi phục ngữ cảnh và tiếp tục chương trình đang bị tạm dừng.

Chu trình thực hiện lệnh có ngắt



4. Bộ nhớ

- Một hệ thống phân cấp bộ nhớ điển hình

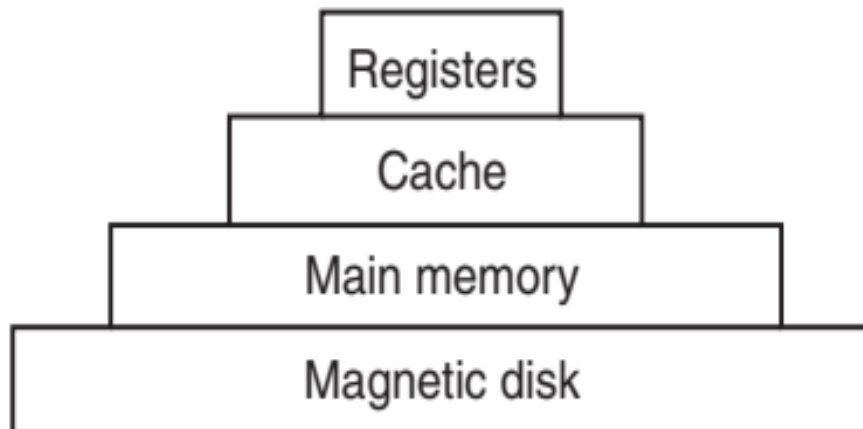
Typical access time

1 nsec

2 nsec

10 nsec

10 msec



Typical capacity

<1 KB

4 MB

1-8 GB

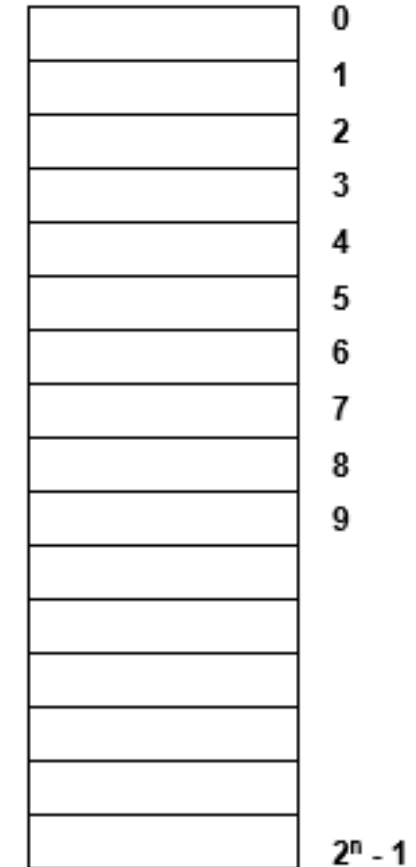
1-4 TB

Bộ nhớ chính

- **RAM (Random Access Memory):** Bộ nhớ truy cập ngẫu nhiên, có thể đọc và ghi dữ liệu lên đó. Dữ liệu trên RAM sẽ mất đi khi tắt máy.
- **ROM (Read Only Memory):** Bộ nhớ chỉ đọc, không thể thay đổi nội dung của nó. Khi tắt máy thì dữ liệu trên ROM vẫn được giữ nguyên.

Tổ chức của bộ nhớ chính

- Bộ nhớ gồm các ngăn nhớ có kích thước bằng nhau
- Mỗi ngăn nhớ được cấu tạo bởi các phần tử nhớ 1 bit
- Các ngăn nhớ được sắp xếp tuần tự trong bộ nhớ và được đánh số lần lượt từ 0, 1, 2... Số hiệu các ngăn nhớ như trên được gọi là địa chỉ vật lý của ngăn nhớ.



Ví dụ: Tổ chức bộ nhớ trong hệ thống 8086

- Mỗi ngăn nhớ có kích thước 1 byte
- Bộ vi xử lý 8086 sử dụng 20 đường dây địa chỉ (bus địa chỉ 20 bit) để liên lạc với bộ nhớ
(địa chỉ các ô nhớ là một dãy dài 20 bit)
- Số lượng cực đại các ô nhớ có thể đánh địa chỉ là 2^{20} (= 1MB)

Ví dụ: Tổ chức bộ nhớ trong hệ thống 8086

- Địa chỉ vật lý của các ngăn nhớ

1111...111111	
...	
0000...0000011	
0000...0000010	
0000...0000001	
0000...0000000	1 ô nhớ

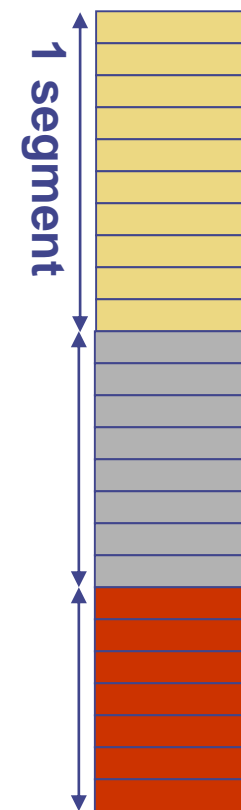
(20 bít)

Địa chỉ dạng hex:

FFFFh	
00003h	
00002h	
00001h	
00000h	1 ô nhớ

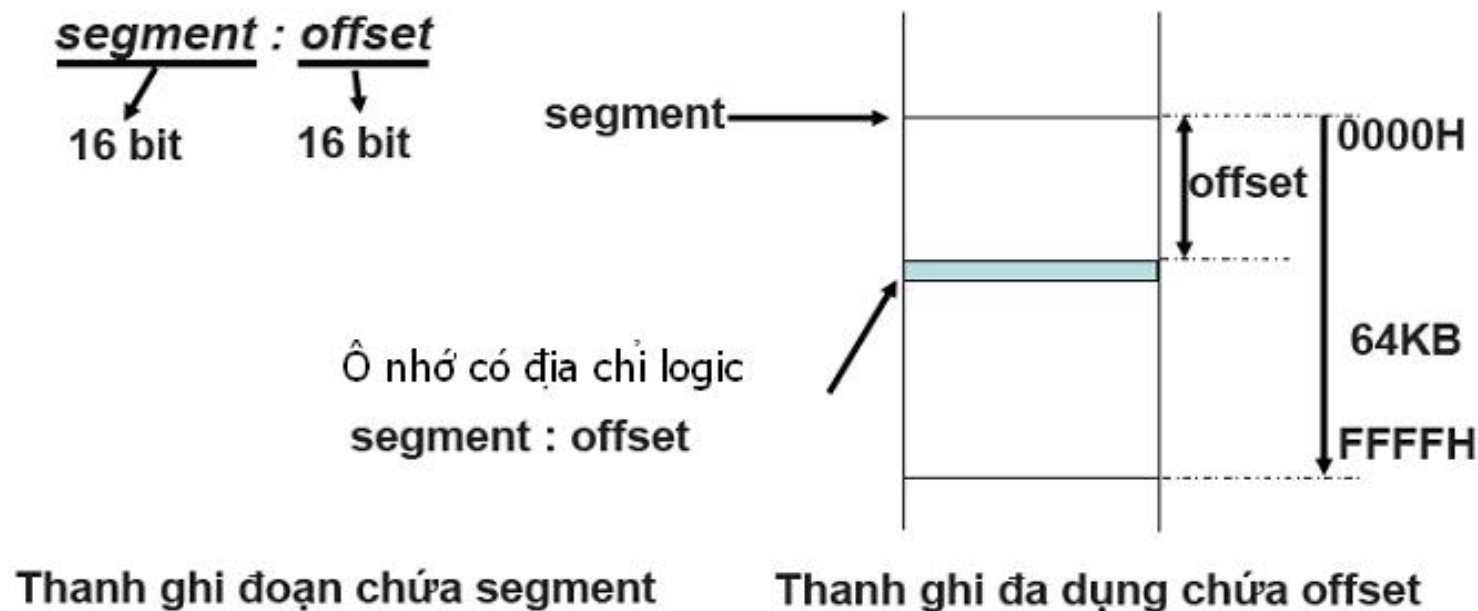
Địa chỉ logic của 8086

- Các thanh ghi của 8086 chỉ dài 16 bít, không thể chứa được địa chỉ dài 20 bít. Do đó người ta phải sử dụng một phương pháp đánh địa chỉ khác, gọi là địa chỉ logic
- Bộ nhớ được chia thành từng đoạn, mỗi đoạn có chiều dài tối đa 64 KB, các đoạn được đánh số lần lượt là 0, 1, 2,...
- Các ô nhớ trong đoạn cũng được đánh số lần lượt là 0, 1, 2,... (cực đại là 65535)



