

Chương 1: Tổng quan về Kiến trúc Máy tính

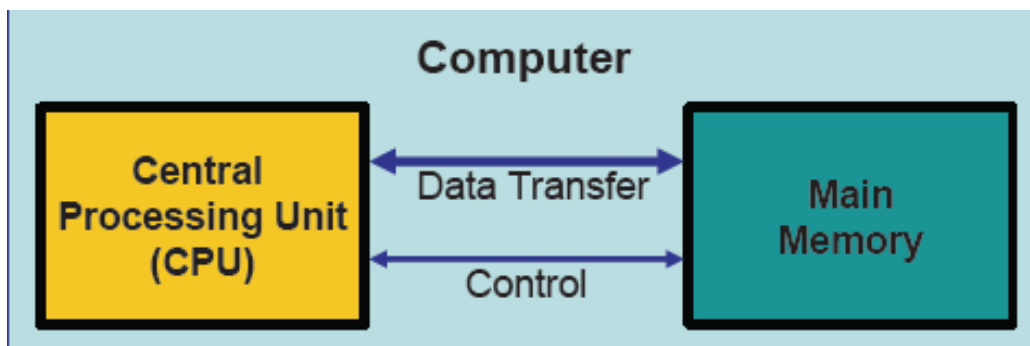
Nội dung

1. Các khái niệm và định nghĩa cơ bản
2. Nguyên lý hoạt động
3. Nội dung môn học
4. Phân loại máy tính
5. Lịch sử phát triển máy tính
6. Tổ chức tổng quát máy tính

Phần 1: Các khái niệm và định nghĩa

* Máy tính (Computer):

Máy tính là thiết bị điện tử xử lý dữ liệu, hoạt động một cách tự động dưới sự điều khiển của chương trình được lưu trữ trong bộ nhớ chính của nó.

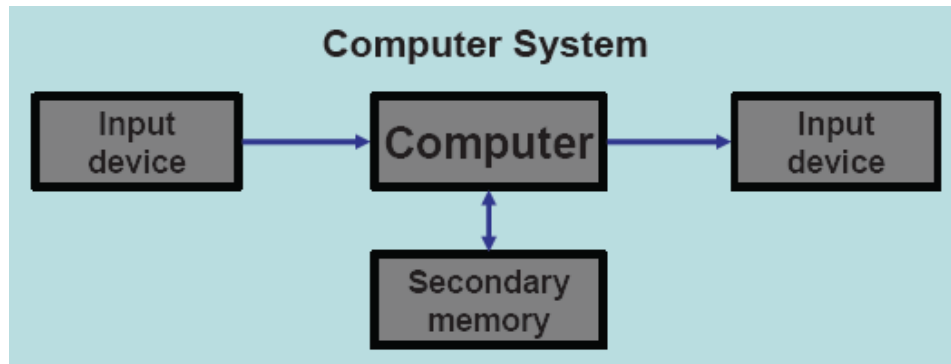


* Hệ thống máy tính (Computer System)

Một hệ thống máy tính bao gồm một máy tính và các thiết bị ngoại vi.

* Thiết bị ngoại vi (Peripherals)

Bao gồm các thiết bị nhập (input devices), thiết bị xuất (output devices) và bộ nhớ thứ cấp (secondary storage).



*** Chương trình (program):**

Danh sách các lệnh (command) hoặc chỉ thị (instruction) để bộ xử lý trong máy tính thi hành.

*** Lệnh và tập lệnh:**

- Bộ xử lý (CPU) trong máy tính được thiết kế để hiểu và thi hành được các lệnh được thiết kế trước của nhà sản xuất CPU
- Tập hợp tất cả các lệnh CPU hiểu được gọi là tập lệnh (instruction set) của CPU đó

*** Lập trình (programming):**

- Việc viết 1 chương trình cho máy tính chạy gọi là lập trình. Người viết chương trình gọi là lập trình viên (programmer)
- Máy tính không thể tự nó giải được 1 bài toán vì cần có chương trình do con người viết ra.
- Người lập trình phải biết cách giải bài toán mới có thể viết chương trình cho máy giải được.

*** Xử lý dữ liệu (data processing):**

- Bao gồm các thao tác: Thu thập, nhập, lưu trữ, tìm kiếm, tính toán, trình bày kết quả.
- Hệ thống máy tính cần có con người tham gia.

*** Ngôn ngữ lập trình:**

- *Ngôn ngữ tự nhiên (natural language):*

+ Do con người sử dụng. Lệ thuộc ngữ cảnh, không có tính chính xác và nhất quán cần thiết cho máy tính

+ Không sử dụng được cho máy tính

- Ngôn ngữ máy (*machine language*)

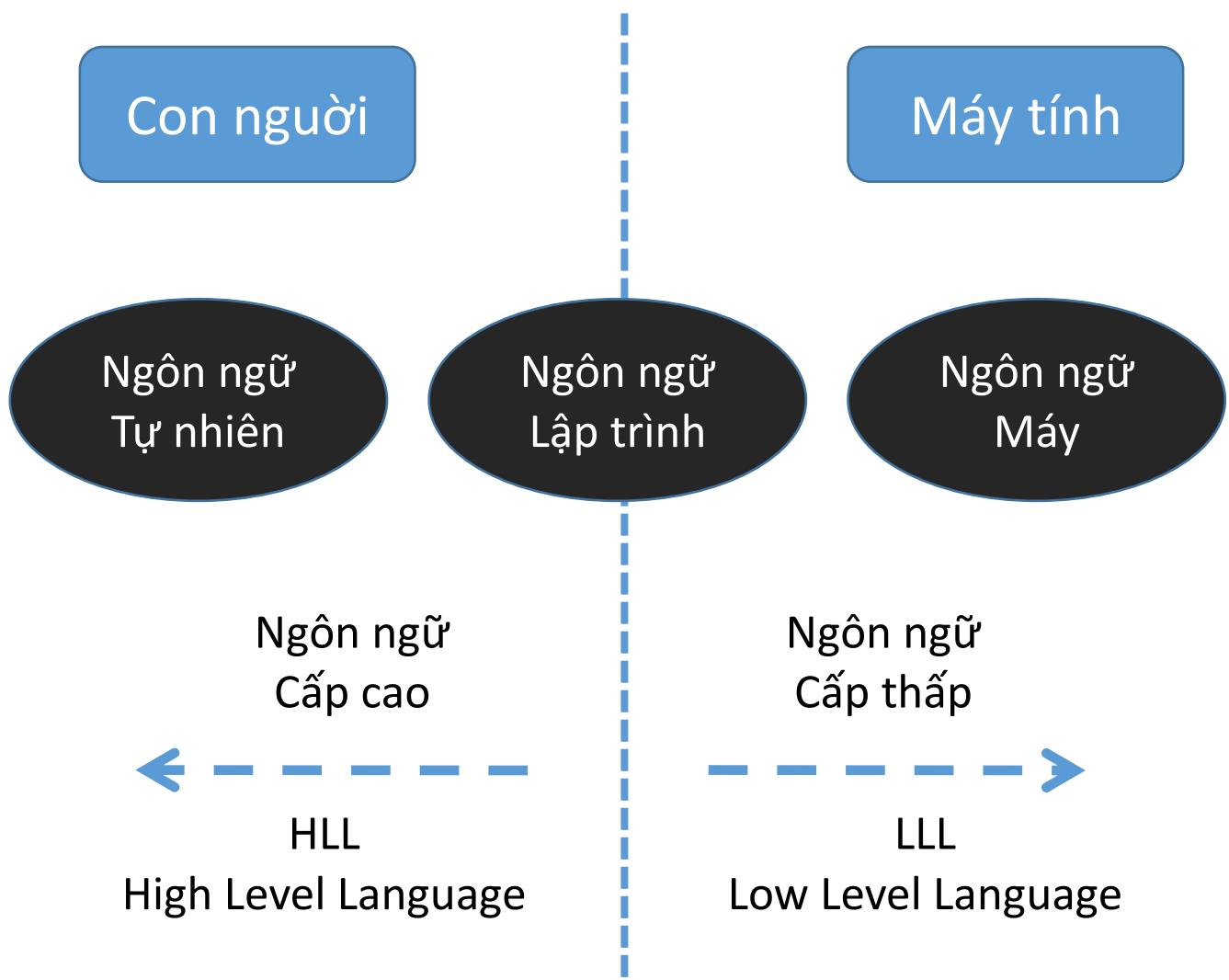
+ Là các ký hiệu nhị phân (số 0 và 1) mà các linh kiện điện tử trong máy tính hiểu và xử lý được.

+ Rất khó khăn khi con người sử dụng trực tiếp.

