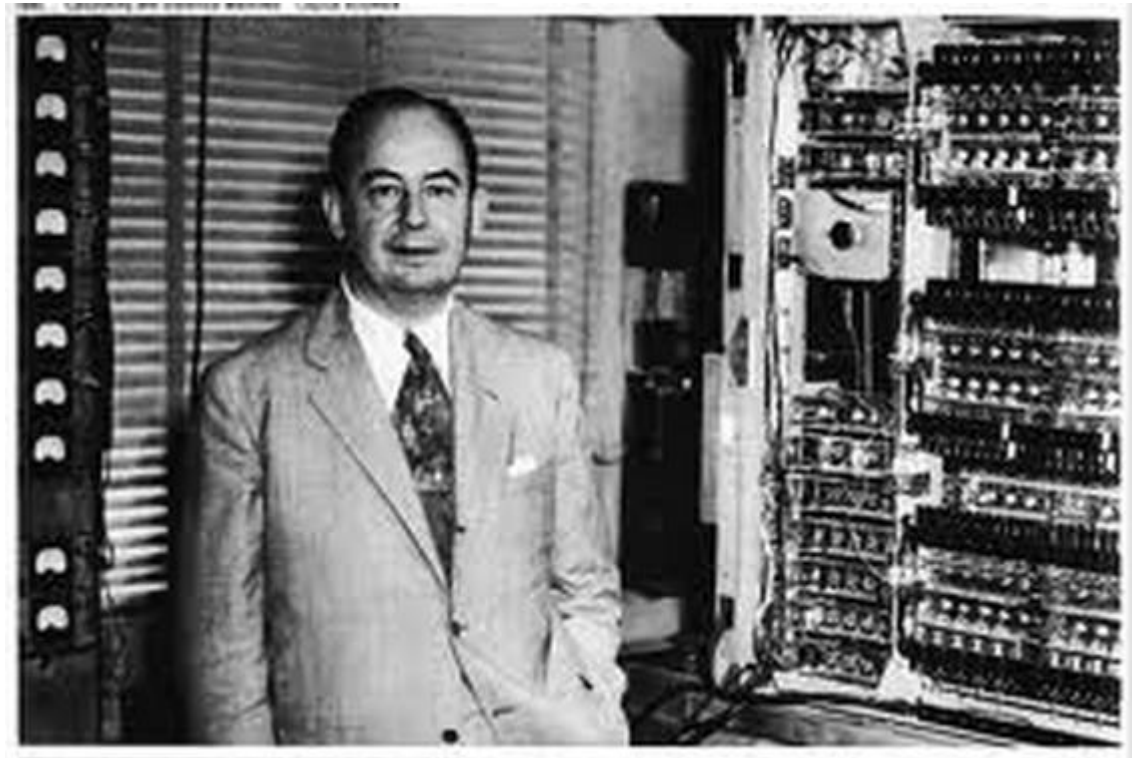


CHƯƠNG 1: GIỚI THIỆU

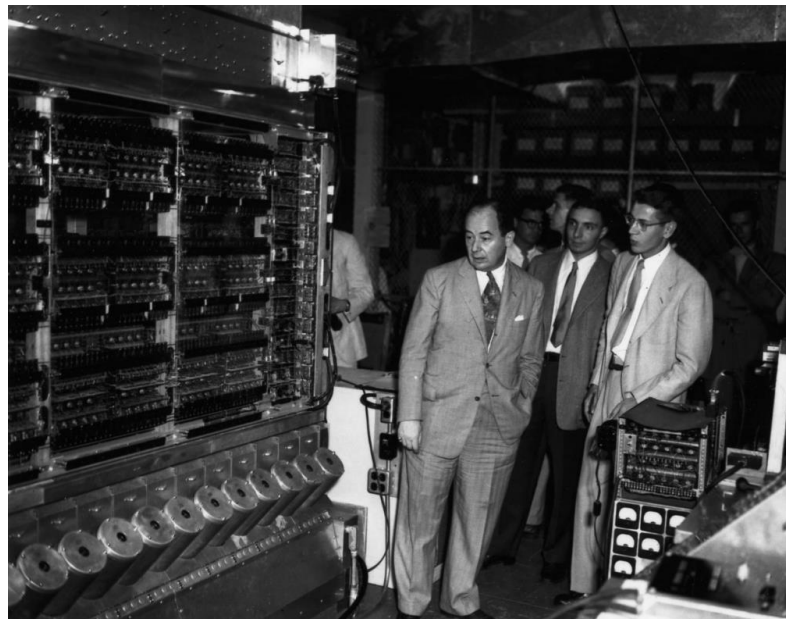
KIẾN TRÚC VON NEUMANN

Được đặt theo tên Nhà toán học người Hungary - John Von Neumann - người đầu tiên đưa ra những yêu cầu của một máy tính điện tử (electronic computer) trong công trình của ông vào năm 1945

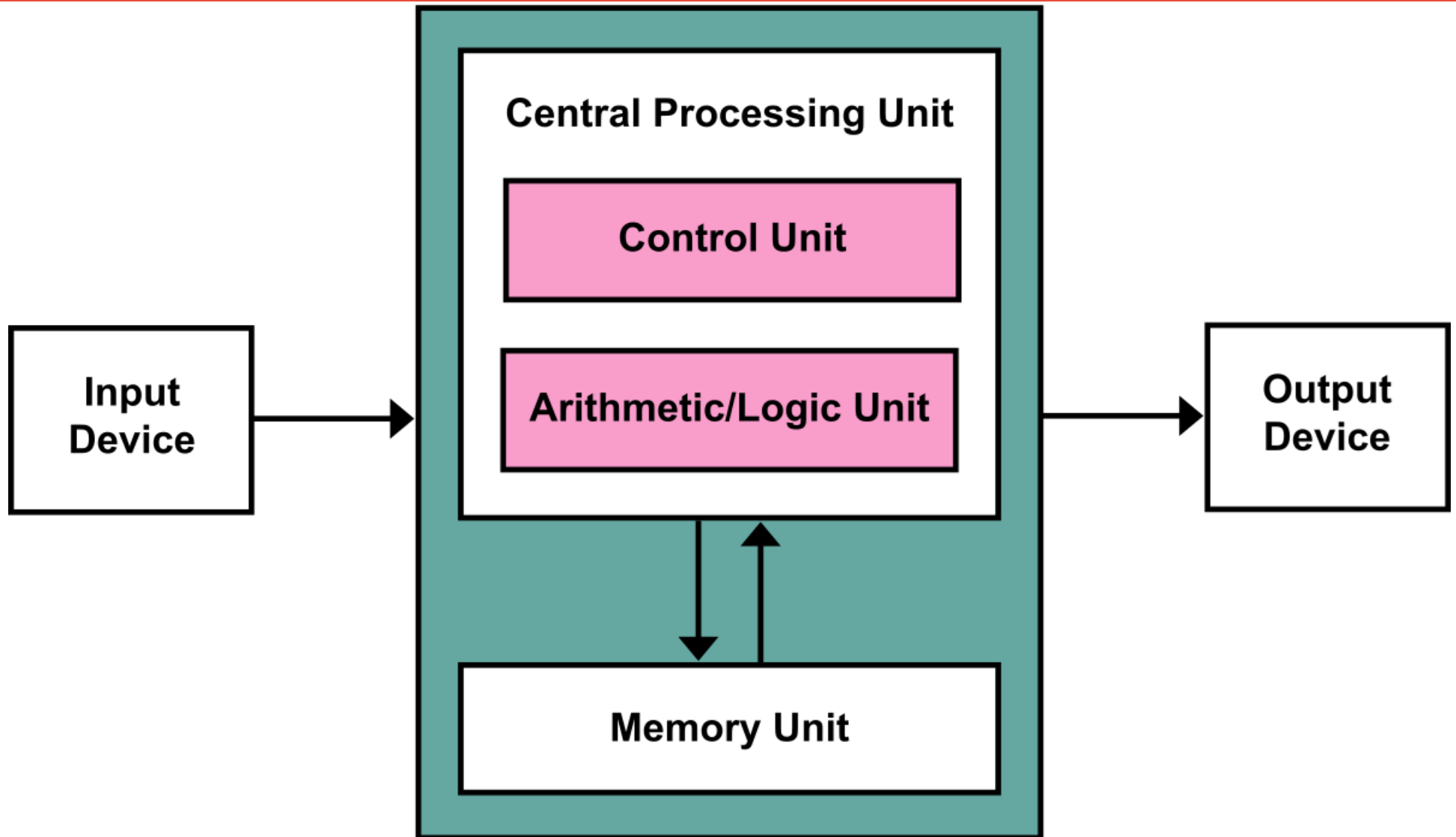


KIẾN TRÚC VON NEUMANN

- Hầu như tất cả các máy tính đều tuân theo thiết kế cơ bản này
- Sự khác nhau giữa các máy chỉ là sự sắp đặt của các hệ thống dây kết nối cứng (hard wiring)