Địa chỉ logic

- Địa chỉ logic của một ô nhớ sẽ gồm 2 phần:
 - Số hiệu đoạn (segment)
 - Vị trí của ô nhớ trong đoạn (offset)

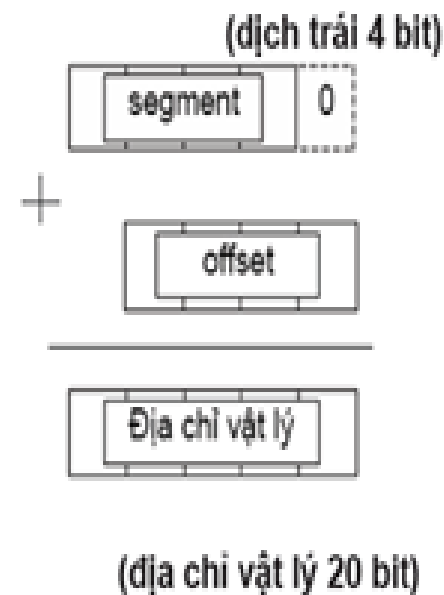
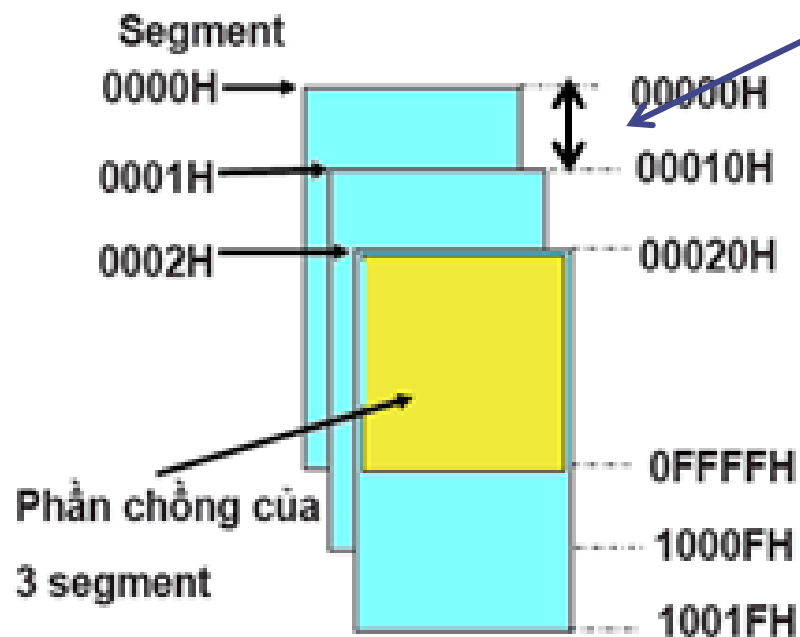


Đổi địa chỉ logic ra địa chỉ vật lý

Địa chỉ vật lý = (segment * 16) + offset

Khoảng cách giữa hai đoạn

$$\frac{2^{20}}{2^{16}} \text{B} = 16\text{B}$$



Ví dụ địa chỉ logic:

- a) Một ô nhớ có địa chỉ segment:offset = 10A2h:34B4h, hãy xác định địa chỉ vật lý của nó
- b) Một ô nhớ có địa chỉ segment:offset = 1A52h:27B4h, hãy xác định địa chỉ vật lý của nó

a) Giải:

- Bước 1: Dịch địa chỉ segment về bên trái 4 bit (tương đương với dịch 1 kí hiệu hex).

Ta thu được: 10A20h

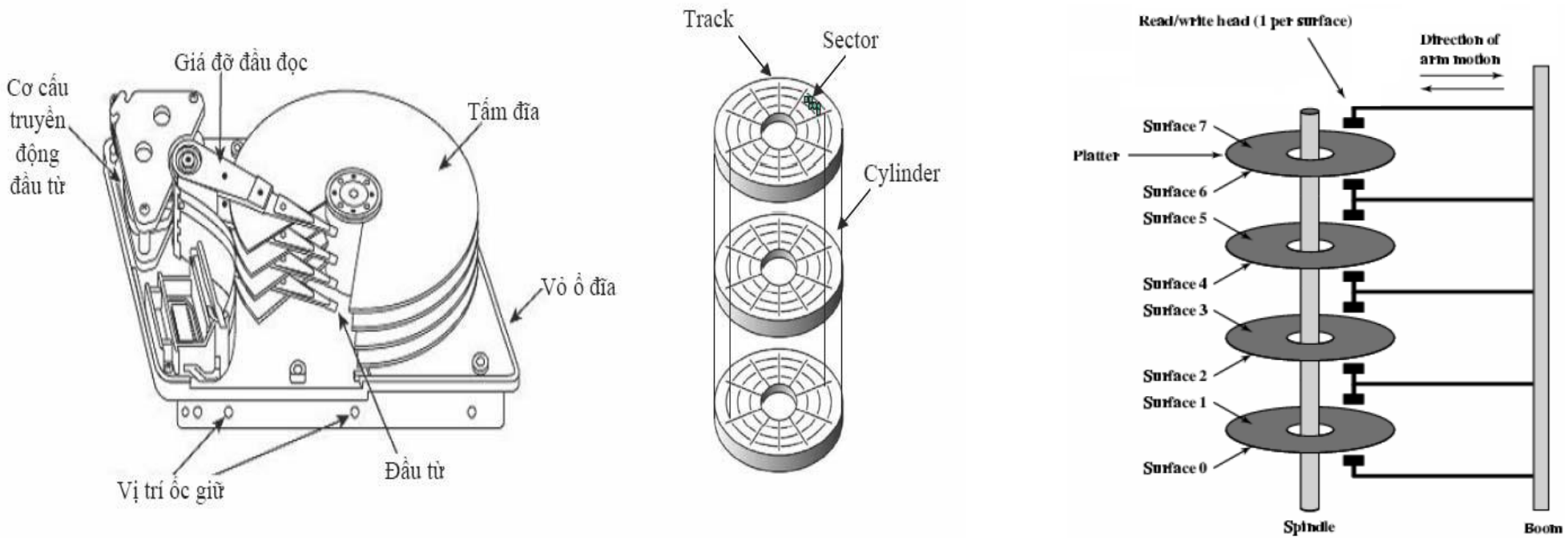
- Bước 2: Lấy giá trị thu được ở bước 1 đem cộng với địa chỉ offset:

$$\begin{array}{r} 10A20h \\ + 34B4h \\ \hline 13ED4h \end{array}$$

Vậy địa chỉ vật lý của ô nhớ đó là 13ED4h

Ổ đĩa (Disk)

