

BÀI 5 TỔ CHỨC MÁY TÍNH

1. BỘ VI XỬ LÝ – CPU

2. GIAO TIẾP GIỮA CPU VÀ THIẾT BỊ NGOẠI VI

3. CẤU TRÚC VI XỬ LÝ 8086/8088

4. CÁCH MÃ HÓA LỆNH CỦA 8086/8088

BÀI 5 TỔ CHỨC MÁY TÍNH

BỘ VI XỬ LÝ – CPU

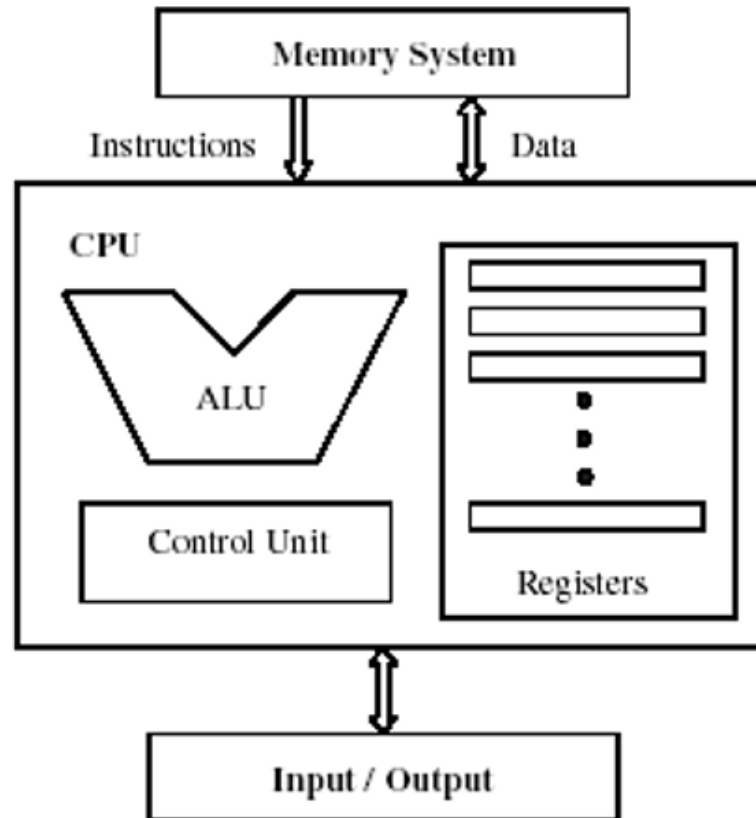
Cấu tạo CPU gồm 3 thành phần :

- **ALU** (Đơn vị luận lý số học): thành phần chịu trách nhiệm tính toán các phép toán số học và logic
- **Thanh ghi**: dùng để chứa các toán hạng, kết quả trả về và địa chỉ truy xuất của CPU
- **CU** (Đơn vị điều khiển): thành phần chịu trách nhiệm điều khiển các trạng thái hoạt động của CPU

BÀI 5 TỔ CHỨC MÁY TÍNH

BỘ VI XỬ LÝ – CPU

Thành phần của CPU:



BÀI 5 TỔ CHỨC MÁY TÍNH

BỘ XỬ LÝ LUẬN LÝ SỐ HỌC - ALU

Các chức năng chủ yếu của ALU:

- Thực hiện các phép toán logic (AND, OR, XOR, NOT) và các phép toán số học (CỘNG, TRỪ, NHÂN, CHIA)
- Thực hiện việc chuyển dữ liệu

BÀI 5 TỔ CHỨC MÁY TÍNH

BỘ ĐIỀU KHIỂN - CU

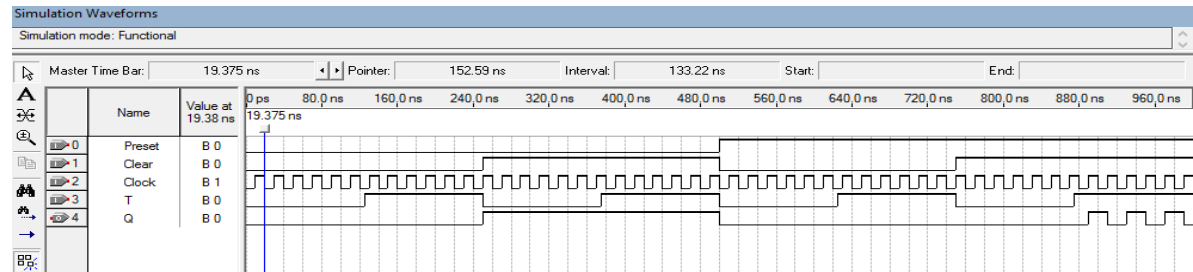
Chịu trách nhiệm liên quan đến việc giải mã và thực thi các lệnh được lấy ra từ trong bộ nhớ lệnh bằng cách đưa ra các tín hiệu điều khiển và định thời cho bộ ALU và các khối phần cứng khác

BÀI 5 TỔ CHỨC MÁY TÍNH

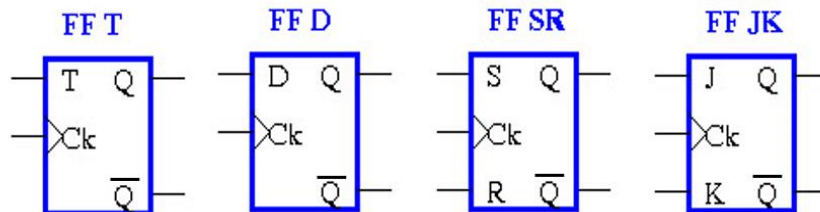
BỘ ĐIỀU KHIỂN - CU

Flip-flop T

| INPUTS | | | | OUTPUTS |
|--------|-------|-------|---|----------------|
| Preset | Clear | Clock | T | Q |
| 0 | 0 | X | X | 0 |
| 0 | 1 | X | X | 1 |
| 1 | 0 | X | X | 0 |
| 1 | 1 | ↑ | 1 | Đảo trạng thái |
| 1 | 1 | ↑ | 0 | Giữ trạng thái |



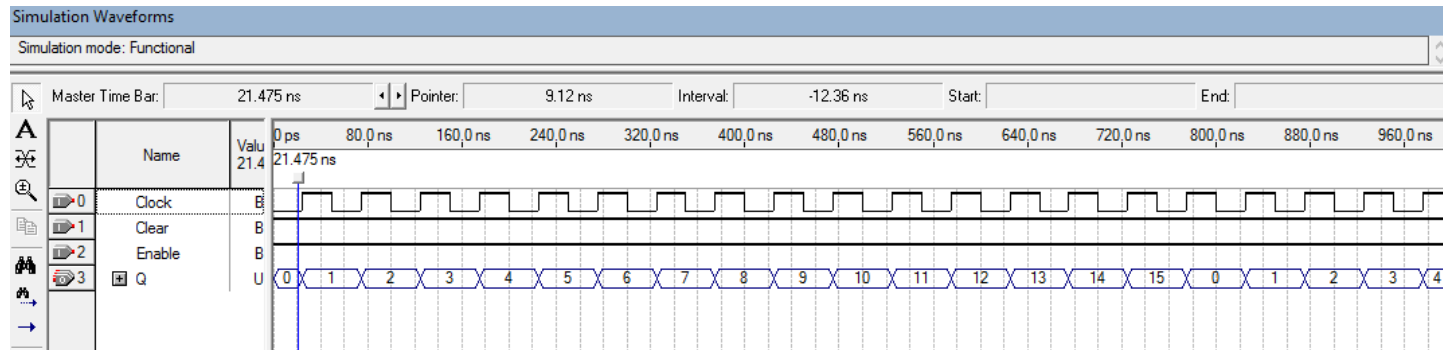
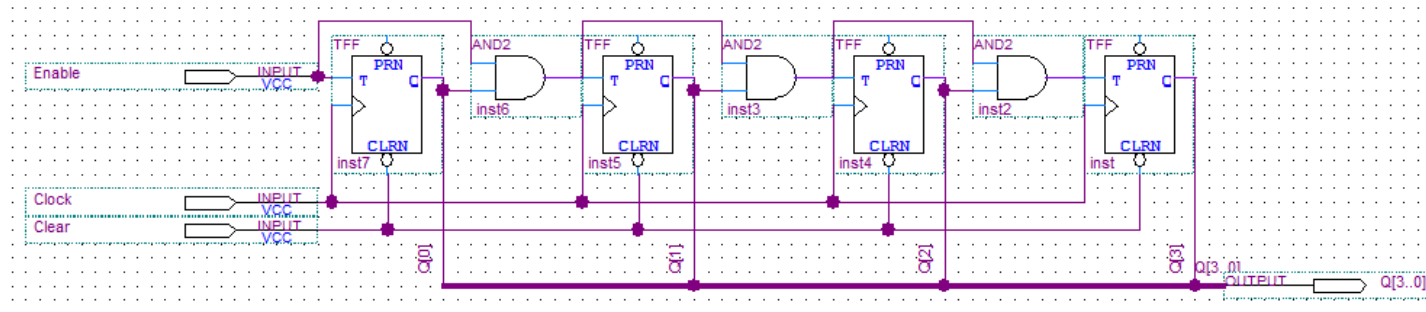
FF là mạch có khả năng lật lại trạng thái ngõ ra tùy theo sự tác động thích hợp của ngõ vào, điều này có ý nghĩa quan trọng trong việc lưu trữ dữ liệu trong mạch và xuất dữ liệu ra khi cần. Có nhiều loại flip-flop khác nhau, chúng được sử dụng rộng rãi trong nhiều ứng dụng.



BÀI 5 TỔ CHỨC MÁY TÍNH

BỘ ĐIỀU KHIỂN - CU

Bộ đếm 4 bit



BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI - REGISTER

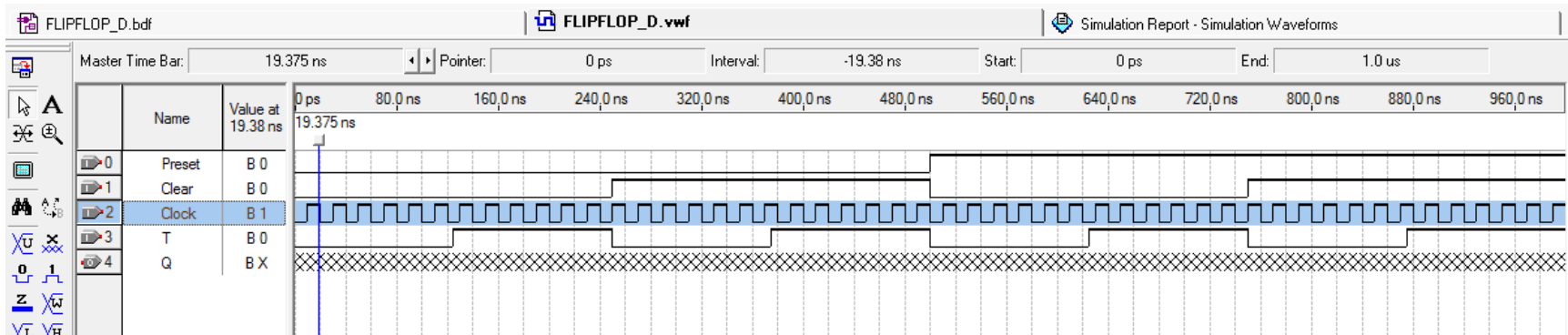
- Thanh ghi là nơi bộ vi xử lý dùng để lưu trữ các giá trị (dữ liệu hoặc địa chỉ)
- Có nhiều loại thanh ghi khác nhau: thanh ghi lệnh, thanh ghi dữ liệu, thanh ghi đa năng, thanh ghi trạng thái, ...
- Các thanh ghi điều khiển và các thanh ghi trạng thái được CU dùng để điều khiển việc thực hiện chương trình nên không thể can thiệp, thay đổi được

BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI - REGISTER

Flip-flop D

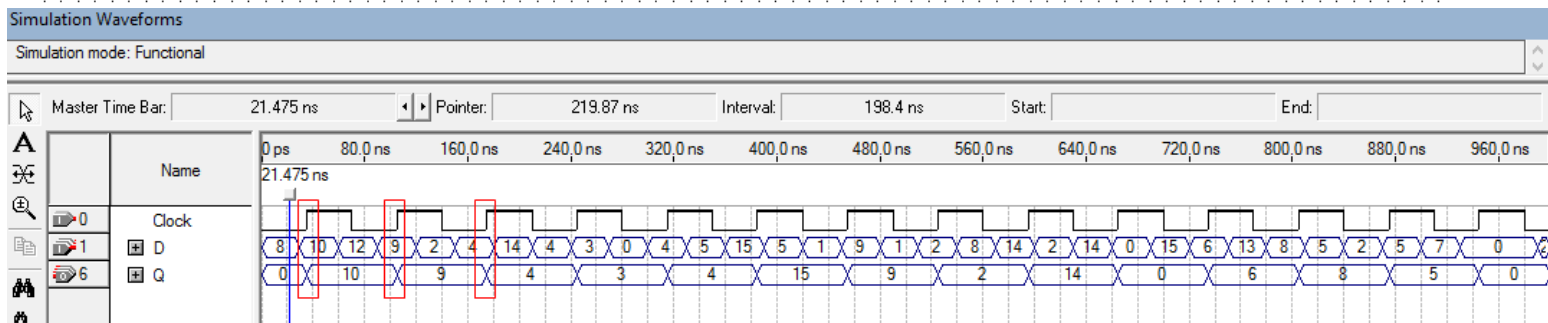
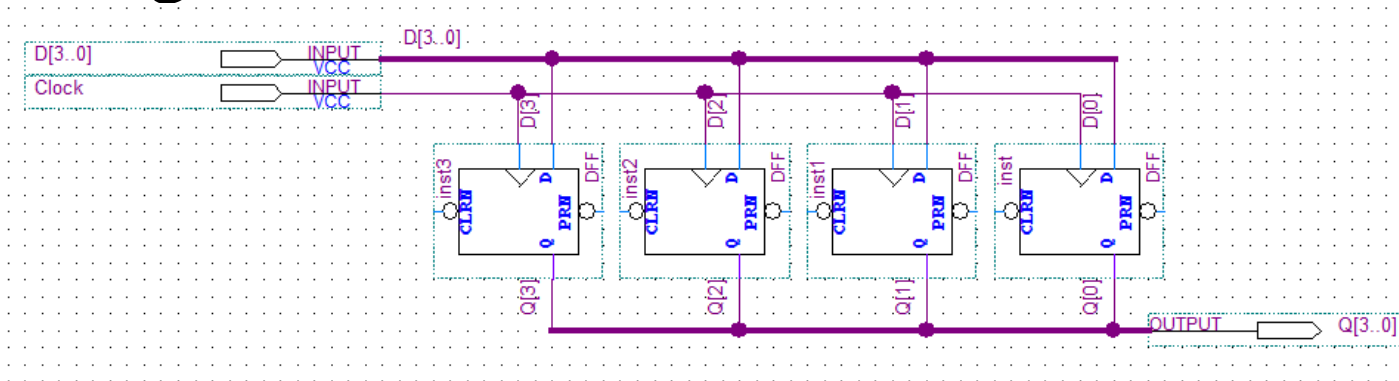
| INPUTS | | | | OUTPUTS |
|--------|-------|-------|---|---------|
| Preset | Clear | Clock | D | Q |
| 0 | 0 | X | X | 0 |
| 0 | 1 | X | X | 1 |
| 1 | 0 | X | X | 0 |
| 1 | 1 | ↑ | 1 | 1 |
| 1 | 1 | ↑ | 0 | 0 |



BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI - REGISTER

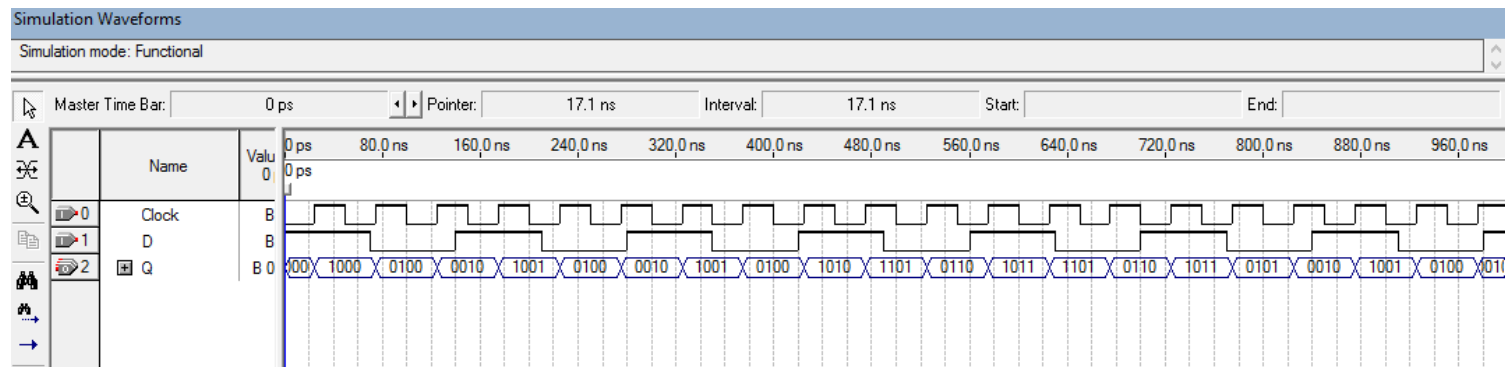
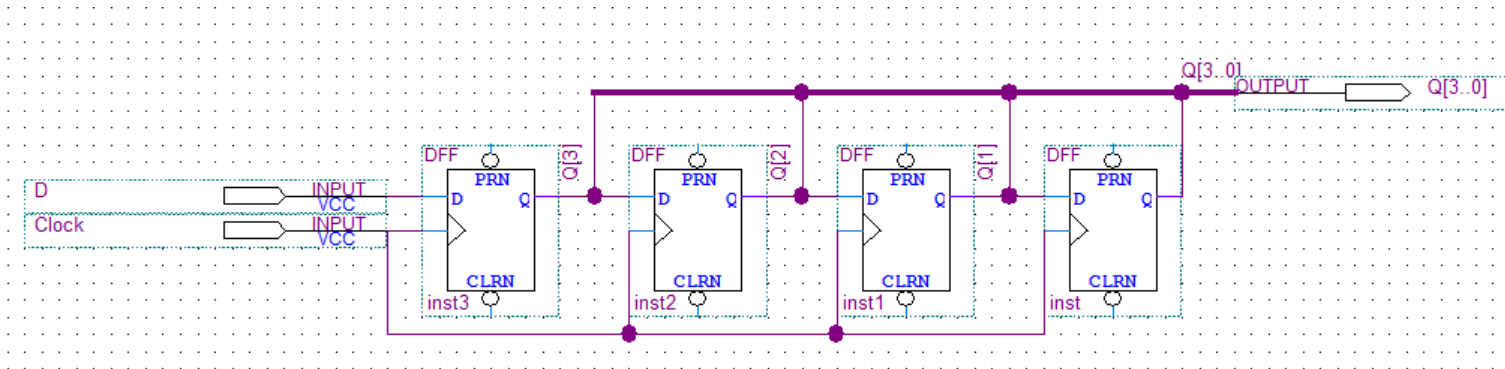
Thanh ghi 4 bit



BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI - REGISTER

Thanh ghi dịch 4 bit



BÀI 5 TỔ CHỨC MÁY TÍNH

GIAO TIẾP GIỮA CPU VÀ THIẾT BỊ NGOẠI VI

- **Phân loại BUS**
- **Mô hình tổng quát**

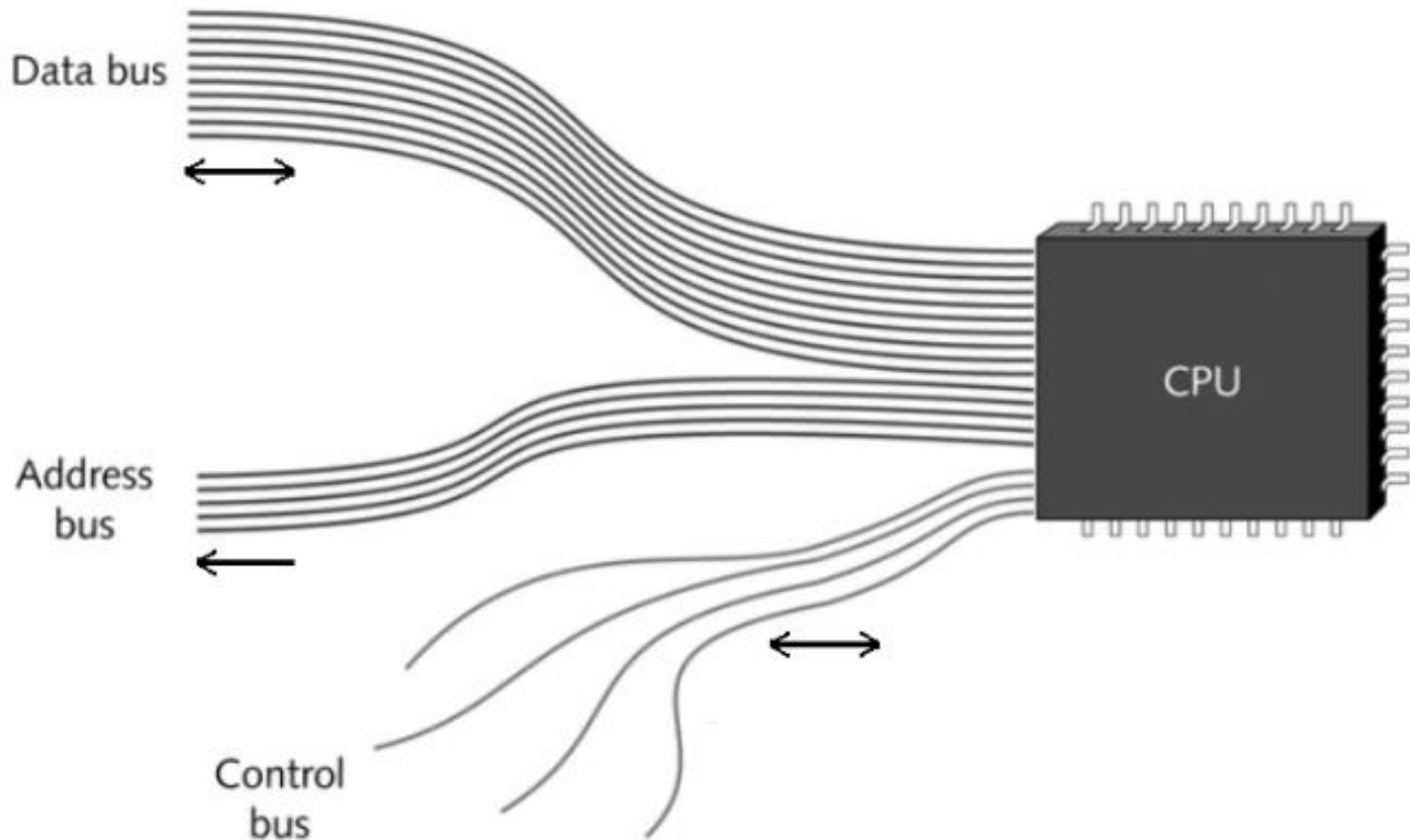
BÀI 5 TỔ CHỨC MÁY TÍNH

PHÂN LOẠI BUS

- **Bus dữ liệu** (data bus): truyền dữ liệu giữa các thành phần trong hệ thống (2 chiều)
- **Bus địa chỉ** (address bus): để CPU xác định các thành phần trong hệ thống (1 chiều)
- **Bus điều khiển** (control bus): để CPU ra lệnh điều khiển cho các thành phần trong hệ thống (2 chiều)

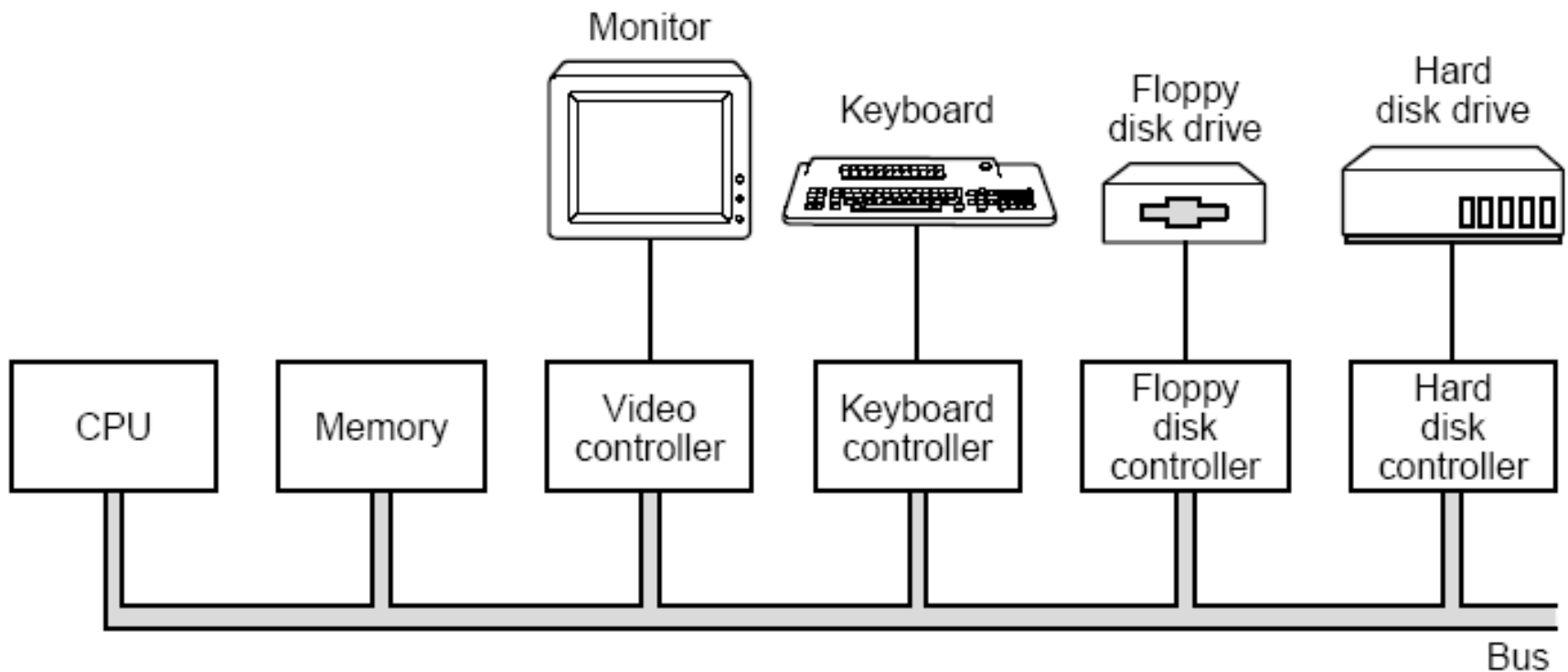
BÀI 5 TỔ CHỨC MÁY TÍNH

PHÂN LOẠI BUS



BÀI 5 TỔ CHỨC MÁY TÍNH

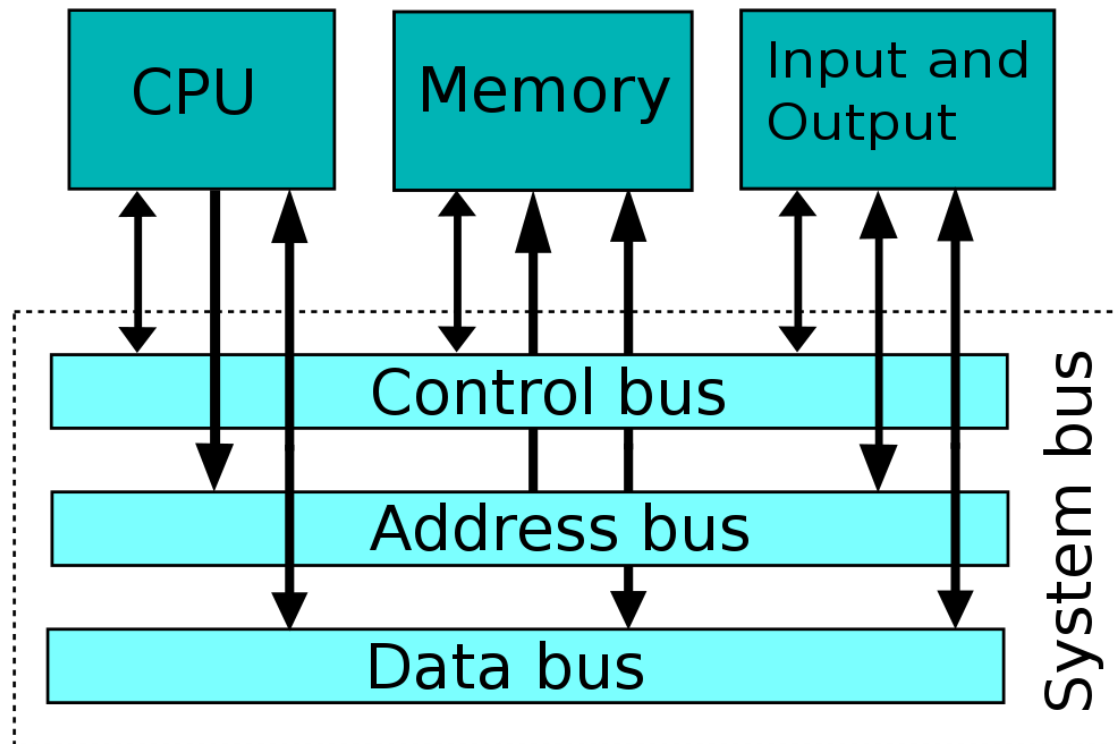
MÔ HÌNH TỔNG QUÁT



BÀI 5 TỔ CHỨC MÁY TÍNH

MÔ HÌNH TỔNG QUÁT

Kiến trúc von Neumann



BÀI 5 TỔ CHỨC MÁY TÍNH

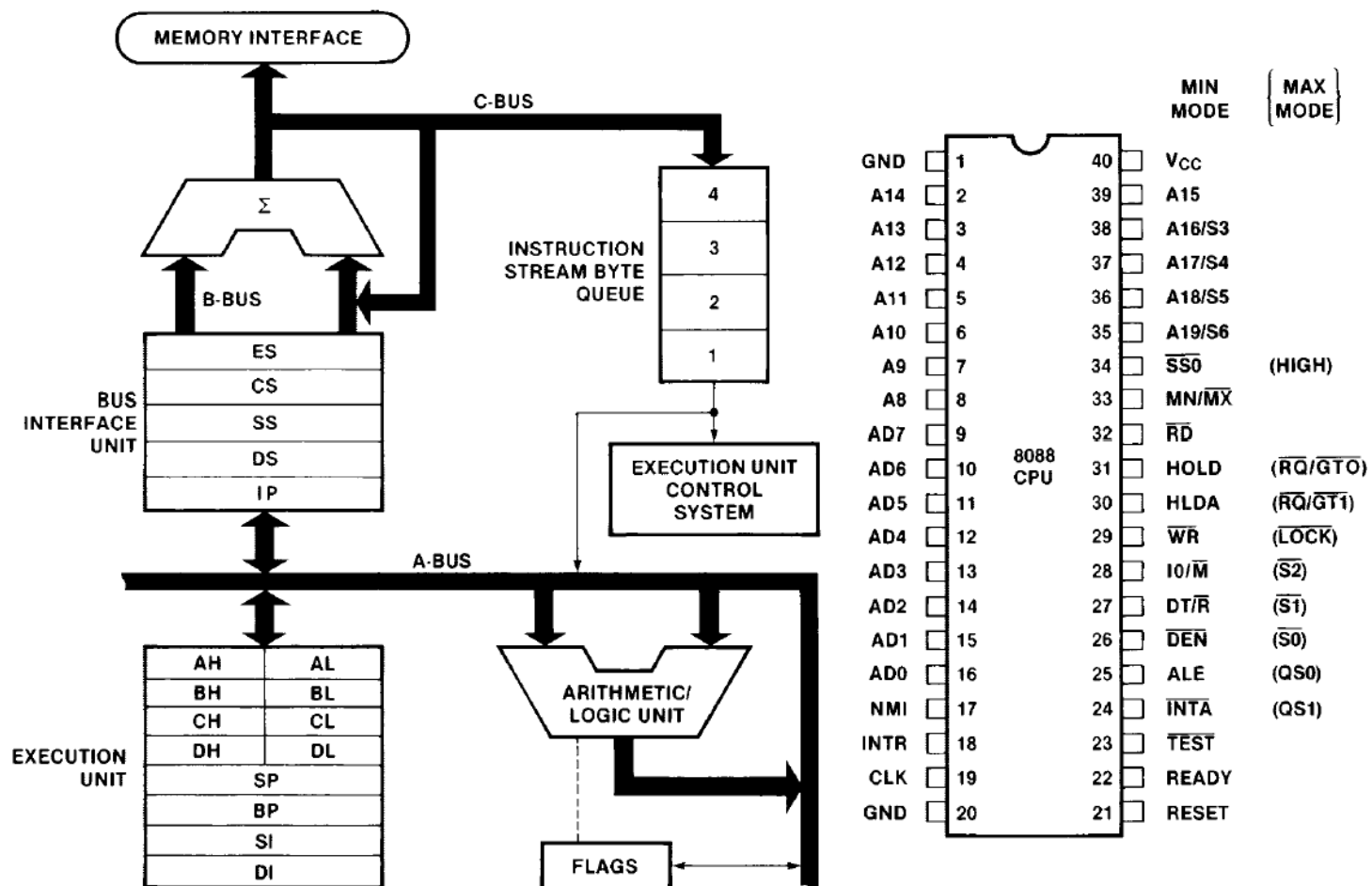
CẤU TRÚC VI XỬ LÝ 8086/8088

Các đặc điểm chính:

- Bus địa chỉ có 20 bit, do đó không gian bộ nhớ mà CPU này có thể định vị được là 2^{20} ô nhớ. Trong các CPU của Intel, mỗi ô nhớ có độ dài là 8 bit, nên không gian bộ nhớ tương ứng của CPU 8086/8088 là 1 Mega bytes
- Các thanh ghi có độ lớn 16 bit, tổng cộng có 14 thanh ghi. Trong đó, các thanh ghi đa năng **AX**, **BX**, **CX**, **DX** được trình bày ở dạng hai thanh ghi 8 bit tương ứng

BÀI 5 TỔ CHỨC MÁY TÍNH

CẤU TRÚC VI XỬ LÝ 8086/8088



BÀI 5 TỔ CHỨC MÁY TÍNH

CẤU TRÚC VI XỬ LÝ 8086/8088

Xuất phát từ việc các thanh ghi bên trong CPU là **16 bit** mà bus địa chỉ của nó lại là **20 bit**. Điều này có nghĩa là CPU sẽ dùng hai thanh ghi **16 bit để tạo thành một địa chỉ 20 bit**. Địa chỉ 20 bit được đưa ra trên Bus địa chỉ được gọi là **địa chỉ vật lý**, còn địa chỉ được ký hiệu bằng hai thanh ghi 16 bit (**segment : offset**) để hình thành địa chỉ vật lý được gọi là **địa chỉ logic**.

BÀI 5 TỔ CHỨC MÁY TÍNH

CẤU TRÚC VI XỬ LÝ 8086/8088

- Cách chuyển đổi từ địa chỉ logic sang địa chỉ vật lý được tính như sau:

$$\text{Địa chỉ vật lý} = \text{Segment} * 16 + \text{Offset}$$

- Ví dụ tìm địa chỉ vật lý từ các địa chỉ logic sau:

$$1234_h : 0005_h$$

-> Địa chỉ vật lý sẽ là: $1234_h * 16 + 0005_h = 12345_h$

BÀI 5 TỔ CHỨC MÁY TÍNH

CẤU TRÚC VI XỬ LÝ 8086/8088

Từ đó, ta có thể nhận thấy :

- Một địa chỉ logic chỉ cho duy nhất một địa chỉ vật lý nhưng một địa chỉ vật lý có thể có nhiều địa chỉ logic.
- Trong CPU 8088/8086 sẽ có các thanh ghi được dùng làm segment (gọi là các **thanh ghi đoạn**) và các thanh ghi dùng là offset trong quá trình tính toán địa chỉ.

BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI ĐOẠN

- **CS** (Code Segment) là thanh ghi đoạn mã lệnh, trỏ đến vùng nhớ chứa mã lệnh của chương trình.
- **DS** (Data Segment) là thanh ghi đoạn dữ liệu, trỏ đến vùng nhớ chứa dữ liệu của chương trình.
- **ES** (Extra Segment) là thanh ghi đoạn dữ liệu phụ, trỏ đến vùng nhớ chứa dữ liệu của chương trình.
- **SS** (Stack Segment) là thanh ghi đoạn ngăn xếp.

BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI ĐA NĂNG

- Thanh ghi **AX** (Accumulator Register - thanh ghi tích lũy): Dùng trong các phép nhân chia, chứa dữ liệu cho các lệnh in, out.
- Thanh ghi **BX** (Base Register - thanh ghi cơ sở): Dùng trong chế độ địa chỉ cơ sở.
- Thanh ghi **CX** (Count register - thanh ghi đếm): Dùng để đếm số lần lặp trong các lệnh lặp.
- Thanh ghi **DX** (Data Register - thanh ghi dữ liệu): Dùng để chứa địa chỉ cho các lệnh in/ out và làm toán hạng trong các lệnh nhân, chia 16 bit. (DX chứa 16 bit cao).

BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI CHỈ SỐ

- Thanh ghi chỉ số nguồn **SI** (Source Index) và thanh ghi chỉ số đích **DI** (Destination Index): Các thanh ghi này có thể được dùng như là các con trỏ để truy xuất gián tiếp đến bộ nhớ. Ngoài ra, các thanh ghi này còn có thể được trong các lệnh chuỗi khi truy xuất đến các chuỗi ký tự.

BÀI 5 TỔ CHỨC MÁY TÍNH

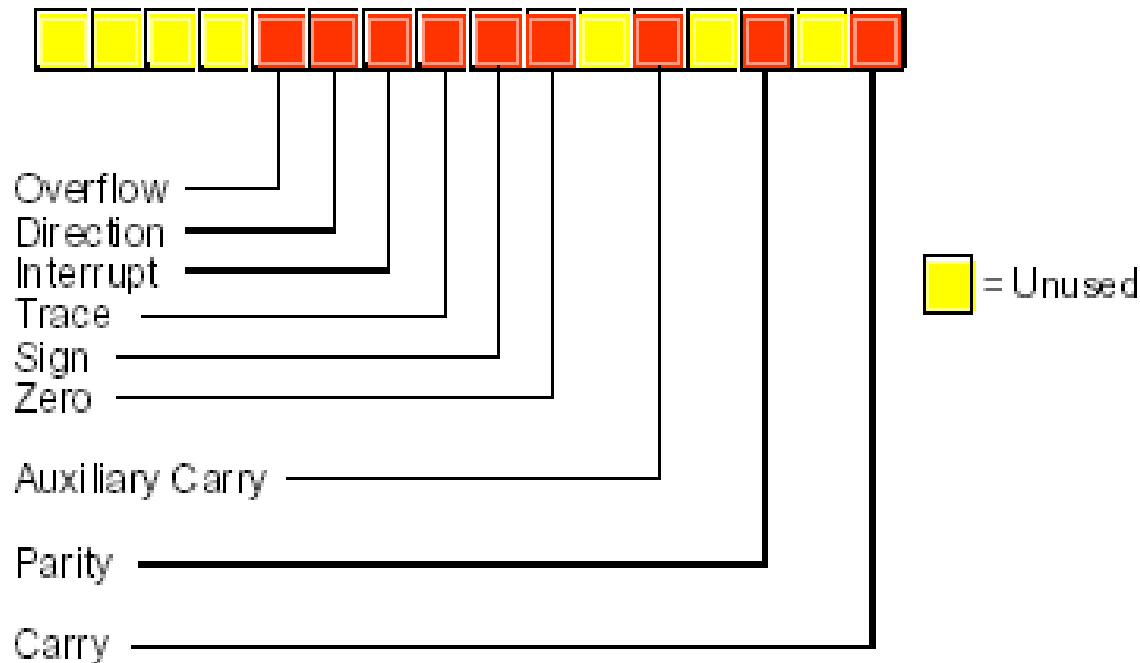
THANH GHI CON TRỎ

- Thanh ghi con trỏ lệnh **IP** (Instruction Pointer): Trỏ đến lệnh tiếp theo mà CPU sẽ thực hiện, địa chỉ đầy đủ là CS: IP.
- Thanh ghi con trỏ ngăn xếp **SP** (Stack Pointer): Trỏ đến ô nhớ nằm trong vùng ngăn xếp, địa chỉ đầy đủ là SS:SP.
- Thanh ghi con trỏ cơ sở **BP** (Base Pointer): Được dùng để truy xuất gián tiếp đến bộ nhớ.

BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI CỜ

Thanh ghi cờ (**Flag**) : Thanh ghi cờ bên trong CPU có nhiều chức năng rất quan trọng như phản ánh kết quả sau khi tính toán, thể hiện trạng thái hoạt động của CPU...



BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI CỜ

- **Cờ tràn** (Overflow Flag - OF): OF = 1 khi kết quả vượt quá khả năng lưu trữ của toán hạng đích (Destination operand) khi xem là số có dấu.
- **Cờ hướng** (DirectionFlag - DF): Dùng cho các lệnh chuỗi, khi DF = 0 thì các CPU 80x86 xử lý các phần tử chuỗi theo chiều tăng của địa chỉ. Khi DF = 1 thì xử lý theo chiều ngược lại.
- **Cờ ngắt** (Interrupt flag - IF): Cho phép các ngắt che được tác động.
- **Cờ bắt** (Trace Flag - TF): Cho phép chế độ chạy từng bước (dùng cho quá trình gỡ rối (Debug)...).

BÀI 5 TỔ CHỨC MÁY TÍNH

THANH GHI CỜ

- **Cờ dấu** (Sign Flag - **SF**): Phản ánh kết quả tính toán dương hay âm. $SF = 1$ khi kết quả âm
- **Cờ Zero** (Zero Flag - **ZF**): Phản ánh kết quả có bằng Zero hay không. $ZF = 1$ khi kết quả bằng Zero.
- **Cờ phụ** (Auxiliary Flag - **AF**): Dùng cho các phép toán BCD (Binary Coded Decimal) đặc biệt, trong khi lập trình ít khi sử dụng đến cờ này.
- **Cờ parity** (Parity Flag **PF**): Phản ánh tính chẵn lẻ của kết quả. $PF = 1$ khi kết quả là số chẵn bit 1.
- **Cờ nhớ** (Carry Flag – **CF**): $CF = 1$ khi có nhớ hoặc mượn ở MSB của kết quả.

BÀI 5 TỔ CHỨC MÁY TÍNH

CÁCH MÃ HÓA LỆNH CỦA 8086/8088

- Lệnh của vi xử lý được ghi bằng các ký tự ở dạng **mã gọi nhớ** để thuận tiện cho người lập trình. Đối với vi xử lý thì lệnh tồn tại ở dạng **nhị phân** (ở dạng các trị giá 0, 1), dạng này còn được gọi là **mã máy**. Một lệnh có độ dài một hoặc vài byte tùy thuộc vào bộ vi xử lý.
- Trong vi xử lý 8088/ 8086 một lệnh có độ dài từ 1 đến 6 byte.

BÀI 5 TỔ CHỨC MÁY TÍNH

CÁCH MÃ HÓA LỆNH CỦA 8086/8088

- Ví dụ: mã gọi nhớ và mã máy của lệnh **MOV**

MOV = Move:

Register/Memory to/from Register

7 6 5 4 3 2 1 0

1 0 0 0 1 0 dw

7 6 5 4 3 2 1 0

mod reg r/m

7 6 5 4 3 2 1 0

7 6 5 4 3 2 1 0

Immediate to Register/Memory

1 1 0 0 0 1 1 w

mod 0 0 0 r/m

data

data if w = 1

Immediate to Register

1 0 1 1 w reg

data

data if w = 1

Memory to Accumulator

1 0 1 0 0 0 0 w

addr-low

addr-high

Accumulator to Memory

1 0 1 0 0 0 1 w

addr-low

addr-high

Register/Memory to Segment Register

1 0 0 0 1 1 1 0

mod 0 reg r/m

Segment Register to Register/Memory

1 0 0 0 1 1 0 0

mod 0 reg r/m

BÀI 5 TỔ CHỨC MÁY TÍNH

CÁCH MÃ HÓA LỆNH CỦA 8088/8086

- Ví dụ tìm mã máy từ mã gợi nhớ:

MOV AL, [SI]

- Ý nghĩa của lệnh này là đưa nội dung của ô nhớ có địa chỉ logic là **DS: SI** vào thanh ghi **AL**. Nên lệnh này có dạng:

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|------------|-------------|-------------|
| 1 | 0 | 0 | 0 | 1 | 0 | d | w | Mod(2 bit) | Reg (3 bit) | R/m (3 bit) |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 00 | 000 | 100 |

- Lệnh này di chuyển dữ liệu 1 byte ($w = 0$), dữ liệu đi đến thanh ghi ($d = 1$), thanh ghi tham gia vào lệnh là AL ($reg = 000$), toán hạng [SI] được xác định với $mod = 00$ và $r/m = 100$. Nên mã máy của lệnh này là **8A04h**.

BÀI 5 TỔ CHỨC MÁY TÍNH

CÁCH MÃ HÓA LỆNH CỦA 8088/8086

- Ví dụ tìm mã máy từ mã gọi nhớ:

MOV SP, BX

- Mã lệnh MOV = 100010
- D=1: gửi tới thanh ghi
- W=1: chuyển word
- Mod=11
- Reg=100: thanh ghi SP
- R/M=011: thanh ghi BX

| 1 | 0 | 0 | 0 | 1 | 0 | d | W | mod | Reg | r/m |
|---|---|---|---|---|---|---|---|-----|-----|-----|
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 11 | 100 | 011 |

Nên mã máy của lệnh này là: **8BE3h**

BÀI 5 TỔ CHỨC MÁY TÍNH

CÁCH MÃ HÓA LỆNH CỦA 8088/8086

Ví dụ tìm mã máy từ mã gọi nhớ:

MOV [BX+9], AL

- Mã lệnh MOV = 100010
- D=0: đi ra từ thanh ghi
- W=0: chuyển 1 byte (thanh ghi AL)
- Mod=01
- Reg=000
- R/M=111: thanh ghi BX

| 1 | 0 | 0 | 0 | 1 | 0 | d | W | mod | Reg | r/m | data | |
|---|---|---|---|---|---|---|---|-----|-----|-----|------|------|
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 01 | 000 | 111 | 0000 | 1001 |

Nên mã máy của lệnh này là: **884709h**