

Chương 5

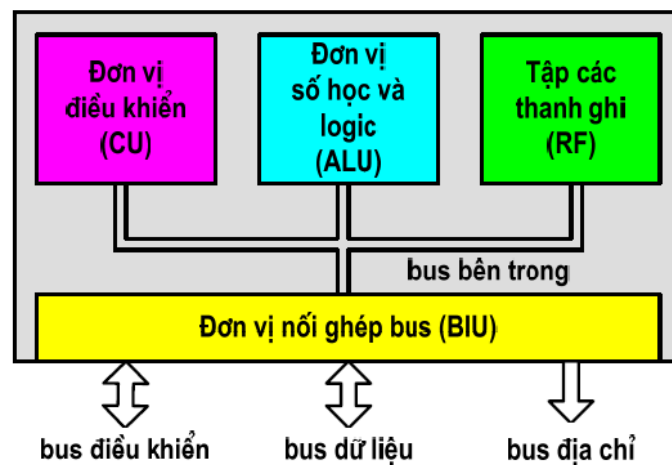
Bộ xử lý trung tâm CPU (Central Processing Unit)

Nội dung

1. Tổ chức của CPU
2. Hoạt động của chu trình lệnh
3. Đơn vị điều khiển
4. Kỹ thuật đường ống lệnh
5. Cấu trúc bộ xử lý tiên tiến

Phần 1: Tổ chức của CPU

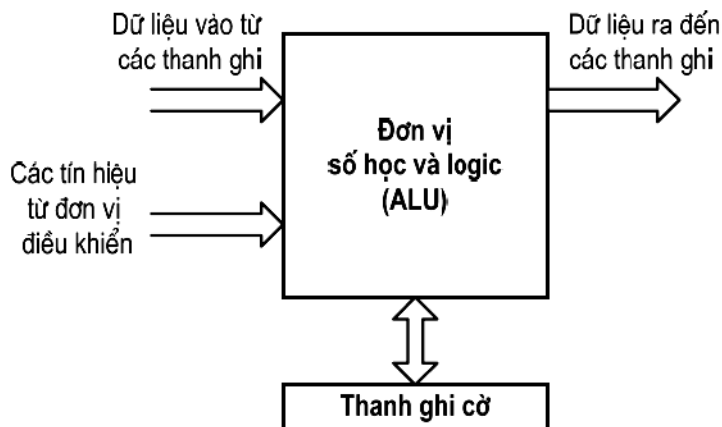
* Cấu trúc cơ bản của CPU:



- Đơn vị điều khiển (Control Unit CU): điều khiển hoạt động của máy tính theo chương trình đã định sẵn.
- Đơn vị số học và logic (Arithmetic and Logic Unit ALU): thực hiện các phép toán số học và phép toán logic.
- Tập thanh ghi (Register File RF): lưu giữ các thông tin tạm thời phục vụ cho hoạt động của CPU.
- Đơn vị nối ghép bus (Bus Interface Unit BIU): kết nối và trao đổi thông tin giữa bus bên trong (internal bus) và bus bên ngoài (external bus).

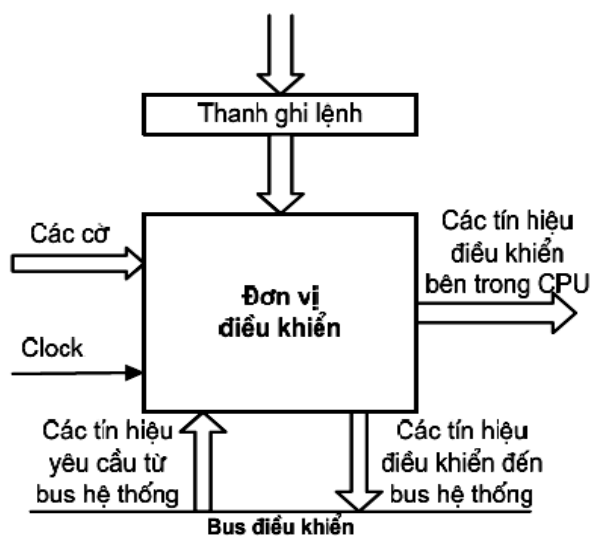
*** Đơn vị số học và luận lý ALU:**

- Thực hiện các phép toán số học và phép toán luận lý:
- Số học: Cộng, trừ, nhân, chia, tăng, giảm, đảo dấu
- Luận lý: AND, OR, XOR, NOT, phép dịch bit,...



*** Đơn vị điều khiển CU:**

- Điều khiển nhận lệnh từ bộ nhớ đưa vào thanh ghi lệnh
- Tăng nội dung của PC để trở sang lệnh kế tiếp
- Giải mã lệnh đã được nhận để xác định thao tác mà lệnh yêu cầu
- Phát ra các tín hiệu điều khiển thực hiện lệnh
- Nhận các tín hiệu yêu cầu từ bus hệ thống và đáp ứng với các yêu cầu đó.



*** Các tín hiệu đưa đến đơn vị điều khiển**

- Clock: tín hiệu xung nhịp từ mạch tạo dao động bên ngoài.
- Mã lệnh từ thanh ghi lệnh đưa đến để giải mã.
- Các cờ từ thanh ghi cờ cho biết trạng thái của CPU.
- Các tín hiệu yêu cầu từ bus điều khiển

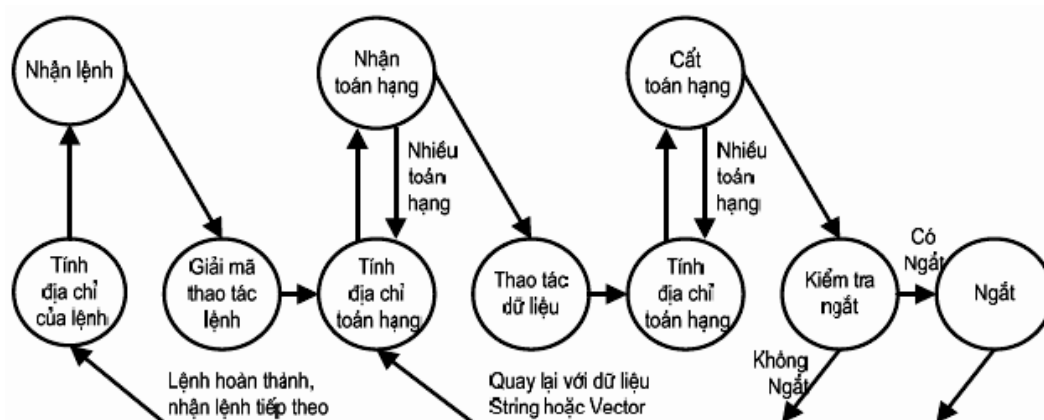
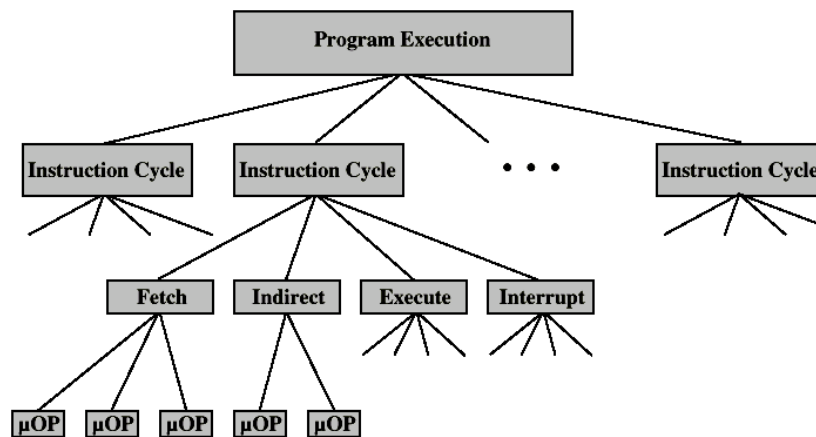
*** Các tín hiệu phát ra từ đơn vị điều khiển**

- Các tín hiệu điều khiển bên trong CPU:
 - Điều khiển các thanh ghi
 - Điều khiển ALU
- Các tín hiệu điều khiển bên ngoài CPU:
 - Điều khiển bộ nhớ
 - Điều khiển các mô đun nhập xuất

