

CHƯƠNG V: CÁC LỆNH NHÂN VÀ CHIA

5.1 Các lệnh nhân

5.1.1 Lệnh MUL (Multiply)

Lệnh này dùng để thực hiện phép nhân đối với các số không dấu.

Cú pháp lệnh:

MUL <Thừa số 1>

<Thừa số 1>: là một thanh ghi hay một biến

+ Nếu <Thừa số 1> có kích thước 1 byte thì <Thừa số 2> sẽ là thanh ghi AL. Lệnh trên sẽ thực hiện phép nhân giữa <Thừa số 1> và <Thừa số 2>, kết quả phép nhân được chứa trong thanh ghi AX (16 bit).

+ Nếu <Thừa số 1> có kích thước 1 word thì <Thừa số 2> sẽ là thanh ghi AX. Kết quả phép nhân được chứa trong hai thanh ghi DX:AX (32 bit).

Ví dụ:

Hãy thực hiện phép nhân hai số: 51 và 5

Giải:

Cách 1:

MOV AL, 51

MOV BL, 5

MUL BL

Kết quả: Tích = AX = 255 = 00FFh (16 bit)

Cách 2:

MOV AX, 51

MOV BX, 5

MUL BX

Kết quả: Tích = DX:AX = 255 = 0000 00FFh (32 bit)

Nhận xét:

Cả hai cách trên đều cho cùng một kết quả: Tích = 255 (8 bit: 1111 1111b). Tuy nhiên cách 1 vẫn phải dùng một thanh ghi 16 bit để chứa kết quả này. Cách 2 quá lãng phí tài nguyên vì nó phải dùng tới 2 thanh ghi 16 bit để chứa một giá trị dài 8 bit!

Vấn đề đặt ra là phải xác định được độ dài của kết quả phép nhân nhằm tránh sự lãng phí tài nguyên trong các thao tác tiếp theo. Việc này được thực hiện bằng cách kiểm tra các cờ CF và OF:

- Trường hợp 1: <Thừa số 1> dài 8 bit:
 - + Nếu sau phép nhân 2 cờ CF/OF = 0: Tích được chứa trong AL (8 bit)
 - + Nếu sau phép nhân 2 cờ CF/OF = 1: Tích được chứa trong AX (16 bit)
- Trường hợp 2: <Thừa số 1> dài 16 bit:
 - + Nếu sau phép nhân 2 cờ CF/OF = 0: Tích được chứa trong AX (16 bit)
 - + Nếu sau phép nhân 2 cờ CF/OF = 1: Tích được chứa trong DX:AX (32 bit)

Sau đây là bảng tổng hợp:

Kích thước <Thừa số 1>	Trạng thái cờ CF/OF	Nơi chứa kết quả nhân
8 bit	0	AL
	1	AX
16 bit	0	AX
	1	DX:AX

5.1.2 Lệnh IMUL (Integer Multiply)

Lệnh này dùng để thực hiện phép nhân đối với các số có dấu.

Cú pháp lệnh:

IMUL <Thừa số 1>

<Thừa số 1>: là một thanh ghi hay một biến

Các vấn đề đã trình bày với lệnh MUL ở trên đều có thể áp dụng cho lệnh IMUL.

Ví dụ:

Hãy thực hiện phép nhân hai số: -64 và 2

Giải:

```
MOV AL, 2
MOV BL, -64
IMUL BL
```

Kết quả: Tích = AX = -128 (thực chất chỉ chứa trong AL vì kết quả dài 8 bit = 80h)

5.2 Các lệnh chia (Divide)

5.2.1 Cú pháp lệnh

Lệnh chia cũng có hai dạng để dùng cho các số không dấu và có dấu:

DIV <Số chia> ;Dùng cho số không dấu

IDIV <Số chia> ;Dùng cho số có dấu

<Số chia>: là một thanh ghi hay một biến

+ Nếu <Số chia> có kích thước 1 byte thì Số bị chia sẽ được chứa trong AX (2 byte).

Kết quả: Thương số chứa trong AL, Số dư chứa trong AH.

+ Nếu <Số chia> có kích thước 2 byte thì Số bị chia sẽ được chứa trong DX:AX (4 byte).

Kết quả: Thương số chứa trong AX, Số dư chứa trong DX.

Ví dụ 1:

Hãy thực hiện phép chia 65 cho 2.

Giải:

```
MOV AX, 65
MOV BL, 2
DIV BL
```

Kết quả: Thương = AL = 32

Số dư = AH = 1

Ví dụ 2:

Hãy thực hiện phép chia -128 cho 4.

Giải:

```
MOV AX, -128
MOV BL, 4
```

IDIV BL

Kết quả: Thương = AL = -32

Số dư = AH = 0

Ví dụ 3:

Hãy thực hiện phép chia -1024 cho 256.

Giải:

Vì số chia bằng 256 (là một số 16 bit) nên số bị chia phải chứa trong DX:AX, tức là DX:AX = -1024.

Nhưng -1024 có thể chứa trọn vẹn trong thanh ghi AX (16 bit), muốn chuyển nó thành số 32 bit cần sử dụng lệnh CWD (Convert Word to Double Word). Lệnh này sẽ chuyển dữ liệu có dấu dạng Word trong AX thành dữ liệu có dấu dài 2 Word trong DX:AX.

MOV AX, -1024

CWD

MOV BX, 256

IDIV BX

Kết quả: Thương = AX = -4

Số dư = DX = 0

5.2.2 Hiện tượng tràn trong phép chia

Ví dụ:

Hãy thực hiện phép chia 512 cho 2.

Giải:

MOV AX, 512

MOV BL, 2

DIV BL

Kết quả phép chia bằng 256, không thể chứa trong thanh ghi AL: Hiện tượng tràn xảy ra. Khi đó hệ thống sẽ đưa ra thông báo: “Divide OverFlow”.