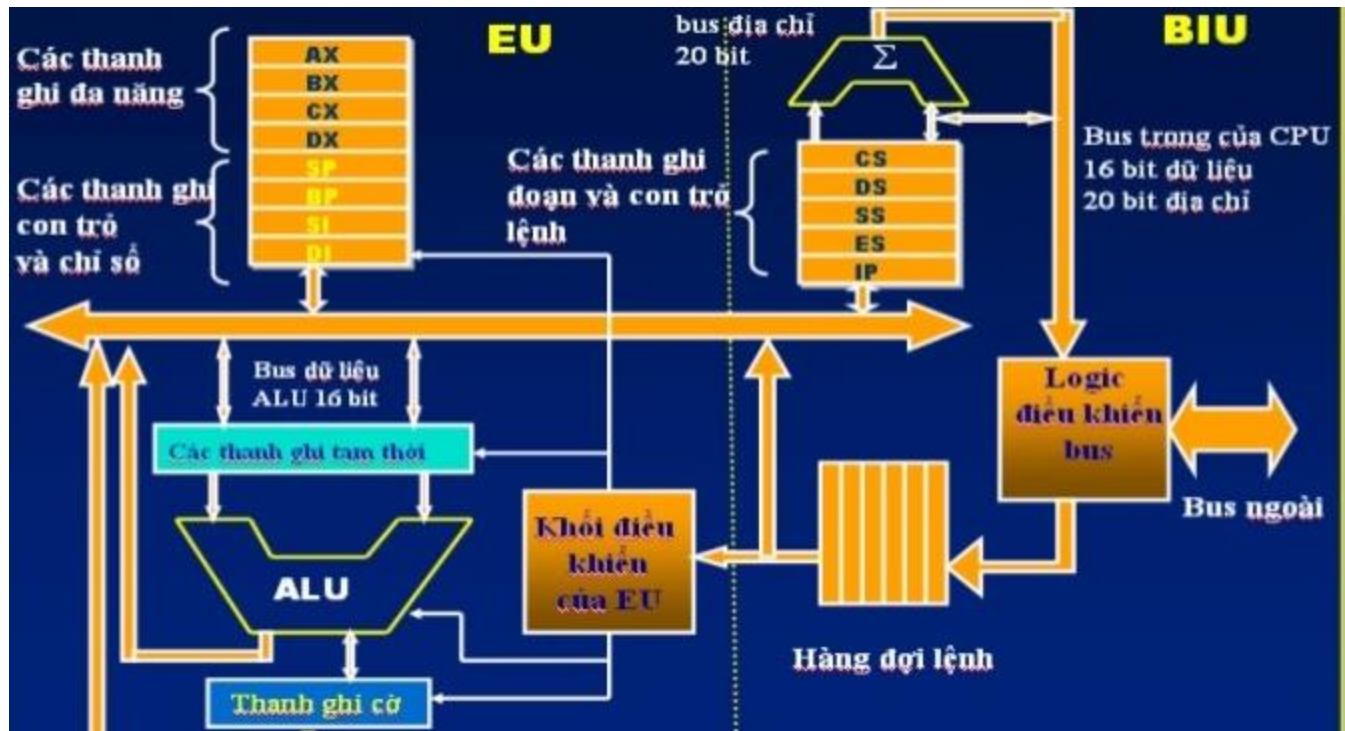


Chương 2: Bộ vi xử lý 8086/8088 của Intel

I. Giới thiệu về bộ vi xử lý 8086/8088



Chương 2: Bộ vi xử lý 8086/8088 của Intel

Thiết kế của 8088 chia làm 2 khối riêng biệt:

- Khối EU (Execution Unit)
- Khối BIU (Bus Interface Unit)

Các thanh ghi bên trong 8088 gồm có 14 thanh ghi 16 bit: AX, BX, CX, DX, DS, CS, ES, SS, IP, SI, DI, SP, BP, FR và được chia làm 4 nhóm thanh ghi cơ bản sau:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

1. Nhóm thanh ghi đa năng

Gồm 4 thanh ghi 16 bit: AX, BX, CX, DX

- Được dùng cho nhiều mục đích khác nhau
- Tuy nhiên mỗi thanh ghi cũng được gán một chức năng chuyên biệt riêng. Chính vì vậy chúng được gán những tên có ý nghĩa.