Phần 2: Hoạt động của chu trình lệnh

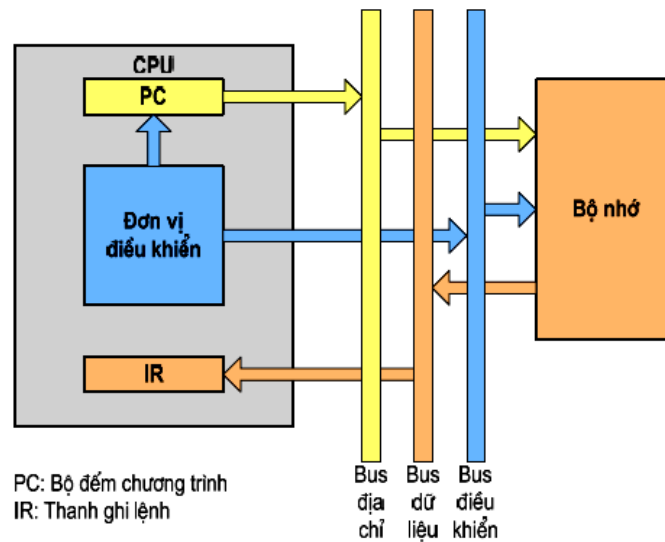
* Chu trình lệnh

- Nhận lệnh (Fetch Instruction FI)
- Giải mã lệnh (Decode Instruction DI)
- Nhận toán hạng (Fetch Operands FO)
- Thực hiện lệnh (Execute Instruction EI)
- Cát toán hạng (Write Operands WO)
- Ngắt (Interrupt Instruction II)



* Nhận lệnh (Fetch):

- CPU đưa địa chỉ của lệnh cần nhận từ bộ đếm chương trình PC ra bus địa chỉ
- CPU phát tín hiệu điều khiển đọc bộ nhớ
- Lệnh từ bộ nhớ được đặt lên bus dữ liệu và được CPU chép vào thanh ghi lệnh IR
- CPU tăng nội dung PC để trở sang lệnh kế tiếp



* Giải mã lệnh (Decode)

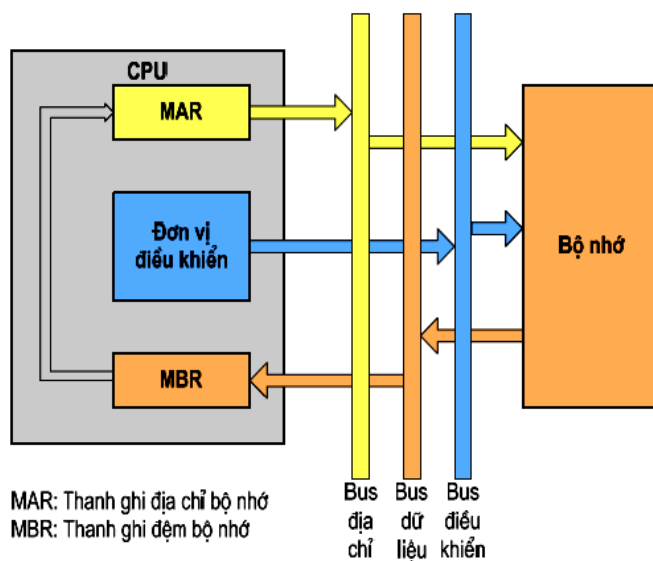
- Lệnh từ thanh ghi lệnh IR được đưa đến đơn vị điều khiển
- Đơn vị điều khiển tiến hành giải mã lệnh để xác định thao tác phải thực hiện
- Giải mã lệnh xảy ra bên trong CPU

* Nhận dữ liệu (Fetch Operand)

- CPU đưa địa chỉ của toán hạng ra bus địa chỉ
- CPU phát tín hiệu điều khiển đọc
- Toán hạng được đọc vào CPU
- Tương tự như nhận lệnh

* Nhận dữ liệu gián tiếp

- CPU đưa địa chỉ ra bus địa chỉ
- CPU phát tín hiệu điều khiển đọc
- Nội dung ngăn nhớ được đọc vào CPU, đó chính là địa chỉ của toán hạng
- Địa chỉ này được CPU phát ra bus địa chỉ để tìm ra toán hạng
- CPU phát tín hiệu điều khiển đọc
- Toán hạng được đọc vào CPU

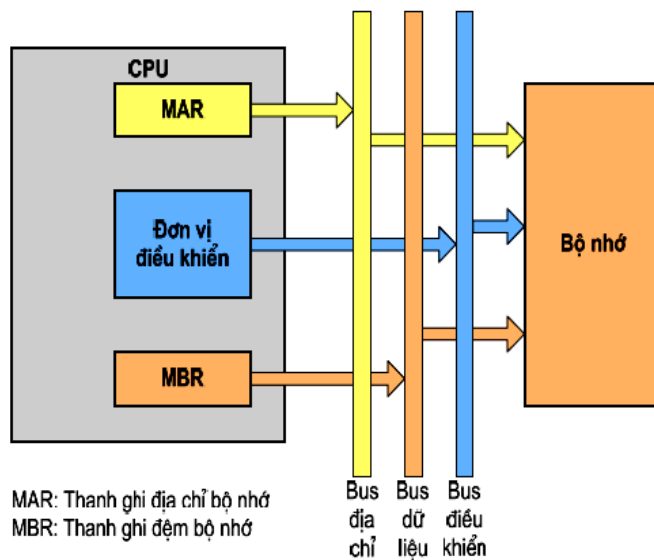


* Thực hiện lệnh (Execute)

- Có nhiều dạng tùy thuộc vào lệnh
- Có thể là:
 - + Đọc/Ghi bộ nhớ
 - + Nhập/ xuất
 - + Chuyển dữ liệu giữa các thanh ghi với nhau
 - + Chuyển dữ liệu giữa thanh ghi và bộ nhớ
 - + Thao tác số học/logic
 - + Chuyển điều khiển (rẽ nhánh)
 - + Ngắt
 - + ...

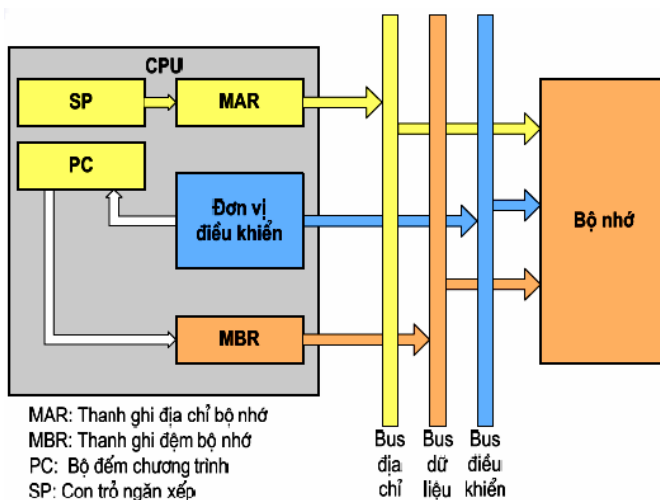
* Ghi toán hạng (Write)

- CPU đưa địa chỉ ra bus địa chỉ
- CPU đưa dữ liệu cần ghi ra bus dữ liệu
- CPU phát tín hiệu điều khiển ghi
- Dữ liệu trên bus dữ liệu được chép đến vị trí xác định



* Ngắt (Interrupt)

- Nội dung của bộ đếm chương trình PC (địa chỉ trở về sau khi ngắt) được đưa ra bus dữ liệu
- CPU đưa địa chỉ (thường được lấy từ con trỏ ngăn xếp SP) ra bus địa chỉ
- CPU phát tín hiệu điều khiển ghi bộ nhớ
- Địa chỉ trở về trên bus dữ liệu được ghi ra vị trí xác định (ở ngăn xếp)
- Địa chỉ lệnh đầu tiên của chương trình con điều khiển ngắt được nạp vào PC