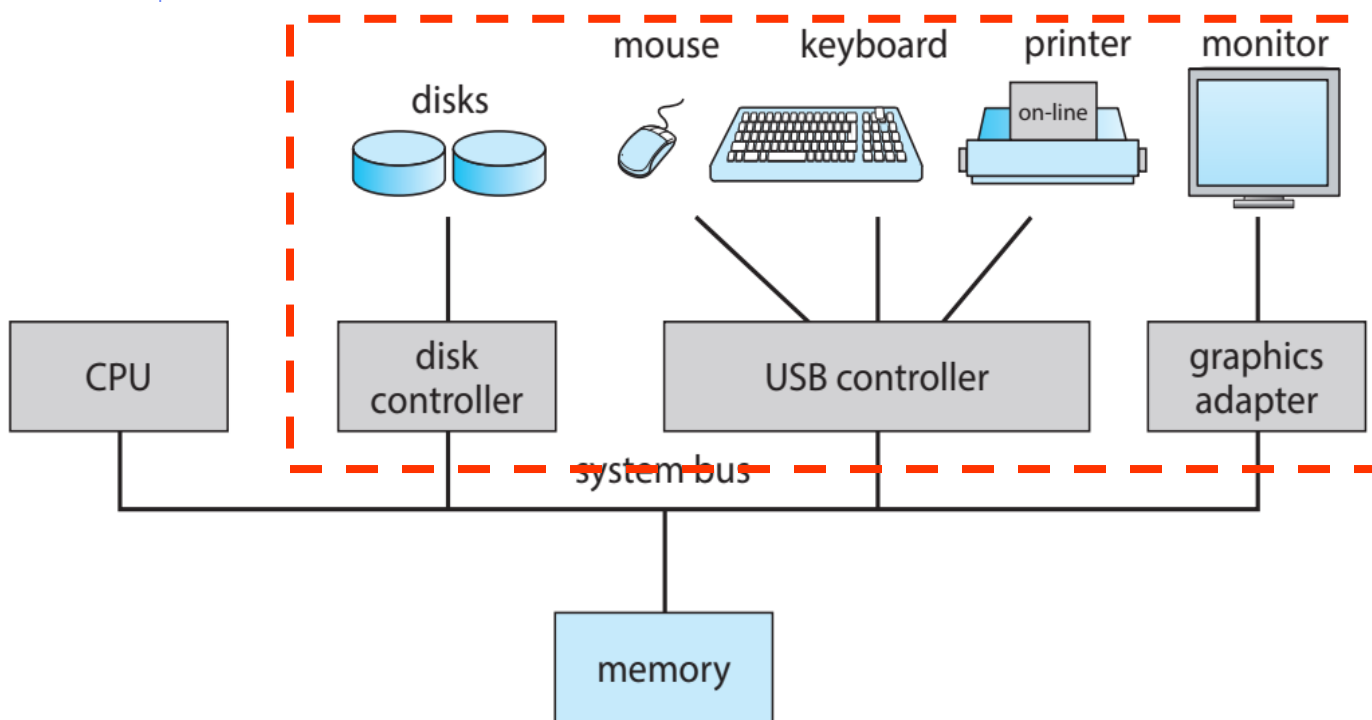
- **Đĩa từ (hay đĩa cứng):** khả năng lưu trữ lớn hơn nhiều so với RAM
- Thông tin được viết lên đĩa trên các đường tròn đồng tâm. Mỗi vành tròn được gọi là **track**. Các track trên cùng vị trí của tay quay tạo thành hình trụ **cylinder**. Mỗi track lại chia làm nhiều **sectors** (thường 512 bytes/sector).



Ổ đĩa (Disk)

- Hầu hết các máy tính đều hỗ trợ quy hoạch kiểu “virtual memory” – bộ nhớ ảo – giúp cho máy tính có thể chạy được chương trình lớn hơn bộ nhớ vật lý bằng cách sử dụng bộ nhớ chính như một kiểu cache cho các thành phần thực thi nặng nhất.
- Bộ **MMU (Memory Management Unit)** sẽ chịu trách nhiệm ánh xạ địa chỉ chương trình sang địa chỉ vật lý trong RAM.

5. Vào/ra dữ liệu



- Mỗi kiểu thiết bị ngoại vi có một bộ điều khiển (device controller) để thực hiện chuyển dữ liệu giữa thiết bị ngoại vi và vùng bộ nhớ đệm cục bộ của nó.
- Hệ điều hành cần tới các trình điều khiển thiết bị “**device drivers**” để điều khiển thiết bị vào ra.
 - Device driver là một module của hệ điều hành để điều khiển một hoặc một nhóm các thiết bị ngoại vi

Thực hiện vào/ra dữ liệu

- Vào/ra dữ liệu bằng ngắt
- Vào/ra dữ liệu bằng truy cập trực tiếp bộ nhớ (DMA – Direct Memory Access)

Vào/ra dữ liệu bằng ngắt

- CPU ghi một số thông tin vào thanh ghi tương ứng của bộ điều khiển thiết bị cần vào/ra dữ liệu.
 - ví dụ thông tin là lệnh ghi ra máy in.
- Bộ điều khiển sẽ chuyển dữ liệu giữa thanh ghi và thiết bị ngoại vi.
- Khi quá trình chuyển dữ liệu kết thúc, bộ điều khiển sinh ra ngắt để thông báo cho hệ thống.

Vào/ra dữ liệu bằng truy cập trực tiếp bộ nhớ (DMA – Direct Memory Access)

- CPU chỉ tham gia để xác lập thông tin ban đầu bằng cách ghi địa chỉ vùng nhớ, vùng đệm, số đếm, sau đó việc trao đổi sẽ diễn ra không cần CPU.
- Khi toàn bộ dữ liệu đã được truyền, bộ điều khiển sẽ sinh ngắt để thông báo.
- Cơ chế này cho phép truyền lượng dữ liệu lớn giữa đĩa và bộ nhớ chính.

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Các thành phần của hệ thống máy tính

1.2. Tổ chức của máy tính

1.3. Khái niệm hệ điều hành

1.4. Lịch sử phát triển hệ điều hành

1.5. Phân loại hệ điều hành

1.6 Hoạt động của hệ điều hành

1.7 Các dịch vụ của hệ điều hành

1.8 Các khái niệm trong hệ điều hành

1.9 Cấu trúc của hệ điều hành

1.3 – Khái niệm hệ điều hành

- Không có định nghĩa thống nhất về hệ điều hành.
- Thông thường, HĐH được định nghĩa thông qua vai trò, mục đích và chức năng trong hệ thống máy tính.
 - Hệ điều hành là hệ thống phần mềm đóng *vai trò trung gian* giữa người sử dụng (user) và phần cứng máy tính (computer hardware) nhằm tạo ra môi trường giúp việc *thực hiện các chương trình một cách thuận tiện và hiệu quả*. Ngoài ra, hệ điều hành còn quản lý và đảm bảo việc *sử dụng phần cứng của máy tính được hiệu quả*.

Hai chức năng cơ bản của hệ điều hành

- Quản lý tài nguyên của hệ thống
- Giả lập một máy tính ảo

Quản lý tài nguyên của hệ thống

- Quản lý tài nguyên đảm bảo cho tài nguyên hệ thống được sử dụng một cách hiệu quả
 - Phân phối các tài nguyên cho các chương trình khi cần thiết
 - Giải quyết tranh chấp: Quyết định thứ tự cấp phát tài nguyên cho những yêu cầu
 - Ví dụ: quản lý tài nguyên bộ nhớ (hữu hạn)
 - ◆ Nhiều chương trình cùng có thể được thực hiện
 - ◆ Tránh truy cập bất hợp lệ
 - ◆ Phải đảm bảo toàn vẹn dữ liệu (dùng chung vùng nhớ: file)

Giả lập một máy tính ảo

- Các thành phần của hệ thống máy tính:
 - Tập lệnh biểu diễn ở dạng ngôn ngữ máy (sử dụng ký hiệu 0 và 1)
 - CPU, bộ nhớ, thiết bị vào/ra, bus: cơ chế hoạt động phức tạp.
- Người sử dụng muốn đối thoại hoặc khai thác máy tính phải hiểu được cơ chế hoạt động của các thành phần máy tính và sử dụng ngôn ngữ máy để giao tiếp.
⇒ quá khó đối với người sử dụng.

Giả lập một máy tính ảo (tiếp)

- HĐH “che dấu” chi tiết phần cứng máy tính bởi một máy tính ảo.
 - Máy tính ảo này có đầy đủ các chức năng của một máy tính thực nhưng đơn giản và dễ sử dụng hơn.
 - Khi cần tác động vào máy tính thực người sử dụng chỉ cần tác động vào máy tính ảo, mọi sự chuyển đổi thông tin điều khiển từ máy tính ảo sang máy tính thực hoặc ngược lại đều do hệ điều hành thực hiện.