Cụ thể như sau:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- *AX (Accumulator)*: thanh ghi chứa.

Các kết quả của các thao tác thường được chứa ở AX (kết quả của phép nhân, chia).

- *BX (Base)*: thanh ghi cơ sở, thường dùng để chứa địa chỉ cơ sở của một dãy các ô nhớ.
- *CX (Count)*: thanh đếm. CX thường được dùng để chứa số lần lặp trong trường hợp các lệnh LOOP (lặp).

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- DX (*Data*): thanh ghi dữ liệu. DX tham gia các thao tác của phép nhân hoặc chia các số 16 bit. DX thường dùng để chứa địa chỉ của các cổng trong các lệnh vào/ ra dữ liệu.

Trong trường hợp không sử dụng hết 16 bit thì các thanh ghi này có thể tách đôi mỗi thanh ghi thành hai thanh ghi 8 bit có tên gọi riêng

Chương 2: Bộ vi xử lý 8086/8088 của Intel

Các thanh ghi tách đôi thành 2 thanh ghi:

AX	AH 8 bit	AL 8 bit
BX	BH	BL
CX	CH	CL
DX	DH	DL

Ví dụ:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

2. Nhóm thanh ghi đoạn

8088 sử dụng một không gian bộ nhớ $2^{20} = 1 \text{ MB}$

- Trong không gian này nó chia thành các vùng nhớ riêng biệt, mỗi vùng chứa một loại dữ liệu riêng.
- Để quản lý không gian các vùng nhớ này, 8088 sử dụng 4 thanh ghi 16 bit đóng vai trò là thanh ghi đoạn có tác dụng chứa địa chỉ đầu của các vùng nhớ:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- CS (Code Segment)
- DS (Data Segment)
- SS (Stack Segment)
- ES (Extra Segment)

Cách tính địa chỉ vật lý:

$$\begin{aligned}\text{Đcvi} &= \text{đc đoạn} * 10\text{H} + \text{đc lệch} \\ &= \text{segment} * 10\text{H} + \text{offset}\end{aligned}$$

Ví dụ:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

3. Nhóm thanh ghi con trỏ và chỉ số:

8088 sử dụng 5 thanh ghi 16 bit đóng vai trò là các thanh ghi con trỏ và chỉ số, có tác dụng chứa **địa chỉ lệch** của các ngăn nhớ trong các vùng nhớ. 5 thanh ghi đó là SP, BP, SI, DI, IP

- BP: con trỏ cơ sở (*Base Pointer*). BP luôn trỏ vào một dữ liệu nằm trong đoạn ngăn xếp SS. Địa chỉ vật lý của dữ liệu trong đoạn ngăn xếp được xác định qua cặp thanh ghi SS:BP

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- SP: con trỏ ngăn xếp (*Stack Pointer*). SP luôn trỏ vào đỉnh hiện thời của ngăn xếp nằm trong đoạn ngăn xếp SS. Địa chỉ vật lý của đỉnh hiện thời ngăn xếp được xác định thông qua cặp thanh ghi SS:SP
- SI: chỉ số gốc hay nguồn (*Source Index*). SI chỉ vào dữ liệu trong đoạn dữ liệu DS, do đó địa chỉ vật lý của ngăn nhớ dữ liệu do SI trỏ đến được xác định thông qua cặp thanh ghi DS:SI

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- DI: chỉ số đích (*Destination Index*). DI chỉ vào dữ liệu trong đoạn dữ liệu DS, do đó địa chỉ vật lý của ngăn nhớ dữ liệu do DI trỏ đến được xác định thông qua cặp thanh ghi DS:DI
- IP: Con trỏ lệnh (*Instruction Pointer*). IP luôn trỏ vào ngăn nhớ lệnh tiếp theo sẽ thực hiện nằm trong đoạn mã lệnh, do đó địa chỉ vật lý của lệnh tiếp theo sẽ thực hiện được xác định thông qua cặp thanh ghi CS:IP