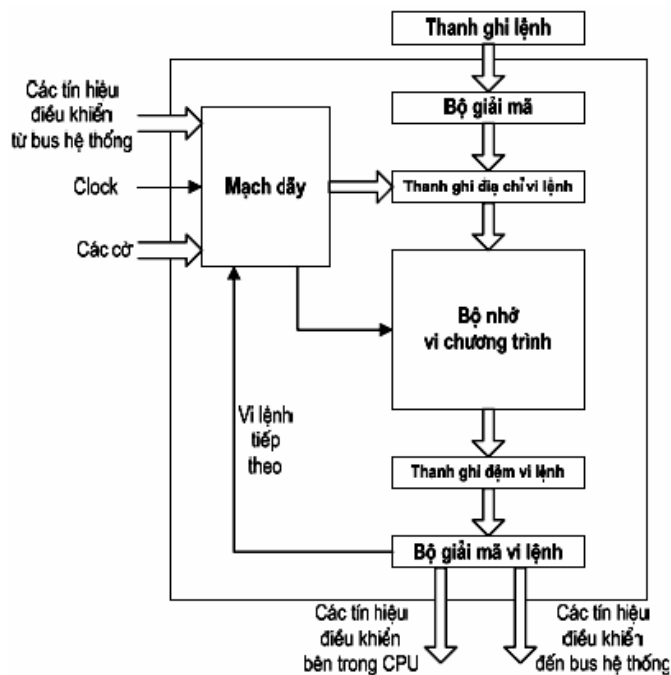
Phần 3: Đơn vị điều khiển

* Gồm 2 loại:

- Đơn vị điều khiển vi chương trình (Microprogrammed Control Unit)
- Đơn vị điều khiển phần cứng (Hardwired Control Unit)

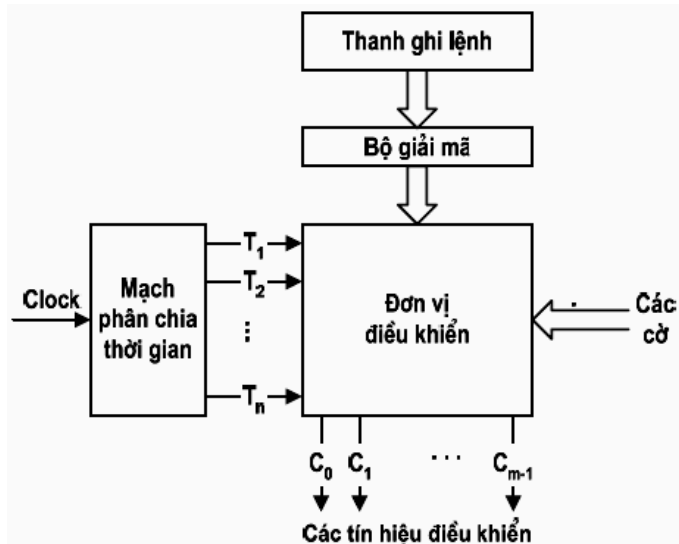
* Đơn vị điều khiển vi chương trình

- Bộ nhớ vi chương trình (ROM) lưu trữ các vi chương trình (microprogram)
- Một vi chương trình bao gồm các vi lệnh (microinstruction)
- Mỗi vi lệnh mã hoá cho một vi thao tác (microoperation)
- Để hoàn thành một lệnh cần thực hiện một hoặc một vài vi chương trình
- Tốc độ chậm



* Đơn vị điều khiển phần cứng

- Sử dụng vi mạch phần cứng để giải mã và tạo các tín hiệu điều khiển thực hiện lệnh
- Tốc độ nhanh
- Đơn vị điều khiển phức tạp



Phần 4: Kỹ thuật đường ống lệnh

* Khái niệm

- Mỗi chu trình lệnh cần thực hiện bằng nhiều thao tác
- Kỹ thuật đơn hướng (Scalar): Thực hiện tuần tự từng thao tác cho mỗi lệnh

=> Chậm

- Kỹ thuật đường ống (Pipeline): Thực hiện song song các thao tác cho nhiều lệnh đồng thời => nhanh hơn

- Ví dụ chu trình 1 lệnh gồm 5 bước:

- Nhận lệnh (I)
- Giải mã lệnh (D)
- Nhận toán hạng (F)
- Thực hiện lệnh (E)
- Cất toán hạng (W)

* So sánh scalar và pipeline

- **Scalar:** Nhiều chu kỳ máy cho 1 lệnh

Chu kỳ	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Lệnh 1	I	D	F	E	W										
Lệnh 2						I	D	F	E	W					
Lệnh 3											I	D	F	E	W

- **Pipeline:** Mỗi chu kỳ máy thực hiện xong 1 lệnh

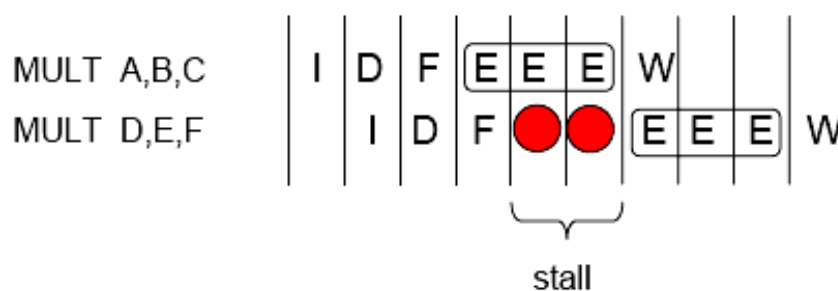
Chu kỳ	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Lệnh 1	I	D	F	E	W										
Lệnh 2		I	D	F	E	W									
Lệnh 3			I	D	F	E	W								
Lệnh 4				I	D	F	E	W							
Lệnh 5					I	D	F	E	W						
Lệnh 6						I	D	F	E	W					
Lệnh 7							I	D	F	E	W				
Lệnh 8								I	D	F	E	W			
Lệnh 9									I	D	F	E	W		
Lệnh 10										I	D	F	E	W	
Lệnh 11											I	D	F	E	W

* Các trở ngại của đường ống lệnh

- Thực tế không thể luôn đạt 1 chu kỳ máy/lệnh do các trở ngại dẫn đến sự gián đoạn của ống lệnh
- Trở ngại cấu trúc: do nhiều công đoạn dùng chung một tài nguyên
- Trở ngại dữ liệu: lệnh sau sử dụng dữ liệu kết quả của lệnh trước
- Trở ngại điều khiển: do các lệnh rẽ nhánh gây ra

* Trở ngại về cấu trúc

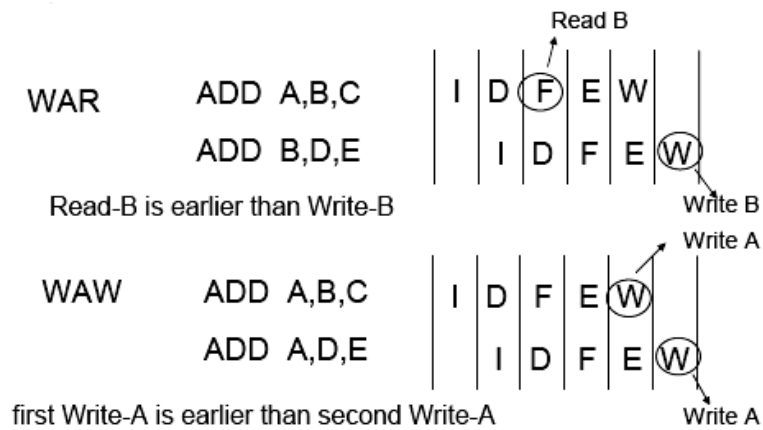
- Nguyên nhân: Dùng chung tài nguyên
- Khắc phục:
 - + Nhân tài nguyên để tránh xung đột
 - + Làm trễ
- Ví dụ 1: Bus dữ liệu truyền lệnh và dữ liệu □ Bus lệnh riêng, bus dữ liệu riêng (cache lệnh và cache dữ liệu)
- Ví dụ 2: Lệnh nhân cần nhiều chu kỳ thực thi (E)



* Trở ngại về dữ liệu

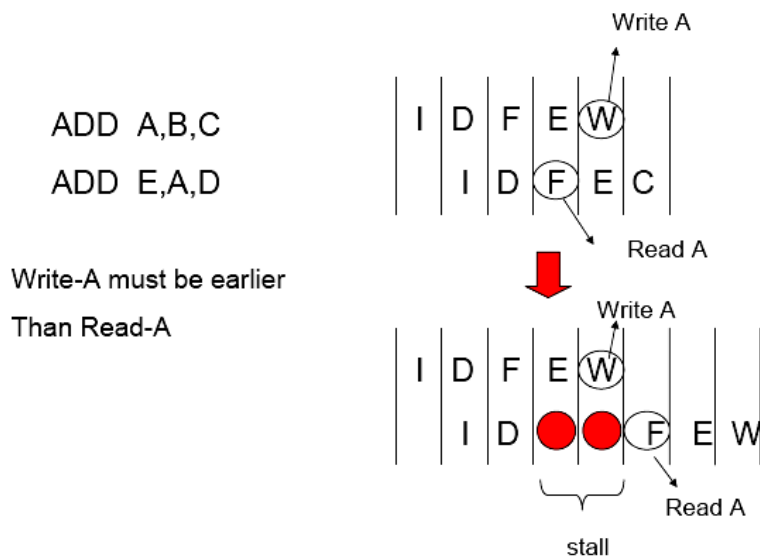
- Nguyên nhân: Lệnh sau sử dụng dữ liệu kết quả của lệnh trước
- Các dạng:

