



**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**



**BÀI GIẢNG MÔN**

# **KỸ THUẬT VI XỬ LÝ**

**Giảng viên:**

**TS. Nguyễn Trung Hiếu**

**Điện thoại/E-mail:**

**0916566268 / [hieunt@ptit.edu.vn](mailto:hieunt@ptit.edu.vn)**

**Học kỳ/Năm biên soạn: Kỳ 1/2016**

# KỸ THUẬT VI XỬ LÝ

---

- Tên môn học: Kỹ thuật vi xử lý

- Số tín chỉ: 3

(45 tiết: 36 tiết LT, 8 tiết BT, 1 tiết Tự học)

- Kiến thức cơ sở:

Tin học cơ sở 2, Điện tử số.



# KỸ THUẬT VI XỬ LÝ

---

## ĐÁNH GIÁ MÔN HỌC

- Điểm chuyên cần: 10%
- Bài tập lớn: 20%
- Kiểm tra giữa kì: 10%
- Kiểm tra cuối kì: 60%



# KỸ THUẬT VI XỬ LÝ

---

## NỘI DUNG

- Chương 1 – Tổng quan về hệ thống vi xử lý
- Chương 2 - Cấu trúc của bộ vi xử lý
- Chương 3 - Kết nối phần cứng hệ thống vi xử lý
- Chương 4 – Vi điều khiển (ARM)
- Chương 5 – Lập trình hợp ngữ cho vi xử lý ARM



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## NỘI DUNG

1. Các hệ đếm và mã hóa thông tin trong máy tính
2. Giới thiệu về vi xử lý
3. Hệ vi xử lý
  - ✓ Cấu trúc hệ vi xử lý
  - ✓ Kiến trúc von-Neumann và Havard
4. Các đặc điểm cấu trúc của vi xử lý
5. Lịch sử phát triển của vi xử lý



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## CÁC HỆ ĐẾM VÀ HỆ THỐNG MÃ HÓA

### 1.1 Các hệ đếm

- Hệ thập phân
- Hệ nhị phân
- Hệ thập lục phân

### 1.2 Các hệ thống mã hoá

- ASCII
- BCD

### 1.3 Giới thiệu một số linh kiện điện tử số cơ bản

- Các cổng logic: AND, OR, XOR, NOT
- Các bộ giải mã



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## HỆ ĐẾM THẬP PHÂN

- Hệ đếm thập phân (Decimal) hay còn gọi là hệ đếm cơ số mười

- Dùng mười ký hiệu:  
1,2,3,4,5,6,7,8,9,0

- Ví dụ:1.1:

$$\begin{aligned} 3978 &= 3 \times 10^3 + 9 \times 10^2 + 7 \times 10^1 + 8 \times 10^0 \\ &= 3000 + 900 + 70 + 8 \end{aligned}$$

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## HỆ ĐẾM NHỊ PHÂN

- Hệ đếm nhị phân (Binary) hay còn gọi là Hệ đếm cơ số hai
- Sử dụng hai ký hiệu (bit): 0 và 1 . Mỗi ký hiệu 0 hoặc 1 được gọi là 1 Bit (Binary Digit- Chữ số nhị phân)
- Kích cỡ của một số nhị phân là số bit của nó
- MSB (Most Significant Bit): Bit sát trái
- LSB (Least Significant Bit): Bit sát phải

Ví dụ 1.2: 1010101010101010

↑  
MSB

↑  
LSB



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## SỐ NHỊ PHÂN KHÔNG DẤU

- Chỉ biểu diễn được các giá trị không âm ( $\geq 0$ )
- Với  $n$ -bit có thể biểu diễn các giá trị từ 0 đến  $2^n - 1$
- Ví dụ 1.3: Giá trị  $V$  của số nhị phân không dấu 1101 được tính:

$$V(1101) = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

$$= 8 + 4 + 0 + 1 = 13$$



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## CHUYỂN TỪ SỐ THẬP PHÂN SANG NHỊ PHÂN

