**MỤC LỤC**

[**DANH SÁCH HÌNH ẢNH** 2](#_Toc515529045)

[**MỞ ĐẦU** 3](#_Toc515529046)

[**PHẦN I: NGUYÊN LÝ HỆ ĐIỀU HÀNH** 4](#_Toc515529047)

[**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT** 4](#_Toc515529048)

[I.Các thiết bị phần cứng liên quan đến tiêu thụ năng lượng điện: 4](#_Toc515529049)

[II.Các khía cạnh của hệ điều hành liên quan đến việc quản lý năng lượng: 10](#_Toc515529082)

[III.Chức năng của Hệ điều hành Windows trong việc tiết kiệm nguồn điện trên các máy tính cá nhân: 13](#_Toc515529103)

[IV.Kết luận: 15](#_Toc515529109)

[**PHẦN II: LẬP TRÌNH MẠNG** 15](#_Toc515529111)

[**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT** 15](#_Toc515529112)

[I.Giao thức TCP/IP: 15](#_Toc515529113)

[II.Mô hình Client-Server: 19](#_Toc515529119)

[III. Cơ chế Socket trong Java: 21](#_Toc515529120)

[IV. Mô hình truyền tin trong Socket: 22](#_Toc515529121)

[V. Một số hàm cơ bản trong Socket: 24](#_Toc515529122)

[**CHƯƠNG 2.PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG** 26](#_Toc515529123)

[I.Phân tích yêu cầu: 26](#_Toc515529124)

[II.Phân tích chức năng: 26](#_Toc515529125)

[**CHƯƠNG 3.TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ** 27](#_Toc515529126)

[I.Môi trường triển khai: 27](#_Toc515529127)

[II.Kết quả thực thi chương trình: 27](#_Toc515529128)

[**KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN** 28](#_Toc515529130)

[I.Những kết quả đạt được 28](#_Toc515529131)

[II.Những vấn đề tồn tại 28](#_Toc515529132)

[III.Hướng phát triển 28](#_Toc515529133)

[**TÀI LIỆU THAM KHẢO** 29](#_Toc515529134)

[**PHỤ LỤC** 30](#_Toc515529135)

# **DANH SÁCH HÌNH ẢNH**

Hình 1. Power consumption of various parts of a notebook computer

Hình 2. The use of zones for backlighting the display. (a) When window 2 is selected, it is not moved. (b) When window 1 is selected, it moves to reduce the number of zones illuminated.

Hình 3. Sơ đồ TCP/IP

Hình 4.Cấu trúc header cua TCP

Hình 5.Mô hình Client-Server

Hình 6. Client gửi yêu cầu kết nối tới Server

Hình 7. Server đồng ý kết nối và tiếp tục lắng nghe

Hình 8. Mô hình tương tác giữa client/server qua socket TCP

Hình 9. Kết quả

# **MỞ ĐẦU**

Đồ án là một trong những môn học chính của Khoa Công nghệ thông tin nói chung và môn học Nguyên lý Hệ điều hành cũng như Lập trình mạng nói riêng. Có thể coi đồ án là một cơ hội, một project hay một bài tập lớn mà các thầy cô giáo dành cho sinh viên của mình. Qua việc làm đồ án, chúng em sẽ:

1.Đi sâu vào nắm vững các một cách có hệ thống các kiến thức đã học được trong quá trình học lý thuyết, làm bài tập cũng như thực hành.

2.Từng bước làm quen với cách làm việc có định hướng, có kế hoạch của giáo viên hướng dẫn và tạo thói quen nghiên cứu độc lập, tìm tòi học hỏi qua các tài liệu tham khảo.

3.Đem quá trình học lý thuyết áp dụng vào công trình nghiên cứu thực tế.

4.Trình bày rõ ràng và khoa học một vấn đề thuộc lĩnh vực mà mình nghiên cứu.

Chính vì lẽ đó mà người làm đồ án môn học phải thực hiện và hoàn thành một khối công việc khá lớn thuộc các lĩnh vực chuyên sâu. Đặc biệt đối với đồ án lần này, hai lĩnh vực mà chúng em phải tìm hiểu là Nguyên lý hệ điều hành và Lập trình mạng. Đây là một đồ án quan trọng giúp chúng em thống kê lại những kiến thức về hai môn học đã và đang hoàn thành, tìm hiểu sâu hơn và vận dụng để làm các chương trình mô phỏng.

Báo cáo này là kết quả làm việc của cá nhân em trong thời gian vừa qua, sau một thời gian dài được giao và chọn đề tài. Mỗi phần trong báo cáo sẽ liên quan đến một lĩnh vực và một đề tài được giao.

Trong quá trình làm đề tài cũng như viết báo cáo, có thể sẽ xuất hiện một số lỗi và sai sót, do đó em rất mong đợi sự góp ý từ các thầy cô.

Em xin chân thành cảm ơn!

Sinh viên

**Lê Cẩm Tiên**

# **PHẦN I: NGUYÊN LÝ HỆ ĐIỀU HÀNH**

**TIÊU ĐỀ: TÌM HIỂU VIỆC QUẢN LÝ NGUỒN NĂNG LƯỢNG ĐIỆN CỦA MÁY TÍNH**

## **CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

### I.Các thiết bị phần cứng liên quan đến tiêu thụ năng lượng điện:

**1.Màn hình:**

**1.1.Màn hình máy tính loại CRT (Cathode Ray Tube):**

Đây là loại màn hình máy tính với nguyên lý ống phóng chùm điện tử (ống CRT, nên thường đặt tên cho loại này là "loại CRT").

a.Ưu điểm:

Thể hiện màu sắc rất trung thực, tốc độ đáp ứng cao, độ phân giải có thể đạt được cao. Phù hợp với games thủ và các nhà thiết kế, xử lý đồ hoạ.

b.Nhược điểm:

Chiếm nhiều diện tích,nặng, tiêu tốn điện năng hơn các loại màn hình khác, thường gây ảnh hưởng sức khoẻ nhiều hơn với các loại màn hình khác.

c.Nguyên lý hiển thị hình ảnh:

Màn hình CRT sử dụng phần màn huỳnh quang dùng để hiển thị các điểm ảnh, để các điểm ảnh phát sáng theo đúng màu sắc cần hiển thị cần các tia điện tử tác động vào chúng để tạo ra sự phát xạ ánh sáng. Ống phóng CRT sẽ tạo ra các tia điện tử đập vào màn huỳnh quang để hiển thị các điểm ảnh theo mong muốn.

**1.2.Màn hình máy tính loại khác:**

**1.2.1. Màn hình cảm ứng:**

Màn hình cảm ứng là các loại màn hình được tích hợp thêm một lớp cảm biến trên bề mặt để cho phép người sử dụng có thể điều khiển, làm việc với máy tính bằng cách sử dụng các loại bút riêng hoặc bằng tay giống như cơ chế điều khiển của một số điện thoại thông minh hay Pocket PC.

Màn hình cảm ứng xuất hiện ở một số máy tính xách tay cùng với [hệ điều hành](https://vi.wikipedia.org/wiki/H%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh) Windows 8. Một số máy tính cho các tụ điểm công cộng cũng sử dụng loại màn hình này phục vụ giải trí, mua sắm trực tuyến hoặc các mục đích khác - chúng được cài đặt hệ điều hành Windows Vista mới nhất.

**1.2.2. Màn hình máy tính sử dụng công nghệ OLED:**

Là công nghệ màn hình mới với xu thế phát triển trong tương lai bởi các ưu điểm: Cấu tạo mỏng, tiết kiệm năng lượng, đáp ứng nhanh, tuổi thọ cao... Về cơ bản, ngoại hình màn hình OLED thường giống màn hình tinh thể lỏng nhưng có kích thước mỏng hơn nhiều do không sử dụng đèn nền.

**2.Đĩa cứng:** (Tên tiếng Anh là: ***Hard Disk Drive***, viết tắt: ***HDD***)

Là thiết bị dùng để lưu trữ [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) trên bề mặt các tấm đĩa hình tròn phủ vật liệu [từ tính](https://vi.wikipedia.org/wiki/T%E1%BB%AB_t%C3%ADnh).

Ổ đĩa cứng là loại bộ nhớ "không thay đổi" (*non-volatile*), có nghĩa là chúng không bị mất [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) khi ngừng cung cấp nguồn [điện](https://vi.wikipedia.org/wiki/%C4%90i%E1%BB%87n) cho chúng.