RAW	ADD A,B,C ADD E,A,D	Write-A must be earlier than Read-A
WAR	ADD A,B,C ADD B,D,E	Read-B must be earlier than Write-B
WAW	ADD A,B,C ADD A,D,E	First Write-A must be earlier Than second Write-A



no conflict at in-order pipeline
conflict at out-of-order pipeline

- RAW



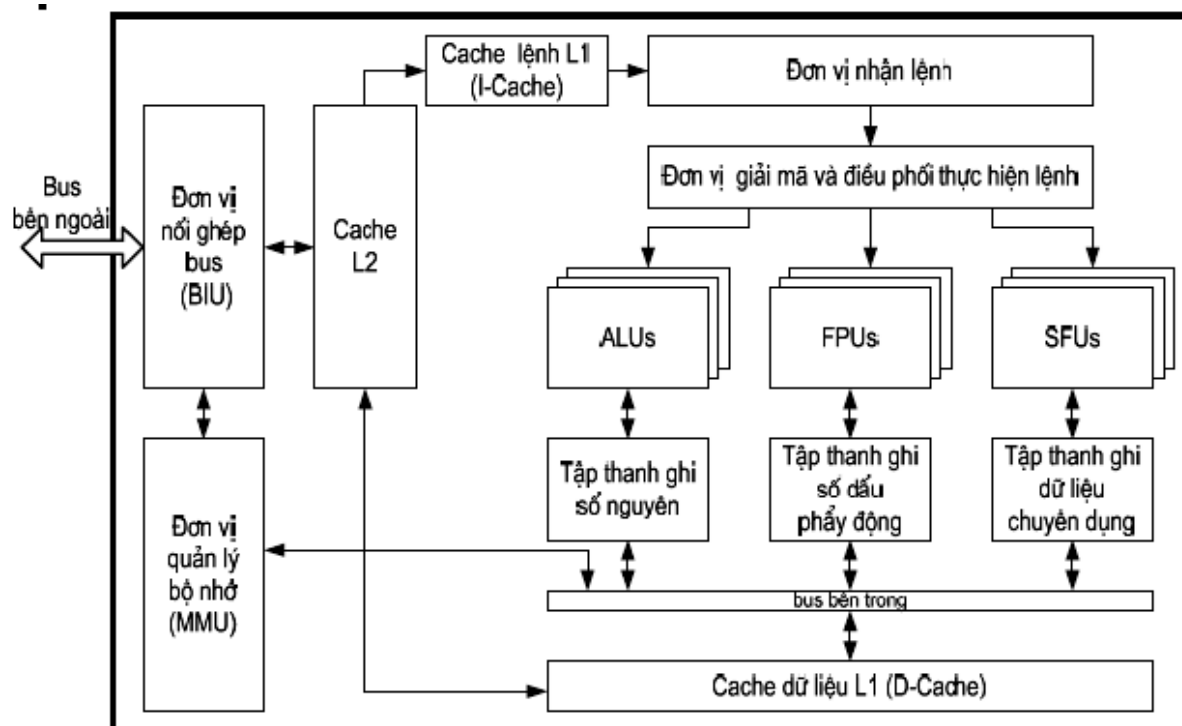
* Trở ngại về điều khiển

- Do lệnh rẽ nhánh gây ra

- Đây là dạng trở ngại gây thiệt hại nhiều nhất cho ống lệnh: toàn bộ các lệnh đang thực thi trong ống phải hủy

Chu kỳ	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Lệnh 1	I	D	F	E	W										
Lệnh 2		I	D	F	E				BRA 25 IF Zero						
Lệnh 3			I	D	F										
Lệnh 4				I	D										
Lệnh 5					I										
Lệnh 25						I	D	F	E	W					
Lệnh 26							I	D	F	E	W				
Lệnh 27								I	D	F	E	W			

Phần 5: Cấu trúc bộ xử lý tiên tiến



*** Các đơn vị xử lý dữ liệu chuyên dụng**

- Các đơn vị số nguyên (ALU)
- Các đơn vị số dấu chấm động (FPU)
- Các đơn vị chức năng đặc biệt (SFU)
 - + Đơn vị xử lý dữ liệu âm thanh
 - + Đơn vị xử lý dữ liệu hình ảnh
 - + Đơn vị xử lý dữ liệu vector

*** Mục đích:** Tăng khả năng xử lý các chức năng chuyên biệt

*** Bộ nhớ cache:**

- Được tích hợp trên chip vi xử lý
- Bao gồm hai đến ba mức cache
- Cache L1 gồm hai phần tách rời:
 - + Cache lệnh (Instruction cache)
 - + Cache dữ liệu (Data cache)
- => Giải quyết xung đột khi nhận lệnh và dữ liệu
- Cache L2 và L3: chung cho lệnh và dữ liệu

*** Mục đích:** Tăng hiệu suất truy cập bộ nhớ chính

*** Đơn vị quản lý bộ nhớ**

- Thường gọi là đơn vị MMU (Memory Management Unit) dùng để quản lý bộ nhớ ảo
- Chuyển đổi địa chỉ ảo (trong chương trình) thành địa chỉ vật lý (trong bộ nhớ)
- Cung cấp cơ chế phân trang/phân đoạn
- Cung cấp chế độ bảo vệ bộ nhớ

*** Mục đích:** Tăng dung lượng bộ nhớ chính bằng cách sử dụng bộ nhớ phụ

*** Các kiến trúc máy tính song song:**

- Nhu cầu giải các bài toán lớn ngày càng nhiều, cần những máy tính cực mạnh có khả năng xử lý tốc độ cao
- Kiến trúc máy tính tuần tự (Von-Neumann) tiến đến giới hạn tốc độ, một bộ xử lý duy nhất khó nâng cao hơn nữa khả năng xử lý
- Các kiến trúc máy tính song song giúp tăng hiệu suất tính toán cho máy tính:
 - + Kiến trúc song song mức lệnh IPL (Instruction-level parallelism) : Tăng số lượng lệnh thi hành được trên cùng 1 đơn vị thời gian
 - + Kiến trúc song song mức xử lý (Machine parallelism) : Tăng số lượng đơn vị xử lý phân cứng
- Cần kết hợp cả 2 kiến trúc song song để tạo ra các máy tính có hiệu suất cao

*** Kiến trúc song song mức lệnh**

- Siêu đường ống (Superpipeline)
 - + Chia mỗi thao tác trong chu trình lệnh ra n bước nhỏ => ống lệnh dài hơn
 - + Cần $1/n$ chu kỳ máy cho mỗi thao tác
- Siêu hướng (Superscalar)
 - + Sử dụng nhiều ống lệnh => CPU gồm nhiều đơn vị chức năng, cho phép thi hành nhiều lệnh đồng thời
 - + Mỗi chu kỳ máy thực hiện được nhiều lệnh
- VLIW (Very Long Instruction Word)
 - + Ghép nhiều lệnh đơn vào 1 từ máy để thực hiện đồng thời
 - + Ví dụ: CPU Itanium họ IA-64 của Intel cho phép ghép 3 lệnh/từ máy gọi là bundle gồm 128 bit