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Các thành phần của hệ thống máy tính

1.2. Tổ chức của máy tính

1.3. Khái niệm hệ điều hành

1.4. Lịch sử phát triển hệ điều hành

1.5. Phân loại hệ điều hành

1.6 Hoạt động của hệ điều hành

1.7 Các dịch vụ của hệ điều hành

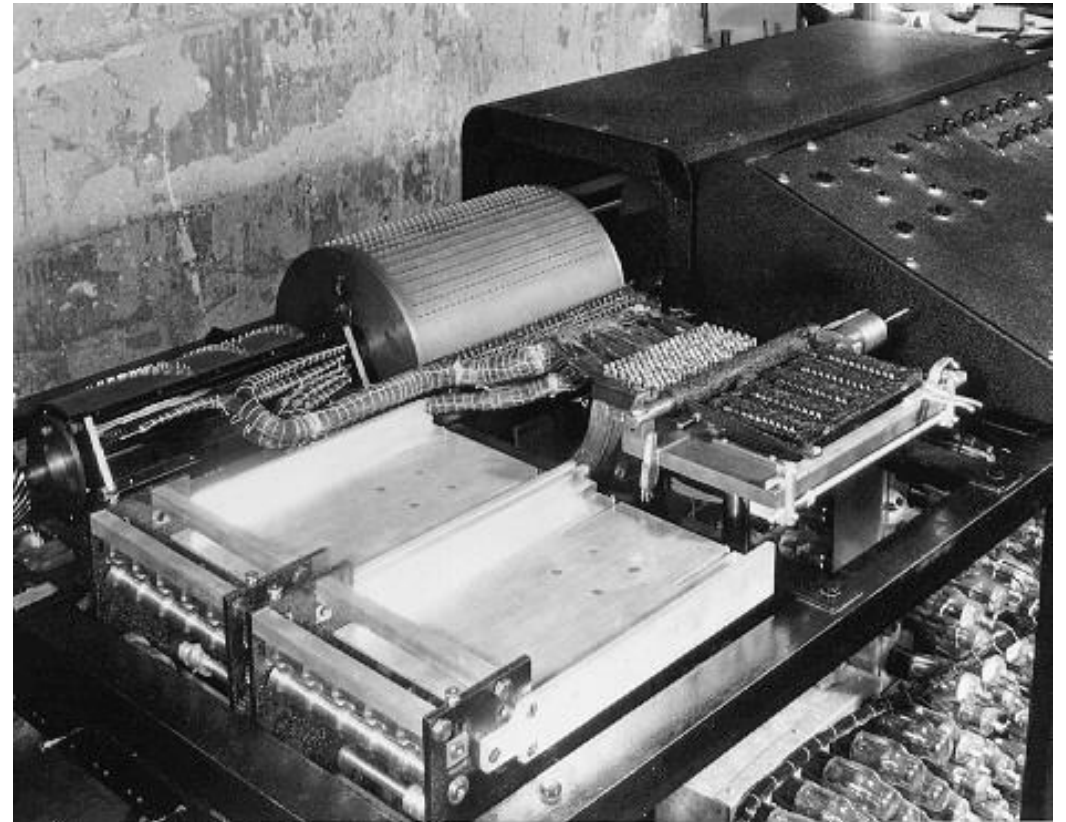
1.8 Các khái niệm trong hệ điều hành

1.9 Cấu trúc của hệ điều hành

1.4 – Lịch sử phát triển hệ điều hành

- **Thế hệ máy tính thứ nhất (1945-1955)**
 - **Dùng đèn chân không và bảng cắm**

Chưa có hệ điều hành

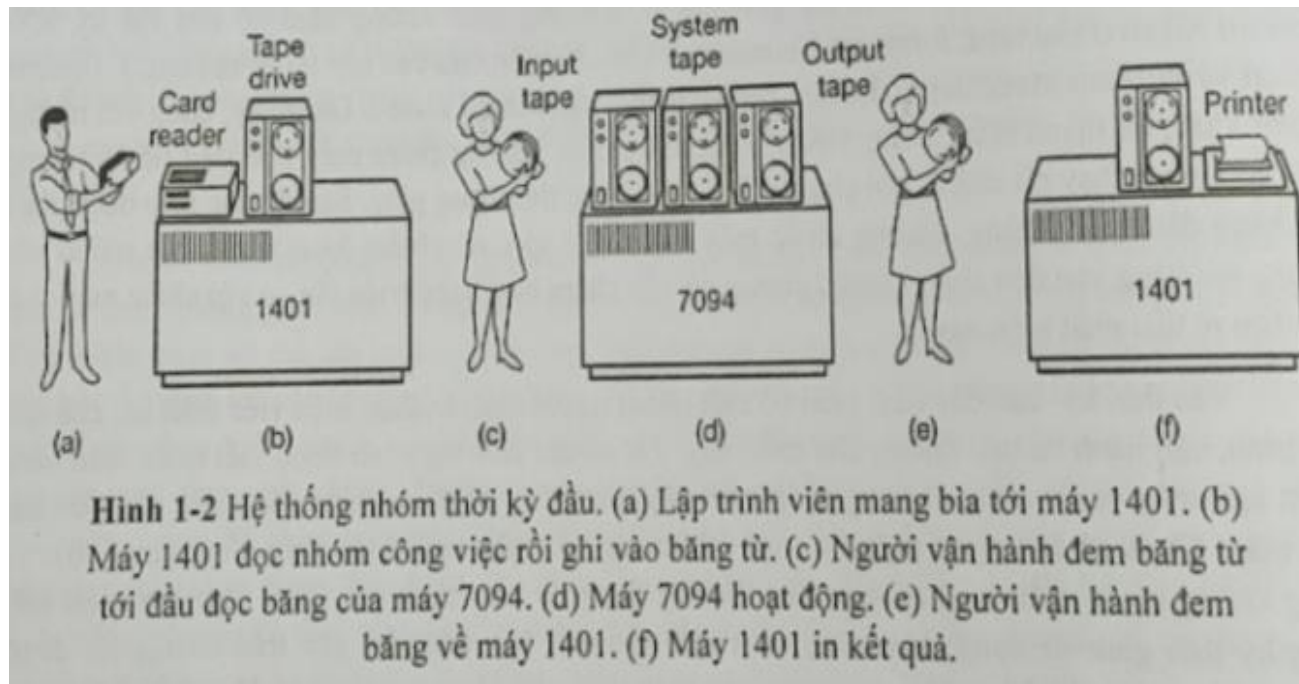


Thế hệ máy tính thứ hai (1955-1965)

Sử dụng transistor và hệ thống nhóm

■ Hệ thống nhóm:

- Lập trình viên chuẩn bị chương trình trên bìa đục lỗ hoặc đĩa từ và giao cho kỹ thuật viên
- Kỹ thuật viên phân chia chương trình thành các nhóm công việc (mỗi nhóm gồm các chương trình có yêu cầu giống nhau)



Thế hệ máy tính thứ hai (1955-1965)

Sử dụng transistor và hệ thống nhóm

- Hệ thống nhóm cần chương trình giám sát
- Chương trình giám sát (dạng đơn giản nhất của hệ điều hành):
 - Được giữ thường xuyên trong bộ nhớ
 - Mỗi khi chương trình của nhóm kết thúc, chương trình giám sát tự động nạp chương trình tiếp theo của nhóm vào máy và cho phép nó chạy
- Các hệ điều hành tiêu biểu:
 - FMS (the Fortran Monitor System)
 - IBSYS, hệ điều hành của IBM cho máy 7094.