Chương 2: Bộ vi xử lý 8086/8088 của Intel

4. Thanh ghi cờ (FR – Flag Register):

8088 sử dụng 1 thanh ghi 16 bit đóng vai trò là thanh ghi cờ, có tác dụng chứa cờ trạng thái phép toán và cờ trạng thái hệ thống, có dạng:



Trong đó: x là chưa định nghĩa

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Cờ hệ thống:

- T hoặc TF (Trap Flag): cờ bẫy. TF = 1 thì CPU làm việc ở chế độ chạy từng lệnh
- I hoặc IF (Interrupt Enable Flag): cờ cho phép ngắt. IF = 1 thì CPU cho phép các yêu cầu ngắt (che được)
- D hoặc DF (Direction Flag): cờ hướng (cờ lùi)

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Cờ phép toán:

- C hoặc CF (Carry Flag): cờ nhớ.
- P hoặc PF (Parity Flag): cờ chẵn lẻ. PF phản ánh tính chẵn lẻ của tổng số bit 1 có trong kết quả. Cờ PF =1 khi tổng số bit 1 trong kết quả là lẻ (odd parity) và PF =0 khi tổng số bit 1 trong kết quả là chẵn (even parity).

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- A hoặc AF (Auxiliary Flag): cờ nhớ phụ rất có ý nghĩa khi ta làm việc với các số BCD (Binary Coded Decimal). AF = 1 khi có nhớ hoặc mượn từ một số BCD thấp (4 bit thấp) sang một số BCD cao (4 bit cao).
- Z hoặc ZF (Zero Flag): cờ rỗng. ZF = 1 khi kết quả = 0 và ZF = 0 khi kết quả $\neq 0$.

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- S hoặc SF (sign flag): cờ dấu. SF = 1 khi kết quả âm và SF = 0 khi kết quả không âm.
- O hoặc OF (Overflow Flag): cờ tràn. OF = 1 khi kết quả là một số bù 2 vượt qua ngoài giới hạn biểu diễn dành cho nó.

Chương 2: Bộ vi xử lý 8086/8088 của Intel

II. Tập lệnh của 8088

1. Nhóm các lệnh dịch chuyển dữ liệu
2. Nhóm các lệnh số học
3. Nhóm các lệnh logic
4. Nhóm các lệnh quay, dịch
5. Nhóm các lệnh so sánh, nhảy

Chương 2: Bộ vi xử lý 8086/8088 của Intel

1. Nhóm các lệnh dịch chuyển dữ liệu

+ Lệnh MOV

Cú pháp: MOV đích, gốc

Ý nghĩa:

Ví dụ:

MOV AL, BL ;chuyển dữ liệu trong BL vào AL

MOV AX, BX ;chuyển dữ liệu trong BX vào AX

MOV [SI], AL ; chuyển dữ liệu trong AL vào ngăn nhớ do SI trỏ đến

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh XCHG

Cú pháp: XCHG đích, gốc

Ý nghĩa:

Ví dụ:

XCHG AL, BL ;đổi dữ liệu trong BL và AL cho nhau

XCHG AX, BX ;đổi dữ liệu trong BX và AX cho nhau

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh LEA

Cú pháp: LEA đích, gốc

Ý nghĩa:

Ví dụ:

LEA DX, st ;chuyển địa chỉ offset của toán
hạng st vào DX

; MOV dx, offset st

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh PUSH

Cú pháp: PUSH gốc

Ý nghĩa:

Ví dụ:

PUSH AX ;đẩy giá trị trong AX vào stack

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh POP

Cú pháp: POP đích

Ý nghĩa:

Ví dụ:

POP AX ;Lấy giá trị trong stack đưa vào AX

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh IN

Cú pháp: IN đích, địa chỉ cổng

Ý nghĩa:

Ví dụ: - Trường hợp cổng có địa chỉ 8 bit: ví dụ 5AH

IN AL, 5AH ;

IN AX, 5AH ;

- Trường hợp cổng có địa chỉ 16 bit: ví dụ 02F9H

MOV DX, 02F9H

IN AL, DX

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh OUT

Cú pháp: OUT địa chỉ cổng, gốc

Ý nghĩa:

Ví dụ: -Trường hợp cổng có địa chỉ 8 bit: ví dụ 5AH

OUT 5AH, AL ;