- Ngôn ngữ dùng ký hiệu/ Hợp ngữ *Symbolic language/ Assembly language* dạng ký hiệu/gọi nhớ của tập lệnh CPU

- Ngôn ngữ lập trình (*programming language*)

Là trung gian giữa ngôn ngữ tự nhiên và ngôn ngữ máy.



*** Chương trình dịch (translator):**

- Máy tính không hiểu được ngôn ngữ lập trình và ngôn ngữ tự nhiên
- Cần phải dịch ngôn ngữ lập trình do con người viết ra ngôn ngữ máy để máy tính thi hành
- Việc dịch có thể thực hiện tự động thông qua 1 chương trình gọi là chương trình dịch
- Bao gồm 2 loại:
 - + Trình Biên dịch (Compiler)
 - + Trình Thông dịch (Interpreter)

*** Thành phần máy tính:**

- Phần cứng (hardware)
 - + Bộ xử lý CPU
 - + Bộ nhớ (Memory)
 - + Thiết bị ngoại vi
- Phần mềm (software)
 - + Hệ thống (system software)
 - + Ứng dụng (application software)
- Phần dẻo (firmware)

Trung gian giữa phần cứng và phần mềm

Phần 2: Nguyên lý hoạt động

*** Mô hình Turing:**

- Là một mô hình máy tính lý thuyết do nhà toán học người Anh Alan Turing đưa ra năm 1936 gọi là máy Turing
- Dùng để kiểm tra khả năng giải các loại bài toán khác nhau bằng các thuật toán trên máy móc

- Luận đề Church-Turing khẳng định mọi hàm toán học tính được thì cũng có thể dùng các máy Turing để tính, và do đó cho phép định nghĩa các khái niệm như sự tính được của hàm hay thuật toán.

* Máy Turing

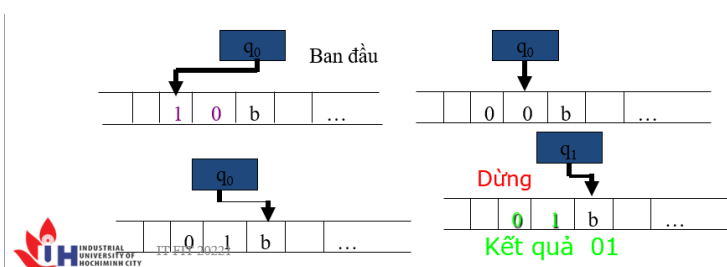
- Gồm 1 dải băng dài vô hạn có nhiều ô.
- 1 đầu đọc/ghi để đọc/ ghi từng ký tự hoặc dịch chuyển trên 1 ô của dải băng.
- 1 khối xử lý chứa tập các trạng thái

* Nguyên lý hoạt động máy Turing

- Máy làm việc theo từng bước rời rạc. Một lệnh của máy như sau : $q_i S_i S_j X q_j$.
- Nghĩa là : trạng thái hiện hành của máy là q_i đầu đọc ghi đang ở ô S_i thì sẽ ghi đề S_j vào ô hiện tại và dịch chuyển hoặc đứng yên theo chỉ thị là X và trạng thái mới của máy là q_j
- Dữ liệu của bài toán là 1 chuỗi các ký hiệu thuộc tập các ký hiệu của máy không kể ký hiệu rỗng b
- Trạng thái trong ban đầu của máy là q_0 được cất vô băng. Quá trình sẽ dừng lại khi trạng thái trong của máy là trạng thái kết thúc q_f .

* Ví dụ máy Turing

- Thực hiện phép toán NOT trên chuỗi các bit 0/1
- Chuỗi dữ liệu nhập ban đầu là 10
- Tập các ký hiệu của máy $\{0,1\}$
- Tập các trạng thái trong $\{q_0, q_1\}$
- Tập lệnh gồm 3 lệnh : $q_0 0 1 R q_0$, $q_0 1 0 R q_0$, $q_0 b b N q_1$



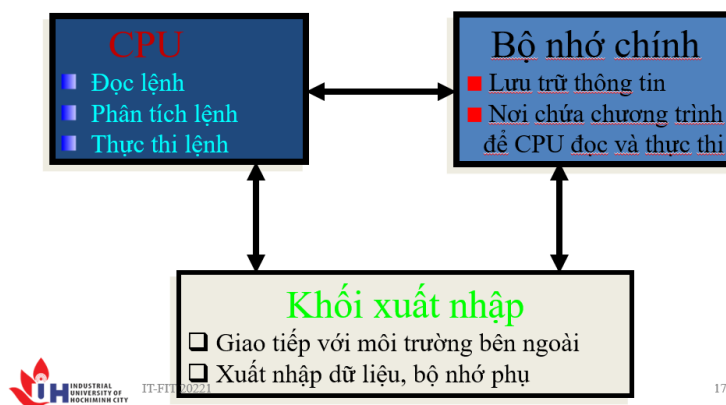
* Mô hình Von Neumann

- Là một mô hình máy tính thực tế do nhà toán học người Mỹ gốc Hungary John Von Neumann đưa ra khi tham gia thiết kế máy tính EDVAC năm 1945.
- Máy gồm 3 khối cơ bản : đơn vị xử lý, bộ nhớ và hệ thống xuất nhập.
- Hiện đang áp dụng cho các máy tính ngày nay.

* Nguyên lý Von Neumann

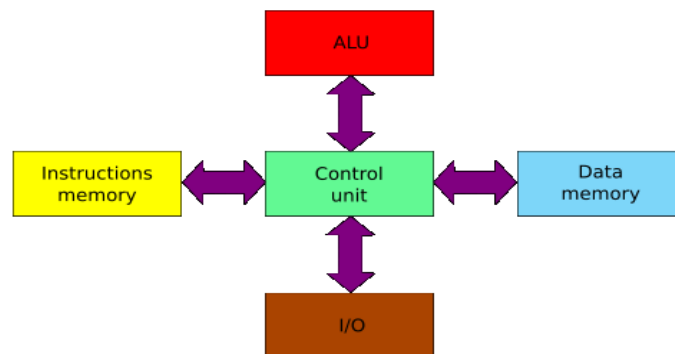
- Chương trình điều khiển xử lý dữ liệu cũng được xem là data và được lưu trữ trong bộ nhớ gọi là chương trình lưu trữ.
- Bộ nhớ chia làm nhiều ô, mỗi ô có 1 địa chỉ (đánh số thứ tự) để có thể chọn lựa ô nhớ trong quá trình đọc ghi dữ liệu. (nguyên lý định địa chỉ)
- Các lệnh được thực hiện t tuần tự nhờ 1 bộ đếm chương trình (thanh ghi lệnh) nằm bên trong đơn vị xử lý.

* Sơ đồ máy tính Von Neumann



* Kiến trúc Harvard

- Được sử dụng trong máy Harvard Mark I (IBM 1944)
- Bao gồm 2 khối bộ nhớ riêng biệt:
 - + Bộ nhớ lệnh chương trình
 - + Bộ nhớ dữ liệu
- Máy tính Von Neumann ngày nay chỉ sử dụng 1 khối bộ nhớ chung cho cả chương trình và dữ liệu
- Tuy nhiên một số loại máy có sử dụng kiến trúc Harvard



* Tại sao sử dụng kiến trúc Von Neumann ?

- Tạo ra các máy tính đa năng, lập trình được
 - Giải các bài toán khác nhau bằng các chương trình khác nhau.
- Các lệnh chương trình được thi hành 1 cách tự động
- Máy tính có thể chế tạo từ các linh kiện điện tử cơ bản
 - + Chức năng xử lý dữ liệu thực hiện bằng các cổng logic
 - + Chức năng lưu trữ dữ liệu thực hiện qua các ô nhớ
 - + Chức năng truyền dữ liệu thực hiện qua các đường truyền dẫn điện