Thế hệ máy tính thứ ba (1965-1980)

sử dụng IC và đa chương trình

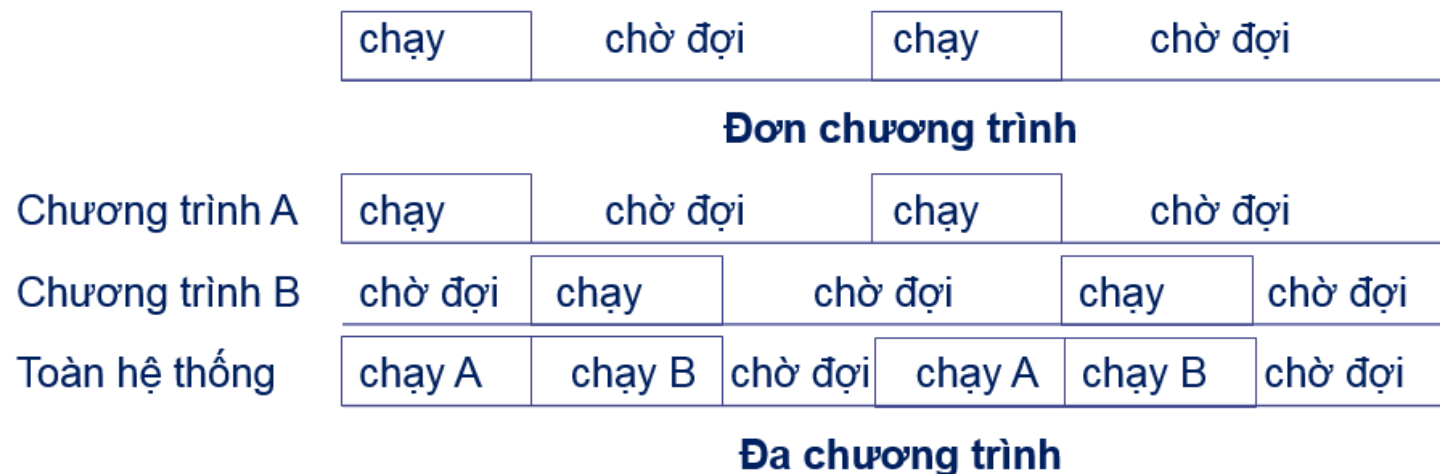
- **Hệ thống đa chương trình**
(multiprogramming system):

- Hệ thống chứa đồng thời nhiều chương trình trong bộ nhớ.
- Khi một chương trình phải dừng lại để thực hiện vào/ra, HĐH sẽ chuyển CPU sang thực hiện chương trình khác

Hệ điều hành
Chương trình A
Chương trình B
Bộ nhớ trống

■ **Hệ thống đa chương trình**(multiprogramming system):

- Sử dụng hiệu quả CPU
- HĐH cho hệ thống đa chương trình:
 - ◆ Quyết định chương trình nào được tải vào bộ nhớ và sẵn sàng thực hiện (chương trình được gọi là tiến trình)
 - ◆ Phân phối CPU cho các tiến trình



■ **Hệ thống đa chương trình:**

- Thời gian từ khi nộp chương trình cho tới khi nhận được kết quả khá lâu

■ **Hệ thống chia sẻ thời gian (Time sharing) (hoặc đa nhiệm): là hệ thống đa chương trình cải tiến**

- Nhiều người dùng nhập thông tin và lệnh từ bàn phím, kết quả sau đó được đưa trực tiếp ra màn hình
- Yêu cầu thời gian đáp ứng (từ khi người dùng gõ lệnh cho tới khi máy tính đáp lại) phải tương đối nhỏ
- CPU lần lượt thực hiện các công việc khác nhau trong khoảng thời gian rất nhỏ.
- Người dùng có cảm giác máy tính chỉ thực hiện chương trình của mình.

Thế hệ máy tính thứ ba (1965-1980)

sử dụng IC và đa chương trình

- **Các hệ điều hành tiêu biểu:**
 - OS/360 - Hệ thống đa chương trình
 - CTSS (Compatible Time Sharing System) – Hệ thống chia sẻ thời gian
 - MULTICS - Hỗ trợ đồng thời hàng trăm người dùng chia sẻ thời gian.
 - UNIX (1994)

Thế hệ máy tính thứ tư (1980 đến nay)

Các máy tính cá nhân (PC)

- Sự phát triển của **mạch tích hợp cỡ lớn** (LSI: Large Scale Integration), các vi mạch có thể chứa hàng ngàn transistor trên một centimet vuông chất bán dẫn, mở đầu cho kỷ nguyên của máy tính cá nhân.
- 1974, Intel cho ra đời bộ **vi xử lý 8080** với thế hệ CPU 8 bit đầu tiên → **Cần hệ điều hành**. Kildall đã xây dựng hệ điều hành CP/M (Control Program for Microcomputers) và thống trị thế giới máy tính trong khoảng 5 năm.
- Đầu những năm 1980, IBM thiết kế máy IBM PC và cần hệ điều hành để chạy. Kildall từ chối gặp IBM. **IBM đề nghị Bill Gates cung cấp cho một hệ điều hành.**

- Gates mua hệ điều hành DOS của công ty địa phương + bộ phiên dịch BASIC của ông, **đặt tên là MS-DOS**. Điểm mấu chốt là, Gates bán MS-DOS cho các công ty máy tính phải gắn liền với phần cứng của họ, khác với việc Kildall cố bán CP/M cho những người dùng cuối.
- CP/M, MS-DOS và các hệ máy tính thời kỳ đầu đều **hoạt động trên cơ sở gõ lệnh từ bàn phím**.
- Steve Jobs, người sáng lập ra Apple tiến hành xây dựng **hệ điều hành với giao diện đồ họa** người dùng GUI – Apple Macintosh (kernel là Unix) với ý tưởng “thân thiện với người dùng” – dành cho người dùng chưa biết gì về máy tính.
- Hệ điều hành kế nhiệm sau MS-DOS của Microsoft, **Windows chịu ảnh hưởng rất mạnh của Macintosh**. Trong 10 năm (từ **1985-1995**), hệ điều hành Windows vẫn chỉ là môi trường đồ họa chạy trên nền DOS.

- Bắt đầu từ hệ điều hành **Windows 95** (từ năm 1995), chỉ sử dụng MS-DOS cho khởi động và chạy các chương trình MS-DOS cũ, đã kết hợp rất nhiều chức năng hệ điều hành. **Windows 95, Windows 98 đều là hệ điều hành 16 bit** chạy phần lớn mã lệnh hợp ngữ (Assembly).
- **Windows NT** (New Technology) chạy trên hệ thống **32 bit**. Version 5 của Windows NT được gọi là Windows 2000 vào năm 1999.
- Sau đó, Microsoft chia các họ Windows ra **dòng cho máy khách và máy chủ riêng**. Dòng cho máy khách gồm Windows XP và cho máy chủ có Windows Server 2003 và Windows Server 2008.
- Các thế hệ **Windows tiếp theo như Vista, 7, 8, 10, 11**.

- Ngoài ra, còn có hệ điều hành **Unix** – mạnh nhất, chạy trên mạng và các máy chủ cho các công ty và **Linux** là hệ điều hành thay thế phổ biến cho Windows.
- Giữa những năm 1980 đánh dấu sự phát triển mạnh mẽ của mạng, các máy tính cá nhân chạy *hệ điều hành mạng* (**Network OS**) và *hệ điều hành phân tán* (**distributed OS**).
- *Hệ điều hành mạng* là hệ điều hành mà người dùng có thể đăng nhập vào máy từ xa và chia sẻ dữ liệu từ máy này sang máy khác.
- *Hệ điều hành phân tán*, ngược lại, là hệ điều hành mà tổ hợp các máy tính độc lập với trình diễn hệ thống như một máy đơn trước người dùng.

Thế hệ máy tính thứ 5 (1990 – nay)

Máy tính di động

- 1970s, chiếc điện thoại cầm tay đầu tiên, nặng xấp xỉ 1kg – “cục gạch”.
- Giữa những năm 1990, chiếc điện thoại Nokia đầu tiên N9000 đã hiện thực hoá ý tưởng **kết hợp tính năng thoại và tính năng máy tính** thành điện thoại thông minh ra đời.
- Hiện nay thì smartphones đã trở nên phổ biến ở khắp mọi nơi.
- Hệ điều hành **Android** của Google thống trị với các ưu điểm là mã nguồn mở
- Hệ điều hành **iOS** của Apple.