OUT 5AH, AX ;

- Trường hợp cổng có địa chỉ 16 bit: ví dụ 02F9H

MOV DX, 02F9H

OUT DX, AL

Chương 2: Bộ vi xử lý 8086/8088 của Intel

2. Nhóm các lệnh số học

+ Lệnh ADD

Cú pháp: ADD đích, gốc

Ý nghĩa:

Ví dụ:

ADD AL, BL ; $AL = AL + BL$

ADD AX, BX ; $AX = AX + BX$

ADD AL, 30H ; $AL = AL + 30H$

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh SUB

Cú pháp: SUB đích, gốc

Ý nghĩa:

Ví dụ:

```
SUB AL, BL ;AL=AL-BL
```

```
SUB AX, BX ;AX = AX-BX
```

```
SUB AL, 30H ; AL =AL-30H
```

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh NEG

Cú pháp: NEG đích

Ý nghĩa:

Ví dụ:

```
MOV AL,5
```

```
NEG AL ; AL = -5
```

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh INC

Cú pháp: INC đích

Ý nghĩa:

Ví dụ:

INC CX; $CX = CX + 1$

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh DEC

Cú pháp: DEC đích

Ý nghĩa:

Ví dụ:

DEC CL ;CL=CL-1

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh MUL

Cú pháp: MUL gốc

Ý nghĩa:

Ví dụ:

+ Lệnh DIV

Cú pháp: DIV gốc

Ý nghĩa:

Ví dụ:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

3. Nhóm các lệnh quay, dịch

+ Lệnh quay trái: ROL/RCL

Cú pháp: ROL đích, n

RCL đích, n

Ý nghĩa:

Ví dụ:

Mov al,95 ; al = 5Fh = 0101 1111b

Mov cl,2 ; số bit quay trái là 2

Rol al,cl ; quay trái al 2 bit, kết quả al = 0111 1101b

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh quay phải: ROR/RCR

Cú pháp: ROR đích, n

RCR đích, n

Ý nghĩa:

Ví dụ:

Mov al,95 ; al = 5Fh = 0101 1111b

Ror AL,1 ; quay phải AL 1 bit
;kết quả AL = 1010 1111b

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh dịch trái: SHL/SAL

Cú pháp: SAL đích, n

SHL đích, n

Ý nghĩa:

Ví dụ:

Mov AL,95 ; al = 5Fh = 0101 1111b

Mov cl,2 ; số bit dịch trái là 2

SHL AL,CL ; dịch trái al 2 bit, kết quả al = 0111 1100b

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh dịch phải: SHR/SAR

Cú pháp: SHR đích, n

 SAR đích, n

Ý nghĩa:

Ví dụ:

Mov al,95 ;al = 5Fh = 0101 1111b

SHR AL,1 ; dịch phải AL 1 bit

 ;kết quả al = 0010 1111b

Chương 2: Bộ vi xử lý 8086/8088 của Intel

4. Nhóm các lệnh logic

+ Lệnh AND

Cú pháp: AND đích, gốc

Ý nghĩa:

Ví dụ:

Mov al,95 ;al = 5Fh =0101 1111b

Mov bl, 71; bl =47h =0100 0111b

and al,bl ;thực hiện phép and giữa al và bl
; kết quả al=0100 0111b

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh OR

Cú pháp: OR đích, gốc

Ý nghĩa:

Ví dụ:

Mov al,95 ;al = 5Fh =0101 1111b

Mov bl, 71; bl =47h =0100 0111b

or al,bl ;thực hiện phép or giữa al và bl
; kết quả al=0101 1111b

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh XOR

Cú pháp: XOR đích, gốc

Ý nghĩa:

Ví dụ:

Mov al,95 ;al = 5Fh =0101 1111b

Mov bl, 71; bl =47h =0100 0111b

xor al,bl ;thực hiện phép xor giữa al và bl
;kết quả al=0001 1000b

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh NOT

Cú pháp: NOT đích

Ý nghĩa:

Ví dụ:

Mov al,95 ;al = 5Fh =0101 1111b

Not al ;thực hiện phép not đối với al
;kết quả al=1010 0000b

Chương 2: Bộ vi xử lý 8086/8088 của Intel

5. Nhóm các lệnh so sánh, nhảy

+ Lệnh CMP

Cú pháp: `cmp` đích, gốc

Ý nghĩa:

Ví dụ:

`Cmp al,50` ; so sánh al với 50

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh JMP

Cú pháp: JMP nhãn

ý nghĩa: nhảy vô điều kiện đến “nhãn”; trong đó “nhãn” là tên do người lập trình đặt và phải kết thúc bằng dấu “:”

Ví dụ

JMP tinhtong ; nhảy đến nhãn tinhtong:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

+ Lệnh LOOP

Cú pháp: LOOP nhãn

ý nghĩa: nhảy đến “nhãn” cho đến khi CX=0 thì kết thúc;
mỗi lần nhảy thì CX tự động giảm 1 đơn vị

Ví dụ

Mov cx, 100

Tinhtong:

.....

LOOP tinhtong ; nhảy đến nhãn “tinhtong”
;với số lần là 100

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- + Lệnh nhảy nếu lớn hơn: JA/JG/JNBE
(Jump if Above/ Jump if Greater than/ Jump if Not Below or Equal)
- + Lệnh nhảy nếu lớn hơn hoặc bằng: JAE/JGE/JNB
- + Lệnh nhảy nếu bé hơn: JB/JNAE/JNGE
- + Lệnh nhảy nếu bé hơn hoặc bằng: JBE/JNA/JNG
- + Lệnh nhảy nếu bằng: JE
- + Lệnh nhảy nếu không bằng (khác): JNE

Chương 2: Bộ vi xử lý 8086/8088 của Intel

III. Các chế độ địa chỉ (*addressing mode*)

Chế độ địa chỉ (*addressing mode*) là cách để CPU tìm thấy toán hạng cho các lệnh của nó khi hoạt động. Một bộ vi xử lý có thể có nhiều chế độ địa chỉ. Các chế độ địa chỉ này được xác định ngay từ khi chế tạo ra bộ vi xử lý và sau này không thể thay đổi được. Bộ vi xử lý 8086 và cả họ 80x86 nói chung có 7 chế độ địa chỉ sau:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- Chế độ địa chỉ thanh ghi (*register addressing mode*).
- Chế độ địa chỉ tức thì (*immediate addressing mode*).
- Chế độ địa chỉ trực tiếp (*direct addressing mode*).
- Chế độ địa chỉ gián tiếp qua thanh ghi (*register indirect addressing mode*).
- Chế độ địa chỉ tương đối cơ sở (*based relative addressing mode*).
- Chế độ địa chỉ tương đối chỉ số (*indexed relative addressing mode*).
- Chế độ địa chỉ tương đối chỉ số cơ sở (*based indexed relative addressing mode*).

Chương 2: Bộ vi xử lý 8086/8088 của Intel

1. Chế độ địa chỉ thanh ghi

Trong chế độ địa chỉ này, người ta dùng các thanh ghi bên trong CPU như là các toán hạng để chứa dữ liệu cần thao tác.

Ví dụ:

MOV BX, DX ; chuyển nội dung DX vào BX

MOV DS, AX ; chuyển nội dung AX vào DS

ADD AL, DL ; $AL = AL + DL$

Chương 2: Bộ vi xử lý 8086/8088 của Intel

2 . Chế độ địa chỉ tức thì

Trong chế độ địa chỉ này, toán hạng đích là một thanh ghi hay một ô nhớ, còn toán hạng nguồn là một hằng số

Ví dụ:

MOV CL, 100 ; chuyển 100 vào CL

MOV AX, 0FF0H ; chuyển 0FF0H vào AX

MOV [BX], 10 ; Nạp 10 vào ô nhớ có địa chỉ DS:BX

Chương 2: Bộ vi xử lý 8086/8088 của Intel

3. Chế độ địa chỉ trực tiếp

Trong chế độ địa chỉ này một toán hạng là **địa chỉ** **lịch** của ô nhớ làm một toán hạng, còn toán hạng kia chỉ có thể là thanh ghi mà không được là ô nhớ.

Ví dụ:

MOV AL, [1234H] ;chuyển nội dung ô nhớ DS:1234
;vào thanh ghi AL.