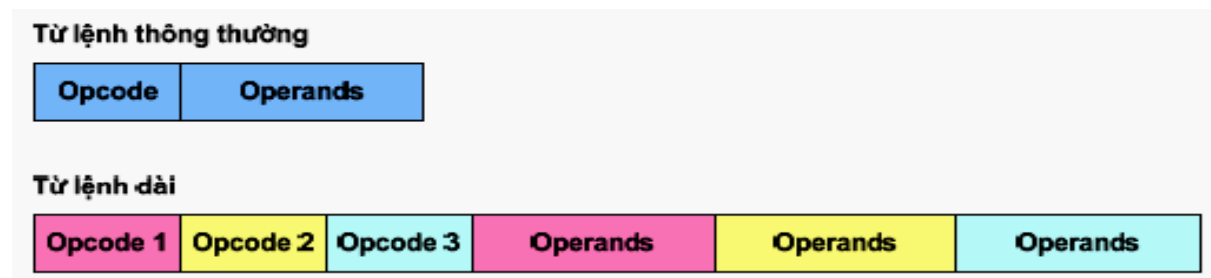
*** Super-pipeline:**

Chu kỳ	1		2		3		4		5		6		7	
Lệnh 1	I1	I2	D1	D2	F1	F2	E1	E2	W1	W2				
Lệnh 2		I1	I2	D1	D2	F1	F2	E1	E2	W1	W2			
Lệnh 3			I1	I2	D1	D2	F1	F2	E1	E2	W1	W2		
Lệnh 4				I1	I2	D1	D2	F1	F2	E1	E2	W1	W2	
Lệnh 5					I1	I2	D1	D2	F1	F2	E1	E2	W1	W2

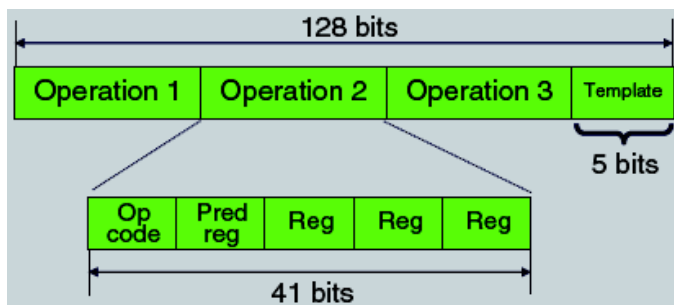
*** Super-scalar:**

Chu kỳ	1	2	3	4	5	6	7	8	9
Lệnh 1	I	D	F	E	W				
Lệnh 2	I	D	F	E	W				
Lệnh 3		I	D	F	E	W			
Lệnh 4		I	D	F	E	W			
Lệnh 5			I	D	F	E	W		
Lệnh 6			I	D	F	E	W		
Lệnh 7				I	D	F	E	W	
Lệnh 8				I	D	F	E	W	
Lệnh 9					I	D	F	E	W
Lệnh 10					I	D	F	E	W