Phần 3: Nội dung môn học

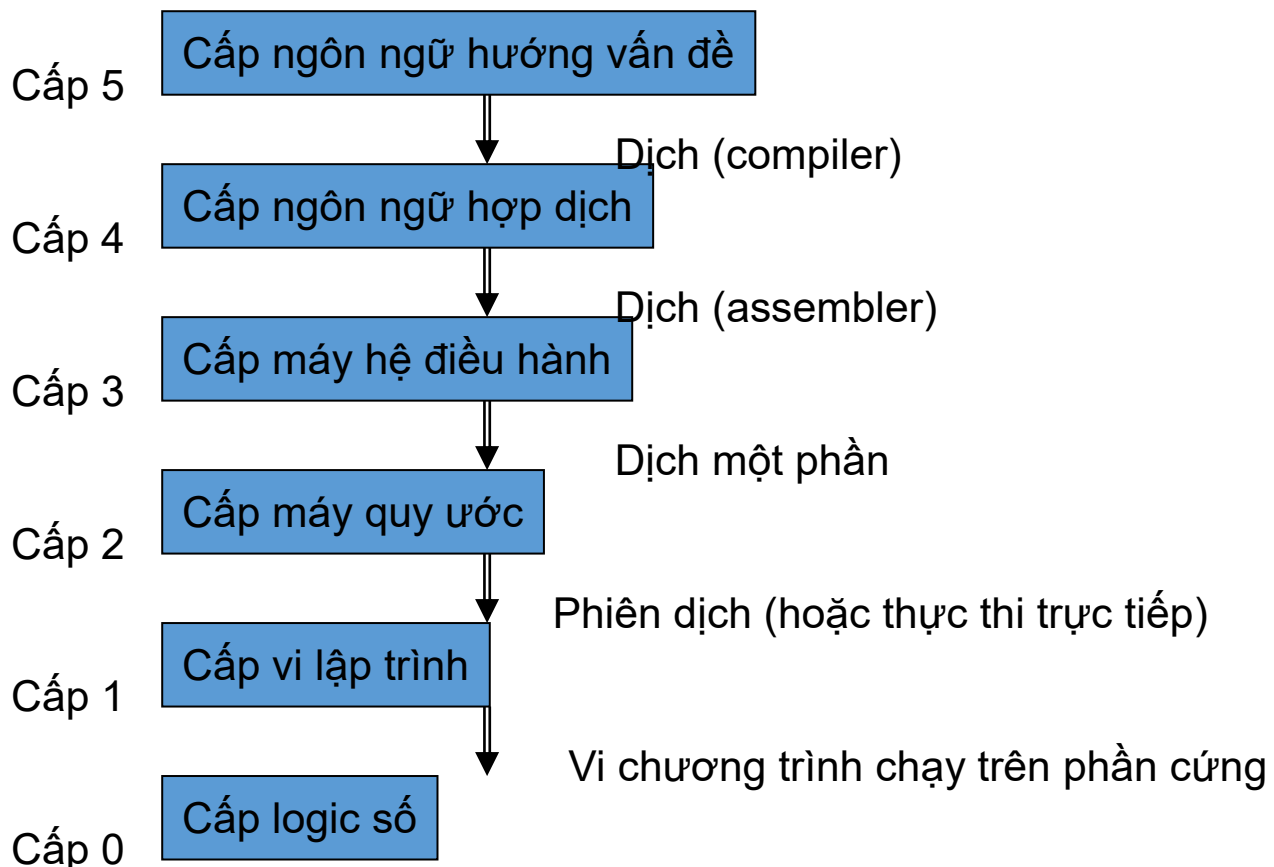
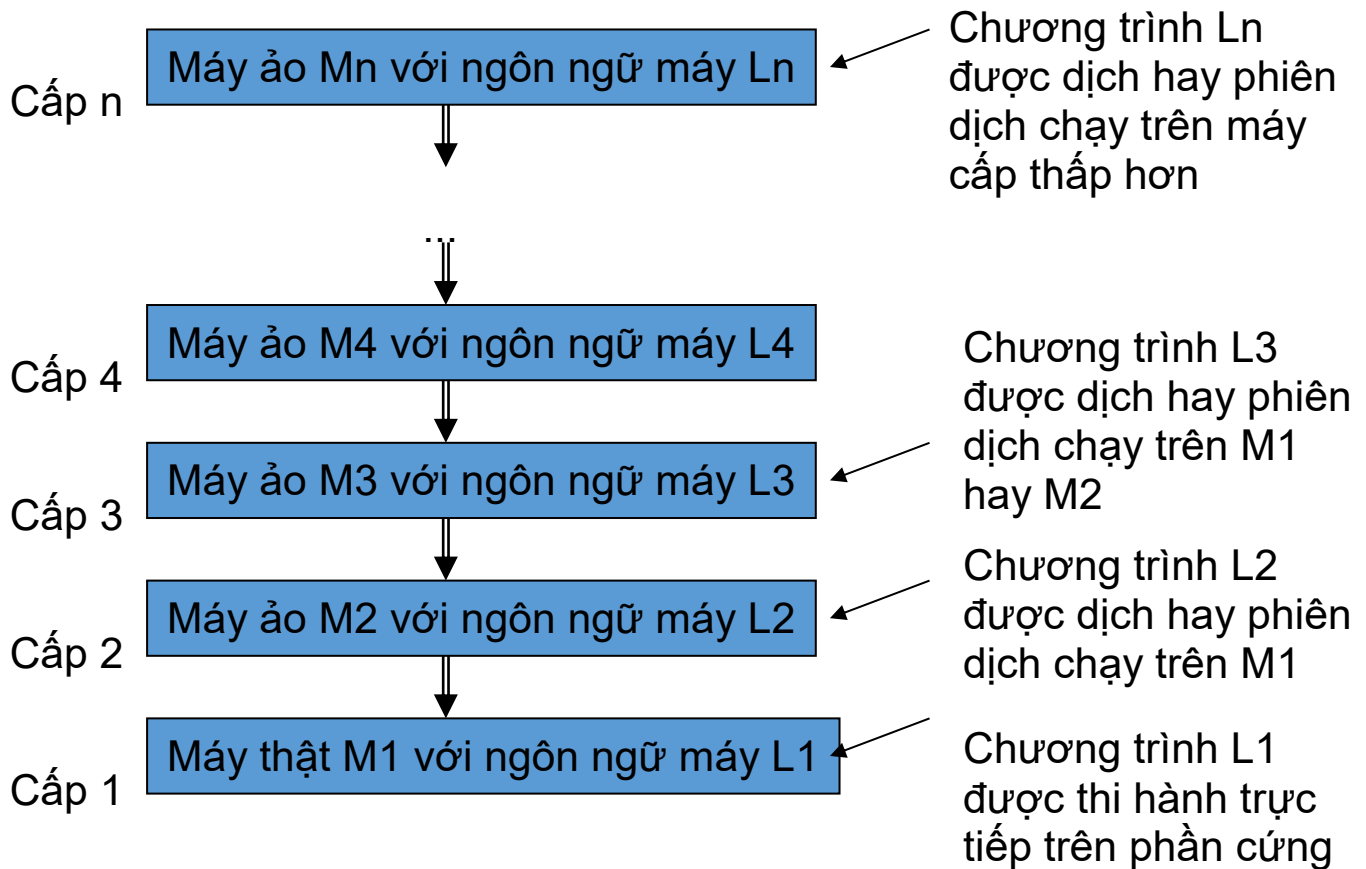
*** Cung cấp các kiến thức cơ bản về nguyên lý hoạt động và tổ chức ở các máy tính số**

- Vấn đề đánh giá hiệu suất
- Kiến trúc tập lệnh
- Tính toán số học
- Đường đi dữ liệu và tín hiệu điều khiển
- Hệ thống bộ nhớ
- Giao tiếp với ngoại vi

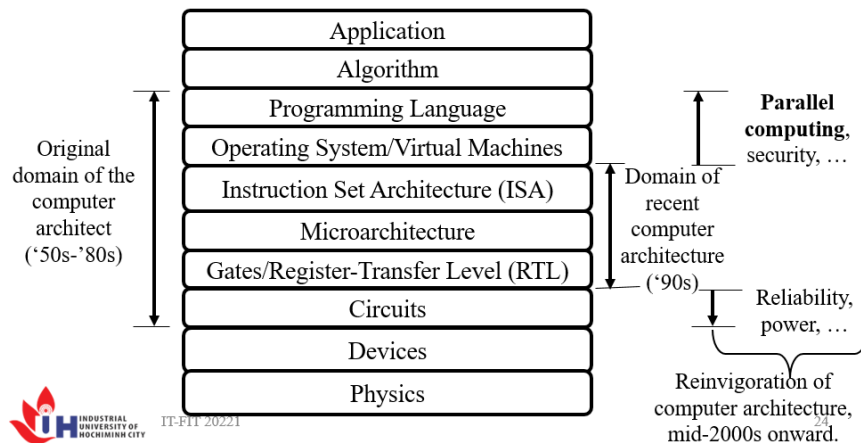
*** Mục đích môn học:**

- Nắm vững các kiến thức cơ bản về tổ chức và hoạt động của máy tính số
- Làm nền tảng để tìm hiểu cấu trúc của các phần tử khác nhau trong một máy tính số
- Vận dụng để thiết kế, cải tạo, sửa chữa các hệ thống ứng dụng vi xử lý–vi điều khiển, vi mạch, hệ thống nhúng
- Vận dụng vào thiết kế, cải tạo các hệ thống phần mềm sao cho hoạt động hiệu quả, phù hợp với kiến trúc phần cứng