MOV [4320H], CX ;chuyển CX vào 2 ô nhớ liên tiếp
;DS:4320 và DS:4321

Chương 2: Bộ vi xử lý 8086/8088 của Intel

4 . Chế độ gián tiếp qua thanh ghi

Trong chế độ địa chỉ này, một toán hạng là một thanh ghi được sử dụng để chứa địa chỉ lệch của ô nhớ chứa dữ liệu; toán hạng còn lại chỉ có thể là thanh ghi mà không được là ô nhớ.

Chương 2: Bộ vi xử lý 8086/8088 của Intel

Ví dụ:

MOV AL, [BX] ; chuyển nội dung ô nhớ có địa chỉ
; DS:BX vào thanh ghi AL.

MOV [SI], CL ; chuyển CL vào ô nhớ có địa chỉ DS:SI.

MOV [DI], AX ; chuyển AX vào 2 ô nhớ liên tiếp tại
; DS:DI và DS: [DI + 1]

Chương 2: Bộ vi xử lý 8086/8088 của Intel

5. Chế độ địa chỉ tương đối cơ sở

Trong chế độ địa chỉ này các thanh ghi cơ sở như BX hoặc BP và hằng số biểu diễn giá trị dịch chuyển (*displacement value*) được dùng để tính địa chỉ hiệu dụng của một toán hạng trong vùng nhớ DS hoặc SS. Sự có mặt của các giá trị dịch chuyển xác định tính tương đối của địa chỉ so với địa chỉ cơ sở.

Chương 2: Bộ vi xử lý 8086/8088 của Intel

MOV CX, [BX] +10 ; chuyển nội dung 2 ô nhớ liên tiếp có địa chỉ DS: [BX + 10] và DS: [BX+11] vào CX.

MOV CX, [BX+10] ; một cách viết khác của lệnh trên.

MOV CX, 10 [BX] ; một cách viết khác của lệnh đầu.

MOV AL, [BP] +5 ; chuyển nội dung ô nhớ SS:[BP+5] vào AL

Chương 2: Bộ vi xử lý 8086/8088 của Intel

6. Chế độ địa chỉ tương đối chỉ số

Trong chế độ địa chỉ này các thanh ghi chỉ số như SI hoặc DI và hằng số biểu diễn giá trị dịch chuyển (*displacement value*) được dùng để tính địa chỉ hiệu dụng của một toán hạng trong vùng nhớ DS hoặc ES (trong các lệnh chuyển chuỗi các byte hoặc từ).

Chương 2: Bộ vi xử lý 8086/8088 của Intel

Ví dụ:

MOV CX, [SI] +10 ; chuyển nội dung 2 ô nhớ liên tiếp có địa chỉ DS: [SI + 10] và DS: [SI+11] vào CX.

MOV CX, [SI+10] ; một cách viết khác của lệnh trên.

MOV CX, 10 [SI] ; một cách viết khác của lệnh đầu.

MOV AL, [DI] +5 ; chuyển nội dung ô nhớ DS:[DI+5] vào AL

Chương 2: Bộ vi xử lý 8086/8088 của Intel

7. Chế độ địa chỉ tương đối chỉ số cơ sở

Kết hợp hai chế độ địa chỉ tương đối chỉ số và tương đối cơ sở ta có chế độ địa chỉ tương đối chỉ số cơ sở. Trong chế độ địa chỉ này ta dùng cả thanh ghi cơ sở lẫn thanh ghi chỉ số và một hằng biểu diễn sự dịch chuyển của địa chỉ để tính địa chỉ của toán hạng. Ta có thể thấy chế độ địa chỉ này rất phù hợp cho việc địa chỉ hoá các mảng hai chiều.