KIẾN TRÚC VON NEUMANN

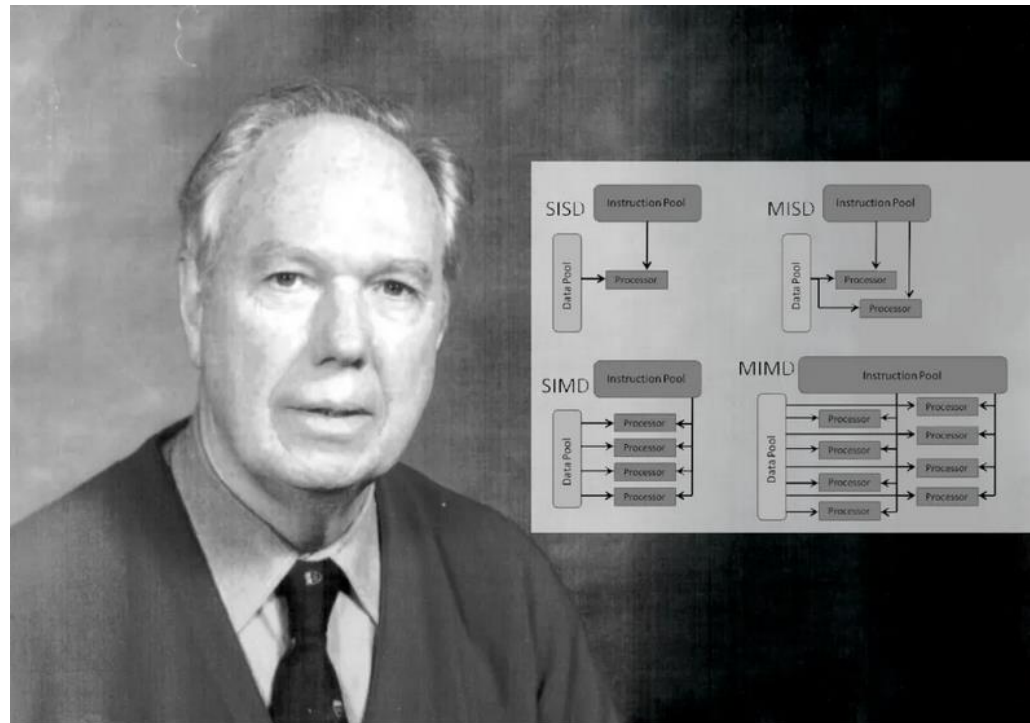


KIẾN TRÚC VON NEUMANN

- Memory: bộ nhớ **đọc ghi ngẫu nhiên** lưu trữ các lệnh chương trình và dữ liệu khi thực thi
- Control unit: lấy lệnh chương trình và dữ liệu từ bộ nhớ, giải mã câu lệnh rồi phối hợp tuần tự các phép toán, các thao tác để hoàn thành nhiệm vụ được lập trình sẵn
- ALU: thực hiện các phép toán số học và luận lý
- Input/output: giao tiếp với người dùng hoặc thiết bị ngoại vi

PHÂN LOẠI THEO FLYNN

- Có nhiều cách để phân loại máy tính song song. Một trong những phân loại được sử dụng rộng rãi có từ năm 1966, đó là phân loại Flynn.

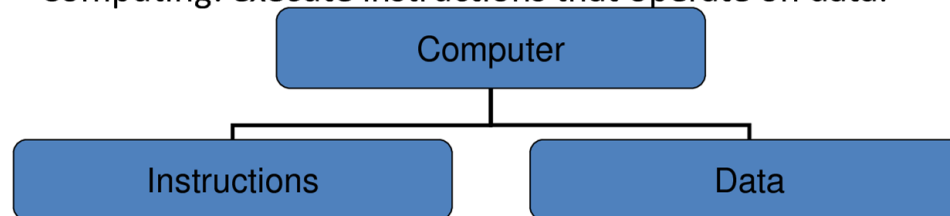


PHÂN LOẠI THEO FLYNN

- Phân loại Flynn phân biệt kiến trúc máy tính song song theo **cách phân lớp** theo **số trạng thái** có thể của **câu lệnh và dữ liệu**.
- Số trạng thái ở đây là Single hay Multiple.

Hardware Parallelism

- Computing: execute instructions that operate on data.



- Flynn's taxonomy (Michael Flynn, 1967) classifies computer architectures based on the number of instructions that can be executed and how they operate on data.

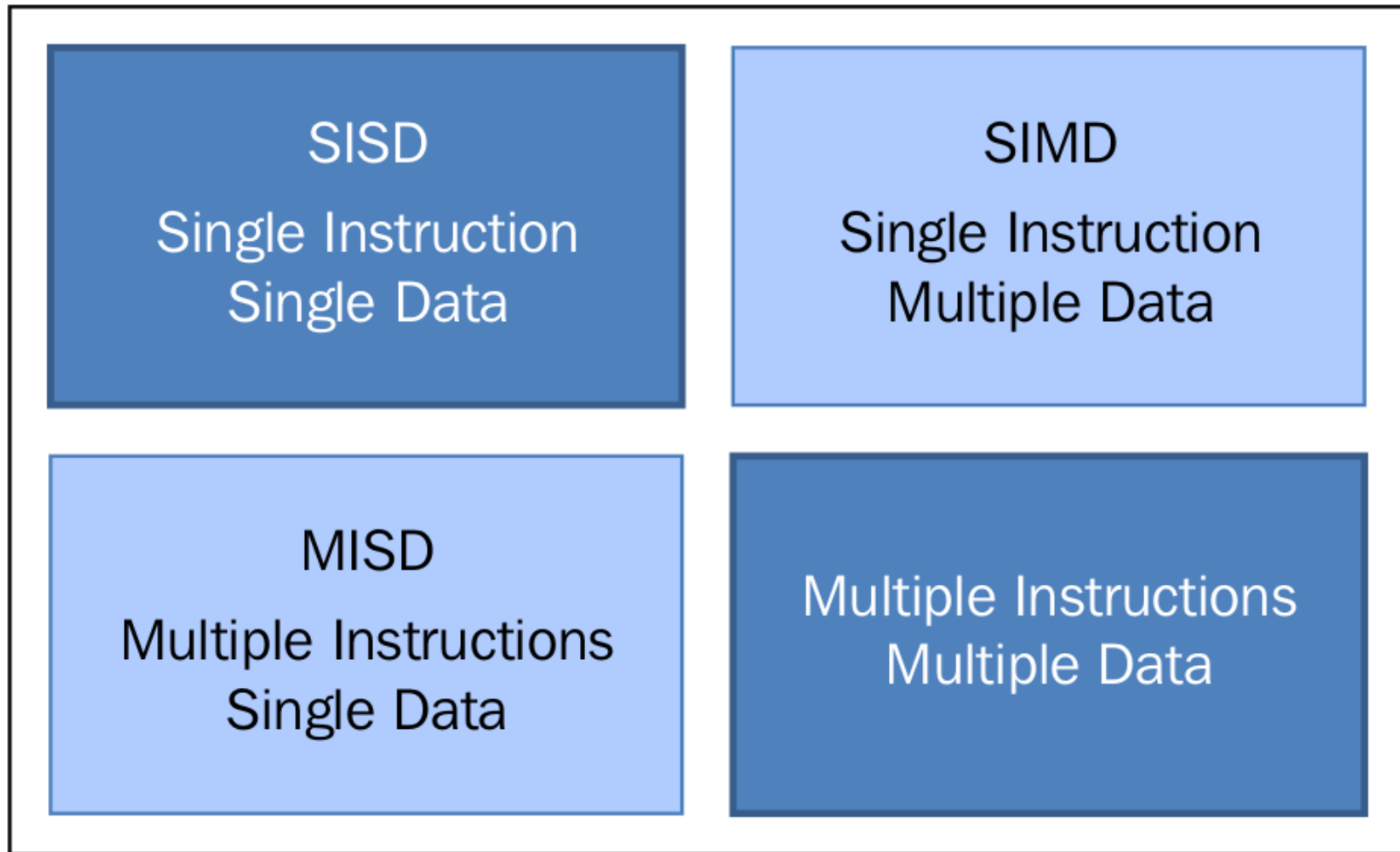
PHÂN LOẠI THEO FLYNN

➤ Dựa trên số lượng câu lệnh và dữ liệu có thể được xử lý đồng thời, hệ thống máy tính được phân thành **bốn** loại:

- ❑ Single instruction, single data (SISD)
- ❑ Single instruction, multiple data (SIMD)
- ❑ Multiple instruction, single data (MISD)
- ❑ Multiple instruction, multiple data (MIMD)