*** Sự phân cấp máy tính theo Tanenbaum:**



*** Các quan niệm về máy tính theo thời gian:**



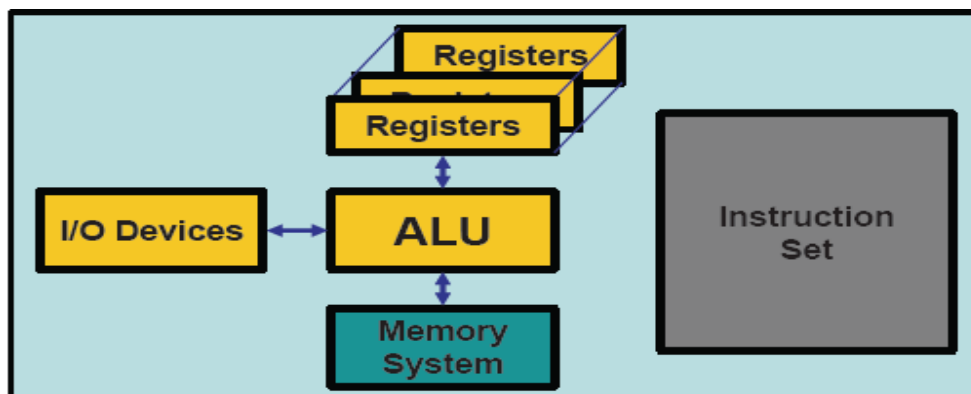
*** Các thuật ngữ thường gặp:**

- Cấu trúc máy tính (computer structure)
- Kiến trúc máy tính (computer architecture)
- Tổ chức máy tính (computer organization)

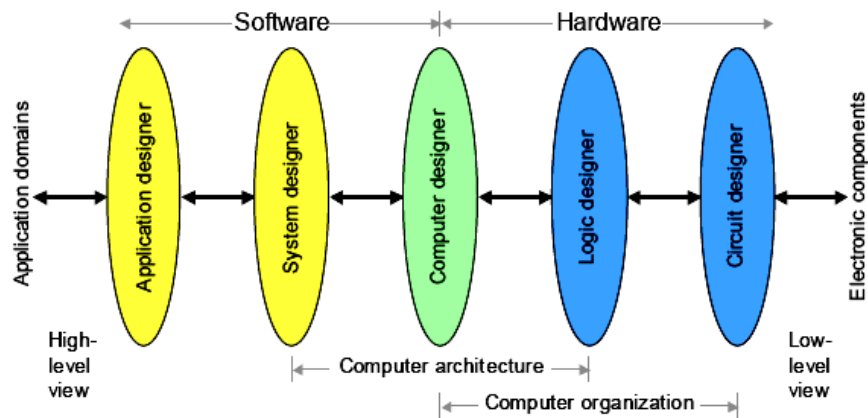
*** Kiến trúc máy tính** liên quan đến các thuộc tính của một hệ thống máy tính có khả năng thấy được đối với người lập trình, hoặc các thuộc tính có ảnh hưởng trực tiếp đến logic thực hiện của chương trình (phần mềm)

*** Tổ chức máy tính** liên quan đến các khối chức năng và sự kết nối giữa chúng để thực hiện các đặc tả kiến trúc (nghĩa là làm thế nào hiện thực các tính năng kiến trúc)

Ví dụ: Tín hiệu điều khiển, giao tiếp giữa máy tính và các thiết bị ngoại vi, công nghệ bộ nhớ, ...(phần cứng)



* Vị trí KTMT và TCMT



* So sánh KTMT và TCMT

- Ví dụ chức năng “nhân”:

+ Kiến trúc: có hay không có lệnh nhân.

+ Tổ chức: một đơn vị thực hiện chức năng “nhân” đặc biệt hay việc dùng nhiều đơn vị “cộng” để thực hiện chức năng “nhân”.

* Nhiều nhà sản xuất máy tính đưa ra dòng (họ) các mẫu máy tính, các máy này có cùng kiến trúc nhưng khác nhau về mặt tổ chức

- Tất cả máy tính họ x86 của Intel có cùng kiến trúc cơ bản

- Họ System/370 của IBM có cùng kiến trúc cơ bản

* Điều này dẫn đến

- Nhiều máy khác nhau trong cùng họ có giá thành và hiệu suất khác nhau

- Tổ chức sẽ thay đổi theo công nghệ

+ Tương thích về chương trình

+ Tối thiểu đối với các máy thế hệ trước (backward)

* Tại sao học KTMT ?

- Để trở thành chuyên nghiệp trong lĩnh vực máy tính ngày nay, bạn không nên xem máy tính như một hộp đen (black box) thực hiện các chương trình bằng ma thuật.
- Bạn nên hiểu các thành phần chức năng của một hệ thống máy tính, đặc tính, hiệu suất và tương tác của chúng.
- Bạn cần hiểu rõ kiến trúc máy tính để có thể xây dựng các chương trình chạy hiệu quả trên máy tính.
- Khi chọn lựa để dùng một hệ thống, bạn phải có khả năng hiểu được ưu và nhược điểm của các thành phần khác nhau, ví dụ tốc độ xung nhịp CPU so với kích thước bộ nhớ.

* Cấu trúc và Chức năng:

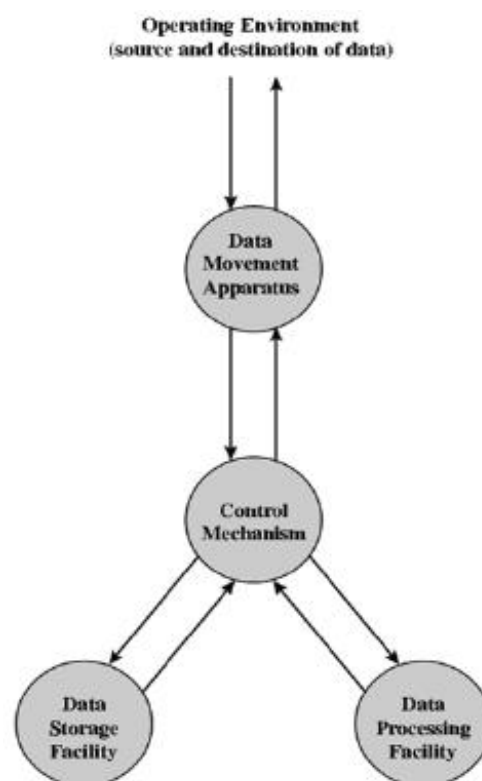
- Nhận biết bản chất phân cấp của các hệ thống phức tạp nhất

Hệ thống phân cấp là tập hợp các hệ thống con có quan hệ với nhau, sao cho mỗi hệ thống con này lại có tính phân cấp về cấu trúc như vậy, cho đến khi chúng ta đạt đến hệ thống con nguyên tử thấp nhất

- Cấu trúc là cách mà các thành phần quan hệ với các thành phần khác
- Chức năng là tác vụ của các thành phần chức năng riêng biệt nằm trong cấu trúc
- Theo cách mô tả, có 2 cách tiếp cận
 - + Bottom-up
 - + Top-down

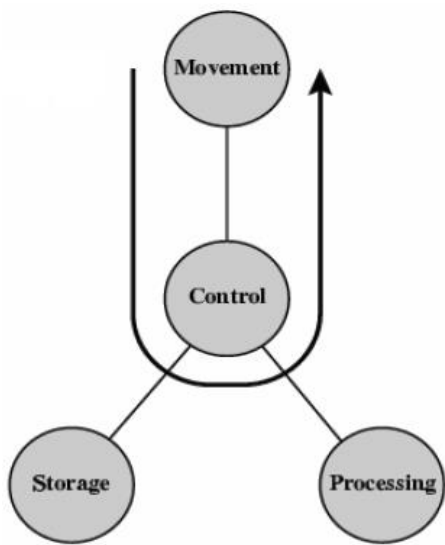
* Chức năng Máy tính

- Xử lý dữ liệu (Data processing)
- Lưu trữ dữ liệu (Data storage)
- Dịch chuyển dữ liệu (Data movement)
- Điều khiển (Control)