Ổ đĩa cứng là một thiết bị rất quan trọng trong hệ thống bởi chúng chứa dữ liệu thành quả của một quá trình làm việc của những người sử dụng máy tính. Những sự hư hỏng của các thiết bị khác trong hệ thống [máy tính](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh) có thể sửa chữa hoặc thay thế được, nhưng [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) bị mất do yếu tố hư hỏng phần cứng của ổ đĩa cứng thường rất khó lấy lại được.

Ổ đĩa cứng là một khối duy nhất, các đĩa cứng được lắp ráp cố định trong ổ ngay từ khi [sản xuất](https://vi.wikipedia.org/wiki/S%E1%BA%A3n_xu%E1%BA%A5t) nên không thể thay thế được các "đĩa cứng" như với cách hiểu như đối với [ổ đĩa mềm](https://vi.wikipedia.org/wiki/%E1%BB%94_%C4%91%C4%A9a_m%E1%BB%81m) hoặc [ổ đĩa quang](https://vi.wikipedia.org/wiki/%E1%BB%94_%C4%91%C4%A9a_quang).

**2.1.Cấu tạo:**

**2.1.1. Cụm đĩa**:

Bao gồm toàn bộ các đĩa, trục quay và động cơ.

-Đĩa từ

-Trục quay: truyền chuyển động của đĩa từ

-Động cơ: được gắn đồng trục với trục quay và các đĩa

**2.1.2.Cụm đầu đọc:**

**-**Đầu đọc (head): Đầu đọc/ghi dữ liệu

-Cần di chuyển đầu đọc (*head arm* hoặc *actuator arm*).

**2.1.3.Cụm mạch điện:**

-Mạch điều khiển: có nhiệm vụ điều khiển động cơ đồng trục, điều khiển sự di chuyển của cần di chuyển đầu đọc để đảm bảo đến đúng vị trí trên bề mặt đĩa.

-Mạch xử lý dữ liệu: dùng để xử lý những [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) đọc/ghi của ổ đĩa cứng.

-Bộ nhớ đệm (*cache* hoặc *buffer*): là nơi tạm lưu [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) trong quá trình đọc/ghi dữ liệu. Dữ liệu trên bộ nhớ đệm sẽ mất đi khi ổ đĩa cứng ngừng được cấp [điện](https://vi.wikipedia.org/wiki/%C4%90i%E1%BB%87n).

-Đầu cắm nguồn cung cấp điện cho ổ đĩa cứng.

-Đầu kết nối giao tiếp với máy tính

-Các cầu đấu thiết đặt (tạm dịch từ *jumper*) thiết đặt chế độ làm việc của ổ đĩa cứng: Lựa chọn chế độ làm việc của ổ đĩa cứng (SATA 150 hoặc SATA 300) hay thứ tự trên các kênh trên giao tiếp IDE (master hay slave hoặc tự lựa chọn), lựa chọn các thông số làm việc khác...

**2.1.4.Vỏ đĩa cứng:**

Vỏ ổ đĩa cứng gồm các phần: Phần đế chứa các [linh kiện](https://vi.wikipedia.org/w/index.php?title=Linh_ki%E1%BB%87n&action=edit&redlink=1) gắn trên nó, phần nắp đậy lại để bảo vệ các linh kiện bên trong, có chức năng chính nhằm định vị các linh kiện và đảm bảo độ kín khít để không cho phép bụi được lọt vào bên trong của ổ đĩa cứng.

Ngoài ra, vỏ đĩa cứng còn có tác dụng chịu đựng sự va chạm (ở mức độ thấp) để bảo vệ ổ đĩa cứng.

**3.CPU: (**Tên tiếng Anh là ***Central Processing Unit***, tạm dịch là ***bộ xử lý trung tâm*)**

Là các mạch điện tử trong một [máy tính](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh), thực hiện các [câu lệnh](https://vi.wikipedia.org/wiki/Ki%E1%BA%BFn_tr%C3%BAc_t%E1%BA%ADp_l%E1%BB%87nh) của [chương trình máy tính](https://vi.wikipedia.org/wiki/Ch%C6%B0%C6%A1ng_tr%C3%ACnh_m%C3%A1y_t%C3%ADnh) bằng cách thực hiện các phép tính số học, logic, so sánh và các hoạt động nhập/xuất dữ liệu (I/O) cơ bản do mã lệnh chỉ ra.

Các hoạt động cơ bản của hầu hết các CPU, không phụ thuộc hình thức vật lý, là thực hiện một chuỗi các [tập lệnh](https://vi.wikipedia.org/wiki/Ki%E1%BA%BFn_tr%C3%BAc_t%E1%BA%ADp_l%E1%BB%87nh) được lưu trữ, gọi là chương trình. Các mã lệnh chờ thực hiện này được lưu giữ trong một số loại bộ nhớ máy tính. Gần như tất cả các CPU có các bước: lấy thông tin, giải mã và thực hiện lệnh khi hoạt động, và được gọi chung là chu kỳ lệnh.

**3.1.Thành phần:**

**3.1.1.Khối điều khiển (Control Unit-CU):**

Là thành phần của CPU có nhiệm vụ thông dịch các lệnh của chương trình và điều khiển hoạt động xử lý, được điều tiết chính xác bởi xung nhịp đồng hồ hệ thống.Phần này là phần cốt lõi của một bộ xử lý được cấu tạo từ các mạch logic so sánh với các linh kiện bán dẫn như [transistor](https://vi.wikipedia.org/wiki/Transistor) tạo thành.

### 3.1.2.Khối tính toán ALU ([Arithmetic logic unit](https://en.wikipedia.org/wiki/Arithmetic_logic_unit)):

### Chức năng thực hiện các phép toán số học và logic sau đó trả lại kết quả cho các thanh ghi hoặc bộ nhớ.

### 3.1.3.Các thanh ghi (Registers):

### Là các bộ nhớ có dung lượng nhỏ nhưng tốc độ truy cập rất cao, nằm ngay trong CPU, dùng để lưu trữ tạm thời các toán hạng, kết quả tính toán, địa chỉ các ô nhớ hoặc thông tin điều khiển. Mỗi thanh ghi có một chức năng cụ thể. Thanh ghi quan trọng nhất là bộ đếm chương trình (PC - Program Counter) chỉ đến lệnh sẽ thi hành tiếp theo.

### 3.1.4.Opcode:

### Phần bộ nhớ chứa mã máy của cpu(không bắt buộc) để có thể thực thi các lệnh trong file thực thi.

### 3.1.5.Phần điều khiển:

### Thực hiện việc điều khiển các khối và điều khiển tần số xung nhịp. Mạch xung nhịp đồng hồ hệ thống dùng để đồng bộ các thao tác xử lý trong và ngoài CPU theo các khoảng thời gian không đổi. Khoảng thời gian chờ giữa hai xung gọi là [chu kỳ](https://vi.wikipedia.org/wiki/Chu_k%E1%BB%B3) xung nhịp. Tốc độ theo đó xung nhịp [hệ thống](https://vi.wikipedia.org/wiki/H%E1%BB%87_th%E1%BB%91ng) tạo ra các xung [tín hiệu](https://vi.wikipedia.org/wiki/T%C3%ADn_hi%E1%BB%87u) chuẩn thời gian gọi là tốc độ xung nhịp – tốc độ đồng hồ tính bằng triệu đơn vị mỗi giây (MHz).Phần này là không cần thiết cho một cpu nhưng hầu hết có trong kiến trúc [cisc](https://vi.wikipedia.org/w/index.php?title=Cisc&action=edit&redlink=1).

### 4.Bộ nhớ: (Tên tiếng Anh là *Computer data storage*)

### Thường được gọi là ổ nhớ (*storage*) hoặc bộ nhớ (*memory*), là một thiết bị công nghệ bao gồm các phần tử máy tính và lưu trữ dữ liệu, được dùng để duy trì dữ liệu số. Nó là một linh kiện cơ bản có chức năng cốt lõi của các máy tính.

### Bộ nhớ máy tính bao gồm các [bộ nhớ điện tĩnh](https://vi.wikipedia.org/wiki/B%E1%BB%99_nh%E1%BB%9B_%C4%91i%E1%BB%87n_t%C4%A9nh) (*non-volatile memory*) để lưu trữ được dữ liệu của [máy tính](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh) một cách lâu dài (khi kết thúc một phiên làm việc của [máy tính](https://vi.wikipedia.org/wiki/M%C3%A1y_t%C3%ADnh) thì dữ liệu không bị mất đi), hoặc [bộ nhớ điện động](https://vi.wikipedia.org/wiki/B%E1%BB%99_nh%E1%BB%9B_%C4%91i%E1%BB%87n_t%C4%A9nh) (*volatile memory*) để lưu dữ liệu tạm thời trong quá trình làm việc của máy tính (khi kết thúc một phiên làm việc của máy tính thì bộ nhớ này bị mất hết [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u)).