* VLIW

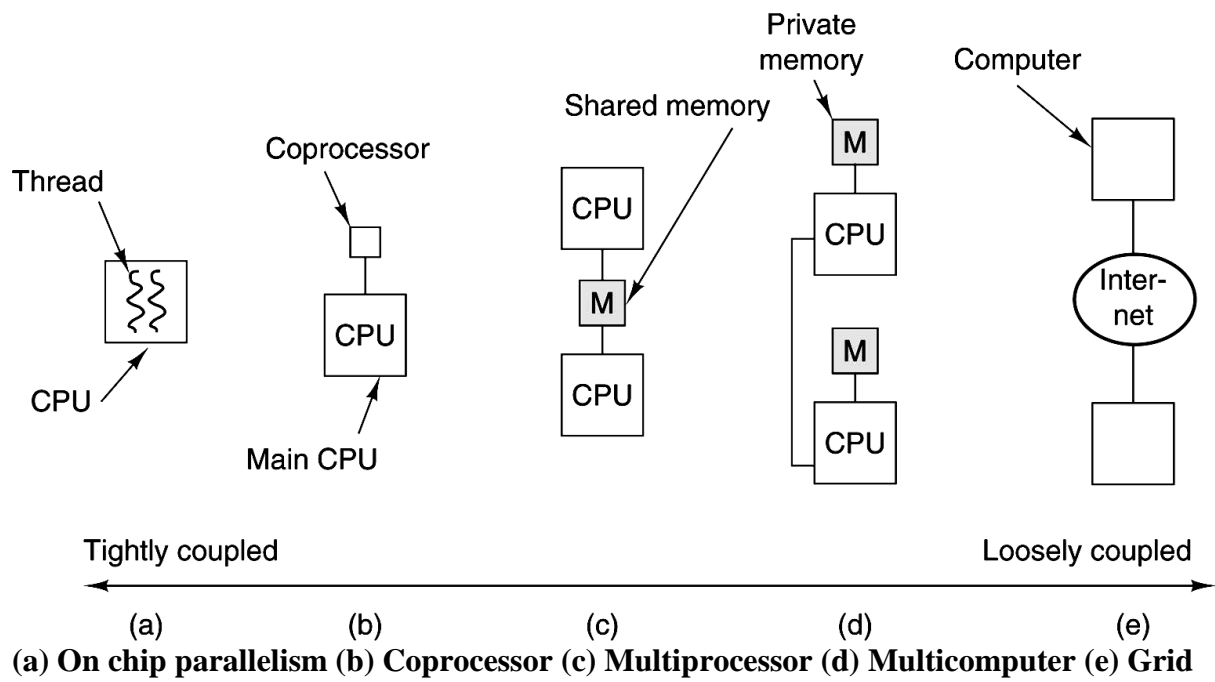


* **Ví dụ:** Khuôn dạng lệnh của CPU Intel Itanium

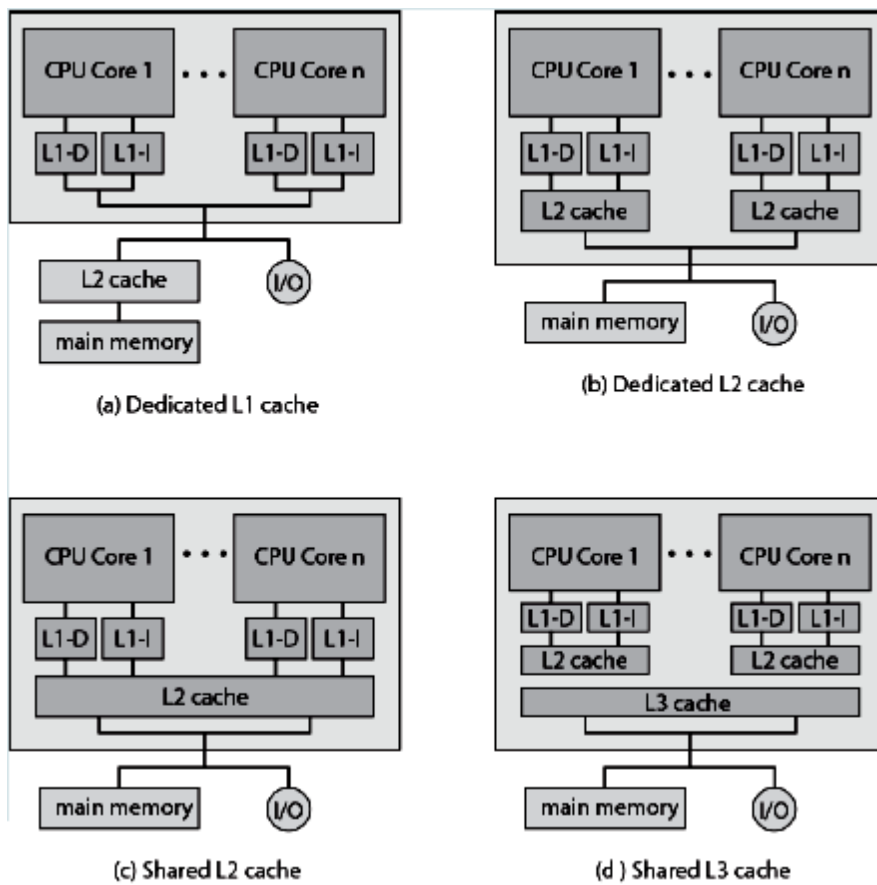


* Kiến trúc song song mức xử lý

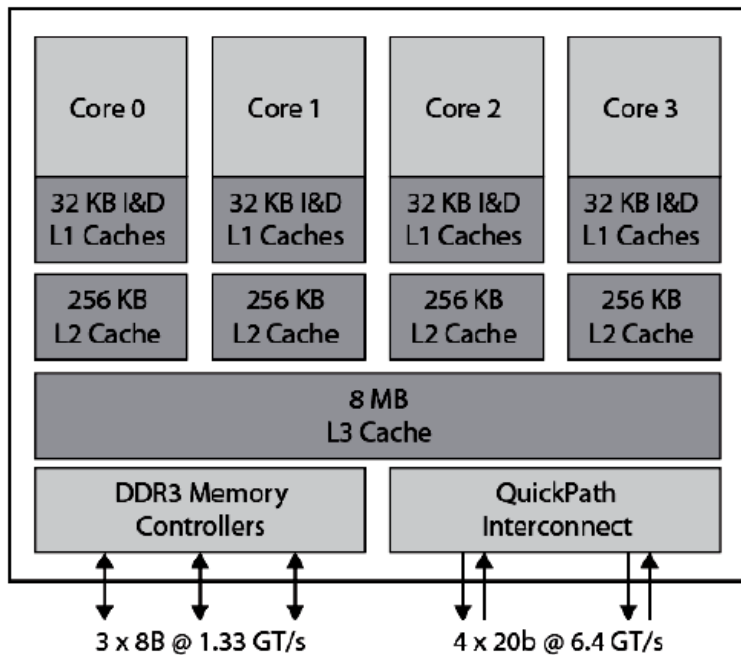
- Tích hợp nhiều bộ xử lý đồng thời để tăng khả năng thi hành chương trình
- Các xu hướng phát triển:
 - + Đa chương (multi-programming)
 - + Đa luồng (multi-threading)
 - + Đa nhân (multi-core)
 - + Đa xử lý (multi-processing)
 - + Đa máy tính (multi-computer)



* Multi-core

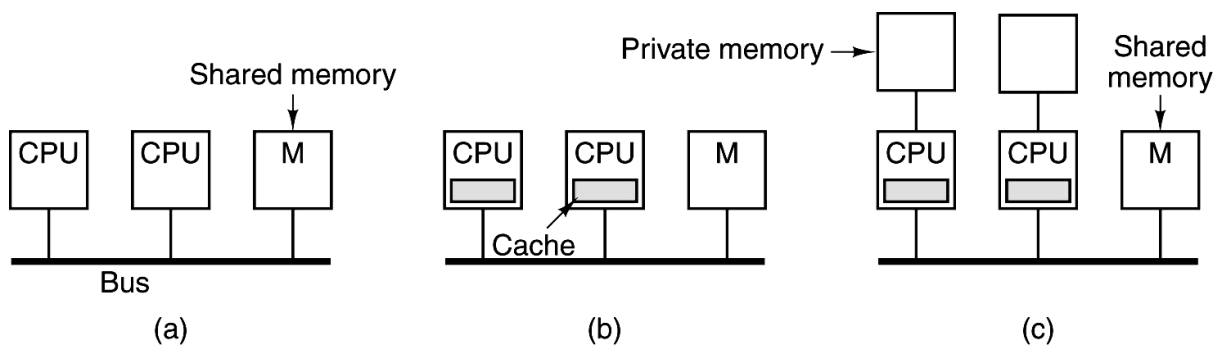


* **Ví dụ:** CPU Intel Core i7 gồm 4 nhân



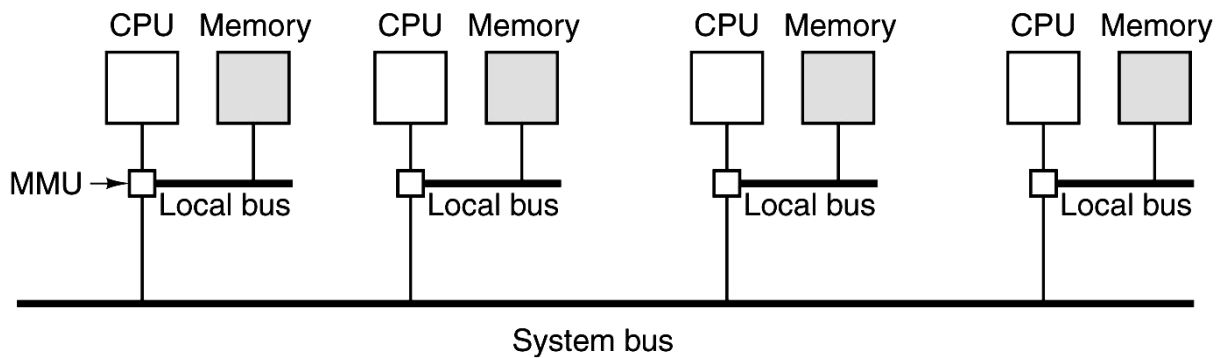
* **Multi-processor:**

- Sử dụng bus chung hoặc switch
- Sử dụng bộ nhớ chung hoặc riêng biệt



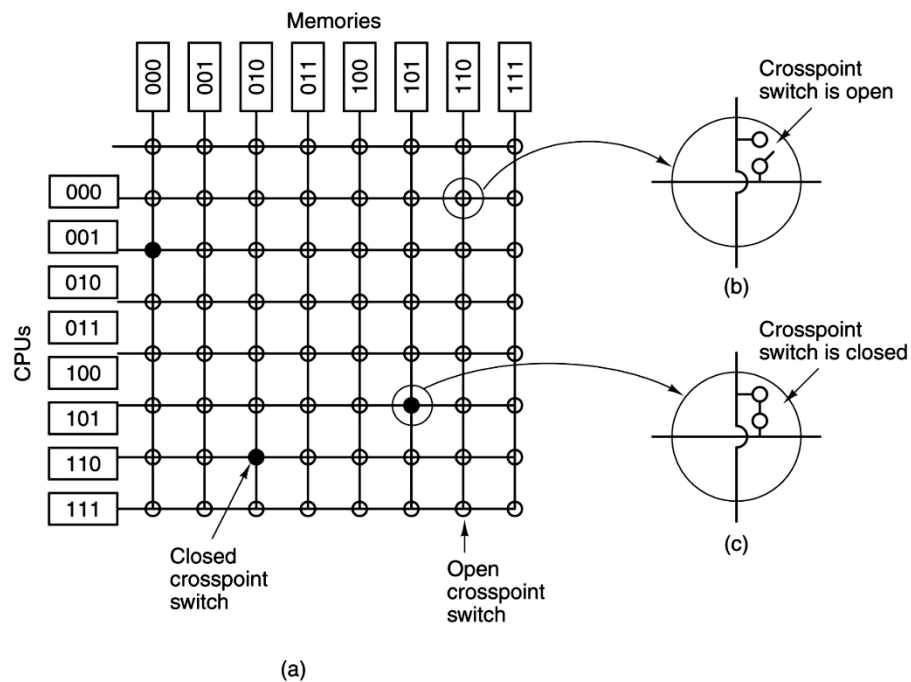
Sơ đồ UMA (Uniform Memory Access) dùng bus chung và bộ nhớ chung