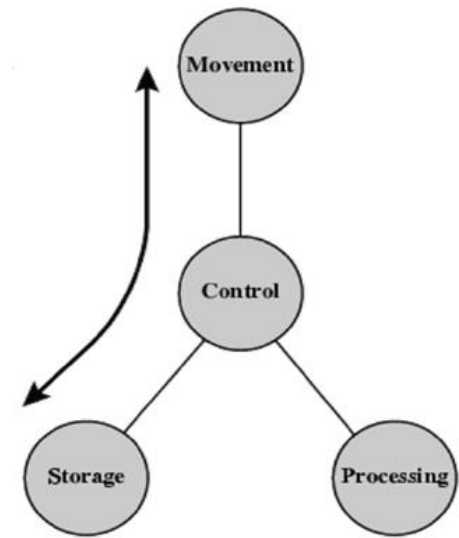


*** Các tác vụ**

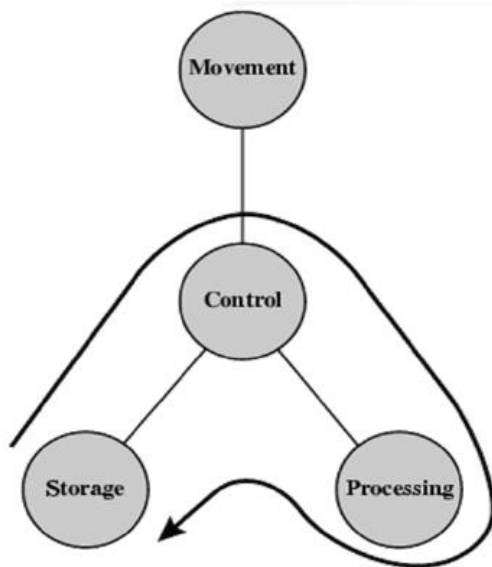
- Dịch chuyển dữ liệu



- Lưu trữ dữ liệu

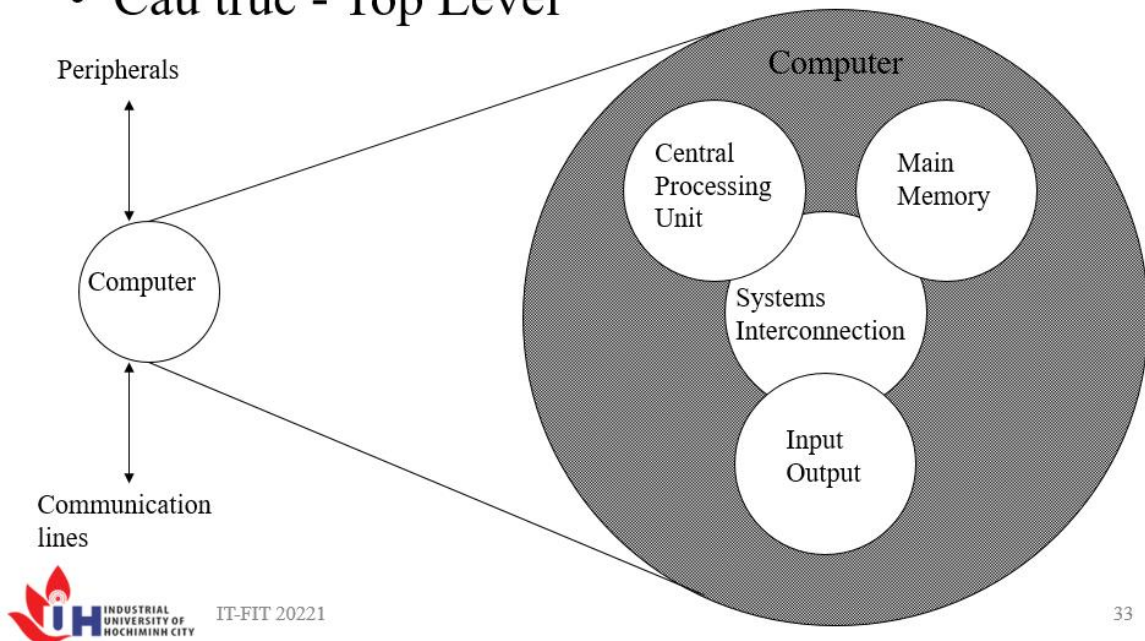


- Xử lý dữ liệu



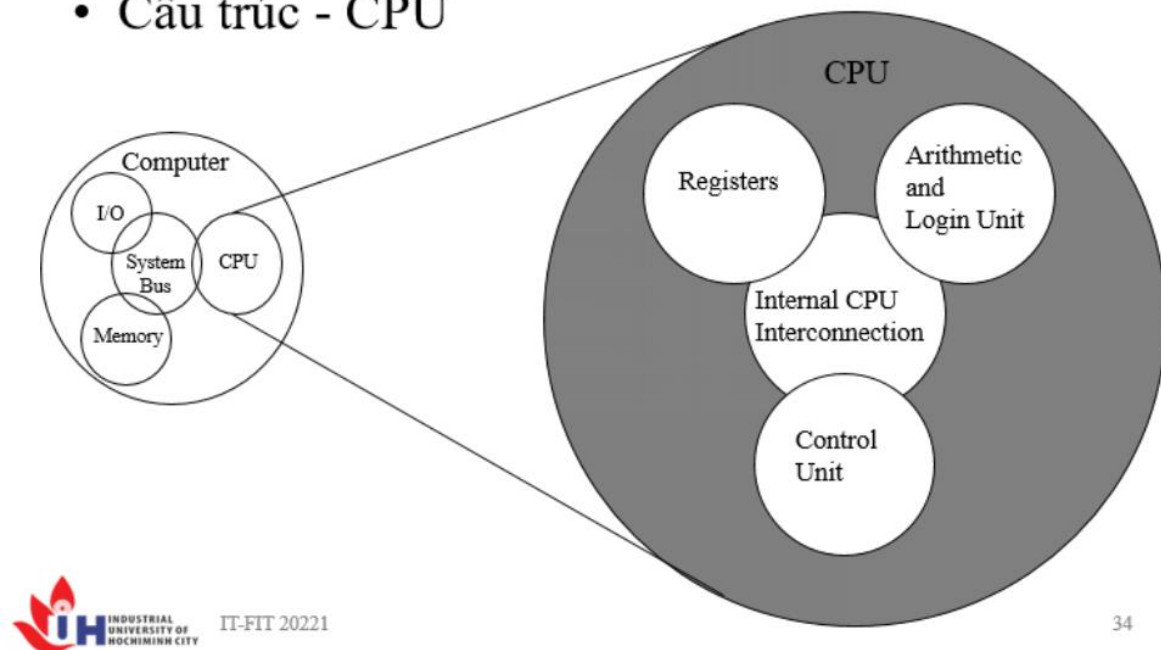
* Cấu trúc – Top Level

• Cấu trúc - Top Level



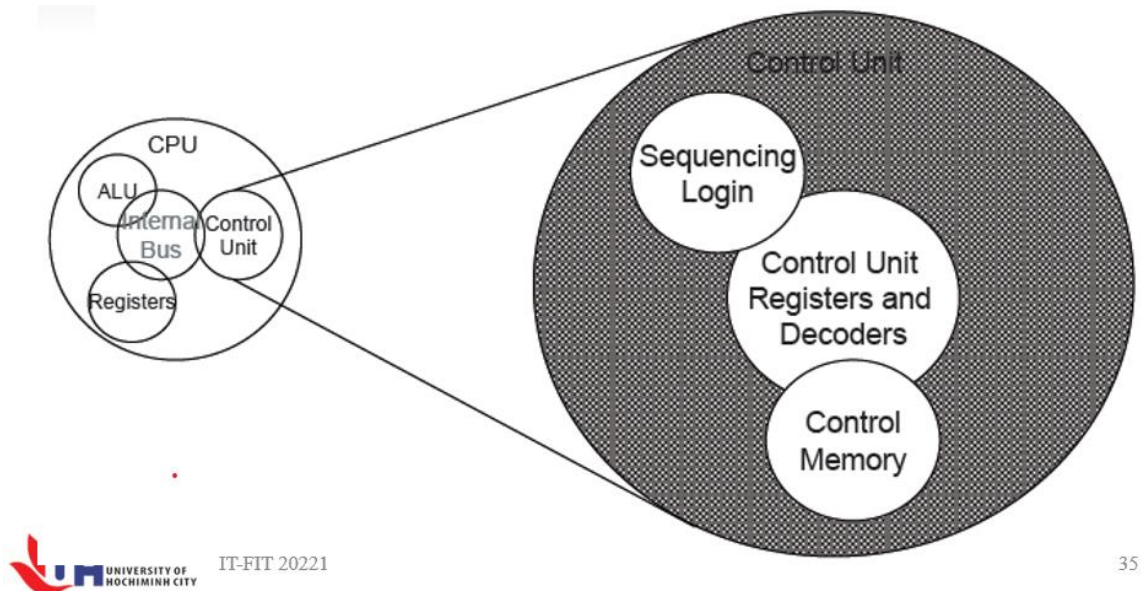
* Cấu trúc – CPU

• Cấu trúc - CPU



* Cấu trúc – Bộ điều khiển

• Cấu trúc – Bộ điều khiển



Phần 4: Phân loại máy tính

* Theo công nghệ

- Máy tính tương tự (analog)
- Máy tính số (digital)

* Theo mức độ sử dụng

- Máy chuyên dụng
- Máy đa dụng

* Theo nguyên lý hoạt động

- Von Neumann
- Phi Von Neumann

* Theo khả năng xử lý

- Máy vi tính (micro computer)
- Máy mini (mini computer)
- Máy tính lớn (main frame)
- Siêu máy tính (super computer)

* Theo thể hệ

- Thế hệ 1
- Thế hệ 2
- Thế hệ 3
- Thế hệ 4 ...

Phần 5: Lịch sử phát triển máy tính

	Year	Name	Made by	Comments
The zeroth generation	1834	Analytical Engine	Babbage	First attempt to build a digital computer
	1936	Z1	Zuse	First working relay calculating machine
	1943	COLOSSUS	British gov't	First electronic computer
	1944	Mark I	Aiken	First American general-purpose computer
The first generation	1946	ENIAC I	Eckert/Mauchley	Modern computer history starts here
	1949	EDSAC	Wilkes	First stored-program computer
	1951	Whirlwind I	M.I.T.	First real-time computer
	1952	IAS	Von Neumann	Most current machines use this design
The second generation	1960	PDP-1	DEC	First minicomputer (50 sold)
	1961	1401	IBM	Enormously popular small business machine
	1962	7094	IBM	Dominated scientific computing in the early 1960s
	1963	B5000	Burroughs	First machine designed for a high-level language
	1964	360	IBM	First product line designed as a family
	1964	6600	CDC	First scientific supercomputer
	1965	PDP-8	DEC	First mass-market minicomputer (50,000 sold)
The third generation	1970	PDP-11	DEC	Dominated minicomputers in the 1970s
	1974	8080	Intel	First general-purpose 8-bit computer on a chip
	1974	CRAY-1	Cray	First vector supercomputer
	1978	VAX	DEC	First 32-bit superminicomputer
The fourth generation	1981	IBM PC	IBM	Started the modern personal computer era
	1985	MIPS	MIPS	First commercial RISC machine
	1987	SPARC	Sun	First SPARC-based RISC workstation
	1990	RS6000	IBM	First superscalar machine

Figure 1-4. Some milestones in the development of the modern digital computer.

*** Thế hệ 1 (1945-1955)**

- Công nghệ chế tạo: Đèn điện tử
- Phần cứng: chưa có. Sau này xuất hiện bằng giấy và phiếu đục lỗ. Chỉ có 1 loại máy mainframe.
- Ngôn ngữ lập trình: chưa có, sử dụng ngôn ngữ máy
- Hệ điều hành: chưa có
- Các máy điển hình: ENIAC, EDVAC, IAS
- Đặc điểm: tính toán chậm, kích thước lớn, tiêu thụ nhiều năng lượng. Chế tạo đơn lẻ.

*** Thế hệ 2 (1955-1965)**

- Công nghệ chế tạo: Transistor
- Phần cứng: Băng từ, máy in. Xuất hiện máy mini
- Ngôn ngữ lập trình: Assembly, Fortran, Cobol
- Hệ điều hành: Đơn giản (Control program, Monitor). Xử lý theo lô, xử lý offline.
- Các máy điển hình: PDP-1, IBM 7094, CDC 6600
- Đặc điểm: Tốc độ nhanh hơn, kích thước thu hẹp nhưng vẫn còn lớn, tiêu thụ ít năng lượng hơn. Sản xuất hàng loạt theo đơn đặt hàng.

*** Thế hệ 3 (1965-1980)**

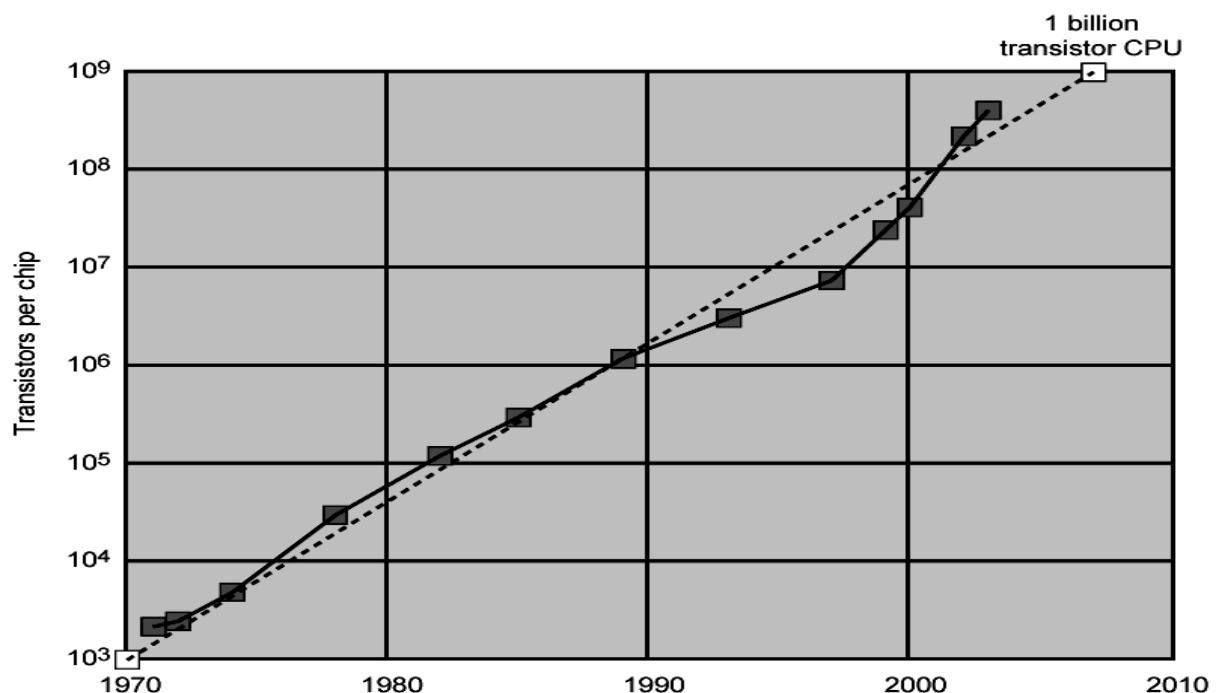
- Công nghệ chế tạo: Mạch tích hợp IC
- Phần cứng: Màn hình, đĩa cứng. Xuất hiện siêu máy tính. Xuất hiện họ các máy tính.
- Ngôn ngữ lập trình: Phát triển mạnh: Algol, Pascal, C
- Hệ điều hành: Xử lý đa chương, chia sẻ thời gian, xuất hiện UNIX
- Các máy điển hình: IBM/360, DEC PDP-11, Cray-1
- Đặc điểm: Tốc độ nhanh, kích thước nhỏ, tiêu thụ ít năng lượng, ngày càng dễ sử dụng

* Thế hệ 4 (1980 - nay)

- Công nghệ chế tạo: Mạch tích hợp IC cỡ lớn (VLSI). Độ tích hợp ngày càng cao theo định luật Moore
- Phần cứng: Phát triển đa dạng. Xuất hiện máy vi tính.
- Ngôn ngữ lập trình: Phát triển mạnh hơn và dễ lập trình hơn: C++, Java, Prolog, Lisp
- Hệ điều hành: Đồ họa, mạng, đa xử lý, xử lý phân tán
- Các máy điển hình: IBM PC, Mac
- Đặc điểm: Tốc độ rất cao, kích thước rất nhỏ, thân thiện với con người, hỗ trợ multimedia,...