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Các thành phần của hệ thống máy tính

1.2. Tổ chức của máy tính

1.3. Khái niệm hệ điều hành

1.4. Lịch sử phát triển hệ điều hành

1.5. Phân loại hệ điều hành

1.6 Hoạt động của hệ điều hành

1.7 Các dịch vụ của hệ điều hành

1.8 Các khái niệm trong hệ điều hành

1.9 Cấu trúc của hệ điều hành

1.5 – Phân loại hệ điều hành

- Các hệ điều hành cho máy tính lớn
- Các hệ điều hành cho máy chủ
- Các hệ điều hành đa xử lý
- Các hệ điều hành cho máy tính cá nhân
- Các hệ điều hành thời gian thực
- Các hệ điều hành nhúng
- Các hệ điều hành cho thẻ thông minh

■ OS cho máy tính lớn (Mainframe)

- **Mainframe** là các máy tính cỡ lớn, được tìm thấy trong các trung tâm dữ liệu của các tập đoàn lớn. Mainframe có khả năng lưu trữ và xử lý khối lượng dữ liệu lớn.
- **Hệ điều hành cho những máy mainframe** cần xử lý nhiều công việc một lúc với lượng vào/ra cực lớn. Nó thường cung cấp ba loại dịch vụ: batch (**xử lý hàng loạt**), transaction processing (**xử lý giao dịch**) và timesharing (**chia sẻ thời gian**).
- **Ví dụ:** OS/390, OS/360, Linux

■ OS cho máy chủ

- Máy chủ, có thể là máy tính cá nhân rất mạnh, là các máy trạm hay thậm chí là máy tính cỡ lớn. Các máy tính này phục vụ nhiều người dùng cùng lúc trên mạng và cho phép người dùng chia sẻ các tài nguyên phần cứng, phần mềm như các dịch vụ in, dịch vụ file, hay Web.
- Hệ điều hành cho máy chủ lưu trữ trang web và xử lý các yêu cầu đến từ các máy khách.
- Ví dụ về một số hệ điều hành cho máy chủ như UNIX, Linux và Windows Server.

■ Hệ điều hành đa xử lý

- Máy tính song song, đa máy tính hay đa xử lý là xu hướng ngày càng phổ biến hiện nay với việc **kết hợp nhiều CPU vào một hệ thống duy nhất** để có được sức mạnh tính toán lớn.
- **Hệ điều hành đa xử lý** dùng cho các hệ thống này thường là **biến thể của các hệ điều hành máy chủ**, với các tính năng đặc biệt cho việc giao tiếp, kết nối và tính nhất quán
- Rất nhiều hệ điều hành nổi tiếng bao gồm Windows và Linux chạy trên đa xử lý.

■ OS dành cho máy tính cá nhân

- Máy tính cá nhân là kiểu máy tính **phổ biến nhất** trong các dòng máy tính.
- Các **hệ điều hành cho máy tính cá nhân hỗ trợ đa chương trình**, cung cấp các tính năng cho người dùng đơn như xử lý văn bản, bảng tính, trò chơi và truy cập Internet.
- **Hệ điều hành phổ biến hiện nay** như Linux, Windows 7, Windows 8, 10 và Apple's OS X.

■ Các hệ điều hành thời gian thực

- Các hệ thống này coi thời gian là một tham số chính.

- ◆ Ví dụ, trong các hệ thống điều khiển xử lý công nghiệp, các máy tính thời gian thực phải lấy dữ liệu về việc xử lý sản phẩm và sử dụng nó để điều khiển thiết bị trong nhà máy.

- Nếu các hành động phải thực hiện vào đúng một thời điểm định trước (hay trong khoảng thời gian định trước), ta gọi đó là **hệ thống thời gian thực cứng**.

- Một hệ thống khác được gọi là **hệ thống thời gian thực mềm**, trong đó đôi khi cho phép được sai lệch về giới hạn thời gian.

- ◆ Ví dụ, hệ thống âm thanh số hay hệ thống đa phương tiện.

■ OS cho hệ thống nhúng (Embedded OS)

- **Hệ thống nhúng** chạy trên các máy tính để điều khiển thiết bị **không-phải-là-máy tính** như lò nướng, lò vi sóng, Tivi, ô tô, đầu ghi DVD, điện thoại truyền thống, máy chơi nhạc MP3...
- **Các hệ điều hành phổ biến** như Embedded Linux, Contiki OS, Tiny OS, RTOS, ...

■ OS cho thẻ thông minh

- Đây là hệ điều hành **nhỏ nhất** chạy trên các thẻ thông minh – thiết bị kích thước thẻ có chứa chip CPU.
- Các hệ điều hành có thể xử lý một **chức năng đơn lẻ** như thanh toán điện tử hoặc một vài chức năng.
- Một số thẻ sử dụng bộ chạy **JVM** (Java Virtual Machine) cài đặt trên ROM để chạy chương trình Java Applets.

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Các thành phần của hệ thống máy tính

1.2. Tổ chức của máy tính

1.3. Khái niệm hệ điều hành

1.4. Lịch sử phát triển hệ điều hành

1.5. Phân loại hệ điều hành

1.6 Hoạt động của hệ điều hành

1.7 Các dịch vụ của hệ điều hành

1.8 Các khái niệm trong hệ điều hành

1.9 Cấu trúc của hệ điều hành

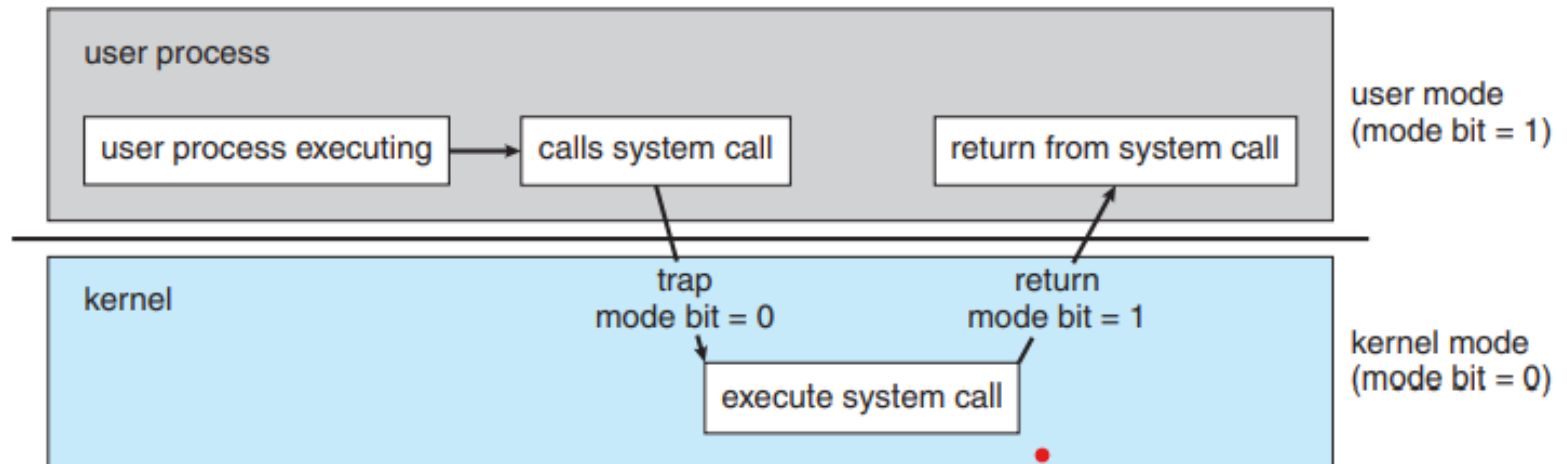
1.6 Hoạt động của hệ điều hành

■ Quá trình khởi động hệ điều hành

- Hệ điều hành có một chương trình nhỏ gọi là chương trình tải hay chương trình mồi (OS loader, hoặc Bootstrap program, hoặc Boot).
 - ◆ OS loader nằm ở một vị trí xác định trên đĩa hoặc thiết bị nhớ ngoài khác.
 - Đối với đĩa, chương trình này nằm ở sector đầu tiên của đĩa