PHÂN LOẠI THEO FLYNN

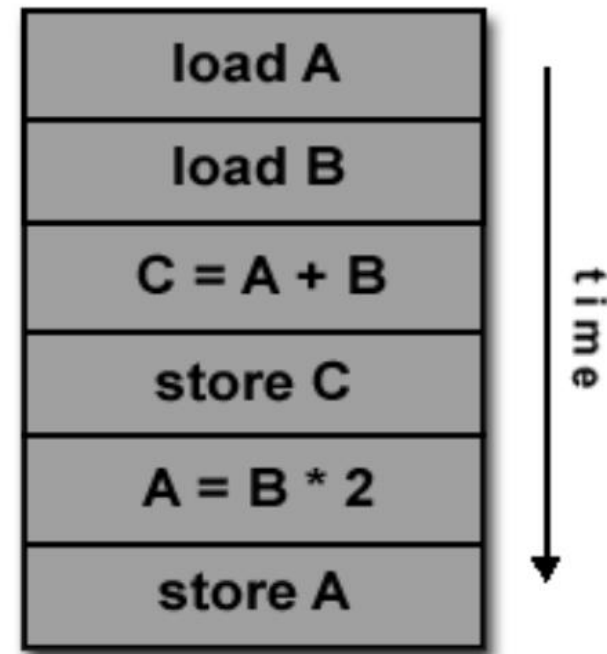
MA TRẬN PHÂN LOẠI THEO FLYNN



PHÂN LOẠI THEO FLYNN

SINGLE INSTRUCTION, SINGLE DATA (SISD)

- Là một **máy tính tuần tự** (non-parallel), **bộ xử lý đơn** (uniprocessor)
- **Single instruction:** **chỉ một dòng lệnh được tác động** bởi CPU trong suốt 1 chu kỳ
- **Single data:** chỉ **một dòng dữ liệu được dùng** như đầu vào trong suốt một chu kỳ
- **Thực thi một lệnh duy nhất** hoạt động trên một luồng dữ liệu
- Các lệnh của máy được xử lý một cách bình đẳng

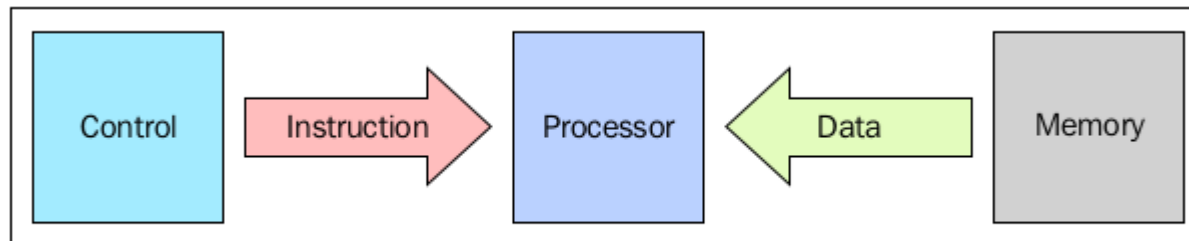


PHÂN LOẠI THEO FLYNN

SINGLE INSTRUCTION, SINGLE DATA (SISD)

➤ Trong một chu kỳ xung nhịp, CPU thực hiện các hoạt động sau:

- ❑ **Tìm nạp** (Fetch): CPU lấy dữ liệu và lệnh từ một vùng bộ nhớ, được gọi là thanh ghi.
- ❑ **Giải mã** (Decode): CPU giải mã lệnh.
- ❑ **Thực thi** (Execute): Lệnh được thực hiện trên dữ liệu. Kết quả của hoạt động được lưu trữ trong một thanh ghi khác.



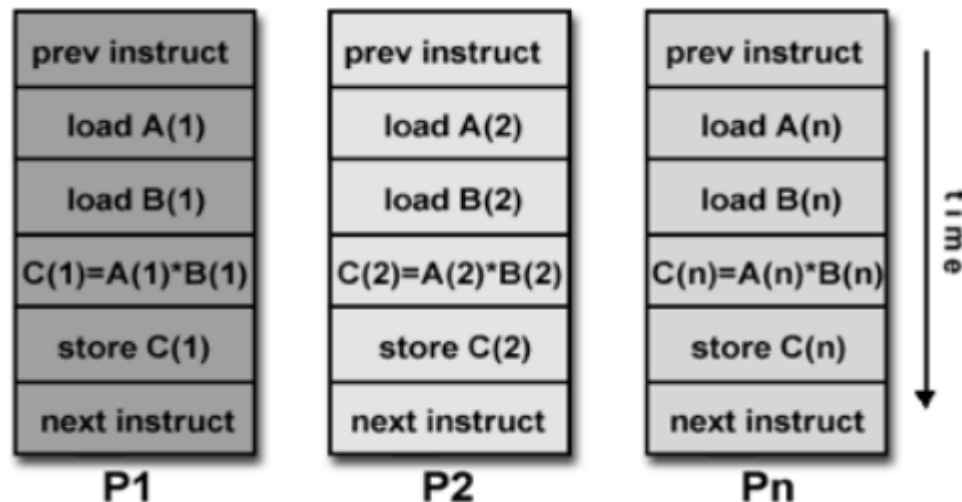
The SISD architecture schema

PHÂN LOẠI THEO FLYNN

SINGLE INSTRUCTION, MULTIPLE DATA (SIMD)

- Loại máy tính song song
- **Single instruction:** tất cả các đơn vị xử lý thi hành cùng câu lệnh ở một chu kỳ
- **Multiple data:** mỗi đơn vị xử lý có thể thực hiện theo tác trên phần tử dữ liệu khác nhau

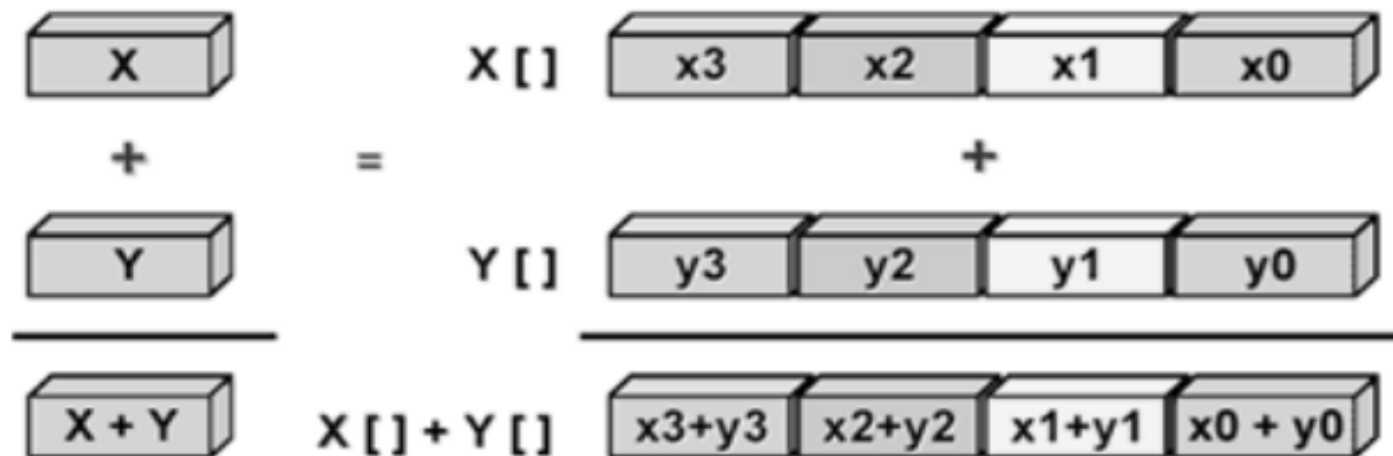
Dữ liệu A, B, C khác nhau
trên các đơn vị xử lý P



PHÂN LOẠI THEO FLYNN

SINGLE INSTRUCTION, MULTIPLE DATA (SIMD)

- Loại tốt nhất phù hợp cho những bài toán chuyên môn đòi hỏi tốc độ xử lý cao như **xử lý đồ thị, hình ảnh**
- Thi hành đồng bộ
- Ví dụ, với câu lệnh $x + y$



PHÂN LOẠI THEO FLYNN

SINGLE INSTRUCTION, MULTIPLE DATA (SIMD)

- Một máy tính SIMD bao gồm **n bộ xử lý giống hệt nhau**, mỗi bộ xử lý **có bộ nhớ cục bộ riêng**, nơi có thể lưu trữ dữ liệu.
- **Tất cả các bộ xử lý hoạt động** dưới sự kiểm soát của **một luồng lệnh duy nhất**; có **n luồng dữ liệu**, **một luồng cho mỗi bộ xử lý**.
- Các bộ xử lý làm việc đồng thời trên mỗi bước và thực hiện cùng một lệnh, nhưng trên các phần tử dữ liệu khác nhau.
- Đây là một ví dụ về **song song mức dữ liệu**.

PHÂN LOẠI THEO FLYNN

SINGLE INSTRUCTION, MULTIPLE DATA (SIMD)

- Kiến trúc SIMD linh hoạt hơn nhiều so với kiến trúc MISD.
- Nhiều vấn đề bao gồm nhiều ứng dụng có thể được giải quyết bằng các thuật toán song song trên máy tính SIMD.
- Các thuật toán cho các máy tính này tương đối dễ thiết kế, phân tích và triển khai.
- **Hạn chế:** chỉ những vấn đề có thể được chia thành một số vấn đề con (tất cả đều giống nhau, mỗi vấn đề sau đó sẽ được giải đồng thời, thông qua cùng một tập lệnh giống nhau) mới có thể được xử lý bằng máy tính SIMD.