* Định luật Moore

Do Gordon Moore, người đồng sáng lập công ty Intel đưa ra từ 1965: “Số lượng transistor trong 1 chip tăng gấp đôi sau 18 tháng”

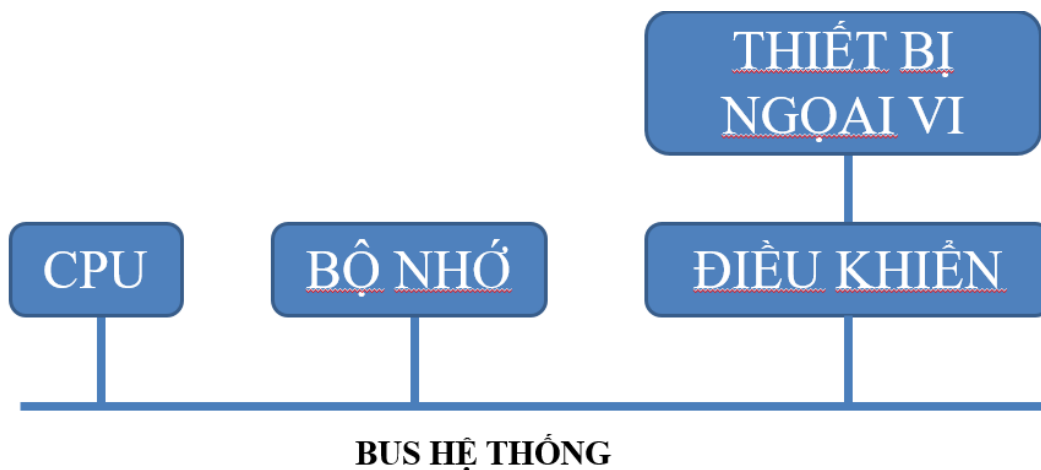


*** Tại sao chưa có thế hệ 5 ?**

- Dự án máy tính thế hệ 5 của Nhật bản
- Xuất hiện máy vi tính
- Xuất hiện mạng Internet toàn cầu
- Xuất hiện các siêu máy tính cực mạnh
- Máy tính đã xâm nhập vào mọi lĩnh vực trong đời sống con người

Phần 5: Lịch sử phát triển máy tính

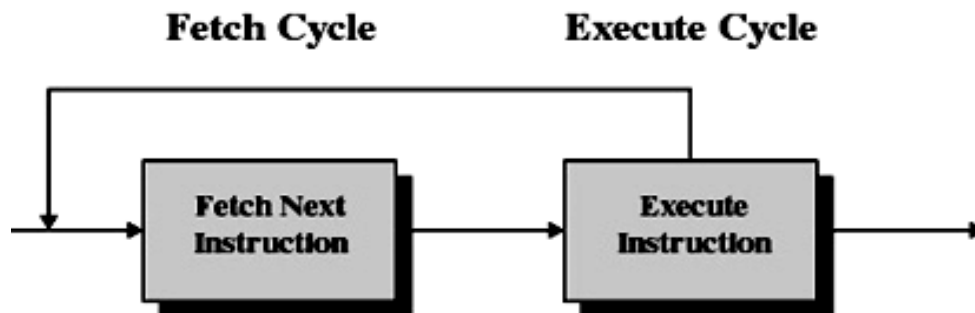
*** Sơ đồ tổ chức tổng quát:**



*** CPU**

- Chức năng: Điều khiển mọi hoạt động bên trong MT và thực hiện các phép tính
- Thành phần:
 - + CU (Control Unit)
 - + ALU (Arithmetic & Logic Unit)
 - + Các thanh ghi (Registers)
- Khả năng xử lý (bit): 8, 16, 32, 64 bit ...
- Tốc độ xử lý (tính toán): ???
- Nhiệm vụ: thi hành chương trình

- Chu trình lệnh: Lấy lệnh, thi hành lệnh

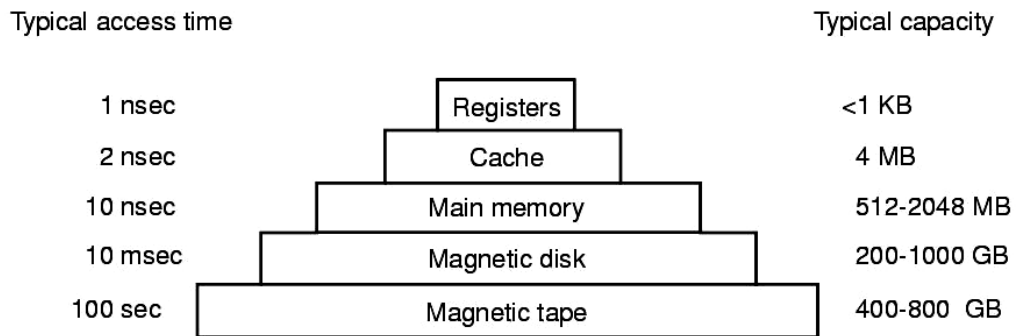


* Bộ nhớ (memory)

- Chức năng: Lưu trữ dữ liệu và chương trình trong máy tính
- Tổ chức : Bộ nhớ được chia ra các ô có kích thước bằng nhau. Mỗi ô có thể lưu trữ 1 byte hoặc 1 từ máy (word). 1 word có thể chứa 2, 4, 8, 16,... byte tùy theo nhà sản xuất máy tính.
- Cần địa chỉ (address) để gán cho các ô nhớ. Mục đích để phân biệt các ô nhớ với nhau khi truy cập dữ liệu
- Phân loại bộ nhớ:
 - + RAM (Random Access Memory)
 - + ROM (Read Only Memory)
 - + Cache
- Phân cấp bộ nhớ

Computer	Bits/cell
Burroughs B1700	1
IBM PC	8
DEC PDP-8	12
IBM 1130	16
DEC PDP-15	18
XDS 940	24
Electrologica X8	27
XDS Sigma 9	32
Honeywell 6180	36
CDC 3600	48
CDC Cyber	60

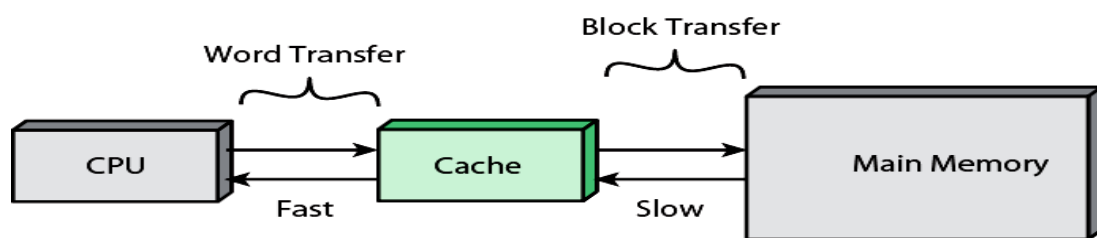
- Khác biệt: Dung lượng, tốc độ truy cập, giá thành



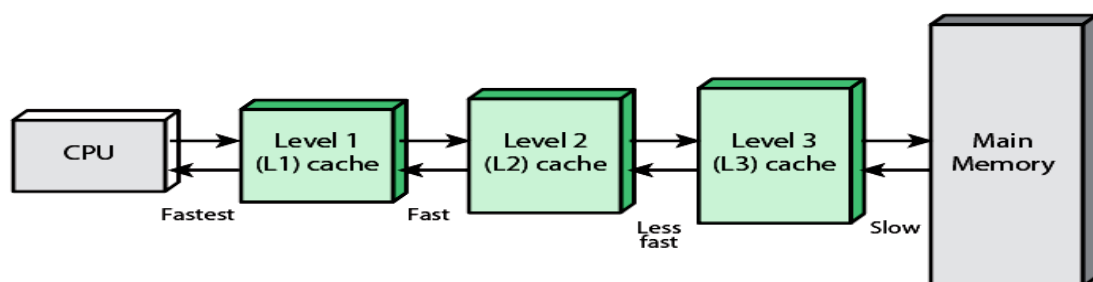
- Đặc điểm các loại bộ nhớ:

Level	1	2	3	4
Name	registers	cache	main memory	disk storage
Typical size	< 1 KB	> 16 MB	> 16 GB	> 100 GB
Implementation technology	custom memory with multiple ports, CMOS	on-chip or off-chip CMOS SRAM	CMOS DRAM	magnetic disk
Access time (ns)	0.25 – 0.5	0.5 – 25	80 – 250	5,000.000
Bandwidth (MB/sec)	20,000 – 100,000	5000 – 10,000	1000 – 5000	20 – 150
Managed by	compiler	hardware	operating system	operating system
Backed by	cache	main memory	disk	CD or tape

- Các mức cache:



(a) Single cache



(b) Three-level cache organization

* Thiết bị ngoại vi (peripherals)

- Chức năng: giao tiếp giữa máy tính với thế giới bên ngoài (con người)

- Nhiệm vụ: chuyển đổi dạng dữ liệu giữa con người và máy tính

- Phân loại:

+ Thiết bị nhập (input devices)

+ Thiết bị xuất (output devices)

+ Thiết bị truyền thông (communication devices)

+ Thiết bị lưu trữ (storage devices)

- Các loại thiết bị lưu trữ

+ Giấy: Băng giấy đục lỗ, Phiếu đục lỗ, ...