1.6 Hoạt động của hệ điều hành

- Chế độ hoạt động kép (**Dual-Mode**) cho phép HĐH tự bảo vệ chính nó và các thành phần khác của hệ thống
 - Chế độ nhân (kernel mode)
 - ◆ chương trình thực hiện có đầy đủ quyền truy cập và điều khiển phần cứng máy tính
 - Chế độ người dùng (user mode)
 - ◆ chương trình bị hạn chế rất nhiều quyền truy cập và điều khiển phần cứng
- Mode bit được cung cấp bởi phần cứng để chuyển giữa chế độ người dùng và chế độ nhân



CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Các thành phần của hệ thống máy tính

1.2. Tổ chức của máy tính

1.3. Khái niệm hệ điều hành

1.4. Lịch sử phát triển hệ điều hành

1.5. Phân loại hệ điều hành

1.6 Hoạt động của hệ điều hành

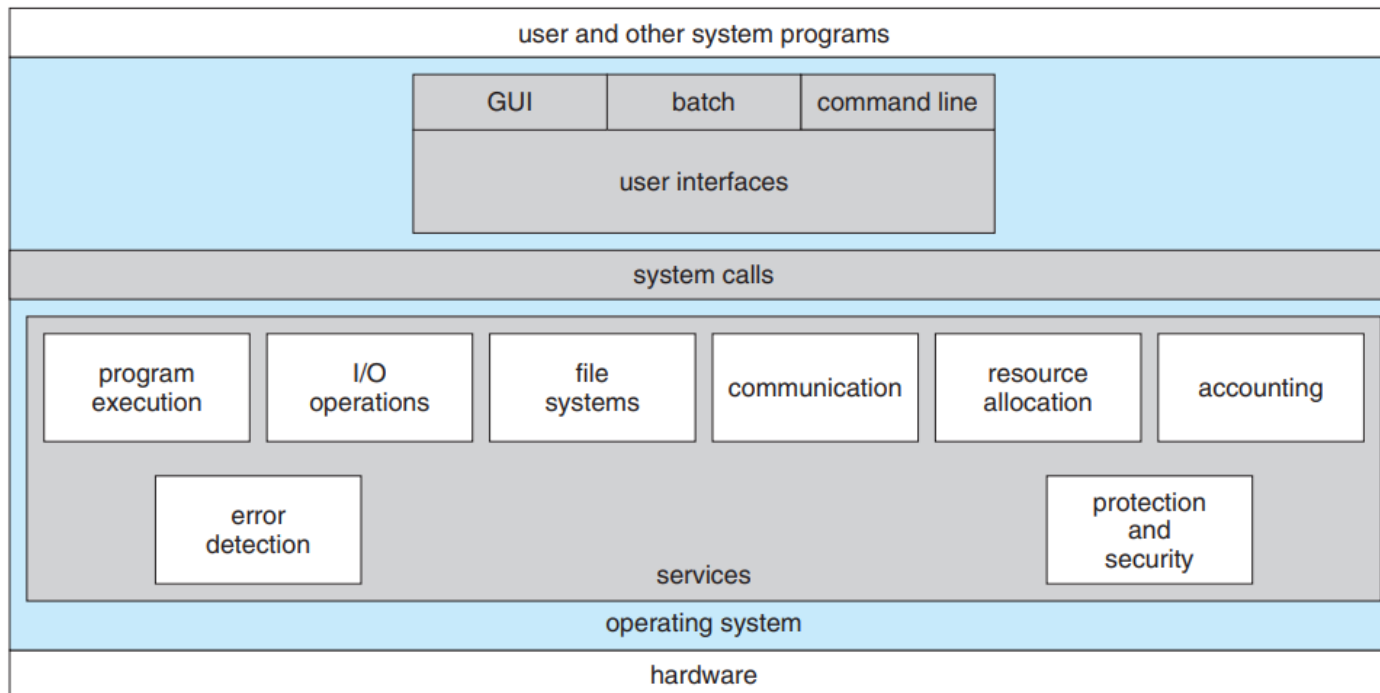
1.7 Các dịch vụ của hệ điều hành

1.8 Các khái niệm trong hệ điều hành

1.9 Cấu trúc của hệ điều hành

1.7 Các dịch vụ của hệ điều hành

- Dịch vụ của hệ điều hành có thể hiểu đơn giản là những công việc mà hệ điều hành thực hiện giúp người dùng hoặc chương trình ứng dụng.



- Một số dịch vụ:
 - Giao diện người dùng (User interface)
 - Thi hành chương trình (program execution)
 - Thao tác nhập xuất (I/O operations)
 - Thao tác trên hệ thống tập tin (systems file)
 - Truyền thông (communication)
 - Phát hiện lỗi (error detection)
 - Cấp phát tài nguyên (resource allocation)
 - Thống kê báo cáo (accounting)
 - Bảo vệ (protection and security)

1.7 Các dịch vụ của hệ điều hành

- Thi hành chương trình (program execution)
 - Nạp chương trình vào bộ nhớ và thi hành. Chấm dứt thi hành theo cách thông thường hay bất thường (lỗi)
- Thao tác nhập xuất (I/O operations)
 - Để tăng tính hiệu quả, chương trình không truy xuất trực tiếp các thiết bị vào/ra. HĐH phải cung cấp phương tiện để thực hiện vào ra.
- Thao tác trên hệ thống tập tin (systems file)
 - Chương trình có khả năng đọc, ghi, tạo, xóa, chép file

1.7 Các dịch vụ của hệ điều hành

- Truyền thông (communication)
 - Trao đổi thông tin giữa các tiến trình đang thực thi trên cùng một máy hoặc trên các máy trong mạng. Truyền thông được thực hiện nhờ bộ nhớ phân chia hoặc bằng kỹ thuật chuyển thông điệp.
- Phát hiện lỗi (error detection)
 - Chỉ ra lỗi tại CPU, bộ nhớ, trong thiết bị vào ra hay trong các chương trình. Mỗi dạng lỗi, HĐH sẽ có cách giải quyết tương ứng
- Cấp phát tài nguyên (resource allocation)
 - Phân phối tài nguyên cho nhiều nhiều công việc thực diện trong cùng thời điểm đảm bảo công bằng và hiệu quả
- Thống kê báo cáo (accounting)
 - Lưu giữ thông tin về loại và số lượng tài nguyên sử dụng, nhằm sử dụng cho tính toán (giá thành sử dụng), nghiên cứu (cải tiến hệ thống)
- Bảo vệ (protection and security)
 - Đảm bảo mọi truy nhập tới các tài nguyên hệ thống đều được kiểm soát

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Các thành phần của hệ thống máy tính

1.2. Tổ chức của máy tính

1.3. Khái niệm hệ điều hành

1.4. Lịch sử phát triển hệ điều hành

1.5. Phân loại hệ điều hành

1.6 Hoạt động của hệ điều hành

1.7 Các dịch vụ của hệ điều hành

1.8 Các khái niệm trong hệ điều hành

1.9 Cấu trúc của hệ điều hành

1.8 Các khái niệm trong hệ điều hành

- **Tiến trình**

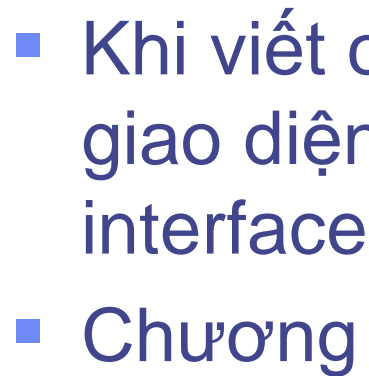
- Tiến trình là một chương trình đang thực hiện

- **Bộ thông dịch lệnh (Shell)**


- Là một chương trình cung cấp giao diện giao tiếp giữa người dùng và hệ điều hành

- **Lời gọi hệ thống (System call)**

- Cung cấp một giao tiếp giữa tiến trình và hệ điều hành
- Tiến trình sử dụng lời gọi hệ thống để yêu cầu một dịch vụ nào đó từ phía hệ điều hành
- Lời gọi hệ thống được thực hiện dưới dạng:
 - ◆ Các câu lệnh trong các ngôn ngữ lập trình cấp thấp
 - ◆ Lệnh gọi ngắt trong hợp ngữ (INT)
 - ◆ Thủ tục gọi hàm API (Application Programming Interface)

- 
- Khi viết chương trình, người lập trình sẽ sử dụng các hàm do giao diện lập trình ứng dụng API (Application programming interface) cung cấp thay vì gọi trực tiếp lời gọi hệ thống.
 - Chương trình dịch (compiler) sau đó sẽ thực hiện việc chuyển đổi lời gọi hàm sang lời gọi hệ thống tương ứng của hệ điều hành.

- Việc sử dụng API có một số ưu điểm so với sử dụng trực tiếp lời gọi hệ thống.
 - **Thứ nhất**, chương trình dễ dàng chuyển sang thực hiện trên hệ thống khác có cùng API. Khi hệ điều hành nâng cấp lời gọi hệ thống bằng cách thêm chức năng mới, chương trình gọi trực tiếp lời gọi hệ thống sẽ phải viết lại, trong khi chương trình sử dụng API thì không.
 - **Thứ hai**, hàm API thường thực hiện thêm một số thao tác so với lời gọi hệ thống tương ứng. Ví dụ, khi thực hiện lời gọi hệ thống để tạo tiến trình mới, cần làm một số thao tác trước và sau khi lời gọi này. Hàm API tương ứng đã chứa sẵn đoạn mã thực hiện các thao tác này, do vậy chương trình gọi hàm API sẽ không phải tự thực hiện các thao tác.
 - **Thứ ba**, hàm API thường hỗ trợ các phiên bản khác nhau của hệ điều hành và tự phát hiện phiên bản phù hợp.

- 
- Application programming interface (API):
 - Win32 API cho Window
 - POSIX API dùng cho UNIX, Linux
 - Java API dùng cho máy ảo Java

CHƯƠNG 1: GIỚI THIỆU CHUNG

1.1. Các thành phần của hệ thống máy tính

1.2. Tổ chức của máy tính

1.3. Khái niệm hệ điều hành

1.4. Lịch sử phát triển hệ điều hành

1.5. Phân loại hệ điều hành

1.6 Hoạt động của hệ điều hành

1.7 Các dịch vụ của hệ điều hành

1.8 Các khái niệm trong hệ điều hành

1.9 Cấu trúc của hệ điều hành

1.9-Cấu trúc của hệ điều hành

- Các thành phần của hệ điều hành
- Một số kiểu cấu trúc hệ điều hành

1. Các thành phần của hệ điều hành

- Hệ điều hành thường gồm các thành phần thực hiện những nhiệm vụ sau:
 - Quản lý tiến trình
 - Quản lý bộ nhớ
 - Quản lý vào ra
 - Quản lý file
 - Hỗ trợ mạng
 - Hệ thống bảo vệ
 - Giao diện với người dùng
- Nhân của hệ điều hành (kernel)
 - Bao gồm những thành phần thực hiện các chức năng cơ bản của hệ điều hành và thường xuyên được giữ trong bộ nhớ.

2. Một số kiểu cấu trúc hệ điều hành

- Cấu trúc hệ điều hành:
 - Hệ điều hành gồm nhiều thành phần và các thành phần này phải được liên kết với nhau
 - Cách thức tổ chức, liên kết các thành phần xác định cấu trúc của hệ điều hành.
- Một số kiểu cấu trúc thường gặp:
 - Cấu trúc nguyên khối (monolithic)
 - Cấu trúc phân lớp
 - Cấu trúc Client-Server

Cấu trúc nguyên khối (monolithic)

- Các chương trình và dữ liệu của HĐH có chung một không gian nhớ
- Hệ điều hành là một chương trình lớn gồm một tập hợp của các chương trình con.
- Mỗi chương trình có thể gọi một chương trình khác bất cứ khi nào nó muốn.

Cấu trúc phân lớp

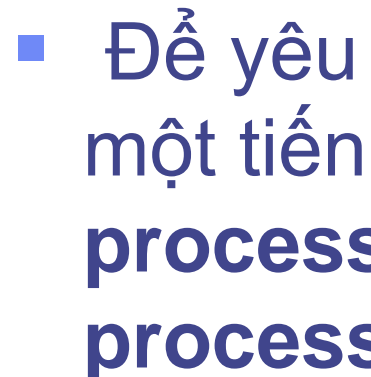
- Các thành phần của hệ điều hành được phân thành các lớp chồng lên nhau:
 - Lớp là sự triển khai của một đối tượng trừu tượng được tạo thành từ dữ liệu và các hoạt động có thể thao tác với dữ liệu đó.
 - Mỗi lớp có thể gọi các hoạt động của lớp ngay bên dưới nó
 - Mỗi lớp chỉ có thể liên lạc được với lớp kề bên trên và lớp kề bên dưới nó.

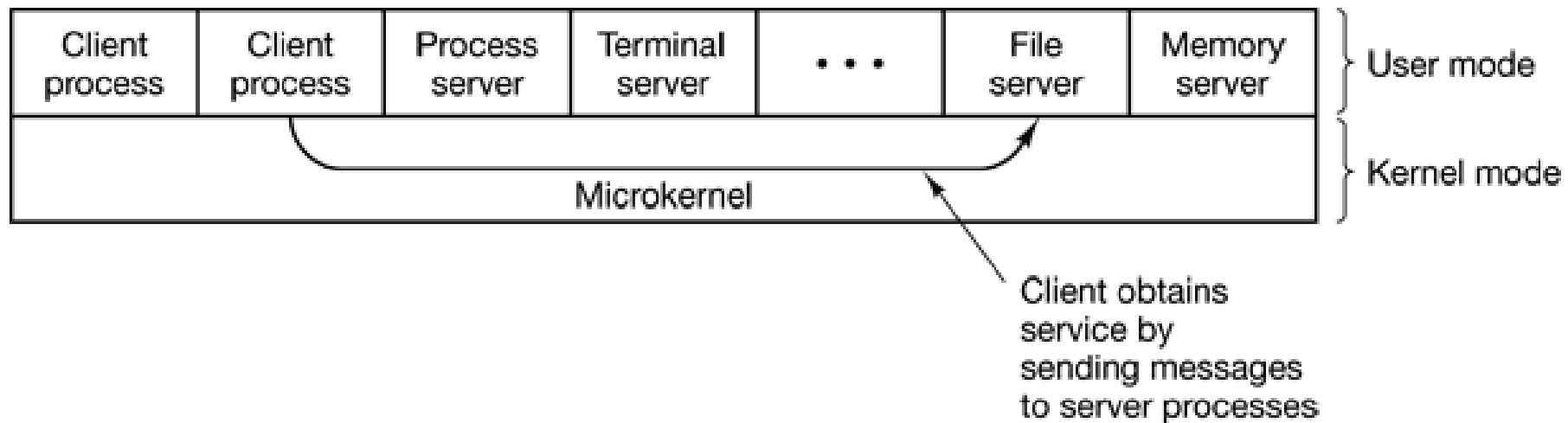
Ví dụ: Cấu trúc hệ điều hành THE

Lớp	Chức năng
5	Hệ thống điều hành
4	Các chương trình của người dùng
3	Quản lý vào/ra
2	Liên lạc giữa tiến trình và hệ thống giao tiếp
1	Quản lý bộ nhớ
0	Phân phối Processor và thực hiện đa chương trình

Cấu trúc Client-Server

- Hệ điều hành được chia nhỏ thành các bộ phận, mỗi bộ phận sẽ điều khiển một chức năng của hệ thống, như dịch vụ file, dịch vụ tiến trình, dịch vụ thiết bị cuối, hay dịch vụ bộ nhớ...
- Nhờ vậy mỗi phần sẽ trở nên nhỏ hơn và dễ quản lý hơn.

- 
- Để yêu cầu một dịch vụ, ví dụ như đọc một block của file, một tiến trình của người dùng (**tiến trình khách – client process**) sẽ gửi yêu cầu tới **tiến trình phục vụ (server process)**, nó sẽ thực hiện công việc và gửi kết quả trở lại.



Phần nhân (kernel) sẽ điều khiển sự liên lạc giữa các tiến trình client và server.



Hết Chương 1