- Sơ đồ NUMA (Non-Uniform Memory Access) dùng bus chung và bộ nhớ riêng

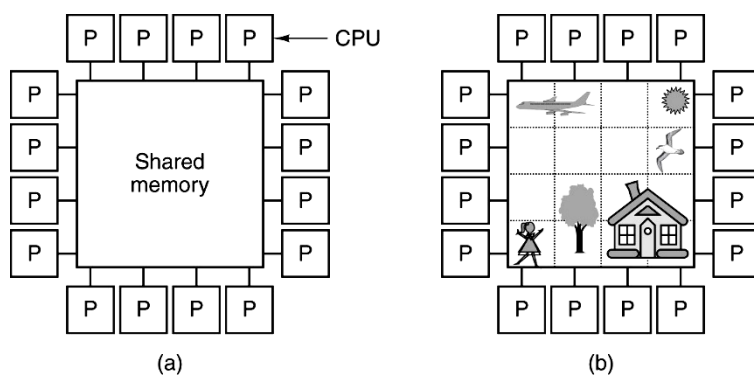


- Sơ đồ UMA (Uniform Memory Access) dùng switch và bộ nhớ riêng

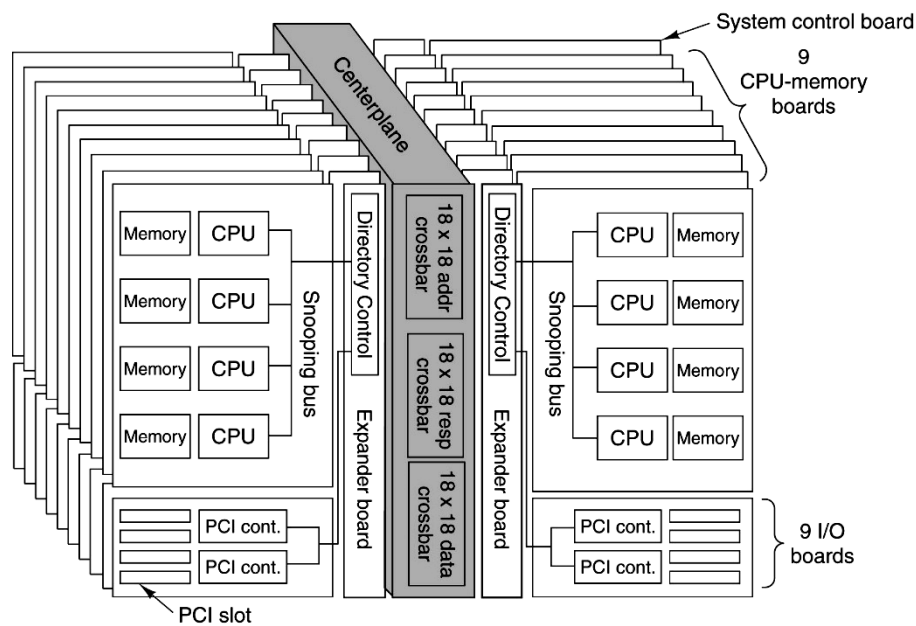
- Còn gọi là hệ thống đa xử lý đối xứng SMP (Symmetric Multi-Processors)



- Sơ đồ multi-processor dùng bộ nhớ chung



* **Ví dụ:** Hệ thống SUN E25K (NUMA multi-processor)

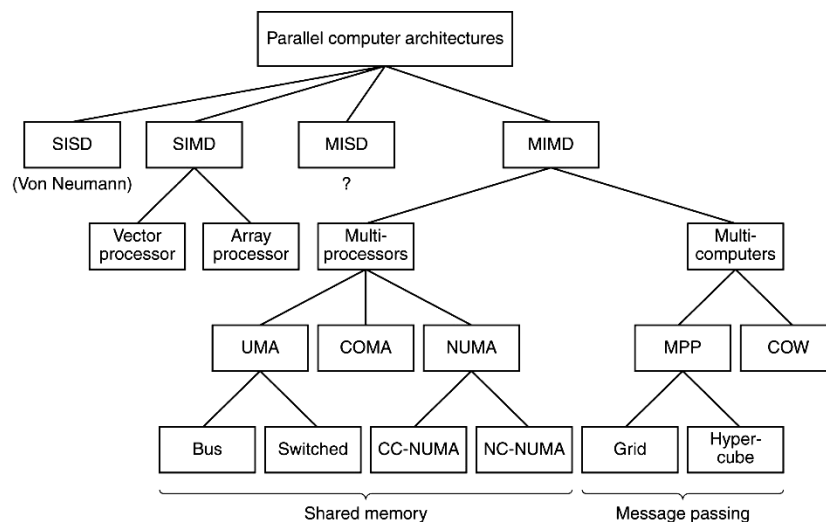


* **Multi-computer**

- Phân loại theo Flynn (1966): Căn cứ vào số lượng lệnh và số lượng dữ liệu có thể xử lý là 1 hay nhiều

- + Single instruction, single data stream – **SISD**
- + Single instruction, multiple data stream – **SIMD**
- + Multiple instruction, single data stream – **MISD**
- + Multiple instruction, multiple data stream- **MIMD**

* **Sơ đồ phân loại Flynn**



* Ví dụ về SIMD

ADD R3 ← R1, R2

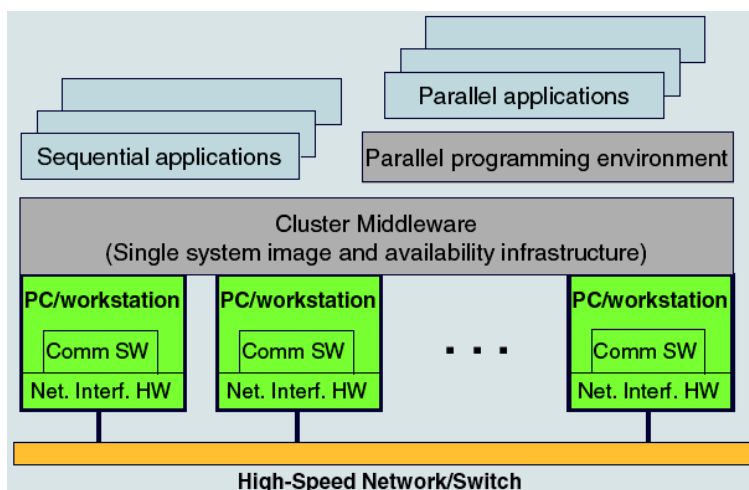
R1	a7	a6	a5	a4	a3	a2	a1	a0
	+	+	+	+	+	+	+	+
R2	b7	b6	b5	b4	b3	b2	b1	b0
	=	=	=	=	=	=	=	=
R3	a7+b7	a6+b6	a5+b5	a4+b4	a3+b3	a2+b2	a1+b1	a0+b0

MULADD R3 ← R1, R2

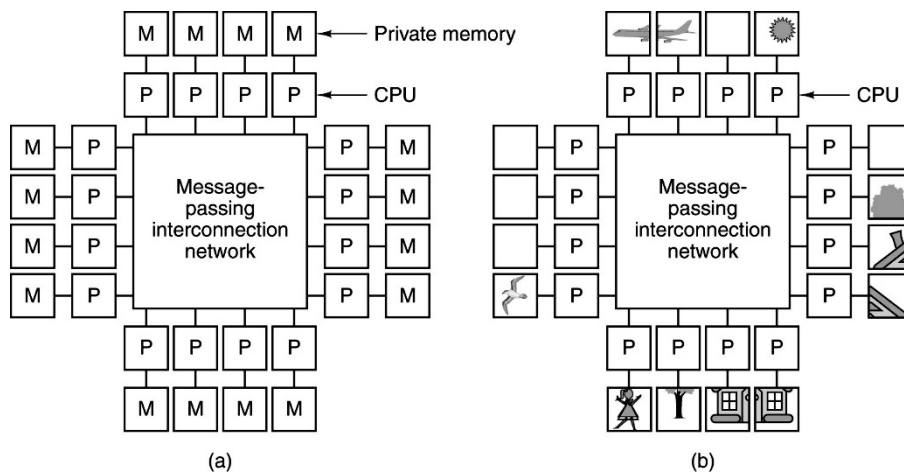
R1	a7	a6	a5	a4	a3	a2	a1	a0
	x&+	x&+	x&+	x&+	x&+	x&+	x&+	x&+
R2	b7	b6	b5	b4	b3	b2	b1	b0
	=	=	=	=	=	=	=	=
R3	(a6×b6)+(a7×b7)	(a4×b4)+(a5×b5)	(a2×b2)+(a3×b3)	(a0×b0)+(a1×b1)				

* Cluster

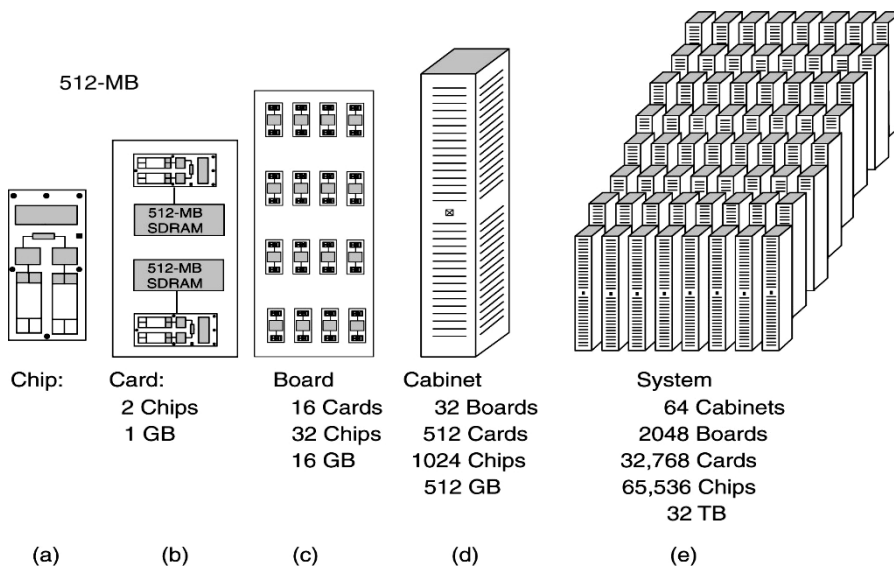
- Là 1 dạng máy tính loại MIMD gồm nhiều máy tính độc lập kết nối qua mạng tốc độ cao, mỗi máy có CPU, BN và IO riêng
- Dùng phương pháp truyền thông báo (Message Passing) để trao đổi thông tin (bằng phần mềm)
 - + MPI (Message Passing Interface)
 - + PVM (Parallel Virtual Machine)
- Gồm 2 loại:
 - + NOW (Network of Workstations) hoặc COW (Cluster of Workstations): Kết nối qua LAN
 - + Grid : Kết nối qua Internet