+ Từ tính

▪ Xuyên từ, Trống từ, Băng từ

▪ Đĩa từ (Đĩa mềm, Đĩa cứng)

+ Quang học

▪ CD/ DVD

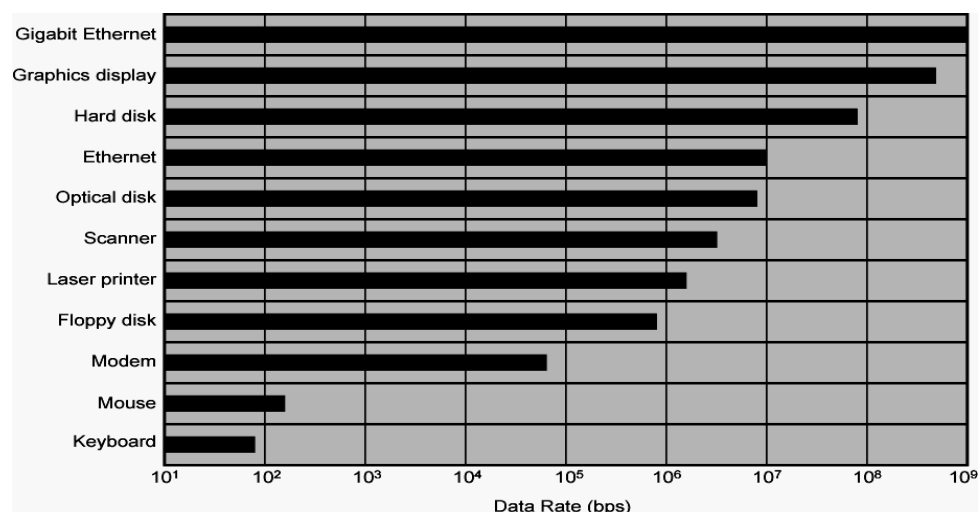
▪ Blue-ray, HD-DVD

+ Quang từ: MO disk

+ Bán dẫn: USB Flash, SSD, thẻ nhớ, ...

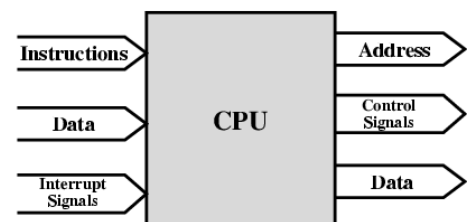
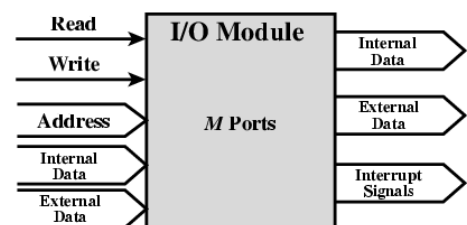
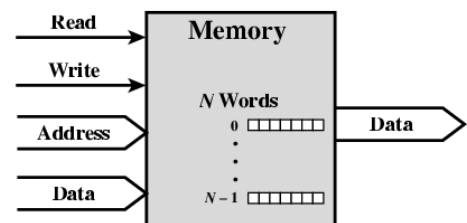
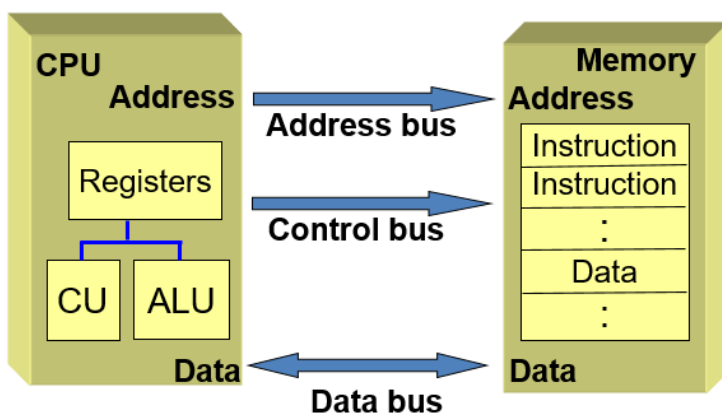
+ Khác: Bubble, Hologram, ...

- Tốc độ truy cập 1 số thiết bị ngoại vi thông dụng:

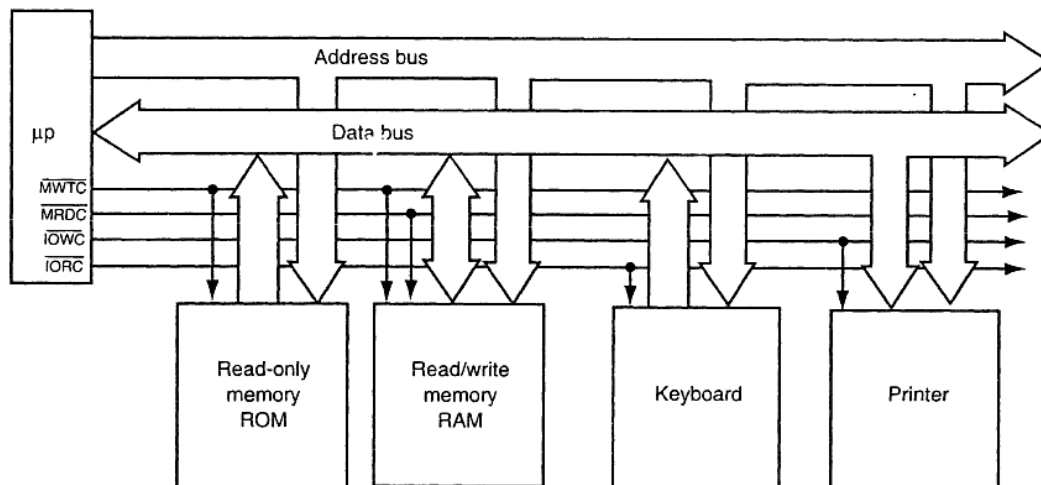


* Bus hệ thống

- Mục đích: Tổ chức dạng bus được dùng để đơn giản hóa việc tổ chức và phân luồng dữ liệu trong hệ thống máy tính
- Chức năng: Liên kết & truyền tín hiệu giữa các thành phần trong MT
- Cần cơ chế sao cho tại một thời điểm, chỉ có 1 thành phần có thể đặt dữ liệu lên bus để tránh tranh chấp bus
- Phân loại:
 - + Data bus
 - + Address bus: Không gian địa chỉ
 - + Control bus
- Mô hình hệ thống máy tính 3 bus:



- Ví dụ về Control Bus



- Đồng bộ bus: Các thành phần trong máy tính phải hoạt động đồng bộ.

- + Mỗi hoạt động cơ bản được chia ra nhiều bước nhỏ
- + Cần 1 trọng tài đánh nhịp để điều khiển từng bước hoạt động
- + Ví dụ: thao tác đọc bộ nhớ được chia ra:

- CPU gửi yêu cầu đọc cho BN (bus điều khiển)
- CPU gửi địa chỉ cần đọc cho BN (bus địa chỉ)
- BN giải mã địa chỉ
- BN xuất dữ liệu cho CPU (bus dữ liệu)

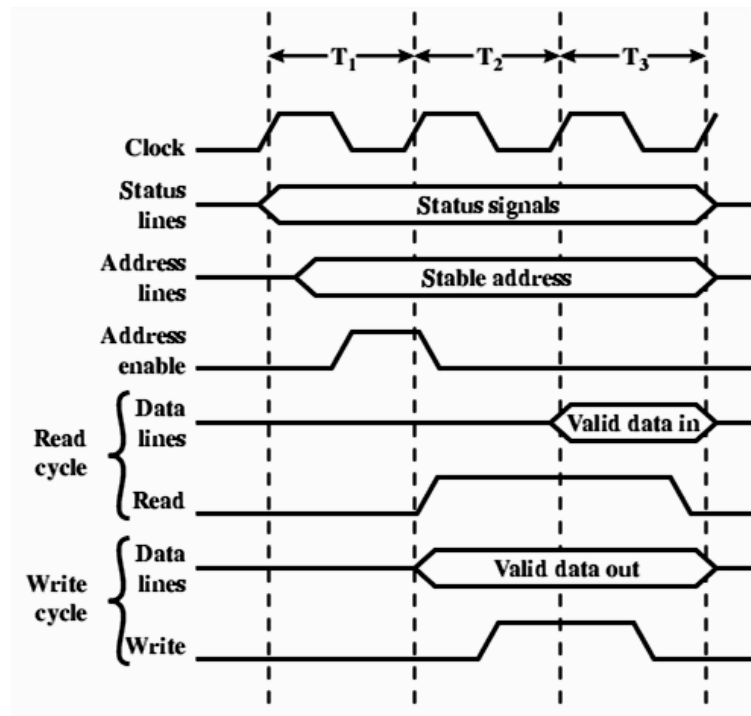
- + Một chu kỳ lệnh gồm nhiều chu kỳ máy (4 như VD trên)

- + Đơn vị đo tốc độ xung nhịp: Hertz (Hz)

- 2 chế độ tốc độ hoạt động trong CPU máy tính

- Sự tồn tại BN cache

- Ví dụ minh họa về sự đồng bộ bus



- Trong thực tế người ta phân bus hệ thống ra nhiều mức tốc độ do có sự chênh lệch của các thành phần trong máy tính