PHÂN LOẠI THEO FLYNN

SINGLE INSTRUCTION, MULTIPLE DATA (SIMD)

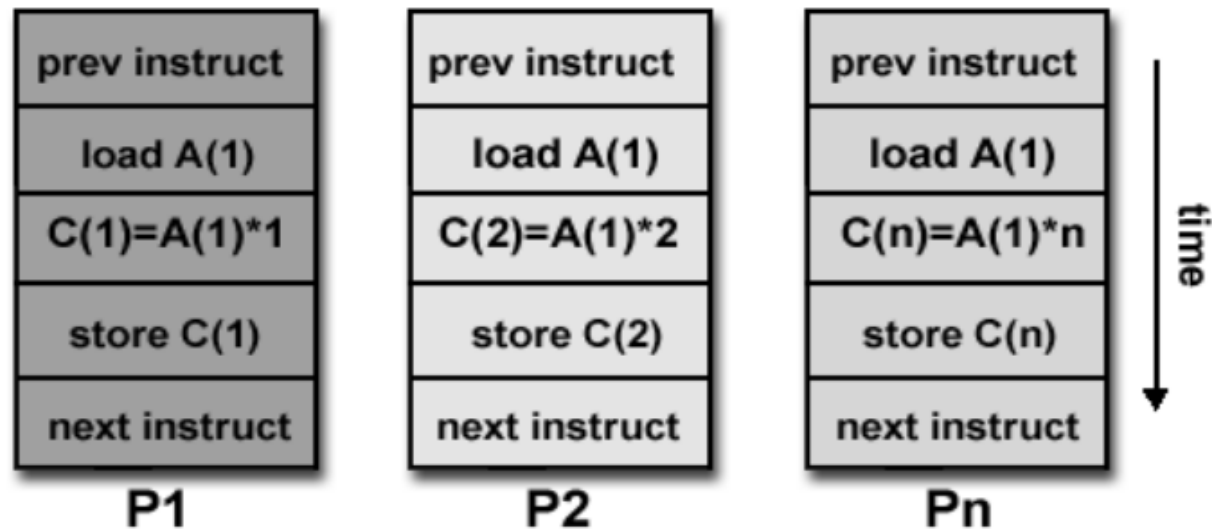
- Với siêu máy tính được phát triển theo mô hình này phải kể đến Máy kết nối - Connection Machine (Cỗ máy tư duy – Thinking Machine - 1985) và MPP (NASA – 1983).
- Lập trình GPU với Python, sự ra đời của đơn vị xử lý đồ họa - Graphics Processor Unit (GPU) hiện đại, được xây dựng với nhiều đơn vị nhúng SIMD đã dẫn đến việc sử dụng rộng rãi hơn mô hình tính toán này.

PHÂN LOẠI THEO FLYNN

MULTIPLE INSTRUCTION, SINGLE DATA (MISD)

- Loại máy tính song song
- Multiple instruction:** mỗi đơn vị xử lý hoạt động trên dữ liệu độc lập thông qua dòng câu lệnh riêng biệt
- Single data:** mỗi dòng dữ liệu duy nhất được đưa vào nhiều đơn vị xử lý

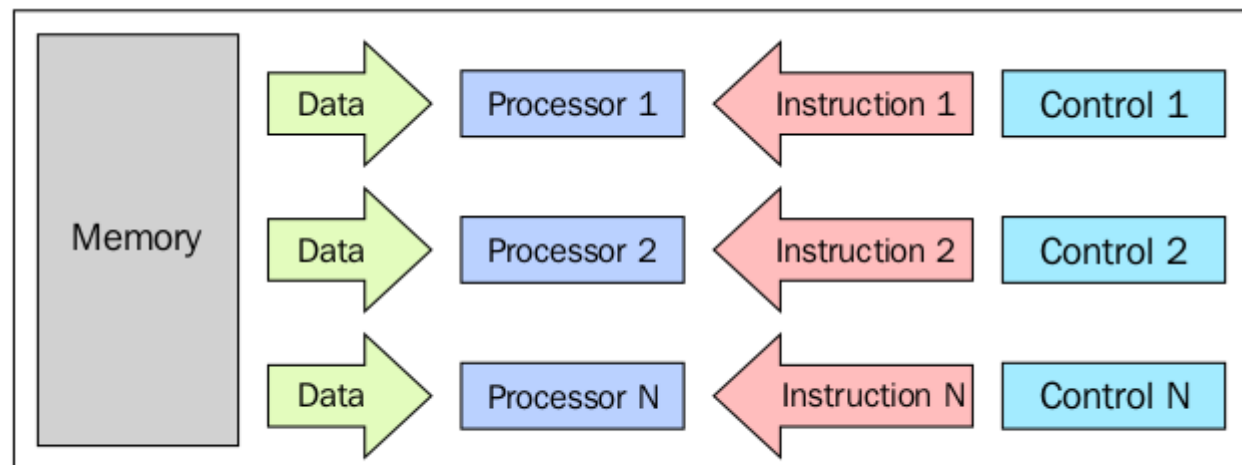
$A(1)$ được đưa vào các P để tính ra các giá trị C khác nhau



PHÂN LOẠI THEO FLYNN

MULTIPLE INSTRUCTION, SINGLE DATA (MISD)

- Trong mô hình này, n bộ xử lý, mỗi bộ xử lý có bộ điều khiển riêng, chia sẻ một bộ nhớ duy nhất.
- Trong mỗi chu kỳ xung nhịp, dữ liệu nhận được từ bộ nhớ được xử lý đồng thời bởi tất cả các bộ xử lý, mỗi bộ xử lý tuân theo các lệnh nhận được từ bộ điều khiển của nó.



The MISD architecture scheme

PHÂN LOẠI THEO FLYNN

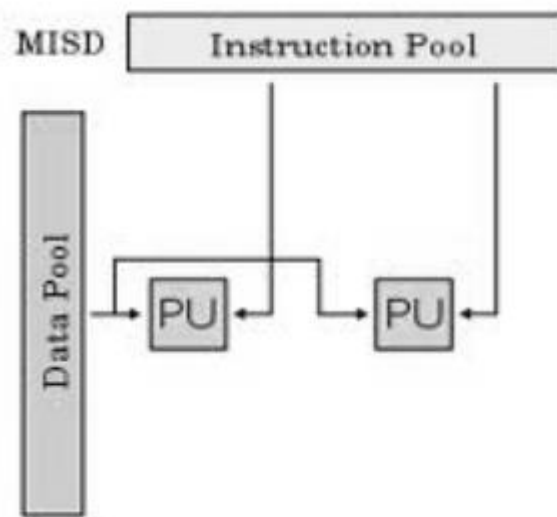
MULTIPLE INSTRUCTION, SINGLE DATA (MISD)

- Trong trường hợp này, tính song song (**song song ở mức lệnh**) có được bằng cách thực hiện một số thao tác trên cùng một phần dữ liệu.
- Các loại vấn đề có thể được giải quyết hiệu quả trong các kiến trúc này khá đặc biệt, chẳng hạn như các vấn đề liên quan đến mã hóa dữ liệu → máy tính **MISD không được sử dụng trong lĩnh vực thương mại.**

PHÂN LOẠI THEO FLYNN

MULTIPLE INSTRUCTION, SINGLE DATA (MISD)

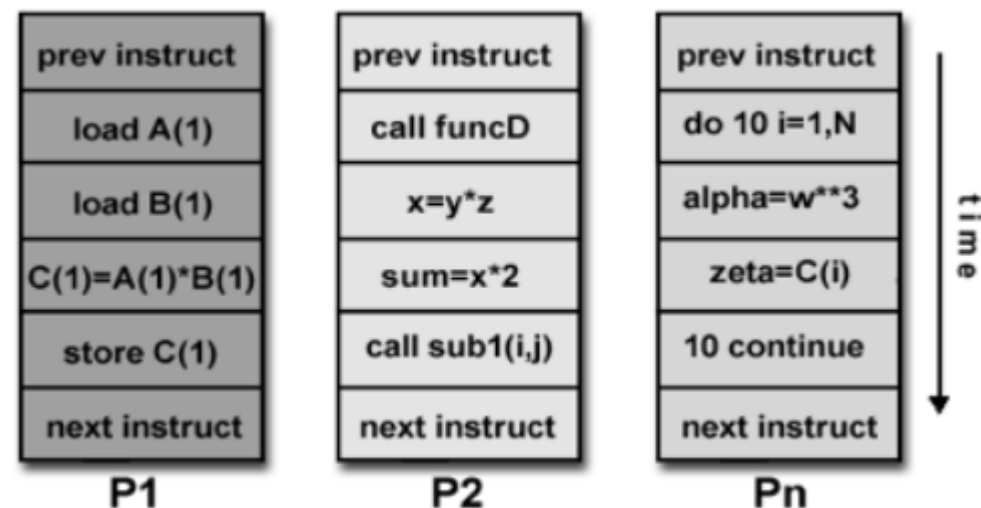
- Có một vài dự án thử nghiệm. Một trong số đó là máy tính Carnegie-Mellon C.mmp (1971)



PHÂN LOẠI THEO FLYNN

MULTIPLE INSTRUCTION, MULTIPLE DATA (MIMD)

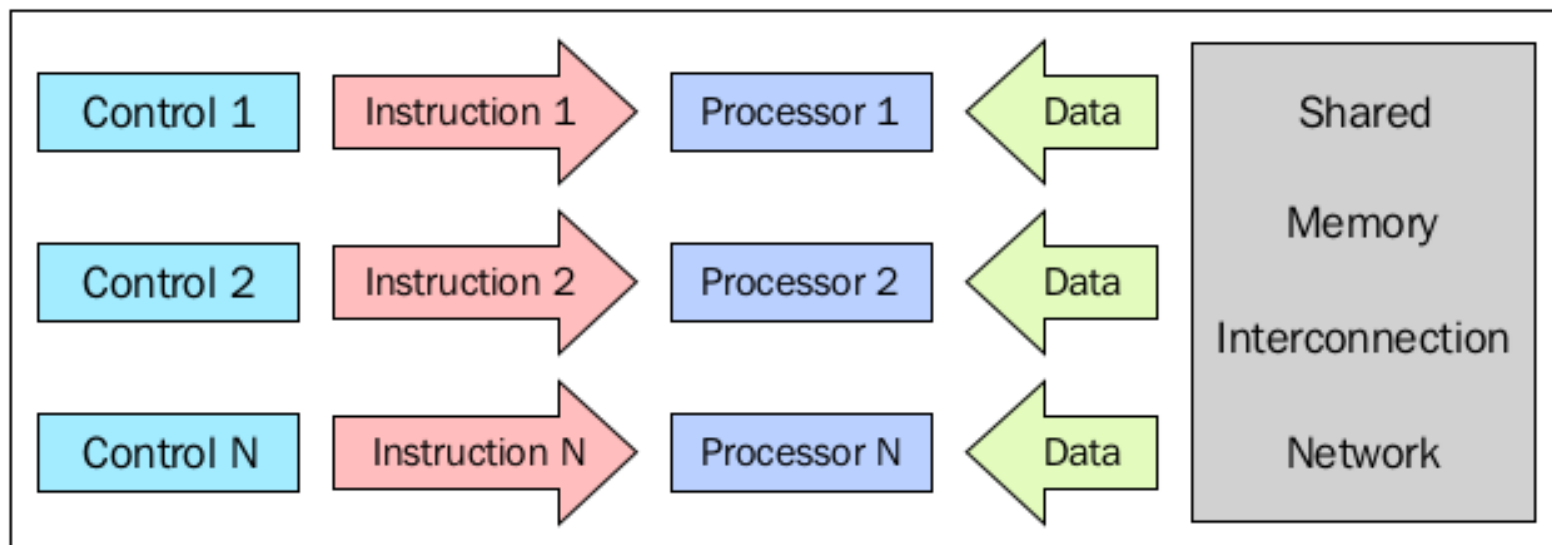
- Loại máy tính song song. Mỗi bộ xử lý P thực hiện một số câu lệnh khác nhau
- **Multiple instruction:** mỗi bộ xử lý có thể thi hành dòng câu lệnh khác nhau
- **Multiple data:** mỗi bộ xử lý có thể làm việc với dòng dữ liệu khác nhau



PHÂN LOẠI THEO FLYNN

MULTIPLE INSTRUCTION, MULTIPLE DATA (MIMD)

- Lớp máy tính song song này là lớp tổng quát nhất và mạnh hơn theo phân loại của Flynn.
- Có n bộ xử lý, n luồng lệnh và n luồng dữ liệu trong này.



PHÂN LOẠI THEO FLYNN

MULTIPLE INSTRUCTION, MULTIPLE DATA (MIMD)

- Mỗi bộ xử lý có bộ điều khiển và bộ nhớ cục bộ riêng, điều này làm cho kiến trúc MIMD mạnh hơn về mặt tính toán so với kiến trúc được sử dụng trong SIMD.
- Mỗi bộ xử lý hoạt động dưới sự kiểm soát của một luồng lệnh do đơn vị điều khiển của chính nó đưa ra → bộ xử lý có khả năng chạy các chương trình khác nhau trên các dữ liệu khác nhau, giải các bài toán con khác nhau và có thể là một phần của bài toán lớn hơn.

PHÂN LOẠI THEO FLYNN

MULTIPLE INSTRUCTION, MULTIPLE DATA (MIMD)

- Trong MIMD, kiến trúc đạt được với sự trợ giúp của cấp độ song song với các luồng và/hoặc quy trình. Điều này cũng có nghĩa là các bộ xử lý thường hoạt động không đồng bộ.
- Các máy tính trong lớp này được sử dụng để giải quyết những vấn đề không có cấu trúc thông thường theo yêu cầu của mô hình SIMD.
- Ngày nay, kiến trúc này được áp dụng cho nhiều PC, siêu máy tính và mạng máy tính.
- Tuy nhiên, các thuật toán không đồng bộ rất khó thiết kế, phân tích và triển khai.

PHÂN LOẠI THEO FLYNN

MULTIPLE INSTRUCTION, MULTIPLE DATA (MIMD)

- Việc thực thi có thể đồng bộ hoặc không đồng bộ, có thể xác định hoặc không xác định (non-deterministic)
- Hiện tại, loại máy tính song song hiện đại nhất, phổ biến nhất thuộc loại này
 - ❑ Hầu hết các siêu máy tính hiện tại
 - ❑ Cụm và lưới (clusters và grids) máy tính song song nối mạng
 - ❑ Máy tính đa xử lý đối xứng (multi-processor SMP)
 - ❑ Multi-core PCs.

MỘT SỐ THUẬT NGỮ

SUPERCOMPUTING/ HIGH PERFORMANCE COMPUTING

- Lĩnh vực sử dụng máy tính nhanh và lớn để giải quyết những bài toán lớn
- Thiết kế các thuật toán để xây dựng chương trình tính toán song song



MỘT SỐ THUẬT NGỮ

NODE

- Có thể coi như là máy tính nằm trong một chiếc hộp độc lập (standalone)
- Thông thường bao gồm nhiều CPUs/processors/cores
- Node cũng có thể là các máy tính nối mạng cùng nhau để bao gồm một siêu máy

MỘT SỐ THUẬT NGỮ

CPU/ Socket/ Processor/ Core

- Đây là những biến thể khác nhau, tùy thuộc vào ngữ cảnh
- CPU là một thành phần thực thi duy nhất từ máy tính
- Nhiều CPU được tích hợp vào một Node
- 1 CPU riêng lẻ được phân thành nhiều Core
- CPU nhiều Core được gọi là Socket (tùy nhà sản xuất)
- 1 Node với nhiều CPU, mỗi CPU chứa nhiều Core

MỘT SỐ THUẬT NGỮ

TASK

- Một công việc giống như một chương trình hoặc một đoạn chương trình tuần tự thực thi bởi một bộ xử lý
- Một chương trình song song bao gồm nhiều Task chạy trên nhiều bộ xử lý

MỘT SỐ THUẬT NGỮ

PIPELINING

- Tách một task thành nhiều bước để thực thi trên nhiều đơn vị xử lý khác nhau
- Khi đó, mỗi đơn vị xử lý có thể nạp một bước mới vào để thực hiện trong khi đang thực hiện một bước trước đó → giống dây chuyền lắp ráp
- Là một loại tính toán song song.

MỘT SỐ THUẬT NGỮ

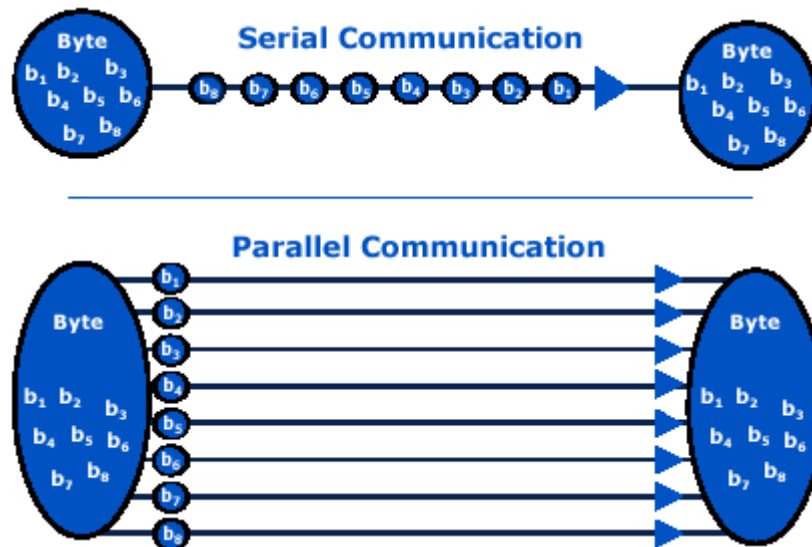
SYMMETRIC MULTI-PROCESSOR (SMP)

- Kiến trúc phần cứng có nhiều bộ xử lý chia sẻ một không gian địa chỉ chung (global address space) và truy cập đến tất cả nguồn tài nguyên
- Là kiến trúc để thực hiện tính toán với bộ nhớ chia sẻ (Share memory)

MỘT SỐ THUẬT NGỮ

COMMUNICATIONS

- Task song song thường cần phải trao đổi dữ liệu
- Có thể thực hiện thông qua bus bộ nhớ chia sẻ hay truyền thông điệp trên một mạng máy tính



MỘT SỐ THUẬT NGỮ

SYNCHRONIZATION

- Sự phối hợp để thực hiện các task song song theo thời gian thực, trong đó việc giao tiếp được diễn ra một cách thường xuyên
- Sự đồng bộ hóa thường liên quan đến việc chờ đợi ít nhất là một task khác, nên thời gian thực thi song song bị kéo dài thêm

MỘT SỐ THUẬT NGỮ

GRANULARITY

- Trong tính toán song song, granularity (độ chi tiết) là độ đo chất lượng của tỷ lệ giữa việc tính toán (computation) và việc giao tiếp (communication)
- Có 2 giá trị
 - **Coarse (thô)**: một lượng công việc tính toán tương đối **lớn** được thực hiện giữa các sự kiện giao tiếp
 - **Fine (tinh)**: một lượng công việc tính toán tương đối **nhỏ** được thực hiện giữa các sự kiện giao tiếp

MỘT SỐ THUẬT NGỮ

OBSERVED SPEEDUP

- Độ tăng tốc quan sát được (Observed speedup) của chương trình song song
- Định nghĩa như là tỷ số giữa thời gian thi hành tuần tự và thời gian thi hành song song
- Một trong những chỉ số đơn giản nhất và được sử dụng rộng rãi nhất khi quan tâm đến hiệu năng (Performance) của một chương trình song song

MỘT SỐ THUẬT NGỮ

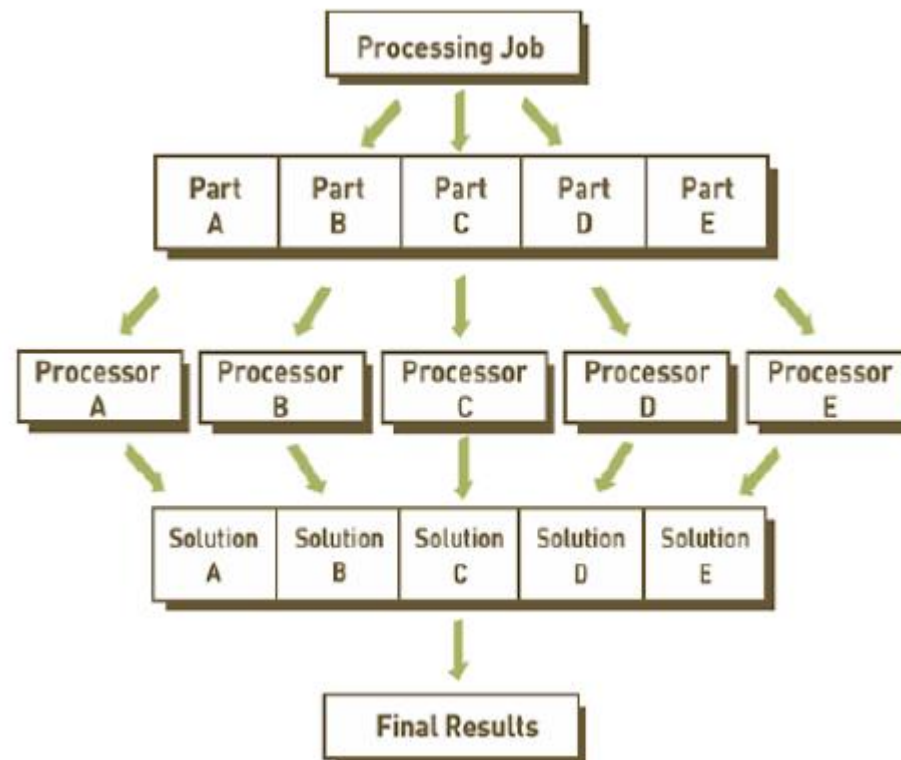
PARALLEL OVERHEAD

- Lượng thời gian cần thiết để phối hợp các task song song
- Bao gồm các yếu tố như:
 - Thời gian khởi động task (start-up time)
 - Đồng bộ hóa
 - Truyền dữ liệu
 - Chi phí phần mềm bị áp đặt bởi trình biên dịch song song, các thư viện, các công cụ, hệ điều hành, ...
 - Thời gian kết thúc task (Termination time)

MỘT SỐ THUẬT NGỮ

MASSIVELY PARALLEL

- Song song quy mô lớn đề cập đến yếu tố phần cứng bao gồm một hệ thống song song có nhiều bộ xử lý.

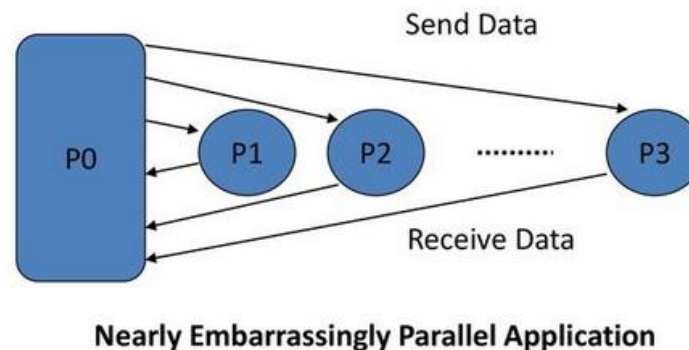
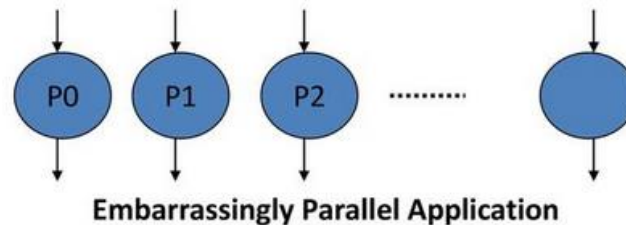


MỘT SỐ THUẬT NGỮ

EMBARRASSINGLY PARALLEL

- Giải quyết nhiều tác vụ tương tự nhưng độc lập một cách đồng thời mà không cần đến sự phối hợp giữa chúng

Embarrassingly Parallel Examples



MỘT SỐ THUẬT NGỮ

SCALABILITY

- Khả năng mở rộng: đề cập đến khả năng (cả phần cứng lẫn phần mềm) của hệ thống song song, qua đó minh chứng sự tăng lên tương ứng của speedup khi bổ sung thêm nhiều bộ xử lý.
- Đôi khi được hiểu theo nghĩa khả năng tạp tỷ lệ giữa bộ xử lý và bộ nhớ để đảm bảo speedup.

MỘT SỐ THUẬT NGỮ

SCALABILITY

- Các yếu tố góp phần cho khả năng mở rộng:
 - Bandwidths và giao tiếp của mạng
 - Thuật toán
 - Chi phí song song liên quan
 - Đặc điểm của chương trình

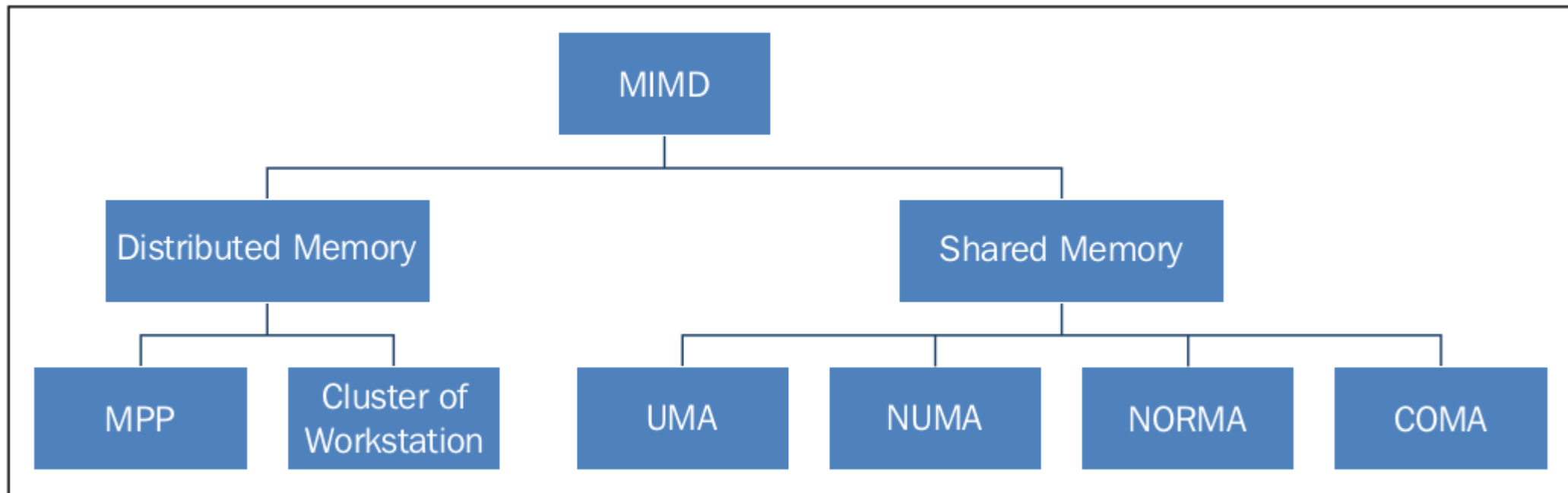
TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

- Một khía cạnh khác cần xem xét để đánh giá một kiến trúc song song là **tổ chức bộ nhớ** hay đúng hơn là **cách thức truy cập dữ liệu**.
- Cho dù bộ xử lý có nhanh đến đâu, nếu bộ nhớ không thể duy trì và cung cấp các lệnh và dữ liệu ở tốc độ đủ, thì hiệu suất sẽ không được cải thiện.
- Vấn đề chính phải được khắc phục để làm cho *thời gian đáp ứng của bộ nhớ tương thích với tốc độ của bộ xử lý* là **thời gian chu kỳ bộ nhớ**, được định nghĩa là **thời gian đã giữa hai thao tác liên tiếp**.

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

- **Thời gian chu kỳ của bộ xử lý thường ngắn hơn** nhiều so với **thời gian chu kỳ của bộ nhớ**.
- Khi bộ xử lý bắt đầu truyền dữ liệu (đến hoặc từ bộ nhớ), bộ nhớ sẽ vẫn được sử dụng trong toàn bộ thời gian của chu kỳ bộ nhớ: trong khoảng thời gian này, không có thiết bị nào khác (bộ điều khiển I/O, bộ xử lý hoặc thậm chí chính bộ xử lý thực hiện yêu cầu) có thể sử dụng bộ nhớ vì nó sẽ được cam kết đáp ứng yêu cầu.

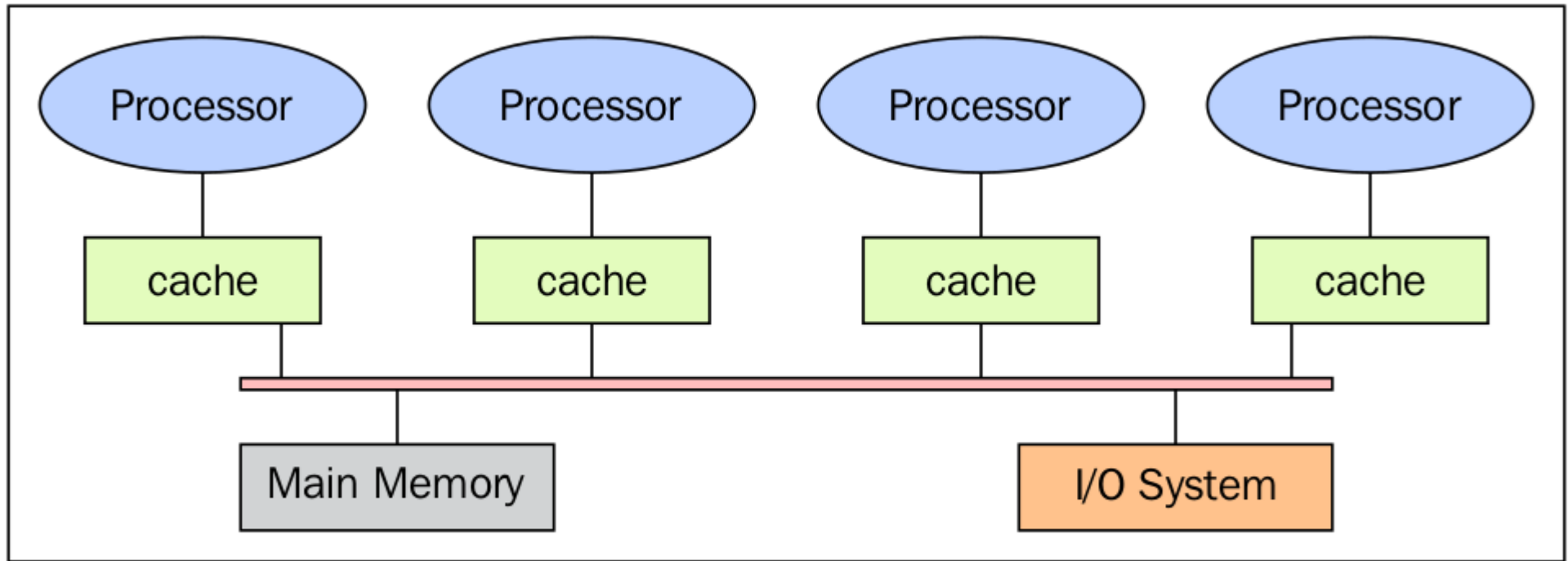
TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)



TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

- **Share memory/ Global memory** - Bộ nhớ chia sẻ (hệ thống bộ nhớ dùng chung): có *bộ nhớ ảo cao* và tất cả các *bộ xử lý* đều có *quyền truy cập như nhau* vào dữ liệu và lệnh trong bộ nhớ này.
- **Distributed memory** - Bộ nhớ phân tán: trong đó mỗi bộ xử lý có một bộ nhớ cục bộ mà các bộ xử lý khác không thể truy cập được.
- Sự khác biệt giữa bộ nhớ dùng chung và bộ nhớ phân tán nằm ở cấu trúc của bộ nhớ ảo hoặc bộ nhớ từ quan điểm của bộ xử lý; ***việc quản lý truy cập bộ nhớ của đơn vị xử lý.***

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)



The shared memory architecture schema

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

- **Sự cố liên kết bộ đệm:** xảy ra khi bộ xử lý sửa đổi dữ liệu được lưu trữ trong hệ thống bộ nhớ được sử dụng đồng thời bởi các bộ xử lý khác.
 - Giá trị mới sẽ chuyển từ bộ đệm của bộ xử lý đã được thay đổi thành bộ nhớ dùng chung;
 - Tuy nhiên, sau đó, nó cũng phải được chuyển đến tất cả các bộ xử lý khác để chúng không hoạt động với giá trị lỗi thời
- > **Thay đổi dữ liệu** ở một vị trí bộ nhớ được thực hiện bởi một bộ xử lý này thì tất cả các **bộ xử lý khác có thể thấy được** sự thay đổi đó
- Yêu cầu triển khai phần cứng có thể xử lý các vấn đề tương tranh và đồng bộ hóa tương tự như các sự cố có lập trình luồng.

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

Những đặc trưng chung của hệ thống bộ nhớ chia sẻ (Share memory):

- Bộ nhớ giống nhau đối với tất cả các bộ xử lý
 - Nhiều bộ xử lý có thể hoạt động độc lập nhưng chia sẻ cùng một tài nguyên bộ nhớ
- Việc đồng bộ hóa được thực hiện bằng cách kiểm soát quyền truy cập của bộ xử lý vào bộ nhớ dùng chung.
 - Trên thực tế, tại một thời điểm chỉ có một bộ xử lý có quyền truy cập vào tài nguyên bộ nhớ.

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

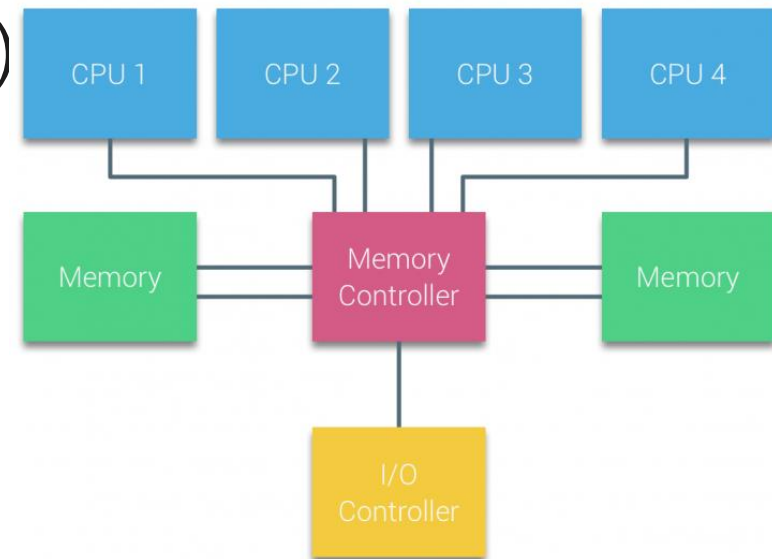
Những đặc trưng chung của hệ thống bộ nhớ chia sẻ (Share memory):

- Vị trí bộ nhớ dùng chung không được thay đổi từ một tác vụ trong khi một tác vụ khác truy cập nó.
- Chia sẻ dữ liệu nhanh chóng; thời gian cần thiết để giao tiếp giữa hai tác vụ bằng với thời gian đọc một vị trí bộ nhớ (nó phụ thuộc vào tốc độ truy cập bộ nhớ).

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

Những đặc trưng chung của hệ thống bộ nhớ chia sẻ (Share memory):

- Máy tính có bộ nhớ chia sẻ có thể phân thành **02 lớp** chính dựa trên **thời gian truy cập bộ nhớ**.
- **UMA** (Uniform Memory Access)
- **NUMA** (Non-Uniform Memory Access)



TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

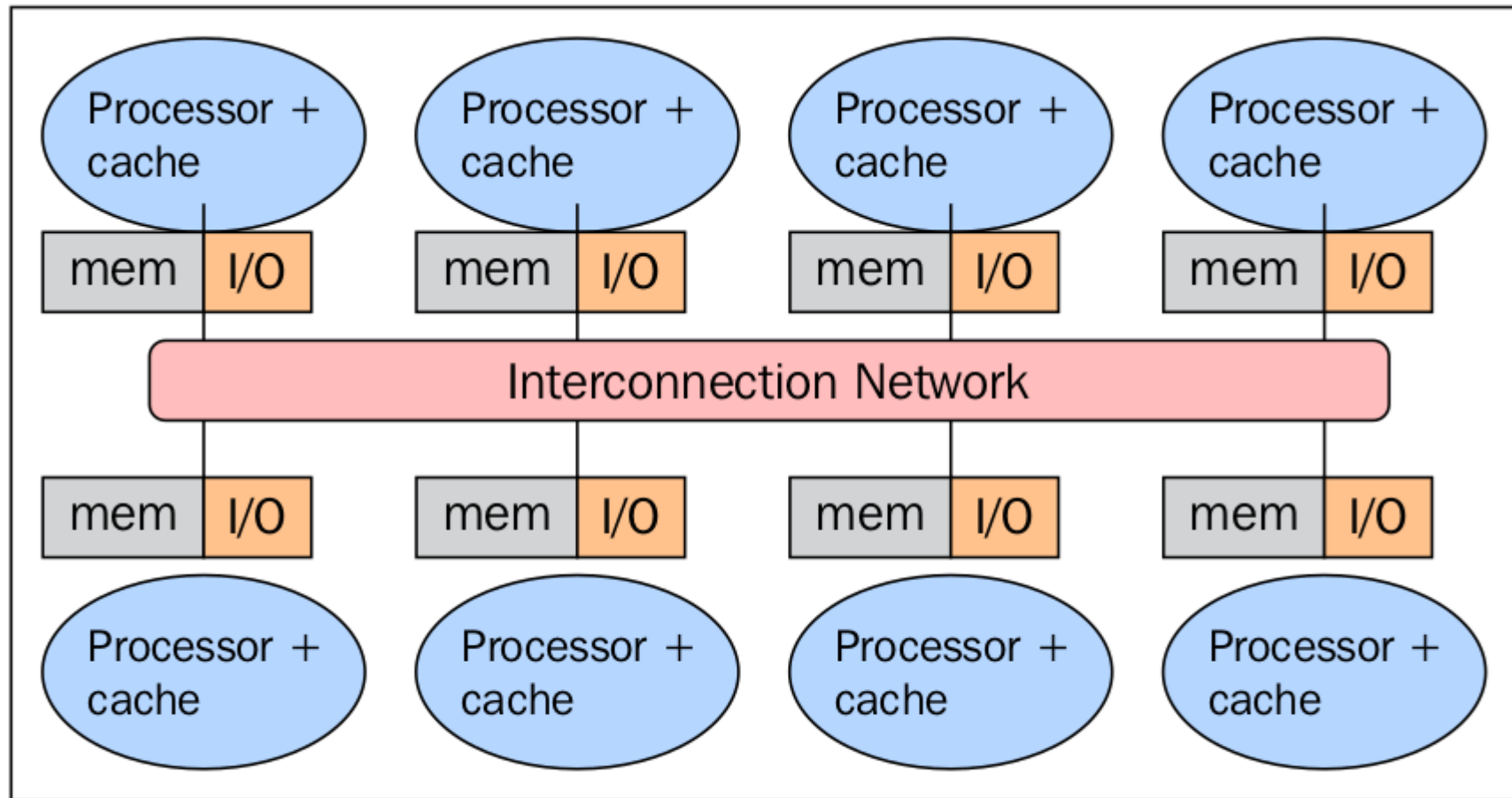
Ưu điểm

- Không gian địa chỉ chung cung cấp cách thức truy cập bộ nhớ thân thiện với người lập trình
- Dữ liệu chia sẻ giữa các task vừa nhanh vừa nhất quán

➤ Hạn chế

- Thiếu khả năng mở rộng (tỷ lệ) giữa bộ nhớ và các CPU
 - Tăng nhiều CPU → tăng lưu lượng truy cập giữa CPU và Share memory
 - Ngay cả với việc quản lý giữa cache và bộ nhớ
- Lập trình viên phải xây dựng sự đồng bộ đảm bảo đúng khi truy cập bộ nhớ chung

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)



The distributed memory architecture scheme

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

- Trong một hệ thống có bộ nhớ phân tán, bộ nhớ được liên kết với từng bộ xử lý và bộ xử lý chỉ có thể xử lý bộ nhớ của chính nó.
- Một số tác giả gọi loại hệ thống này là "đa máy tính", phản ánh thực tế rằng bản thân các phần tử của hệ thống là những hệ thống hoàn chỉnh nhỏ gồm bộ xử lý và bộ nhớ.
- Mỗi bộ xử lý có thể sử dụng toàn bộ băng thông của bộ nhớ cục bộ của riêng mình mà không có bất kỳ sự can thiệp nào từ các bộ xử lý khác.

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

- Không có giới hạn nội tại đối với số lượng bộ xử lý, kích thước của hệ thống chỉ bị giới hạn bởi mạng được sử dụng để kết nối các bộ xử lý.
- Không có vấn đề về tính nhất quán của bộ đệm.
- Hạn chế: giao tiếp giữa các bộ xử lý khó thực hiện hơn.
 - Nếu một bộ xử lý yêu cầu dữ liệu trong bộ nhớ của bộ xử lý khác, thì hai bộ xử lý nhất thiết phải trao đổi tin nhắn thông qua giao thức truyền tin (message-passing protocol).

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

Những đặc trưng chung của hệ thống bộ nhớ phân tán (Distributed memory):

- Bộ nhớ được phân phối vật lý giữa các bộ xử lý
 - Mỗi bộ nhớ cục bộ chỉ có thể truy cập trực tiếp bởi bộ xử lý của nó.
- Đồng bộ hóa đạt được bằng cách di chuyển dữ liệu (ngay cả khi đó chỉ là thông báo) giữa các bộ xử lý (giao tiếp).

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

Các tính năng chính của hệ thống bộ nhớ phân tán (Distributed memory):

- Việc chia nhỏ dữ liệu trong bộ nhớ cục bộ ảnh hưởng đến hiệu suất của máy → cần làm cho việc chia nhỏ chính xác, để giảm thiểu giao tiếp giữa các CPU.
- Ngoài ra, bộ xử lý điều phối các hoạt động phân tách và tổng hợp phải giao tiếp hiệu quả với các bộ xử lý hoạt động trên các phần riêng lẻ của cấu trúc dữ liệu.
- Giao thức truyền tin được sử dụng để CPU có thể giao tiếp với nhau thông qua việc trao đổi các gói dữ liệu.
- Các tin nhắn là các đơn vị thông tin rời rạc; theo nghĩa là chúng có một danh tính được xác định rõ ràng, vì vậy luôn có thể phân biệt chúng với nhau.

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

Ưu điểm

- Bộ nhớ tỷ lệ với số bộ xử lý
 - Tăng số lượng bộ xử lý thì kích thước bộ nhớ cũng tăng tương ứng
- Mỗi bộ xử lý nhanh chóng truy cập đến bộ nhớ riêng, không cần bất kỳ sự can thiệp nào và không phát sinh thêm chi phí

TỔ CHỨC BỘ NHỚ (MEMORY ORGANIZATION)

Hạn chế

- Lập trình viên chịu trách nhiệm nhiều chi tiết liên quan đến việc truyền dữ liệu giữa các bộ xử lý
- Khó khăn khi ánh xạ cấu trúc dữ liệu tồn tại (trên cơ sở bộ nhớ toàn cục) đến tổ chức của bộ nhớ phân tán này
- Thời gian truy cập bộ nhớ không đồng nhất

MÔ HÌNH LẬP TRÌNH SONG SONG (PARALLEL PROGRAMMING MODEL)

Các mô hình lập trình song song được sử dụng phổ biến:

- The shared memory model: mô hình bộ nhớ chia sẻ
- The multithread model: mô hình đa luồng
- The distributed memory/message passing model: mô hình phân tán/ truyền thông điệp
- The data parallel model: mô hình song song dữ liệu

LƯU Ý

- Kiến trúc Von Neumann
- Phân loại theo Flynn
- Tổ chức bộ nhớ
- Mô hình lập trình song song