Các thiết bị lưu trữ [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) cho bộ nhớ lâu dài bao gồm: [Đĩa cứng](https://vi.wikipedia.org/wiki/%E1%BB%94_%C4%91%C4%A9a_c%E1%BB%A9ng), [Đĩa mềm](https://vi.wikipedia.org/wiki/%C4%90%C4%A9a_m%E1%BB%81m), [Đĩa quang](https://vi.wikipedia.org/wiki/%C4%90%C4%A9a_quang), Băng từ, [ROM](https://vi.wikipedia.org/wiki/B%E1%BB%99_nh%E1%BB%9B_ch%E1%BB%89_%C4%91%E1%BB%8Dc), các loại bút nhớ...

Các thiết bị lưu trữ [dữ liệu](https://vi.wikipedia.org/wiki/D%E1%BB%AF_li%E1%BB%87u) tạm thời trong quá trình làm việc: [RAM](https://vi.wikipedia.org/wiki/RAM) máy tính, [Cache](https://vi.wikipedia.org/wiki/Cache)...

Hầu hết các bộ nhớ nêu trên thuộc loại bộ nhớ có thể truy cập dữ liệu ngẫu nhiên, riêng băng từ là loại bộ nhớ truy cập tuần tự.

Bộ nhớ máy tính có thể chia thành hai dạng: Bộ nhớ trong (main memory) và bộ nhớ ngoài (secondary memory).

### 4.1.Bộ nhớ trong:

### Là các loại bộ nhớ nằm nội bộ bên trong thùng máy. Còn có tên gọi khác là bộ nhớ chính (*Main Memory*)

### 4.1.1.Bộ nhớ đệm nhanh (Cache memory):

### - Tốc độ truy xuất nhanh

### - Thường nằm trong CPU, một số cache cũ có thể nằm ngoài CPU: như các cache trên đế cắm kiểu slot 1, hoặc cache dạng thanh, có thể tháo rời giống như các thanh [RAM](https://vi.wikipedia.org/wiki/RAM) ngày nay

### - Bao gồm Cache L1 và Cache L2, Cache L3 (*L3 chỉ có ở một số CPU*) có tốc độ truy xuất gần bằng tốc độ truyền dữ liệu trong [CPU](https://vi.wikipedia.org/wiki/CPU)

### 4.1.2.Bộ nhớ chính (Main memory):

### - Bộ nhớ [RAM](https://vi.wikipedia.org/wiki/RAM) (*Random Access Memory*), hay [Bộ nhớ truy cập ngẫu nhiên](https://vi.wikipedia.org/wiki/RAM): Tốc độ truy cập nhanh, lưu trữ dữ liệu tạm thời, dữ liệu sẽ bị mất đi khi bị cắt nguồn điện

**-** Bộ nhớ [ROM](https://vi.wikipedia.org/wiki/B%E1%BB%99_nh%E1%BB%9B_ch%E1%BB%89_%C4%91%E1%BB%8Dc) (*Read Only Memory*), hay [Bộ nhớ chỉ đọc](https://vi.wikipedia.org/wiki/B%E1%BB%99_nh%E1%BB%9B_ch%E1%BB%89_%C4%91%E1%BB%8Dc): Lưu trữ các chương trình mà khi mất nguồn điện cung cấp sẽ không bị (xóa) mất. Ngày nay còn có công nghệ [FlashROM](https://vi.wikipedia.org/w/index.php?title=FlashROM&action=edit&redlink=1) tức bộ nhớ [ROM](https://vi.wikipedia.org/wiki/ROM) không những chỉ đọc mà còn có thể ghi lại được, nhờ có công nghệ này BIOS được cải tiến thành [FlashBIOS](https://vi.wikipedia.org/w/index.php?title=FlashBIOS&action=edit&redlink=1).

**4.2.Bộ nhớ ngoài:**

Là bộ nhớ máy tính gắn bên ngoài thùng máy, có thể dùng để mang đi lại được.

### Bao gồm:

### -Bộ nhớ từ: đĩa cứng, đĩa mềm,...

### -Bộ nhớ quang: CD, DVD, …

### -Bộ nhớ bán dẫn:  flash disk, thẻ nhớ...

### -Các loại bộ nhớ dựa trên công nghệ FlashROM: Kết hợp với chuẩn giao tiếp máy tính [USB](https://vi.wikipedia.org/wiki/USB) (*Universal Serial Bus*) tạo ra các bộ nhớ máy tính di động thuận tiện và đa năng như: Các thiết bị giao tiếp USB lưu trữ dữ liệu, thiết bị giao tiếp USB chơi nhạc số, chơi video số; khóa bảo mật qua giao tiếp [USB](https://vi.wikipedia.org/wiki/USB); thẻ nhớ... Dung lượng thiết bị lưu trữ FlashROMđã lên tới 32GB ([Samsung](https://vi.wikipedia.org/wiki/Samsung),[Intel](https://vi.wikipedia.org/wiki/Intel) công bố năm 2005), trong tương lai, có thể FlashROM sẽ dần thay thế các ổ đĩa cứng, các loại đĩa CD, DVD,…

### 5.Nhận xét:

### Khi nghĩ đến năng lượng điện trên các máy tính, chúng ta vẫn thường nghĩ đến pin, nhưng pin không có đủ năng lượng để cấp nguồn cho các máy tính làm việc trong nhiều giờ liền. Cách phương pháp mà hầu hết các nhà cung cấp máy tính sử dụng để bảo tồn pin là thiết kế các thiết bị CPU, bộ nhớ và I/O có nhiều trạng thái: on, sleeping, hibernating, and off. Khi thiết bị không được sử dụng trong một thời gian ngắn, nó sẽ được tự động chuyển sang chế độ sleep để giảm thiểu sự tiêu hao năng lượng. Khi nó không được sử dụng trong một khoảng thời gian dài hơn, nó có thể được chuyển sang trạng thái hibernating (ngủ đông), nhằm giảm tiêu thụ năng lượng nhiều hơn. Nhược điểm ở đây là việc khôi phục lại thiết bị khỏi trạng thái hibernating thường tốn nhiều thời gian và năng lượng hơn là khôi phục khỏi trạng thái sleep. Cuối cùng, khi một thiết bị tắt, nó không làm gì và không tiêu thụ năng lượng. Không phải tất cả các thiết bị đều có tất cả các trạng thái này, nhưng khi chúng thực hiện, nó phụ thuộc vào hệ điều hành để quản lý các trạng thái chuyển tiếp vào đúng thời điểm.

### Một số máy tính có hai hoặc thậm chí ba trạng thái khi tắt nguồn. Một trong số đó có thể đặt toàn bộ máy tính trong trạng thái sleep, từ đó nó có thể được bật lại nhanh chóng bằng cách gõ một kí tự hoặc di chuyển chuột. Trường hợp khác, có thể đặt máy tính vào chế độ ngủ đông, từ đó việc bật lại sẽ mất nhiều thời gian hơn. Trong cả hai trường hợp, các nút này thường không làm gì ngoại trừ việc gửi tín hiệu đến hệ điều hành.

### Việc quản lý năng lượng sẽ đặt ra một số vấn đề mà hệ điều hành phải giải quyết. Một trong số đó có liên quan đến trạng thái ngủ đông có chọn lọc và tạm thời tắt thiết bị hoặc ít nhất giảm mức tiêu thụ điện năng khi máy tính đang nhàn rỗi. Các vấn đề được đặt ra bao gồm:

### Những loại phần cứng nào có thể được điều khiển? Chúng có bật / tắt hay có trạng thái trung gian không? Bao nhiêu năng lượng thì được đưa vào trong các trạng thái năng lượng thấp? Năng lượng có bị tiêu hao để khởi động lại thiết bị không? Mất bao lâu để nạp lại năng lượng pin của máy tính?....

### Nhiều nhà khoa học nghiên cứu máy tính xách tay để xem nguồn điện tiêu hao đi đâu? Li et al. (1994) cũng như Lorch và Smith (1998) đã thực hiện các phương pháp đo trên các máy tính khác nhau và đưa kết luận như hình 1. Tuy nhiên, họ chỉ nhận định rằng ba nguồn năng lượng hàng đầu là màn hình, đĩa cứng và CPU. Mặc dù những con số này không đồng nhất, nhưng có thể bởi vì các máy tính đến từ các hãng khác nhau của nên có yêu cầu năng lượng khác nhau, có vẻ như rõ ràng rằng màn hình hiển thị, đĩa cứng và CPU là những mục tiêu để tiết kiệm năng lượng.