- Ví dụ 1.4

Chuyển 25 sang nhị phân không dấu. Dùng phương pháp  
**chia 2 liên tiếp**

Kết quả là: 11001

# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## SỐ NHỊ PHÂN CÓ DẤU

- Biểu diễn được cả các giá trị âm
- Còn gọi là Số bù hai
- Với n-bit có thể biểu diễn các giá trị từ  $-2^{(n-1)}$  đến  $2^{(n-1)} - 1$
- Ví dụ 1.5: Giá trị V của số nhị phân có dấu 1101 được tính:

$$\begin{aligned} V(1101) &= -1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= -8 + 4 + 0 + 1 = -3 \end{aligned}$$

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## CÁCH TÍNH ĐỔI SỐ

- Tổng của một số với đối số của nó bằng 0
- Ví dụ 1.6

Đối số của số nhị phân có dấu 10011101?

|          |                 |
|----------|-----------------|
| 10011101 | Số có dấu (-99) |
| 01100010 | Lấy bù 1        |
| + 1      | Cộng 1          |
| -----    |                 |
| 01100011 | Kết quả (+99)   |



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## ĐỔI SỐ THẬP PHÂN SANG NHỊ PHÂN CÓ DẤU

- Với số dương: Giống như chuyển thập phân sang nhị phân không dấu rồi thêm bit 0 vào **sát bên trái**
- Ví dụ: Chuyển 25 sang nhị phân có dấu:  
Kết quả: **0**11011
- Với số âm: Chuyển đối số sang nhị phân có dấu rồi lấy bù 2

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## ĐỔI SỐ THẬP PHÂN SANG NHỊ PHÂN CÓ DẤU

Ví dụ 1.7 : Chuyển  $-26$  sang nhị phân

1. Chuyển đổi số:  $+26 = 11010$

2. Đưa 0 vào sát trái:  $011010$

3. Bù 1:  $100101$

4. Cộng 1:  $+ 1$

-----  
 $-26 = 100110$



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## ĐỔI SỐ THẬP PHÂN SANG NHỊ PHÂN CÓ DẤU

Ví dụ : Thực hiện phép trừ các số nhị phân sau:

$$1011.0100 - 1001.0111 = ?$$

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## SỐ THẬP LỤC PHÂN

- Quen gọi là số Hexa (Hexadecimal) hay còn gọi là hệ đếm cơ số mười sáu
- Sử dụng 16 ký hiệu để biểu diễn:  
**0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F**
- Mỗi ký hiệu tương ứng với **4-bit**
- Mục đích: **Biểu diễn số nhị phân ở dạng ngắn gọn**

1111.0000 = F0

1010.1010 = AA

0101.0101 = 55

Nhị phân Thập lục phân



# CHƯƠNG 1-GIỚI THIỆU CHUNG

## CHUYỂN ĐỔI THẬP LỤC PHẦN VÀ NHỊ PHẦN

### ■ Ví dụ 1.8

Chuyển số hexa 2F8 và ABBA sang nhị phân

Thay thế mỗi ký hiệu hexa bằng 4-bit tương ứng với nó

|      |      |      |      |
|------|------|------|------|
|      | 2    | F    | 8    |
|      | 0010 | 1111 | 1000 |
| A    | B    | B    | A    |
| 1010 | 1011 | 1011 | 1010 |

■ Kết quả      2F8h                      =      001011111000b  
                    ABBAh                    =      1010101110111010b



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## CHUYỂN ĐỔI LỤC PHÂN VÀ NHỊ PHÂN

- Ví dụ : Hãy biểu diễn số -97 dưới dạng số hex với
  - 8 bit?
  - 16 bit?

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## CHUYỂN ĐỔI LỤC PHÂN VÀ NHỊ PHÂN

Giải:

Dưới dạng 8 bit:

$$\begin{array}{rcll} 97 & = & 0110.0001 & \\ \text{Bù 1 của } 97 & = & 1001.1110 & \\ & & + 1 & \\ -97 & = & 1001.1111 & = 9Fh \end{array}$$

Dưới dạng 16 bit:

$$\begin{array}{rcll} 97 & = & 0000.0000.0110.0001 & \\ \text{Bù 1 của } 97 & = & 1111.1111.1001.1110 & \\ & & + 1 & \\ -97 & = & 1111.1111.1001.1111 & = FF9Fh \end{array}$$

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## CÁC HỆ THỐNG MÃ HÓA-MÃ ASCII

- ASCII: **A**merican **S**tandard **C**ode for **I**nformation **I**nterchange.
- Nó thường được dùng để hiển thị văn bản trong máy tính và các thiết bị thông tin khác : Gồm **ký tự hiển thị được** và **ký tự điều khiển**
- Bảng mã ASCII gồm 128 mã được biểu diễn bởi 7 bit. Bit thứ 8 có thể là bit parity.
  - Các chữ cái in và thường: **A..Z** và **a..z**
  - Các chữ số thập phân: **0,1,...,9**
  - Các dấu chấm câu: **;, , . :** *vân vân*
  - Các ký tự đặc biệt: **\$ & @ / {** *vân vân*
  - Các ký tự điều khiển: **carriage return (CR) , line feed (LF), beep,** *vân vân*

# CHƯƠNG 1-GIỚI THIỆU CHUNG

| Decimal Hex ASCII |    |     | Decimal Hex ASCII |    |    | Decimal Hex ASCII |    |   | Decimal Hex ASCII |    |   |
|-------------------|----|-----|-------------------|----|----|-------------------|----|---|-------------------|----|---|
| 0                 | 0  | NUL | 32                | 20 |    | 64                | 40 | @ | 96                | 60 | ` |
| 1                 | 1  | SOH | 33                | 21 | !  | 65                | 41 | A | 97                | 61 | a |
| 2                 | 2  | STX | 34                | 22 | "  | 66                | 42 | B | 98                | 62 | b |
| 3                 | 3  | ETX | 35                | 23 | #  | 67                | 43 | C | 99                | 63 | c |
| 4                 | 4  | EOT | 36                | 24 | \$ | 68                | 44 | D | 100               | 64 | d |
| 5                 | 5  | ENQ | 37                | 25 | %  | 69                | 45 | E | 101               | 65 | e |
| 6                 | 6  | ACK | 38                | 26 | &  | 70                | 46 | F | 102               | 66 | f |
| 7                 | 7  | BEL | 39                | 27 | '  | 71                | 47 | G | 103               | 67 | g |
| 8                 | 8  | BS  | 40                | 28 | (  | 72                | 48 | H | 104               | 68 | h |
| 9                 | 9  | HT  | 41                | 29 | )  | 73                | 49 | I | 105               | 69 | i |
| 10                | A  | LF  | 42                | 2A | *  | 74                | 4A | J | 106               | 6A | j |
| 11                | B  | VT  | 43                | 2B | +  | 75                | 4B | K | 107               | 6B | k |
| 12                | C  | FF  | 44                | 2C | ,  | 76                | 4C | L | 108               | 6C | l |
| 13                | D  | CR  | 45                | 2D | -  | 77                | 4D | M | 109               | 6D | m |
| 14                | E  | SOH | 46                | 2E | .  | 78                | 4E | N | 110               | 6E | n |
| 15                | F  | SI  | 47                | 2F | /  | 79                | 4F | O | 111               | 6F | o |
| 16                | 10 | DLE | 48                | 30 | 0  | 80                | 50 | P | 112               | 70 | p |
| 17                | 11 | DC1 | 49                | 31 | 1  | 81                | 51 | Q | 113               | 71 | q |
| 18                | 12 | DC2 | 50                | 32 | 2  | 82                | 52 | R | 114               | 72 | r |
| 19                | 13 | DC3 | 51                | 33 | 3  | 83                | 53 | S | 115               | 73 | s |
| 20                | 14 | DC4 | 52                | 34 | 4  | 84                | 54 | T | 116               | 74 | t |
| 21                | 15 | NAK | 53                | 35 | 5  | 85                | 55 | U | 117               | 75 | u |
| 22                | 16 | SYN | 54                | 36 | 6  | 86                | 56 | V | 118               | 76 | v |
| 23                | 17 | ETB | 55                | 37 | 7  | 87                | 57 | W | 119               | 77 | w |
| 24                | 18 | CAN | 56                | 38 | 8  | 88                | 58 | X | 120               | 78 | x |
| 25                | 19 | EM  | 57                | 39 | 9  | 89                | 59 | Y | 121               | 79 | y |
| 26                | 1A | SUB | 58                | 3A | :  | 90                | 5A | Z | 122               | 7A | z |
| 27                | 1B | ESC | 59                | 3B | ;  | 91                | 5B | [ | 123               | 7B | { |
| 28                | 1C | FS  | 60                | 3C | <  | 92                | 5C | \ | 124               | 7C |   |
| 29                | 1D | GS  | 61                | 3D | =  | 93                | 5D | ] | 125               | 7D | } |
| 30                | 1E | RS  | 62                | 3E | >  | 94                | 5E | ^ | 126               | 7E | ~ |
| 31                | 1F | US  | 63                | 3F | ?  | 95                | 5F | _ | 127               | 7F | □ |

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## BIT, NIBBLE, BYTE, WORD

- Bit: Một chữ số nhị phân 0 hoặc 1
- Nibble: 4-bit (nửa byte)
- Byte: 8-bit (Còn gọi là Octet)
- Word (Từ): 16-bit
- Double Word (Từ kép): 32-bit
- $K = 2^{10} = 1024$   
Kb (kilôbit) = 1024 bit = 128 byte  
KB (kilôbyte) = 1024 byte  
Kbps (Kilobit per second): Kilôbit trên giây
- $M = 2^{20} = 1024 K = 1048576$   
Mb (Mêgabit) = 1024 Kb = 1048576 bit  
MB (Mêgabyte) = 1024 KB = 1048576 byte
- $G = 2^{30} = 1024 M = 1048576 K$   
Gb (Gigabit) = 1024 Mb = 1048576 Kb  
GB (Gigabyte) = 1024 MB = 1048576 KB
- $T = 2^{40} = 1024 G = 1048576 M$   
Tb (Terabit) = 1024 Gb = 1048576 Mb  
TB (Terabyte) = 1024 GB = 1048576 MB



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## GIỚI THIỆU VỀ VI XỬ LÝ

- Vi xử lý (microprocessor):
  - Là vi mạch có mức tích hợp rất lớn (Very Large Scale Integrated circuit) có thể lập trình được.
  - Dưới sự điều khiển của chương trình trong bộ nhớ, vxl thực hiện các phép tính số học, logic, dịch, quay, trao đổi dữ liệu với các thiết bị bên ngoài,...
- Chức năng của vi xử lý: Xử lý dữ liệu
  - Tính toán
  - Vận chuyển dữ liệu



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## GIỚI THIỆU VỀ VI XỬ LÝ

- Phân loại vi xử lý:
  - Vi xử lý đa năng (general purpose microprocessor):
    - Chứa bên trong chip tất cả các thành phần chủ yếu cho tính toán và điều khiển
    - Không bao gồm bộ nhớ và cổng vào/ra
  - Vi điều khiển (microcontrollers):
    - Chứa bên trong chip tất cả các thành phần chủ yếu cho tính toán và điều khiển
    - Bao gồm bộ nhớ và cổng vào/ra
    - Tất cả các thành phần của vi điều khiển được tích hợp trên một chip đơn





# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

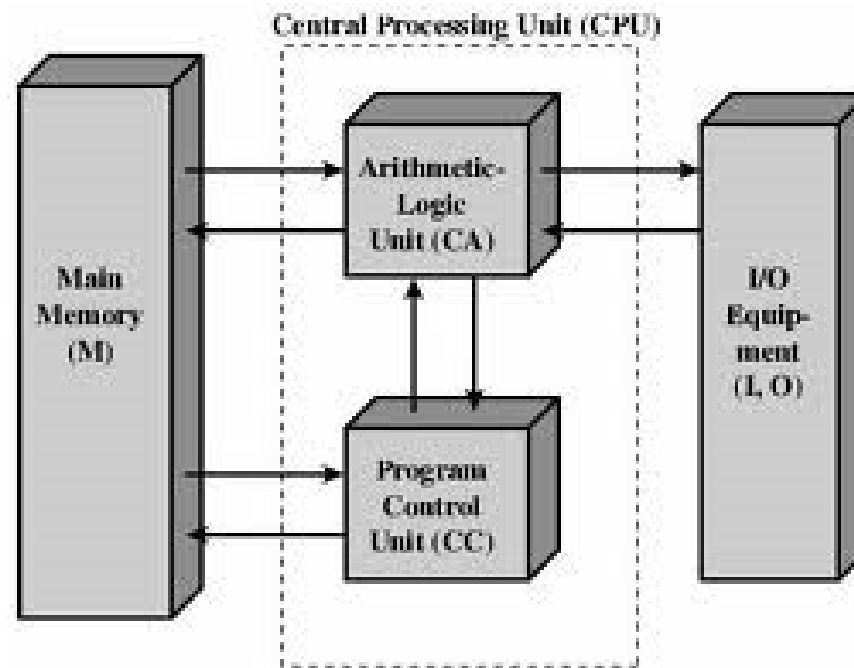
## GIỚI THIỆU VỀ VI XỬ LÝ

- Thực hiện lệnh của vi xử lý:
  - Vi xử lý thực hiện các tác vụ dựa trên các chỉ dẫn (instructions), hay còn gọi là các lệnh.
  - Một tập hợp các lệnh được sắp xếp theo một trật tự nào đó được gọi là chương trình.
  - Chương trình thường được lưu trên ổ đĩa. Chương trình được nạp vào bộ nhớ khi được kích hoạt. Vi xử lý sẽ đọc và thực hiện từng lệnh của chương trình trong bộ nhớ.
- Quá trình thực hiện một lệnh:
  - Đọc lệnh (Instruction Fetch)
  - Giải mã lệnh (Instruction Decode)
  - Thực hiện lệnh (Instruction Execution)
  - Lưu kết quả của lệnh (nếu có) (Memory Access)

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## MÁY VI TÍNH VÀ HỆ THỐNG VI XỬ LÝ

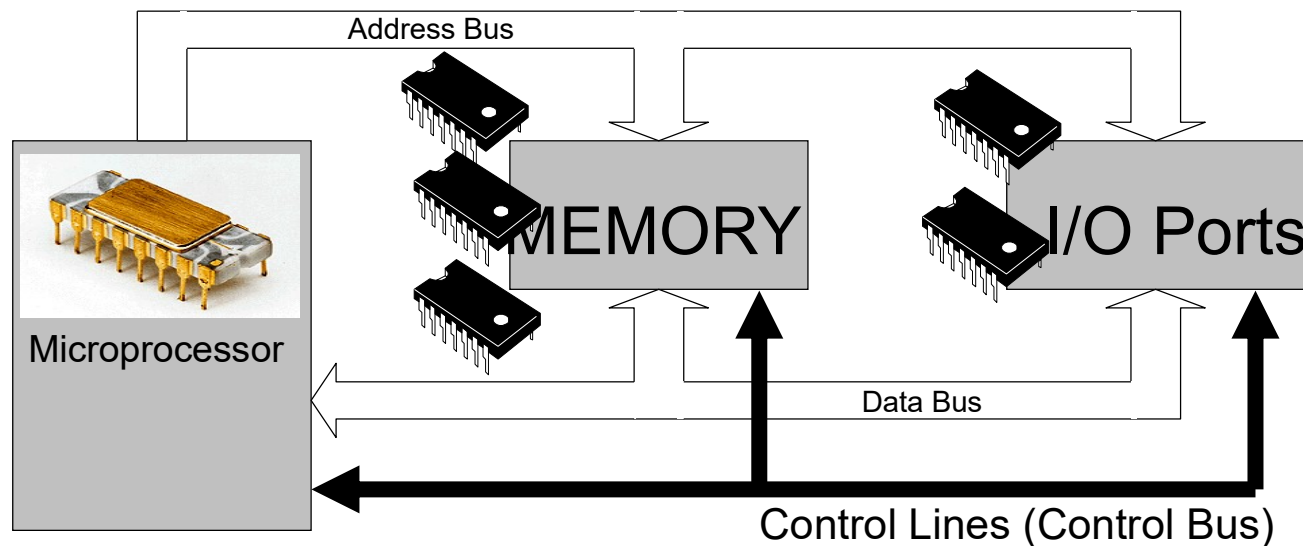
- Máy vi tính (microcomputer)
  - Là hệ thống gồm 5 khối phần cứng: vào, ra, bộ nhớ, đơn vị số học-logic (ALU) và điều khiển (Control Unit).
  - Vi xử lý là đơn vị xử lý trung tâm của máy vi tính gồm hai khối phần cứng là ALU và CU



# CHƯƠNG 1-GIỚI THIỆU CHUNG

## MÁY VI TÍNH VÀ HỆ THỐNG VI XỬ LÝ

- Hệ vi xử lý
  - Là một hệ thống bao gồm bộ vi xử lý kết hợp với các bộ phận điện tử khác như bộ nhớ và các bộ phối ghép vào/ra.
  - Lưu ý: “hệ vi xử lý” mang ý nghĩa tổng quát hơn so với thuật ngữ “máy vi tính”, vì máy vi tính chỉ là một trong những ứng dụng cụ thể của hệ vi xử lý.





# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## MÁY VI TÍNH VÀ HỆ THỐNG VI XỬ LÝ

- Hệ vi xử lý bao gồm
  - CPU (Central Processing Unit): Bộ xử lý trung tâm có nhiệm vụ tính toán và điều khiển
    - ALU (Arithmetic and Logic Unit): Khối tính toán
    - CU (Control Unit): Khối điều khiển
  - Bộ nhớ (Memory): Lưu trữ dữ liệu (data) và lệnh (instruction) cho CPU xử lý
    - Bộ nhớ ROM: lưu lệnh và dữ liệu của hệ thống
    - Bộ nhớ RAM: lưu lệnh và dữ liệu của hệ thống và của người dùng
  - Thiết bị vào (Input): Tiếp nhận dữ liệu và thông tin điều khiển, chuyển cho CPU xử lý
  - Thiết bị ra (Output): Kết xuất thông tin ra màn hình, máy in,...hoặc lưu trữ thông tin trên đĩa từ, đĩa quang,...



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## HỆ VI XỬ LÝ: KIẾN TRÚC Von-Neumann

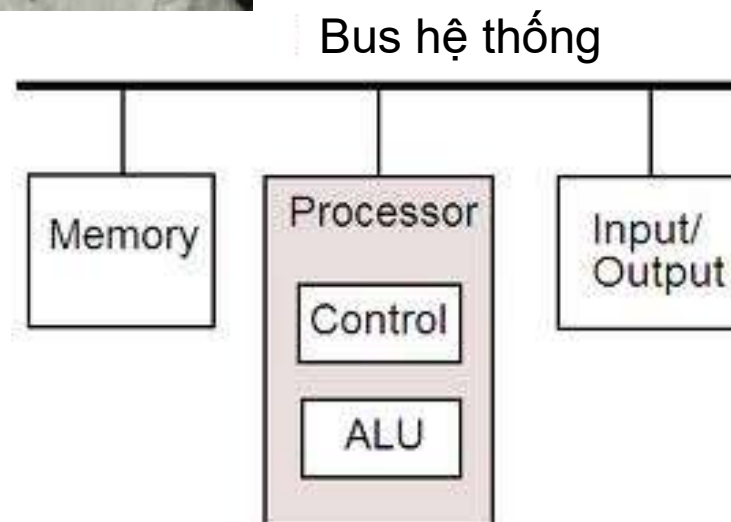
- Các máy tính điện tử số thường có hai loại kiến trúc: kiến trúc Von-Neumann và kiến trúc Havard
- Kiến trúc Von-Neumann được nhà toán học Jonh Von-Neumann đưa ra vào năm 1945.
- Các loại máy tính Von-Neumann dựa trên 3 khái niệm cơ bản:
  - Dữ liệu (data) và các lệnh lưu trong 1 bộ nhớ đọc/ghi.
  - Các nội dung của bộ nhớ đọc/ghi được đánh địa chỉ theo vùng, không phụ thuộc vào loại dữ liệu mà bộ nhớ lưu giữ.
  - Quá trình thực hiện các lệnh xảy ra theo tuần tự

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## HỆ VI XỬ LÝ: KIẾN TRÚC Von-Neumann



Hungarian-born John von Neumann (1903-1957)



Kiến trúc máy tính von-Neumann  
hiện đại



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## HỆ VI XỬ LÝ: KIẾN TRÚC Von-Neumann

- Chu trình thực hiện lệnh (đọc – giải mã – thực hiện):
  - Đọc lệnh từ bộ nhớ chính vào trong thanh ghi lệnh của CPU
  - Thay đổi bộ đếm chương trình để trở tới lệnh tiếp theo
  - Xác định kiểu lệnh vừa đọc vào từ bộ nhớ
  - Nếu lệnh có yêu cầu dữ liệu, xác định vị trí dữ liệu
  - Đọc dữ liệu (nếu có) vào một thanh ghi bên trong của CPU
  - Thực hiện lệnh
  - cất giữ kết quả thực hiện lệnh
- Bus hệ thống bao gồm:
  - Bus địa chỉ (Address bus)
  - Bus dữ liệu (Data bus)
  - Bus điều khiển (Control bus)



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

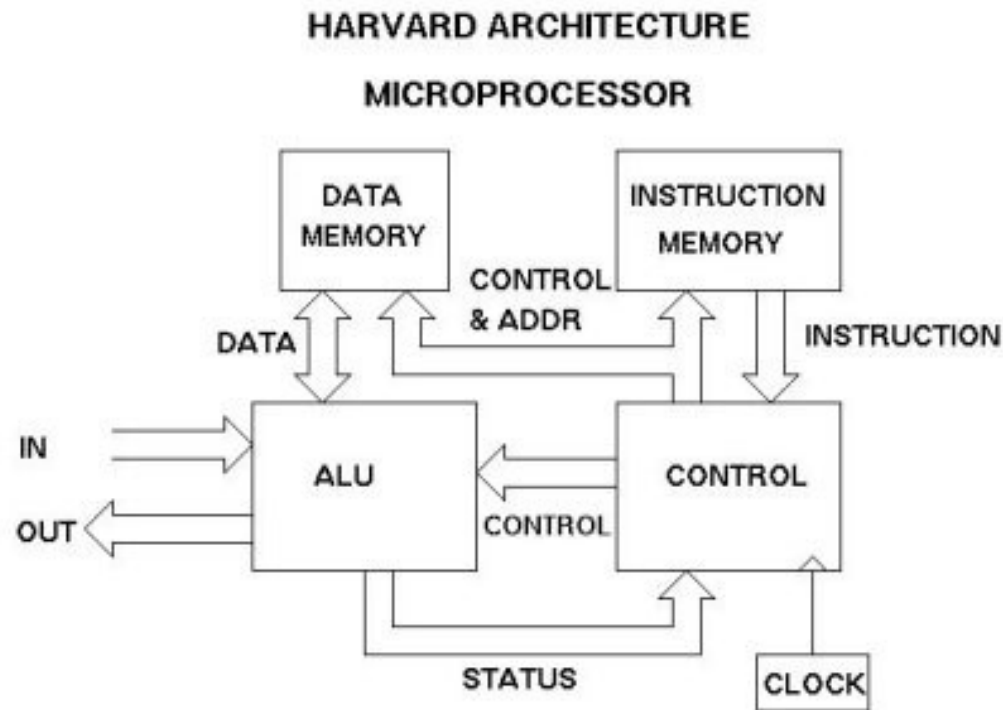
## HỆ VI XỬ LÝ: KIẾN TRÚC Harvard

- Kiến trúc Harvard có sự phân chia bộ nhớ chính thành 2 phần:
  - Bộ nhớ chỉ chứa chương trình (program memory)
  - Bộ nhớ chỉ chứa dữ liệu
- CPU sử dụng 2 hệ thống bus:
  - Bus giao tiếp với bộ nhớ chương trình
  - Bus giao tiếp với bộ nhớ dữ liệu
- So sánh 2 kiến trúc:
  - Kiến trúc Harvard phức tạp hơn kiến trúc von-Neumann
  - Trong kiến trúc Harvard, kết nối giữa CPU với 2 bộ nhớ được thực hiện qua bus riêng biệt nên tốc độ xử lý dữ liệu nhanh hơn, hiệu suất cao hơn



# CHƯƠNG 1-GIỚI THIỆU CHUNG

## HỆ VI XỬ LÝ: KIẾN TRÚC Harvard



- Do tính dễ thiết kế nên kiến trúc Harvard được áp dụng rộng rãi cho thiết kế các máy vi tính cá nhân.



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## CÁC ĐẶC ĐIỂM CẤU TRÚC CỦA BỘ VI XỬ LÝ

- Công suất vi xử lý:
  - Độ dài từ dữ liệu (data word length):
    - Phụ thuộc vào thế hệ vi xử lý: 4, 8, 16, 32, 64 bits
    - Độ rộng của các thanh ghi, bus trong, bus ngoài thường bằng độ dài từ dữ liệu
    - Độ dài từ xử lý lớn sẽ tăng khả năng biểu diễn dữ liệu, tăng tốc độ tính toán.
  - Khả năng đánh địa chỉ (addressing capacity):
    - Quyết định dung lượng bộ nhớ mà vxl có thể đánh địa chỉ
    - Khả năng đánh địa chỉ của vxl theo số bit địa chỉ

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## CÁC ĐẶC ĐIỂM CẤU TRÚC CỦA BỘ VI XỬ LÝ

- Công suất vi xử lý:
  - Tốc độ xử lý lệnh (instruction execution speed):
    - Thường được đo bằng tốc độ thực hiện:
      - Triệu lệnh dấu phẩy động (MFLOPS – Millions of Floating Point Operations Per Second) hoặc
      - Triệu lệnh/giây (MIPS-Millions of Instruction Per Second)
  - MIPS được tính theo kiến trúc von-Neumann

$$\text{MIPS} = (f \times N) / (M + T)$$

Trong đó:

f: Tần số làm việc của bộ vxl

N: Số lượng các bộ ALU độc lập trong vxl

M: Số lượng các vi lệnh trung bình của một lệnh (trung bình 4-7 vi lệnh/lệnh)

T: Hệ số thời gian truy cập bộ nhớ



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

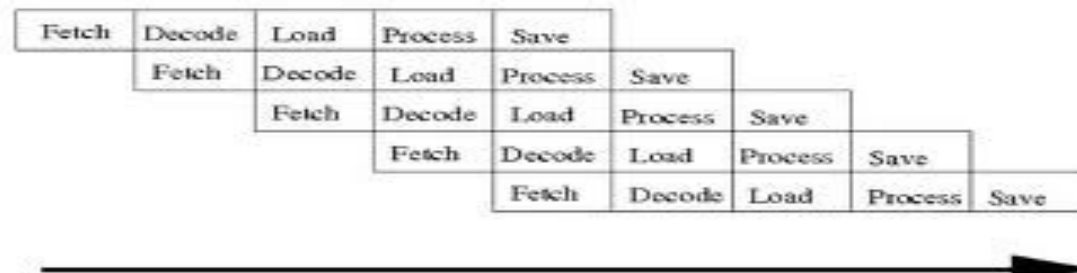
## CÁC ĐẶC ĐIỂM CẤU TRÚC CỦA BỘ VI XỬ LÝ

- Các đặc tính nâng cao:
  - Xử lý song song (parallel processing):
    - Kết hợp nhiều vi xử lý trong một máy tính
    - Tăng tốc độ xử lý dữ liệu, đặc biệt với các hệ thống máy chủ
  - Đồng xử lý (co-processing):
    - Sử dụng bộ đồng xử lý để tăng tốc độ: chuyên tính toán số dấu phẩy động
    - Các module xử lý dấu phẩy động được tích hợp vào vxl trong các vxl hiện đại
  - Kỹ thuật nhớ đệm (cache)
    - Bộ nhớ được tổ chức theo mô hình phân cấp

# CHƯƠNG 1-GIỚI THIỆU CHUNG

## CÁC ĐẶC ĐIỂM CẤU TRÚC CỦA BỘ VI XỬ LÝ

- Các đặc tính nâng cao:
  - Kỹ thuật đường ống (pipelining technique)
    - Một lệnh được chia thành nhiều bước
    - Các bước của nhiều lệnh được thực hiện xen kẽ nhau tại các bộ phận khác nhau của vi xử lý



- Tăng độ rộng bus: tăng tốc độ xử lý nhờ tăng độ dài từ dữ liệu xử lý.



# CHƯƠNG 1-GIỚI THIỆU CHUNG

---

## LỊCH SỬ PHÁT TRIỂN VI XỬ LÝ

- Intel:
  - Công ty Fairchild Semiconductor (được thành lập năm 1957) đã phát minh ra IC đầu tiên vào năm 1959
  - Năm 1968, Robert Noyce, Gordon Moore và Andrew tách ra khỏi Fairchild Semiconductor và thành lập INTEL
  - Các thế hệ của vi xử lý intel: Intel 4004 (4 bit), 8080 (8 bit), 8085, 8086 (16 bit), 80186, 80286,..., Intel Core 2 Duo,...
- ARM
  - Nội dung chi tiết trong chương 2