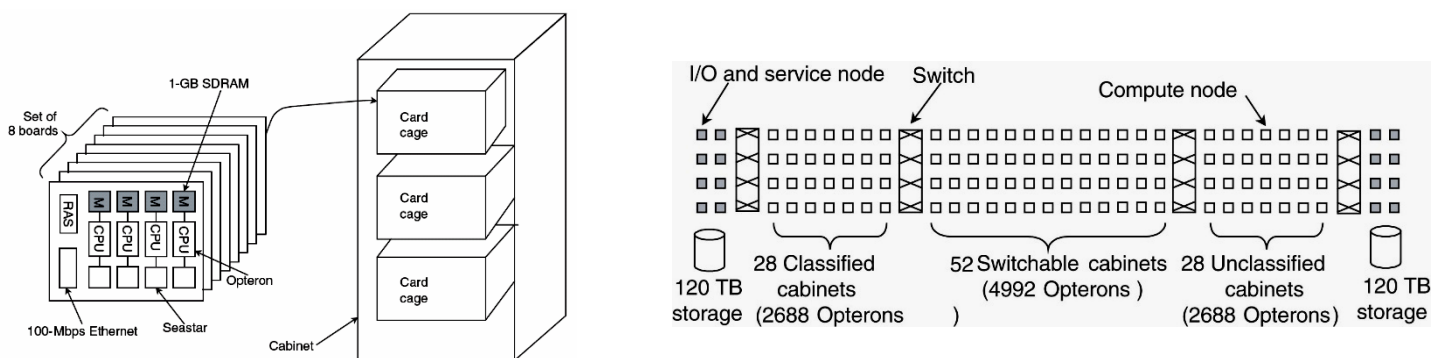
* Message-passing multi-computer



* Ví dụ: Siêu máy tính Bluegen của IBM



* Ví dụ: Siêu máy tính Red Storm của Cray



*** So sánh 2 siêu máy tính Bluegen & Red Storm:**

Item	BlueGene/L	Red Storm
CPU	32-Bit PowerPC	64-Bit Opteron
Clock	700 MHz	2 GHz
Compute CPUs	65,536	10,368
CPUs/board	32	4
CPUs/cabinet	1024	96
Compute cabinets	64	108
Teraflops/sec	71	41
Memory/CPU	512 MB	2-4 GB
Total memory	32 TB	10 TB
Router	PowerPC	Seastar
Number of routers	65,536	10,368
Interconnect	3D torus $64 \times 32 \times 32$	3D torus $27 \times 16 \times 24$
Other networks	Gigabit Ethernet	Fast Ethernet
Partitionable	No	Yes
Compute OS	Custom	Custom
I/O OS	Linux	Linux
Vendor	IBM	Cray Research
Expensive	Yes	Yes

*** Top 10 siêu máy tính 06/2010 trên trang top500.org**

Rank	Site	Computer
1	Oak Ridge National Laboratory United States	Jaguar - Cray XT5-HE Opteron Six Core 2.6 GHz Cray Inc.
2	National Supercomputing Centre in Shenzhen China (Thâm Quyến)	Nebulae (Tinh Vân) - Dawning TC3600 Blade, Intel X5650 Dawning
3	DOE/NNSA/LANL United States	Roadrunner - BladeCenter QS22/LS21 Cluster, PowerXCell 8i 3.2 Ghz / Opteron DC 1.8 GHz, Voltaire Infiniband IBM
4	National Institute for Computational Sciences/University of Tennessee United States	Kraken XT5 - Cray XT5-HE Opteron Six Core 2.6 GHz Cray Inc.
5	Forschungszentrum Juelich (FZJ) Germany	JUGENE - Blue Gene/P Solution IBM
6	NASA/Ames Research Center/NAS United States	Pleiades - SGI Altix ICE 8200EX/8400EX, Xeon HT QC 3.0 Ghz SGI
7	National SuperComputer Center in Tianjin/NUDT China (Thiên Tân)	Tianhe-1 (Tinh Hà) - NUDT TH-1 Cluster, Xeon E5540/E5450 NUDT
8	DOE/NNSA/LLNL United States	BlueGene/L - eServer Blue Gene Solution IBM
9	Argonne National Laboratory United States	Intrepid - Blue Gene/P Solution IBM
10	National Renewable Energy Laboratory United States	Red Sky - Sun Blade x6275, Xeon X55xx 2.93 Ghz, Infiniband Sun

*** Top 10 siêu máy tính 06/2011 trên trang top500.org**

Rank	Site	Computer
1	RIKEN Advanced Institute for Computational Science - Japan	K computer, SPARC64 VIIIfx 2.0GHz Fujitsu
2	National Supercomputing Center in Tianjin (Thiên Tân) – China	Tianhe-1A (Tinh Hà) X5670 2.93Ghz 6C, NVIDIA GPU NUDT
3	DOE/SC/Oak Ridge National Laboratory United States	Jaguar - Cray XT5-HE Opteron 6-core 2.6 GHz Cray Inc.
4	National Supercomputing Centre in Shenzhen (Thâm Quyến) – China	Nebulae (Tinh Vân) Intel X5650, NVidia Tesla C2050 GPU Dawning
5	GSIC Center, Tokyo Institute of Technology Japan	TSUBAME 2.0 G7 Xeon 6C X5670, Nvidia GPU, NEC/HP
6	DOE/NNSA/LANL/SNL United States	Cielo - Cray XE6 8-core 2.4 GHz Cray Inc.
7	NASA/Ames Research Center/NAS United States	Pleiades Xeon HT QC 3.0/Xeon 5570/5670 2.93 Ghz SGI
8	DOE/SC/LBNL/NERSC United States	Hopper - Cray XE6 12-core 2.1 GHz Cray Inc.
9	Commissariat à l'Energie Atomique (CEA) France	Tera-100 - Bull bullx super-node S6010/S6030 Bull SA
10	DOE/NNSA/LANL United States	Roadrunner - PowerXCell 8i 3.2 Ghz / Opteron DC 1.8 GHz IBM

*** Top 10 siêu máy tính 06/2012 trên trang top500.org**

Rank	Site	Computer
1	DOE/NNSA/LLNL United States	Sequoia - BlueGene/Q , Power BQC 16C 1.60 GHz, Custom IBM
2	RIKEN Advanced Institute for Computational Science Japan	K computer , SPARC64 VIIIfx 2.0GHz, Tofu interconnect Fujitsu
3	DOE/SC/Argonne National Laboratory United States	Mira - BlueGene/Q , Power BQC 16C 1.60GHz, Custom IBM
4	Leibniz Rechenzentrum Germany	SuperMUC - iDataPlex DX360M4 , Xeon E5-2680 8C 2.70GHz, Infiniband FDR IBM
5	National Supercomputing Center in Tianjin China	Tianhe-1A - NUDT YH MPP , Xeon X5670 6C 2.93 GHz, NVIDIA 2050 NUDT
6	DOE/SC/Oak Ridge National Laboratory United States	Jaguar - Cray XK6 , Opteron 6274 16C 2.200GHz, Cray Gemini interconnect , NVIDIA 2090 Cray Inc.
7	CINECA Italy	Fermi - BlueGene/Q , Power BQC 16C 1.60GHz, Custom IBM
8	Forschungszentrum Juelich (FZJ) Germany	JuQUEEN - BlueGene/Q , Power BQC 16C 1.60GHz, Custom IBM
9	CEA/TGCC-GENCI France	Curie thin nodes - Bullx B510 , Xeon E5-2680 8C 2.700GHz, Infiniband QDR Bull
10	National Supercomputing Centre in Shenzhen (NSCS) China	Nebulae - Dawning TC3600 Blade System , Xeon X5650 6C 2.66GHz, Infiniband QDR , NVIDIA 2050 Dawning