### 

### 

### *Hình 1. Power consumption of various parts of a notebook computer*

### II.Các khía cạnh của hệ điều hành liên quan đến việc quản lý năng lượng:

### Hệ điều hành đóng một vai trò quan trọng trong quản lý năng lượng. Nó kiểm soát tất cả các thiết bị, do đó, nó phải quyết định tắt những gì và khi nào thì tắt. Nếu nó tắt thiết bị và thiết bị đó lại cần sử dụng một cách nhanh chóng, có thể sẽ mất một quãng thời gian khá lâu khi khởi động lại gây phiền nhiễu. Mặt khác, nếu mất nhiều thời gian để tắt một thiết bị, năng lượng sẽ bị lãng phí.

### 1.Màn hình:

### Một trong những thành phần có lẽ tiêu tốn nhiều năng lượng điện nhất có lẽ phải nói đến màn hình. Để có một hình ảnh sắc nét, tươi sáng, màn hình phải được chiếu sáng và phải mất một lượng năng lượng đáng kể. Nhiều hệ điều hành cố gắng tiết kiệm năng lượng ở đây bằng cách tắt màn hình khi không có hoạt động nào trong vài phút. Thường thì người dùng có thể quyết định khoảng thời gian tắt máy là bao lâu. Tắt màn hình là một trạng thái sleep vì nó có thể được khởi động lại (từ RAM) gần như ngay lập tức khi bất kỳ phím nào được nhấn hoặc con trỏ chuột được di chuyển.

### Một cải tiến có thể được đề xuất bởi Flinn và Satyanarayanan (2004). Họ đề xuất có màn hình hiển thị bao gồm một số khu vực có thể được cung cấp độc lập lên hoặc xuống. Trong hình 2, chúng tôi mô tả 16 vùng, sử dụng dấu gạch ngang dòng để tách chúng. Khi con trỏ nằm trong cửa sổ 2, như trong hình 2(a), chỉ có bốn khu vực ở góc dưới bên phải phải được thắp sáng. Cửa sổ thứ 12 có thể tối, tiết kiệm 3/4 công suất màn hình. Khi người dùng di chuyển con trỏ đến cửa sổ 1, các vùng cho cửa sổ 2 có thể là tối và các khu vực phía sau cửa sổ 1 có thể được bật lên. Tuy nhiên, vì cửa sổ 1 nằm trong 9 khu vực, cần nhiều năng lượng hơn. Nếu trình quản lý cửa sổ có thể cảm nhận được những gì đang xảy ra, nó có thể tự động di chuyển cửa sổ 1 để vừa với bốn vùng, với một loại hành động theo vùng, như trong Hình 2(b).

### Để giảm từ 9/16 toàn bộ năng lượng đến còn 4/16 toàn bộ năng lượng, trình quản lý cửa sổ phải hiểu quản lý năng lượng hoặc có khả năng chấp nhận hướng dẫn từ một số phần khác của hệ thống. Thậm chí sẽ là khả năng chiếu sáng một phần cửa sổ không hoàn toàn đầy đủ (ví dụ: cửa sổ chứa dòng văn bản ngắn có thể được giữ ở phía bên tay phải).

### 

### 

### *Hình 2. The use of zones for backlighting the display. (a) When window 2 is selected, it is not moved. (b) When window 1 is selected, it moves to reduce the number of zones illuminated.*

### 2.Đĩa cứng:

### Một phần khác cũng không kém trong việc tiêu hao năng lượng là Đĩa cứng. Phải mất năng lượng đáng kể để giữ nó quay ở tốc độ cao, ngay cả khi không có kết nối. Nhiều máy tính, đặc biệt máy tính xách tay, quay đĩa xuống sau một số phút nhất định không hoạt động. Khi cần sử dụng, nó được kéo lên trở lại. Điều không tốt là đĩa sẽ rơi vào trạng thái ngủ đông thay vì ngủ vì phải mất một vài giây để quay lại nó, điều này gây ra việc phải mất thời gian chờ đợi cho người dùng. Ngoài ra, khởi động lại đĩa cũng mất đi một phần năng lượng đáng kể. Kết quả là, mỗi đĩa sẽ có một thời gian đặc trưng, ​​Td, đó là điểm hòa vốn, thường là trong khoảng từ 5 đến 15 giây. Giả sử rằng kết nối đĩa tiếp theo dự kiến ​​sẽ mất một thời gian t trong tương lai. Nếu t <Td, sẽ mất ít năng lượng hơn để giữ cho đĩa quay thay vì quay nó xuống và sau đó xoay nó nhanh lên. Nếu t> Td, năng lượng được tiết kiệm làm cho nó quay đĩa xuống và sau đó lên một lần nữa. Trong thực tế, hầu hết các hệ thống được bảo vệ và dừng đĩa chỉ sau vài phút không hoạt động. Một cách khác để tiết kiệm năng lượng ở đĩa cứng là có một bộ đệm đĩa trong RAM. Nếu ghi vào đĩa có thể được đệm trong bộ nhớ cache, đĩa không phải khởi động lại chỉ để xử lý ghi. Đĩa có thể vẫn tắt cho đến khi bộ nhớ cache đầy hoặc một lỗi bỏ lỡ xảy ra.

### Một cách khác để tránh khởi động đĩa không cần thiết là cho hệ điều hành tiếp tục chạy các chương trình được thông báo về trạng thái đĩa bằng cách gửi tin nhắn hoặc tín hiệu. Một số chương trình có ghi tùy ý có thể bị bỏ qua hoặc bị trì hoãn.

### 3.CPU:

### CPU cũng có thể được quản lý để tiết kiệm năng lượng. CPU trong một máy tính xách tay ở trạng thái sleep có thể khiến việc tiêu thụ năng lượng gần như bằng 0. Điều duy nhất nó có thể làm trạng thái này là khởi động lại khi xảy ra gián đoạn. Do đó, bất cứ khi nào CPU không hoạt động, hoặc chờ I/O hoặc vì không có việc gì để làm, nó sleep.

### Trên nhiều máy tính, có một mối quan hệ giữa điện áp CPU, chu kỳ đồng hồ, và sử dụng năng lượng. Điện áp CPU thường có thể bị giảm trong phần mềm, không những giúp tiết kiệm năng lượng mà còn làm giảm chu kỳ đồng hồ (xấp xỉ tuyến tính). Vì tiêu thụ điện năng tỷ lệ thuận với bình phương của điện áp, giảm điện áp làm đôi khiến cho CPU nhanh gấp đôi nhưng chỉ tốn 1/4 năng lượng.

### Tính chất này có thể được khai thác cho các chương trình với thời hạn được xác định rõ. Ví dụ như nếu người dùng đang gõ 1 kí tự/giây, nhưng công việc cần thiết để xử lý kí tự mất 100 msec, nó tốt hơn cho hệ điều hành để phát hiện thời gian nghỉ dài và làm chậm CPU xuống theo hệ số 10. Trong thời gian ngắn, chạy từ từ tiết kiệm năng lượng hơn là chạy nhanh. Việc giảm tỷ lệ lõi CPU không phải lúc nào cũng làm cho CPU giảm hoạt động.

### 4.Bộ nhớ:

### Tồn tại hai lựa chọn có thể tiết kiệm năng lượng với bộ nhớ. Đầu tiên, bộ nhớ cache có thể được xả và sau đó tắt. Nó luôn luôn có thể được tải lại từ bộ nhớ chính mà không làm mất thông tin. Việc tải lại có thể được thực hiện linh hoạt và nhanh chóng, vì vậy khi tắt bộ nhớ cache là đang rơi vào trạng thái ngủ. Một lựa chọn quyết liệt hơn là viết nội dung của bộ nhớ chính vào đĩa, sau đó tắt chính bộ nhớ chính. Cách này gọi là ngủ đông, vì hầu như tất cả năng lượng có thể được cắt giảm vào bộ nhớ, tuy nhiên nó mất một thời gian tải lại đáng kể, đặc biệt là nếu đĩa bị tắt, quá tải. Khi bộ nhớ bị cắt, CPU phải được tắt cũng như phải thực thi ROM. Nếu CPU tắt, ngắt sẽ đánh thức nó lên đã làm cho nó nhảy vào mã trong ROM để bộ nhớ có thể được nạp lại trước khi được sử dụng.

### 5.Truyền thông không dây:

### Ngày càng nhiều máy tính xách tay có kết nối không dây với bên ngoài thế giới (ví dụ: Internet). Thiết bị phát và thu vô tuyến được yêu cầu thường là hạng nhất. Đặc biệt, nếu máy thu radio luôn bật để lắng nghe email đến, pin có thể tiêu hao khá nhanh. Mặt khác, nếu đài phát thanh bị tắt, ví dụ, 1 phút không hoạt động, các tin nhắn đến có thể bị bỏ qua, điều này rõ ràng là không mong muốn. Một giải pháp hiệu quả cho vấn đề này đã được Kravets và Krishnan đề xuất (1998). Điểm chính của giải pháp mà họ khai thác thực tế là máy tính di động giao tiếp với các trạm gốc cố định có bộ nhớ lớn và đĩa mà không ràng buộc năng lượng. Những gì họ đề xuất là để máy tính di động gửi tin nhắn đến trạm gốc khi nó sắp tắt radio. Từ thời điểm đó, trạm cơ sở lưu các thông điệp gửi đến trên đĩa của nó. Máy tính di động có thể cho biết rõ ràng nó đang lên kế hoạch ngủ bao lâu, hoặc đơn giản là thông báo cho trạm gốc khi nó bật lại radio. Tại thời điểm đó, bất kỳ tin nhắn tích lũy nào có thể được gửi đến nó. Các tin nhắn gửi đi được tạo ra trong khi radio bị tắt được đệm trên máy tính di động. Nếu bộ đệm bị lấp đầy, radio sẽ được bật và hàng đợi được truyền đến trạm gốc. Khi nào nên tắt radio? Một khả năng là cho phép người dùng hoặc chương trình ứng dụng tự quyết định. Một cách khác là tắt nó sau một vài giây nghỉ. Khi nào nó nên được bật lại? Một lần nữa, người dùng hoặc chương trình có thể quyết định hoặc có thể được bật định kỳ để kiểm tra lưu lượng truy cập đến và truyền tải bất kỳ tin nhắn xếp hàng nào. Tất nhiên, nó cũng nên được bật khi bộ đệm đầu ra gần đầy.

### 6.Quản lý pin:

### Trước đây, pin chỉ được cung cấp từ hiện tại cho đến khi hết hoàn toàn, tại thời gian nó dừng lại. Không còn nữa. Bây giờ thiết bị di động sử dụng pin thông minh, có thể giao tiếp với hệ điều hành. Theo yêu cầu từ hoạt động hệ thống, nó có thể báo cáo về những thứ như điện áp tối đa, điện áp hiện tại, phí tối đa, phí hiện tại, tốc độ thoát tối đa, tốc độ xả hiện tại và hơn thế nữa. Hầu hết các thiết bị di động đều có các chương trình có thể chạy để truy vấn và hiển thị tất cả các tham số này. Pin thông minh cũng có thể được hướng dẫn để thay đổi nhiều hoạt động khác nhau đối với các thông số dưới sự kiểm soát của hệ điều hành. Một số máy tính xách tay có nhiều pin. Khi hệ điều hành phát hiện rằng một pin sắp hết, nó phải sắp xếp cho một lần cắt giảm, mà không gây ra bất kỳ trục trặc nào trong quá trình chuyển đổi. Khi pin cuối cùng bật chân cuối cùng của nó, nó vào hệ điều hành để cảnh báo người dùng và sau đó gây ra một thứ tự tắt máy, ví dụ, đảm bảo rằng hệ thống tập tin không bị hỏng.

### III.Chức năng của Hệ điều hành Windows trong việc tiết kiệm nguồn điện trên các máy tính cá nhân:

### Lịch sử quản lý tiêu thụ điện năng bao gồm cả việc tắt màn hình hiển thị và dừng ổ đĩa quay. Các cơ sở quản lý năng lượng mới hơn bao gồm giảm tiêu thụ điện năng các thành phần khi hệ thống không được sử dụng, bằng cách chuyển các thiết bị riêng lẻ sang trạng thái chờ hoặc thậm chí tắt nguồn hoàn toàn bằng các công tắc nguồn mềm. Đa xử lí sẽ tắt các CPU riêng lẻ khi chúng không cần thiết và thậm chí tốc độ xung nhịp của CPU đang chạy có thể được điều chỉnh xuống để giảm công suất tiêu thụ. Khi bộ xử lý không hoạt động, mức tiêu thụ điện năng cũng giảm vì nó không cần phải làm gì ngoài việc chờ một sự gián đoạn xảy ra. Windows hỗ trợ chế độ tắt đặc biệt gọi là ngủ đông, sao chép tất cả bộ nhớ vật lý vào đĩa và sau đó giảm mức tiêu thụ điện năng xuống một lượng nhỏ để ít hao pin. Bởi vì tất cả trạng thái bộ nhớ được ghi vào đĩa, bạn thậm chí có thể thay pin một máy tính xách tay trong khi nó đang ngủ đông. Khi hệ thống tiếp tục sau khi ngủ đông, nó khôi phục trạng thái bộ nhớ đã lưu (và khởi động lại các thiết bị I / O). Điều này giúp máy tính trở lại trạng thái tương tự trước khi ngủ đông, mà không cần phải đăng nhập lại và khởi động tất cả các ứng dụng và dịch vụ đang chạy. Windows tối ưu hóa quá trình này bằng cách bỏ qua các trang chưa sửa đổi được sao lưu bằng đĩa và nén các trang bộ nhớ khác để giảm lượng băng thông I / O cần thiết. Thuật toán ngủ đông tự động tự điều chỉnh để cân bằng giữa I / O và thông lượng bộ vi xử lý.

### Với thế hệ đa bộ xử lý hiện tại, cả chế độ ngủ đông và tiếp tục có thể được thực hiện trong vài giây ngay cả trên các hệ thống có nhiều gigabyte RAM. Một thay thế cho chế độ ngủ đông là chế độ chờ, nơi trình quản lý nguồn giảm toàn bộ hệ thống xuống trạng thái nguồn thấp nhất có thể, chỉ sử dụng đủ năng lượng để làm mới RAM động. Bởi vì bộ nhớ không cần phải được sao chép vào đĩa, điều này hơi nhanh hơn ngủ đông trên một số hệ thống. Mặc dù có sẵn chế độ ngủ đông và chế độ chờ, nhiều người dùng vẫn ở trong thói quen tắt máy tính của họ khi họ làm việc xong. Windows sử dụng chế độ ngủ đông để thực hiện tắt và khởi động giả, được gọi là HiberBoot, nhanh hơn nhiều so với tắt và khởi động bình thường. Khi người dùng yêu cầu hệ thống tắt máy, HiberBoot ghi lại người dùng và sau đó ngủ đông hệ thống tại thời điểm họ sẽ thường đăng nhập lại. Sau đó, khi người dùng bật lại hệ thống, HiberBoot sẽ tiếp tục hệ thống tại điểm đăng nhập. Đối với người dùng, có vẻ như tắt máy là rất, rất nhanh vì hầu hết các bước khởi tạo hệ thống đều bị bỏ qua.

### Tất nhiên, đôi khi hệ thống cần thực hiện tắt máy thực để sửa lỗi vấn đề hoặc cài đặt bản cập nhật cho hạt nhân. Nếu hệ thống được yêu cầu khởi động lại hơn là tắt máy, hệ thống phải thực sự tắt máy và thực hiện khởi động bình thường.

### Trên điện thoại và máy tính bảng, cũng như thế hệ máy tính xách tay, máy tính mới nhất các thiết bị dự kiến ​​sẽ luôn luôn tiêu thụ ít năng lượng. Để cung cấp nhiều trải nghiệm hiện đại, Windows triển khai phiên bản quản lý nguồn điện đặc biệt được gọi là CS (kết nối dự phòng). CS có thể thực hiện trên các hệ thống có phần cứng mạng đặc biệt có khả năng nghe lưu lượng truy cập trên một tập hợp nhỏ các kết nối bằng ít điện năng hơn nếu CPU đang chạy. Hệ thống CS luôn xuất hiện bật, sắp ra khỏi CS ngay sau khi màn hình được bật bởi người dùng. Đã kết nối chế độ chờ khác với chế độ chờ thông thường vì hệ thống CS cũng sẽ thoát ra khỏi chế độ chờ khi nó nhận được một gói tin trên một kết nối được giám sát. Nếu tình trạng pin bắt đầu chạy thấp, hệ thống CS sẽ chuyển sang trạng thái ngủ đông tránh hết pin và có thể mất dữ liệu người dùng.