Chương 2: Bộ vi xử lý 8086/8088 của Intel

Ví dụ:

MOV AX, [BX]+ [SI]+8 ;chuyển 2 ô nhớ liên tiếp có địa chỉ DS:[BX+SI+8] và DS:[BX+SI+9] vào AX

MOV AX, [BX+SI+8] ; một cách viết khác của lệnh trên

MOV CL, [BP+DI+5] ; chuyển ô nhớ SS:[BP+DI+5] vào CL

Chương 2: Bộ vi xử lý 8086/8088 của Intel

IV. Cách mã hóa lệnh của 8088

- Các lệnh của 8088 thường được mô tả theo các ký tự dưới dạng gợi nhớ
- Các lệnh mà bộ vi xử lý hiểu được đó là các số 0 hoặc 1
- Cần phải xây dựng cách mã hóa các lệnh mô tả thành các lệnh mã máy (0, 1)

Chương 2: Bộ vi xử lý 8086/8088 của Intel

Ví dụ:

MOV – *Move*

CMP – *Comparision*

JMP – *Jump*

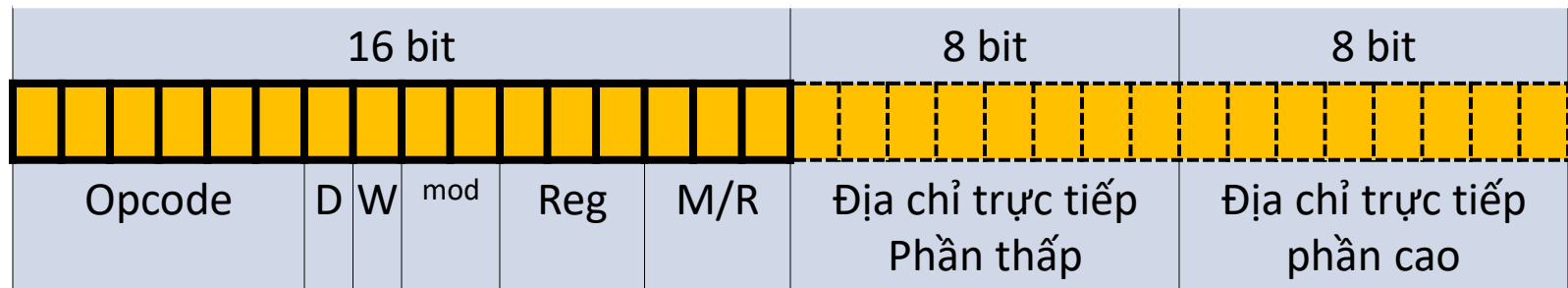
ADD – *Addition*

.....

Trong phạm vi chương trình ta xem xét cách mã hóa lệnh sau: MOV *đích, gốc*

Chương 2: Bộ vi xử lý 8086/8088 của Intel

Để mã hóa lệnh này 8088 cần tối thiểu 2 Byte, tối đa 4 Byte có dạng:



Trong đó:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- Opcode: MOV = 100010
- D (*Direction*):
- W (*Word*):
- Reg (*Register*): mã hóa thành ghi, nơi dữ liệu được chuyển đến hay chuyển đi

Bảng mã hóa REG:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

Thanh ghi		REG
W=0	W=1	
AL	AX	000
CL	CX	001
DL	DX	010
BL	BX	011
AH	SP	100
CH	BP	101
DH	SI	110
BH	DI	111

Chương 2: Bộ vi xử lý 8086/8088 của Intel

- MOD và M/R kết hợp với nhau tạo ra bảng gồm các chế độ địa chỉ như sau:

Chương 2: Bộ vi xử lý 8086/8088 của Intel

	00	01	10	11	
				W=0	W=1
000	[BX] + [SI]	[BX] + [SI] + d8	[BX] + [SI] + d16	AL	AX
001	[BX] + [DI]	[BX] + [DI] + d8	[BX] + [DI] + d16	CL	CX
010	[BP] + [SI]	[BP] + [SI] + d8	[BP] + [SI] + d16	DL	DX
011	[BP] + [DI]	[BP] + [DI] + d8	[BP] + [DI] + d16	BL	BX
100	[SI]	[SI] + d8	[SI] + d16	AH	SP
101	[DI]	[DI] + d8	[DI] + d16	CH	BP
110	D16	[BP] + d8	[BP] + d16	DH	SI
111	[BX]	[BX] + d8	[BX] + d16	BH	DI

Chương 2: Bộ vi xử lý 8086/8088 của Intel

Trong đó:

- d8, d16:
- Các cột có mod = 00, 01, 10:
- Cột có mod = 11:

Ví dụ:

Hãy giải thích và mã hóa các lệnh sau: