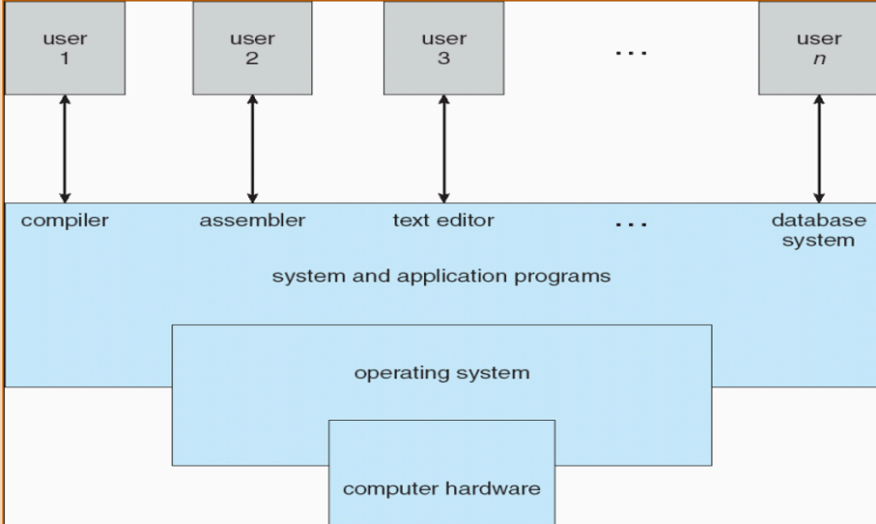
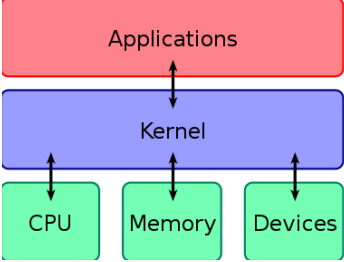
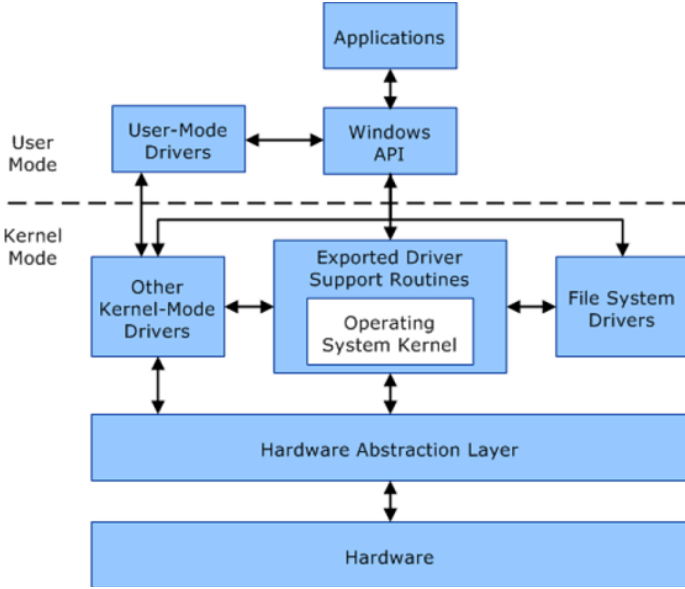


Hệ điều hành là gì
<div>- Là tập phần mềm đóng vai trò là thành phần trung gian giữa người dùng và hệ thống phần cứng máy tính</div> <div>- Mục đích của hệ điều hành:</div> <div>+ Thực thi chương trình người dùng dễ dàng hơn</div> <div>+ Sử dụng hệ thống máy tính hiệu quả, thuận tiện hơn</div>

Các thành phần của một hệ thống máy tính	
<div>- Phần cứng: Cung cấp các tài nguyên cơ bản (CPU, bộ nhớ, các thiết bị vào ra)</div> <div>- Hệ điều hành: Điều khiển và điều phối việc sử dụng phần cứng máy tính phục vụ các chương trình ứng dụng của người dùng.</div> <div>- Các chương trình ứng dụng: Sử dụng các tài nguyên máy tính để giải quyết các vấn đề tính toán của người dùng</div> <div>- Người dùng: (user - đối tượng sử dụng) Con người, máy móc hay các hệ thống máy tính khác</div> <div>- Ví dụ 1 số chương trình hệ thống: CMD, File Explorer,...</div>	

Quản lý tiến trình	
<div>- Một chương trình không làm gì trừ khi các chỉ thị của nó được thực thi bởi 1 CPU</div> <div>- Một tiến trình:</div> <div>+ Có thể được xem như 1 chương trình đang thực thi.</div> <div>+ Là 1 công việc hay chương trình chia thời</div> <div>+ Cần các tài nguyên xác định gồm: thời gian CPU, bộ nhớ, tập tin, các thiết bị xuất /nhập để hoàn thành tác vụ của nó.</div>	<div>- Các tài nguyên này được cấp cho tiến trình khi nó được tạo ra, hay được cấp phát tới nó khi nó đang chạy.</div> <div>- Khi tiến trình này kết thúc, hệ điều hành sẽ đòi lại bất cứ tài nguyên nào có thể dùng lại.</div>

Định nghĩa hệ điều hành
<div>- Bộ phân phối tài nguyên: Quản lý và điều phối tài nguyên</div> <div>- Bộ điều khiển chương trình: Điều khiển thực thi chương trình của người dùng và điều khiển thao tác của các thiết bị vào ra</div>

Kernel		
<div>- Chương trình “nhân” (kernel): là hạt nhân - trái tim của hệ điều hành. Bất cứ điều gì xảy ra trên máy tính đều đi qua nó</div>  <div>- Kernel luôn được thực thi khi hệ thống máy tính hoạt động</div>	<div>- Vai trò của Kernel:</div> <div>+ Trong HĐH, Kernel đóng vai trò như 1 dịch giả (translator)</div> <div>+ Nó chuyển đổi các yêu cầu input - output từ phần mềm thành tập lệnh cho CPU và GPU</div> <div>+ Là 1 lớp ở giữa phần mềm và phần cứng</div> <div>+ Kernel quản lý: CPU, GPU, Bộ nhớ, Thiết bị và thông tin vào ra, nguồn tài nguyên,...</div> <div>- Người dùng chỉ có thể truy cập không gian kernel thông qua việc sử dụng các cuộc gọi hệ thống (system call). Nếu một chương trình cố gắng truy cập trực tiếp, nó sẽ dẫn đến lỗi.</div>	 <div>- Nếu bạn khởi chạy 1 chương trình, UI sẽ gửi yêu cầu tới Kernel. Kernel sau đó gửi yêu cầu tới CPU, Bộ nhớ để gán sức mạnh xử lý, bộ nhớ và những thứ khác để ứng dụng có thể chạy trơn tru ở UI</div>

Các hệ thống máy tính điển hình		
<div>- Các hệ Mainframe: (Máy tính mainframe là gì và người ta sử dụng nó như thế nào?)</div> <div>+ Các hệ xử lý theo lô đơn giản</div> <div>+ Các hệ xử lý theo lô, đa chương trình</div> <div>+ Các hệ phân chia thời gian</div> <div>- Các hệ máy tính cá nhân</div> <div>- Các hệ song song, các hệ phân tán, các hệ thời gian thực</div>		
Các hệ xử lý theo lô đơn giản		
<div>- Có thể coi là hệ điều hành đầu tiên, tương đối đơn giản</div> <div>+ Nhiệm vụ của HĐH: truyền quyền điều khiển tuần tự cho các “công việc” (job) trong lô (batch)</div> <div>+ “Công việc”: chương trình, dữ liệu, các thông tin điều khiển...</div>	<div>- Phân phối bộ nhớ trong hệ xử lý theo lô đơn giản:</div>	<div>- Lô (Batch): là 1 file chứa nhiều lệnh</div> <div>copy con sasa.bat</div> <div>@echo Hello</div> <div>md N01</div>

+ Nhược điểm: không tận dụng CPU một cách hiệu quả	<div>Operating System</div> <div>User Program Area</div>	cd N01 @echo con cac pause Ctrl + Z: //để dừng nhập cho file .bat
--	--	--

Các hệ thống xử lý theo lô đa chương trình

<div>- Vấn đề: Làm sao để giữ CPU luôn bận rộn?</div> <div>- Cơ sở: Một chương trình người dùng không thể cùng một lúc sử dụng cả CPU và các thiết bị vào ra</div> <div>- Tư tưởng chính:<div>+ Lưu đồng thời nhiều công việc trong bộ nhớ trong</div><div>+ Hệ điều hành chọn công việc để thực hiện</div><div>+ Trong trường hợp công việc đang phải đợi thực hiện một thao tác nào đó (ví dụ thao tác vào/ra) Hệ điều hành sẽ chọn việc khác để thực hiện</div></div> <div>- Đặc điểm:<div>+ Tương đối phức tạp</div><div>+ Khái niệm “Nhóm công việc” (job pool)</div><div>+ Lập lịch công việc: chọn các công việc để chuyển vào bộ nhớ trong</div><div>+ Quản lý lưu trữ: lưu cùng lúc một số công việc trong bộ nhớ trong</div><div>+ Lập lịch CPU: chọn thực thi một trong các công việc đang ở bộ nhớ trong</div></div>	<div>Phân phối bộ nhớ trong các hệ thống xử lý theo lô đa chương trình</div> <div><div>0</div><div>Operating System</div><div>Job 1</div><div>Job 2</div><div>Job 3</div><div>Job 4</div><div>512K</div></div>
--	--

Các hệ phân chia thời gian (Time Sharing)

<div>- Vì sao?</div> <div>+ Các hệ xử lý theo lô, đa chương trình cung cấp một môi trường trong đó các tài nguyên hệ thống được sử dụng một cách hiệu quả, nhưng không cung cấp cho người dùng khả năng tương tác với hệ thống</div> <div>+ Các hệ phân chia thời gian là sự mở rộng của các hệ xử lý theo lô, đa chương trình</div> <div>- Tư tưởng chính</div> <div>+ Chuyển đổi quyền xử lý giữa các chương trình thường xuyên hơn</div> <div>+ Thời gian phản ứng ~ 1 giây hoặc ít hơn</div> <div>+ Cho phép chia sẻ đồng thời một máy tính giữa nhiều người dùng</div> <div>+ Khái niệm “tiền trình”: chương trình được nạp vào bộ nhớ và đang được thực thi: Vào/ra tương tác -> phụ thuộc “people speech” ví dụ tốc độ nhập dữ liệu</div>	<div>- Đặc điểm:</div> <div>+ Phức tạp hơn hệ xử lý theo lô, đa chương trình</div> <div>+ Quản lý bộ nhớ và bảo vệ</div> <div>+ Tráo đổi các công việc từ đĩa cứng và bộ nhớ (swap in/swap out)->phương pháp bộ nhớ ảo</div> <div>+ Hệ thống file trên một số đĩa cứng -> quản lý đĩa cứng</div> <div>+ Thực thi đồng thời -> Lập lịch CPU</div> <div>+ Giao tiếp và đồng bộ hoá</div> <div>+ Giải quyết bế tắc</div>
<div>- Các hệ điều hành: Window, Linux, MacOS,... đều là hệ phân chia theo thời gian</div>	

Quản lý bộ nhớ

<div>- Bộ nhớ chính là trung tâm điều hành của một máy tính hiện đại</div> <div>- Bộ nhớ chính là một kho chứa dữ liệu có khả năng truy xuất nhanh được chia sẻ bởi CPU và các thiết bị xuất /nhập</div> <div>- Hệ điều hành có nhiệm vụ cho các hoạt động sau khi đề cập tới việc quản lý bộ nhớ</div> <div>+ Giữ vết về phần nào của bộ nhớ hiện đang được dùng và tiến trình nào đang dùng.</div> <div>+ Quyết định tiến trình nào được nạp vào bộ nhớ khi không gian bộ nhớ trở nên sẵn dùng.</div> <div>+ Cấp phát và thu hồi không gian bộ nhớ khi được yêu cầu</div>

Các hệ máy tính cá nhân

<div>- Xuất hiện những năm 1970</div> <div>- Hướng tới sự tiện dụng của người dùng</div> <div>- Các hệ điều hành cho máy tính cá nhân</div> <div>+ Microsoft Windows, Apple Macintosh</div> <div>+ Linux, Unix-like OS cho PCs</div> <div>+ Kế thừa sự phát triển của hệ điều hành cho các hệ Mainframe</div>

Các hệ song song, các hệ phân tán, các hệ thời gian thực

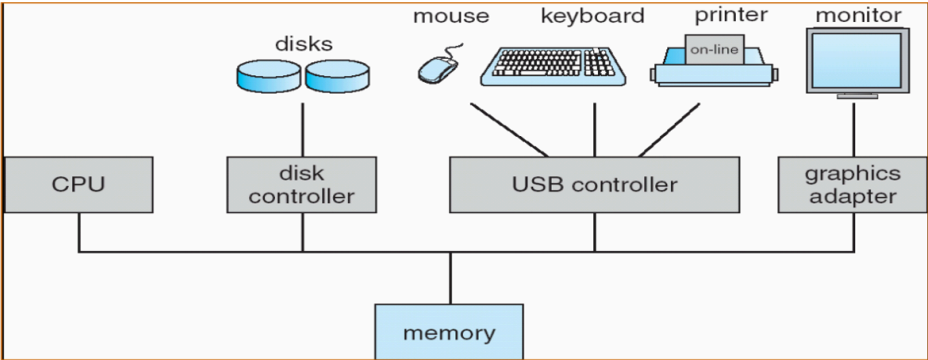
<div>Các hệ song song</div> <div>- Còn gọi là hệ đa xử lý (multiprocessor systems)</div> <div>- Một hệ thống có nhiều bộ xử lý, giao tiếp “gần”, chia sẻ computer bus, clock ...</div> <div>- Phân loại:<div>+ Đa xử lý đối xứng (SMP): Mỗi bộ xử lý có một phiên bản sao chép hệ điều hành, giao tiếp với nhau peer-to-peer</div><div>+ Đa xử lý không đối xứng (AMP): Mỗi bộ xử lý được gán một nhiệm vụ. Bộ xử lý chủ (master) sắp xếp công việc và quản lý các máy phục phục vụ (slave)</div></div>
--

+ Các hệ peer-to-peer	
-----------------------	--

Tổ chức hệ thống máy tính

Các thao tác trong hệ thống máy tính

- Một hệ thống máy tính gồm một hoặc nhiều CPU và một số bộ điều khiển thiết bị kết nối với nhau thông qua một bus chung, chia sẻ một bộ nhớ chung



Các thao tác:

- Khởi động (người dùng)
- Chương trình môi (BIOS) (thường nằm trong ROM hay EPROM)
- + Khởi tạo: thanh ghi CPU, các bộ điều khiển thiết bị, nội dung bộ nhớ
- + Tải hệ điều hành (chương trình môi phải biết địa chỉ bắt đầu của hệ điều hành) vào bộ nhớ trong
- + Chuyển quyền thực thi cho hệ điều hành

- Hệ điều hành
- + Thực hiện tiến trình đầu tiên (init) và chờ đợi các “sự kiện” (các “ngắt” từ phần cứng/phần mềm)

- Các thiết bị vào ra và CPU có thể thực thi đồng thời; tương tranh (tương tranh: 2 tiến trình cùng chạy 1 lúc, có thể dẫn đến xung đột)) các chu kì bộ nhớ

- Mỗi bộ điều khiển thiết bị (device controller) chịu trách nhiệm một loại thiết bị xác định (vì các loại thiết bị hoạt động theo cách khác nhau, không thể vừa điều khiển bàn phím lại vừa điều khiển chuột)

- Mỗi bộ điều khiển thiết bị có một bộ đệm (bộ đệm - buffer,

- CPU chuyển dữ liệu từ/vào bộ nhớ ra/từ các bộ đệm
- Thao tác vào ra (I/O) là các thao tác từ thiết bị đến bộ đệm của bộ điều khiển
- Các bộ điều khiển thiết bị báo cho CPU biết chúng đã hoàn thành các tác vụ của chúng bằng cách làm sinh ra một tín hiệu ngắt (interrupt)

- Các yêu cầu đi từ CPU đến bộ nhớ đệm (buffer) rồi mới được thực hiện vì tốc độ của các yêu cầu là khác nhau
- 1 cổng CPU cùng lúc cấm được 127 thiết bị (chuột, bàn phím,...)

- Có 3 phương pháp điều khiển vào/ra:
 - + Polling - hỏi vòng/ bầu cử
 - + Interrupt (ngắt)
 - + DMA: không do CPU điều khiển, trao đổi dữ liệu

Cấu trúc lưu trữ

- Các thanh ghi
- + Thanh ghi lệnh
- + Các thanh ghi nội tại khác

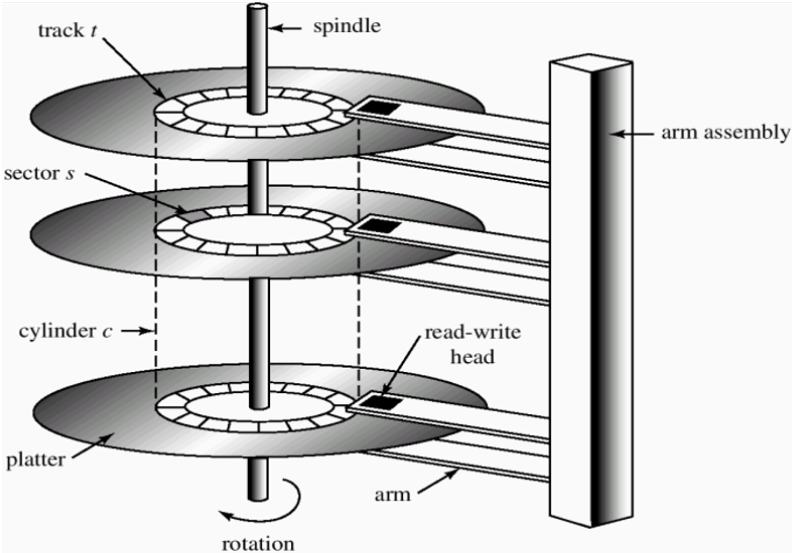
- Bộ nhớ chính (RAM)
- + Vùng lưu trữ lớn duy nhất mà CPU có thể truy nhập trực tiếp
- + Tương tác giữa CPU và BNC thông qua một loạt các thao tác load/store

- Thiết bị lưu trữ thứ cấp
- + Ví dụ: đĩa từ, ổ bán dẫn

- Dung lượng tăng, tốc độ giảm: Thanh ghi → RAM → Thiết bị lưu trữ thứ cấp

- Ổ cứng mới mua về phải thực hiện 2 việc:
 - + Phân vùng
 - + Định dạng (Format)

- Cơ chế đọc đĩa



- 1 sector chỉ chứa được 512byte

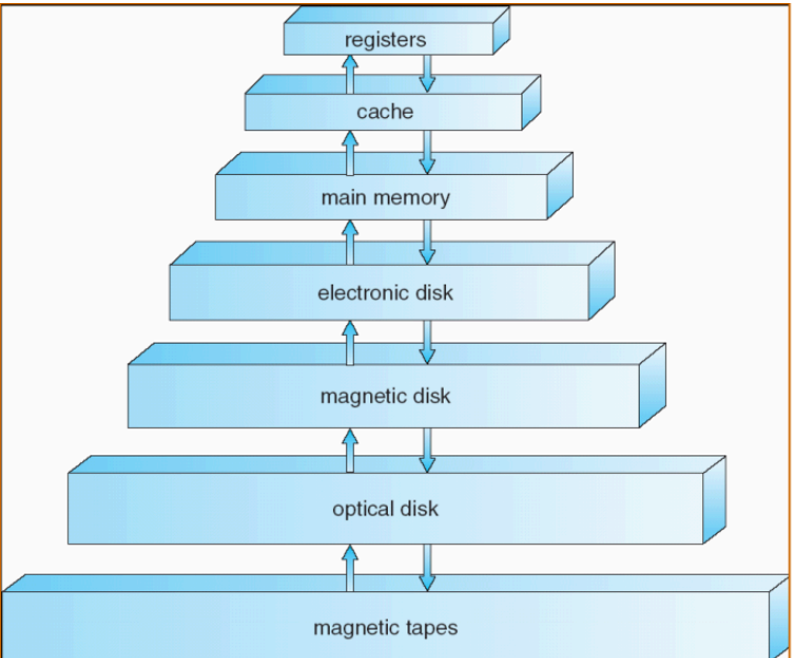
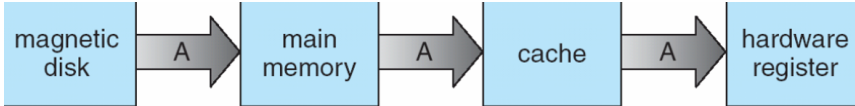
Phân cấp thiết bị lưu trữ

- Các hệ thống lưu trữ được phân cấp theo các tiêu chí về.
 - + Tốc độ
 - + Giá thành
 - + Tính không ổn định (Volatility)

- Thứ tự: Thiết bị ngoại vi (HDD, SSD) - DRAM - SRAM (cache ở trong SRAM) - CPU

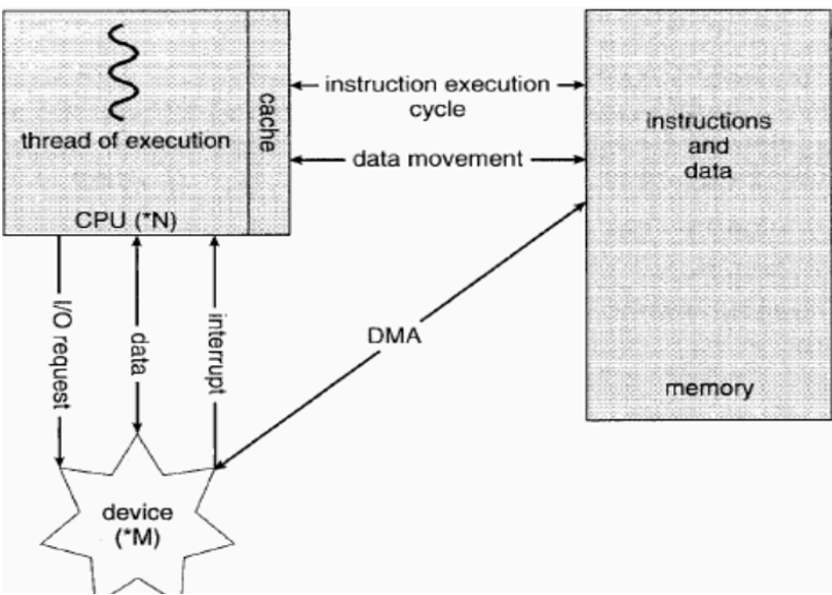
- Caching – sao chép thông tin vào thiết bị lưu trữ nhanh hơn; bộ nhớ chính có thể được xem là cache của bộ nhớ thứ cấp
 - + Sử dụng bộ nhớ tốc độ cao để lưu trữ dữ liệu mới được truy cập → Cần một chiến lược quản lý cache
 - + Caching làm nảy sinh một cấp độ mới trong phân cấp lưu trữ → đảm bảo tính nhất quán của dữ liệu được lưu trữ cùng lúc ở nhiều nơi (với các cấp độ truy cập khác nhau)

Từ đĩa đến thanh ghi



Cấu trúc vào/ra

- Các thiết bị lưu trữ chỉ là một loại thiết bị vào/ra
- Các bộ điều khiển thiết bị (device controller)
- + Có thể có nhiều hơn một thiết bị được gắn với 1 bộ điều khiển thiết bị (ví dụ: SCSI)
- + Mỗi bộ điều khiển thiết bị có một bộ đệm
- + Chịu trách nhiệm giữa các thiết bị ngoại vi và bộ đệm
- Trình điều khiển thiết bị (device driver)
- + Thường được cung cấp bởi Hệ điều hành
- + Tương ứng với mỗi bộ điều khiển thiết bị là một trình điều khiển thiết bị
- + Cung cấp một giao diện truy nhập đến thiết bị cho các thành phần khác của hệ điều hành.



Các thao tác trong hệ điều hành

Các nhiệm vụ của ngắt

- Ngắt chuyển điều khiển đến dịch vụ ngắt, thông qua một vectơ ngắt – nơi chứa địa chỉ của tất cả các dịch vụ ngắt
- Trong kiến trúc ngắt, ta phải lưu giữ địa chỉ của lệnh tại đó có tín hiệu ngắt ngắt → cho việc khôi phục lại quá trình tính toán sau khi xử lý ngắt
- Trong khi một ngắt đang được xử lý, các ngắt khác sẽ bị từ chối để tránh hiện tượng “lost interrupt”
- Một trap hay exception là một ngắt của chương trình người dùng, sinh ra do lỗi hoặc một yêu cầu đặc biệt của người dùng.
- Hệ điều hành điều khiển bởi ngắt.

Xử lý ngắt

- Hệ điều hành bảo quản trạng thái của CPU bằng cách lưu lại các thanh ghi, bộ đếm chương trình...
- Xác định loại tín hiệu ngắt, gọi dịch vụ ngắt
- + Tìm kiếm tuần tự theo thông tin ngắt
- + Sử dụng vector ngắt
- Dịch vụ ngắt
- + Phân tích thông tin ngắt (interrupt information)
- + Gọi trình xử lý tín hiệu ngắt tương ứng

Timer

- Timer được sử dụng để ngăn các tiến trình chiếm tài nguyên quá lâu
- + Sinh ra ngắt sau một đơn vị thời gian
- + Hệ điều hành sử dụng một biến đếm: Trừ dần biến đếm đi 1; Biến đếm bằng 0 → sinh ngắt

Cơ chế dual-mode (hai chế độ)

- Hỗ trợ phần cứng cho việc tách biệt ít nhất hai chế độ thao tác
- User mode – thực thi dưới tư cách người dùng.
- Monitor mode (còn gọi là kernel mode hay system mode) – thực thi dưới tư cách hệ điều hành.
- Bit mode được thêm vào phần cứng để chỉ mode hiện thời: monitor (0) or user (1).
- Khi có ngắt hoặc là phát sinh lỗi, phần cứng được chuyển qua monitor mode