### Nhiều ứng dụng ngày nay được triển khai với cả mã và dịch vụ cục bộ trong đám mây. Windows cung cấp WNS (Dịch vụ thông báo của Windows) cho phép dịch vụ của bên thứ ba để đẩy thông báo tới thiết bị Windows trong CS mà không yêu cầu phần cứng mạng CS để lắng nghe cụ thể các gói từ máy chủ của bên. Thông báo WNS có thể báo hiệu các sự kiện quan trọng thời gian, chẳng hạn như sự xuất hiện của tin nhắn văn bản hoặc cuộc gọi VoIP. Khi một gói WNS đến, bộ xử lý sẽ phải được bật để xử lý nó, nhưng khả năng của phần cứng mạng CS phân biệt giữa lưu lượng truy cập từ các kết nối khác nhau có nghĩa là bộ vi xử lý không cần phải đánh thức mọi gói dữ liệu ngẫu nhiên đến giao diện mạng.

### IV.Kết luận:

### Việc quản lý nguồn năng lượng điện trong máy tính là một việc vô cùng quan trọng. Hiện nay lượng người dung máy tính càng nhiều đòi hỏi các nhà sản xuất phải cải tiến càng ngày càng tốt hơn sản phẩm của mình. Vì thế, việc tiết kiệm nguồn năng lượng điện của máy tính cũng được chú trọng và nâng cao hơn.

# **PHẦN II: LẬP TRÌNH MẠNG**

**TIÊU ĐỀ: SỬ DỤNG SOCKET TRONG JAVA XÂY DỰNG CHƯƠNG TRÌNH TRAO ĐỔI THÔNG TIN THEO MÔ HÌNH CLIENT-SERVER**

## **CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

### I.Giao thức TCP/IP:

TCP/IP là tên chung cho một tập hợp hơn 100 giao thức được sử dụng để kết nối các máy tính vào mạng, trong đó hai giao thức chính là TCP (Transmission Control Protocol) và IP (Internet Protocol). Trong phạm vi Internet, thông tin không được truyền tải như một dòng riêng biệt từ máy tính này tới máy tính khác. Thay vào đó, dữ liệu được chia thành những gói nhỏ gọi là packet.

### Các packet này được gửi trên mạng máy tính. Công việc của IP là chuyển chúng đến các máy tính ở xa. Tại trạm cuối, TCP nhận các packet và kiểm tra lỗi. Nếu một lỗi xuất hiện, TCP yêu cầu gói riêng biệt đó phải được gửi lại. Chỉ khi tất cả các packet đã nhận được là đúng, TCP sẽ sử dụng số thứ tự để tạo lại thông tin ban đầu.

*Hình 3.* *Sơ đồ TCP/IP*

**1.Giao thức IP(Internet Protocol - Giao thức Liên mạng):**

Là một giao thức hướng dữ liệu được sử dụng bởi các máy chủ nguồn và đích để truyền dữ liệu trong một liên mạng chuyển mạch gói.

Dữ liệu trong một liên mạng IP được gửi theo các khối được gọi là các gói (packet hoặc datagram). Cụ thể, IP không cần thiết lập các đường truyền trước khi một máy chủ gửi các gói tin cho một máy khác mà trước đó nó chưa từng liên lạc với.

Giao thức IP cung cấp một dịch vụ gửi dữ liệu không đảm bảo (còn gọi là cố gắng cao nhất), nghĩa là nó hầu như không đảm bảo gì về gói dữ liệu.

Gói dữ liệu có thể đến nơi mà không còn nguyên vẹn, nó có thể đến không theo thứ tự (so với các gói khác được gửi giữa hai máy nguồn và đích đó), nó có thể bị trùng lặp hoặc bị mất hoàn toàn. Nếu một phần mềm ứng dụng cần được bảo đảm, nó có thể được cung cấp từ nơi khác, thường từ các giao thức giao vận nằm phía trên IP.

Các thiết bị định tuyến liên mạng chuyển tiếp các gói tin IP qua các mạng tầng liên kết dữ liệu được kết nối với nhau. Việc không có đảm bảo về gửi dữ liệu có nghĩa rằng các chuyển mạch gói có thiết kế đơn giản hơn. (Lưu ý rằng nếu mạng bỏ gói tin, làm đổi thứ tự hoặc làm hỏng nhiều gói tin, người dùng sẽ thấy hoạt động mạng trở nên kém đi. Hầu hết các thành phần của mạng đều cố gắng tránh để xảy ra tình trạng đó. Đó là lý do giao thức này còn được gọi là cố gắng cao nhất. Tuy nhiên, khi lỗi xảy ra không thường xuyên sẽ không có hiệu quả đủ xấu đến mức người dùng nhận thấy được.)

Giao thức IP rất thông dụng trong mạng Internet công cộng ngày nay. Giao thức tầng mạng thông dụng nhất ngày nay là IPv4; đây là giao thức IP phiên bản 4. IPv6 được đề nghị sẽ kế tiếp IPv4: Internet đang hết dần địa chỉ IPv4, do IPv4 sử dụng 32 bit để đánh địa chỉ (tạo được khoảng 4 tỷ địa chỉ); IPv6 dùng địa chỉ 128 bit, cung cấp tối đa khoảng 3.4×1038 địa chỉ . Các phiên bản từ 0 đến 3 hoặc bị hạn chế, hoặc không được sử dụng. Phiên bản 5 được dùng làm giao thức dòng (stream) thử nghiệm. Còn có các phiên bản khác, nhưng chúng thường dành là các giao thức thử nghiệm và không được sử dụng rộng rãi.

Địa chỉ IP được chia thành 4 số giới hạn từ 0 - 255. Mỗi số được lưu bởi 1 byte - > IP có kích thước là 4byte, được chia thành các lớp địa chỉ. Có 3 lớp là A, B, và C. Nếu ở lớp A, ta sẽ có thể có 16 triệu điạ chỉ, ở lớp B có 65536 địa chỉ. Ví dụ: Ở lớp B chúng ta có tất cả các địa chỉ từ 132.25.0.0 đến 132.25.255.255. Phần lớn các địa chỉ ở lớp A là sở hữu của các công ty hay của tổ chức. Một ISP thường sở hữu một vài địa chỉ lớp B hoặc C. Ví dụ: Nếu địa chỉ IP của bạn là 132.25.23.24 thì bạn có thể xác định ISP của bạn là ai. (có IP là 132.25.x.)

**2.Giao thức TCP(Transmission Control Protocol):**

### Là một trong các giao thức cốt lõi của bộ giao thức TCP/IP. Sử dụng TCP, các ứng dụng trên các máy chủ được nối mạng có thể tạo các "kết nối" với nhau, mà qua đó chúng có thể trao đổi dữ liệu hoặc các gói tin. Giao thức này đảm bảo chuyển giao dữ liệu tới nơi nhận một cách đáng tin cậy và đúng thứ tự. TCP còn phân biệt giữa dữ liệu của nhiều ứng dụng (chẳng hạn, dịch vụ Web và dịch vụ thư điện tử) đồng thời chạy trên cùng một máy chủ.

### TCP hỗ trợ nhiều giao thức ứng dụng phổ biến nhất trên Internet và các ứng dụng kết quả, trong đó có WWW, thư điện tử và Secure Shell .

### Trong bộ giao thức TCP/IP, TCP là tầng trung gian giữa giao thức IP bên dưới và một ứng dụng bên trên. Các ứng dụng thường cần các kết nối đáng tin cậy kiểu đường ống để liên lạc với nhau, trong khi đó, giao thức IP không cung cấp những dòng kiểu đó, mà chỉ cung cấp dịch vụ chuyển gói tin không đáng tin cậy. TCP làm nhiệm vụ của tầng giao vận trong mô hình OSI đơn giản của các mạng máy tính.

### Các ứng dụng gửi các dòng gồm các byte 8-bit tới TCP để chuyển qua mạng. TCP phân chia dòng byte này thành các đoạn (segment) có kích thước thích hợp (thường được quyết định dựa theo kích thước của đơn vị truyền dẫn tối đa (MTU) của tầng liên kết dữ liệu của mạng mà máy tính đang nằm trong đó). Sau đó, TCP chuyển các gói tin thu được tới giao thức IP để gửi nó qua một liên mạng tới mô đun TCP tại máy tính đích. TCP kiểm tra để đảm bảo không có gói tin nào bị thất lạc bằng cách gán cho mỗi gói tin một "số thứ tự" (sequence number). Số thứ tự này còn được sử dụng để đảm bảo dữ liệu được trao cho ứng dụng đích theo đúng thứ tự. Mô đun TCP tại đầu kia gửi lại "tin báo nhận" (acknowledgement) cho các gói tin đã nhận được thành công; một "đồng hồ" (timer) tại nơi gửi sẽ báo time-out nếu không nhận được tin báo nhận trong khoảng thời gian bằng một round-trip time (RTT), và dữ liệu (được coi là bị thất lạc) sẽ được gửi lại. TCP sử dụng checksum (giá trị kiểm tra) để xem có byte nào bị hỏng trong quá trình truyền hay không; giá trị này được tính toán cho mỗi khối dữ liệu tại nơi gửi trước khi nó được gửi, và được kiểm tra tại nơi nhận.

*Hình 4.* *Cấu trúc header cua TCP*

### II.Mô hình Client-Server:

Mô hình được phổ biến nhất và được chấp nhận rộng rãi trong các hệ thống phân tán là mô hình client/server. Trong mô hình này sẽ có một tập các tiến trình mà mỗi tiến trình đóng vai trò như là một trình quản lý tài nguyên cho một tập hợp các tài nguyên cho trước và một  tập hợp các tiến trình client trong đó mỗi tiến trình thực hiện một tác vụ nào đó cần truy xuất tới tài nguyên phần cứng hoặc phần mềm dùng chung. Bản thân các trình quản lý tài nguyên cần phải truy xuất tới các tài nguyên dùng chung  được quản lý bởi một tiến trình khác, vì vậy một số tiến trình vừa là tiến trình client vừa là tiến trình server. Các tiến trình phát ra các yêu cầu tới các server bất kỳ khi nào chúng cần truy xuất tới một trong các tài nguyên của các server. Nếu yêu cầu là đúng đắn thì server sẽ thực hiện hành động được yêu cầu và gửi một đáp ứng trả lời tới tiến trình client.

Mô hình client/server cung cấp một cách tiếp cận tổng quát để chia sẻ tài nguyên trong các hệ thống phân tán. Mô hình này có thể được cài đặt bằng rất nhiều môi trường phần cứng và phần mềm khác nhau. Các máy tính được sử dụng để chạy các tiến trình client/server có nhiều kiểu khác nhau và không cần thiết phải phân biệt giữa chúng; cả tiến trình client và tiến trình server đều có thể chạy trên cùng một máy tính. Một tiến trình server có thể sử dụng dịch vụ của một server khác.

Mô hình truyền tin client/server hướng tới việc cung cấp dịch vụ. Quá trình trao đổi dữ liệu bao gồm:

1. Truyền một yêu cầu từ tiến trình client tới tiến trình server

2. Yêu cầu được server xử lý

3. Truyền đáp ứng cho client

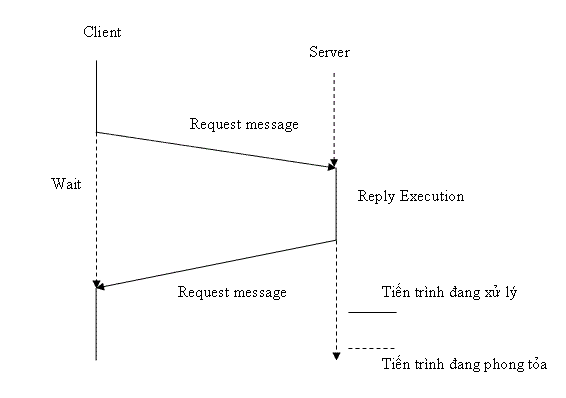
Mô hình truyền tin này liên quan đến việc truyền hai thông điệp và một dạng đồng bộ hóa cụ thể giữa client và server. Tiến trình server phải nhận thức được thông điệp được yêu cầu ở bước một ngay khi nó đến và hành động phát ra yêu cầu trong client phải được tạm dừng (bị phong tỏa) và buộc tiến trình client ở trạng thái chờ cho tớ khi nó nhận được đáp ứng do server gửi về ở bước ba.

Mô hình client/server thường được cài đặt dựa trên các thao tác cơ bản là gửi (send) và nhận (receive).

Quá trình giao tiếp client và server có thể diễn ra theo một trong hai chế độ: bị phong tỏa (blocked) và không bị phong tỏa (non-blocked).

\*Chế độ bị phong tỏa (blocked):

Trong chế độ bị phong tỏa, khi tiến trình client hoặc server phát ra lệnh gửi dữ liệu (send), việc thực thi của tiến trình sẽ bị tạm ngừng cho tới khi tiến trình nhận phát ra lệnh nhận dữ liệu (receive)



*Hình 5. Mô hình Client-Server*

Tương tự đối với tiến trình nhận dữ liệu, nếu tiến trình nào đó (client hoặc server) phát ra lệnh nhận dữ liệu, mà tại thời điểm đó chưa có dữ liệu gửi tới thì việc thực thi của tiến trình cũng sẽ bị tạm ngừng cho tới khi có dữ liệu gửi tới.

\*Chế độ không bị phong tỏa (non-blocked):

Trong chế độ này, khi tiến trình client hay server phát ra lệnh gửi dữ liệu thực sự, việc thực thi của tiến trình vẫn được tiến hành mà không quan tâm đến việc có tiến trình nào phát ra lệnh nhận dữ liệu đó hay không.

Tương tự cho trường hợp nhận dữ liệu, khi tiến trình phát ra lệnh nhận dữ liệu, nó sẽ nhận dữ liệu hiện có, việc thực thi của tiến trình vẫn được tiến hành mà không quan tâm đến việc có tiến trình nào phát ra lệnh gửi dữ liệu tiếp theo hay không.

### III. Cơ chế Socket trong Java:

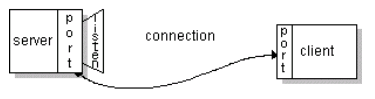
Một socket là một điểm cuối của thông tin hai chiều liên kết giữa hai chương trình đang chạy trên mạng. Những lớp socket được dùng để đại diện cho kết nối giữa một chương trình client và một chương trình server. Trong Java gói Java.net cung cấp hai lớp Socket và ServerSocket để thực hiện kết nối giữa client và server.

Thông thường thì server sẽ chạy trên một máy đặc biệt và có một socket giới hạn trong 1 Portnumber đặc biệt.

Phía client: client được biết hostname của máy mà server đang chạy và port number mà server đang lắng nghe. Để tạo một yêu cầu kết nối client sẽ thử hẹn gặp server ở trên máy của server thông qua port number. Client cũng cần xác định chính nó với server thông qua local port number.

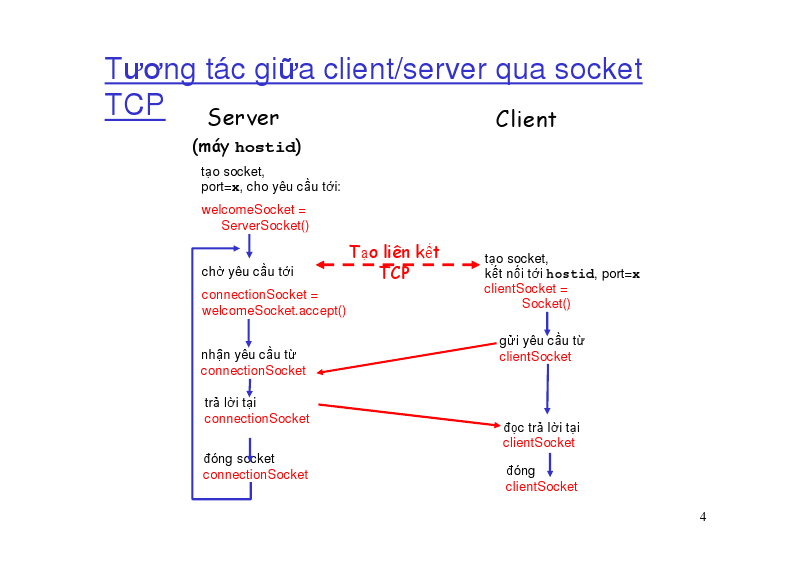


*Hình 6.* *Client gửi yêu cầu kết nối tới Server*

Nếu mọi thứ tốt đẹp thì server sẽ đồng ý kết nối. khi đồng ý kết nối thì server sẽ tạo ra một socket mới để nói chuyện với client và cũng tạo ra một socket khác để tiếp tục lắng nghe.

*Hình 7. Server đồng ý kết nối và tiếp tục lắng nghe*

### IV. Mô hình truyền tin trong Socket:

*Hình 8.* *Mô hình tương tác giữa client/server qua socket TCP*

Một socket có thể thực hiện bảy thao tác cơ bản:

-Kết nối với một máy ở xa (ví dụ, chuẩn bị để gửi và nhận dữ liệu)

-Gửi dữ liệu

-Nhận dữ liệu

-Ngắt liên kêt

-Gán cổng

-Nghe dữ liệu đến

-Chấp nhận liên kết từ các máy ở xa trên cổng đã  được gán

Lớp Socket của Java được sử dụng bởi cả client và server, có các phương thức tương ứng với bốn thao tác đầu tiên. Ba thao tác cuối chỉ cần cho server để chờ các client liên kết với chúng. Các thao tác này được cài đặt bởi lớp ServerSocket. Các socket cho client thường được sử dụng theo mô hình sau:

Một socket mới được tạo ra bằng cách sử dụng hàm  Socket().

Socket cố gắng liên kết với một host ở xa.

Mỗi khi liên kết được thiết lập, các host ở xa nhận các luồng vào và luồng ra từ socket, và sử dụng các luồng này để gửi dữ liệu cho nhau. Kiểu liên kết này được gọi là song công (full-duplex)-các host có thể nhận và gửi dữ liệu đồng thời. Ý nghĩa của dữ liệu phụ thuộc vào giao thức.

Khi việc truyền dữ liệu hoàn thành, một hoặc cả hai phía ngắt liên kết. Một số giao thức, như HTTP, đòi hỏi mỗi liên kết phải bị đóng sau mỗi khi yêu cầu được phục vụ. Các giao thức khác, chẳng hạn FTP, cho phép nhiều yêu cầu được xử lý trong một liên kết đơn.

### V. Một số hàm cơ bản trong Socket:

**5.1.Class mô tả về socket:**

- Tạo một socket

Socket(InetAddress address, int port)

Socket(String host, int port)

Socket(InetAddress address, int port, InetAddress, localAddr, int localPort)

Socket(String host, int port, InetAddress, localAddr, int localPort)

Socket()

-Lấy thông tin về một socket

InetAddress getInetAddress() : trả về địa chỉ mà socket kết nối đến

int getPort() : trả về địa chỉ mà socket kết nối đến.

InetAddress getLocalAddress() : trả về địa chỉ cục bộ.

int getLocalPort() : trả về địa chỉ cục bộ.

-Sử dụng Streams

public OutputStream getOutputStream() throws IOException: Trả về một output stream cho việc viết các byte đến socket này.

public InputStream getInputStream() throws IOException : Trả về một input stream cho việc đọc các byte từ socket này.

ServerSocket class

**5.2.Class mô tả ServerSocket:**

**-**Tạo một ServerSocket

ServerSocket(intport) throws IOException ServerSocket(intport, intbacklog) throws IOException ServerSocket(intport, intbacklog, InetAddressbindAddr) throws IOException

-Các phương thức trong ServerSocket

Socket accept() throws IOException: lắng nghe một kết nối đến socket này và có chấp nhận nó hay không

void close() throws IOException: Ðóngsocket.

InetAddress getInetAddress() : trả về địa chỉ cục bộ của socket

int getLocalPort() : trả về port mà server đang lắng nghe.

## **CHƯƠNG 2.PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

### I.Phân tích yêu cầu:

- Yêu cầu của bài toán: Xây dựng mô hình Client – Server ứng dụng CHAT

- Xây dựng chương trình Server

Tạo một TCP Socket và gắn vào một cổng.

Xây dựng một chương trình Server đa tuyến (Listener) để cho phép nhiều Client kết nối tới Server.Mỗi tuyến đảm nhận liên lạc với Client.

Chờ và lắng nghe yêu cầu kết nối từ Client.

Chấp nhận kết nối và nhận Socket tương ứng.

Tạo thông điệp trả lời

Gửi thông điệp trả lời về client.

-Xây dựng chương trình Client:

Mở một socket nối kết đến Server đã biết địa chỉ IP (hay tên miền) và số hiệu  
cổng.  
Lấy Stream nhập và Stream xuất được gán với socket  
Trao đổi dữ liệu với Server nhờ vào các Stream nhập và Stream xuất.  
Tham khảo protocol của dịch vụ để định dạng đúng dữ liệu trao đổi với  
Server.  
Đóng socket trước khi kết thúc chương trình.

### II.Phân tích chức năng:

**-**Nhập tên và nhấn nút Login để đăng nhập vào hệ thống

-Nhập nọi dung chat vào phần textarea

-Nhấn Enter hoặc Send để gửi đi

-Nhất Logout để thoát

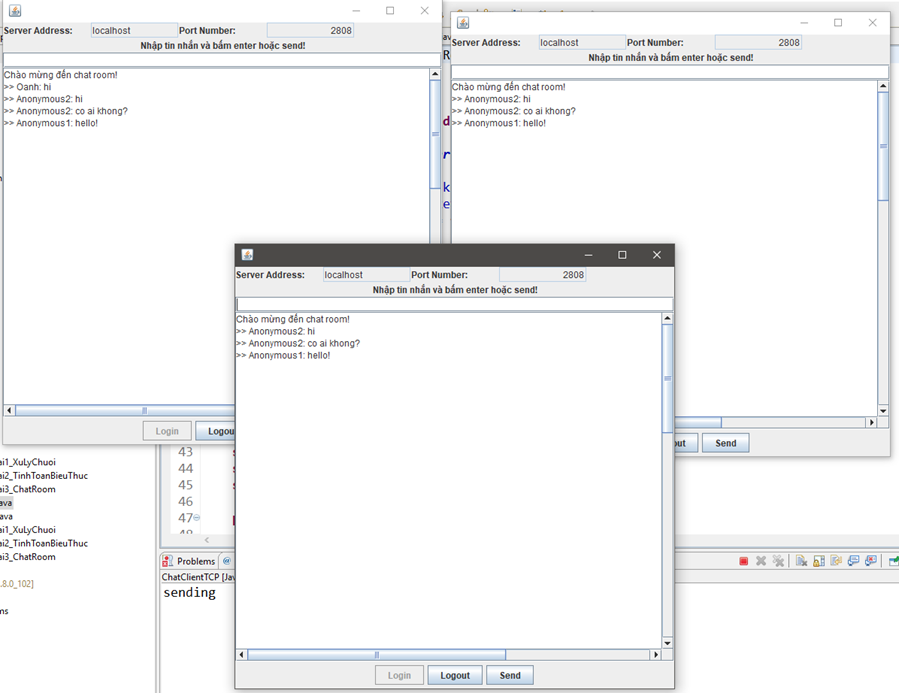
## **CHƯƠNG 3.TRIỂN KHAI VÀ ĐÁNH GIÁ KẾT QUẢ**

### I.Môi trường triển khai:

- Hệ điều hành Window 10

-NetBeans IDE 8.0

### II.Kết quả thực thi chương trình:



# *Hình 9. Kết quả*

# 

# **KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN**

### I.Những kết quả đạt được

- Mô phỏng được mô hình Client-Server

- Thực hiện được yêu cầu bài đề ra (CHAT).

- Dùng giao thức TCP/IP trong truyền dữ liệu.

- Thực hiện được việc kiểm tra trong nối kêt giữa Client-Server.

### II.Những vấn đề tồn tại

- Chương trình Chat còn đơn giản

- Giao diện sơ sài

- Chỉ thực hiện được nối kết giữa Client-Server(theo mô hình).

### III.Hướng phát triển

- Hoàn thiện giao diện đẹp và phù hợp với người dùng.

- Mở rộng ứng dụng trong chương trình Chat như :

+ Xử lý truyền file thông qua chương trình Chat

+ Xử lý gửi hình ảnh, hiện hình ảnh lên

- Phát triển trên mạng rộng ngoài LAN.

Cuối cùng em xin chân thành cám ơn cô Trần Hồ Thủy Tiên đã hướng dẫn và nhiệt tình giúp đỡ cho chúng em trong quá trình làm đồ án để em hoàn thành đồ án một cách tốt nhất.

**TÀI LIỆU THAM KHẢO**

[1] Nguyễn Phương Lan-Hoàng Hải, Lập trình Linux, tập1, NXB Giáo dục,2001  
[2] Đỗ Duy Việt-Nguyễn Hoàng Thanh Ly, Linux kernel[3] Nguyễn Thanh Thủy, Nhập môn hệ điều hành Linux[4] Trần Hồ Thuỷ Tiên, Bài giảng Nguyên lý hệ điều hành, Khoa CNTT Trường Đại học Bách khoa Đà Nẵng.  
[5] Andrew S. Tannenbaum, Modern Operating Systems, Prentice Hall 2nd.

[6] Giáo trình Lập trình Java, thầy Mai Văn Hà, Khoa CNTT Đại học BKDN.

[7] Giáo trình Lập trình mạng, thầy Mai Văn Hà, Khoa CNTT Đại học BKDN.

**PHỤ LỤC**