

Práctica 2: El juego de instrucciones

Tipos de instrucciones básicas

Transferencia de datos

Operaciones aritméticas

Operaciones lógicas

Transferencia de control

Interrupciones

Activación de banderas (flags)

- Operadores que devuelven valores
 - **\$**: Devuelve el desplazamiento (offset) de la instrucción o directiva donde se encuentra. Suele usarse para calcular el tamaño de cadenas de caracteres.
 - **OFFSET** y **SEG**: devuelven el desplazamiento y número de segmento de una variable.
 - Ejemplos:

TEXT0		DB	"Hola qué tal"
LONG_TEXT0	DB	\$-TEXT0	


```
mov AX, SEG TEXT0  
mov DX, OFFSET TEXT0
```

Operaciones aritméticas

MUL: multiplica el operando por **AX** (operando de 8 bits) o el par **DX:AX** (operando de 16 bits).

MUL operando

IMUL: multiplica con signo.

IMUL operando

DIV: divide **AX** (operando de 8 bits) o **DX:AX** (operando de 16 bits) por el operando sin signo.

Cociente en **AL** y resto en **AH** (operando de 8 bits).

Cociente en **AX** y resto en **DX** (operando de 16 bits).

DIV operando

IDIV: divide con signo.

IDIV operando

Imul: multiplicar con signo

Syntax:

imul op8

imul op16

op8: 8-bit register or memory

imul BL

imul TABLA[9] (elementos tipo DB)

op16: 16-bit register or memory

imul BX

imul TABLA[8] (elementos tipo DW)

Acción:

Si operando es op8, signed $AX = AL * op8$

Si operando es op16, signed $DX::AX = AX * op16$

Flags Affected: OF, SF=?, ZF=?, AF=?, PF=?, CF

Idiv: dividir con signo

Syntax:

idiv op8

idiv op16

op8: 8-bit register or memory

idiv bl

idiv tabla[10] (definido como DB)

op16: 16-bit register or memory

idiv BX.

idiv value (variable definida como DW)

Acción:

Si operando es op8, signed $AL = AX / op8$ (cociente) y $AH = AX \% op8$ (resto)

Si operando es op16, signed $AX = DX::AX / op16$ y $DX = DX::AX \% op16$ (resto)

Interrupciones

Las interrupciones son llamadas a rutinas del sistema (normalmente servicios del BIOS o del SO).

Estas rutinas están “residentes” en memoria.

Las posiciones de memoria donde empiezan las rutinas se guardan en una tabla en memoria que se **encuentra al principio de la memoria en DOS**: desde la dirección 0 a la 3FFh.

Cada 4 bytes de esta tabla constituyen un **vector de interrupción** (offset y segmento donde comienza la rutina de servicio a esa interrupción).

Imprimir un caracter en pantalla

INT 21H AH = 2H

Descripcion: La función de esta rutina es visualizar un caracter en pantalla.

Uso: Entrada: AH = 2H

DL contiene el caracter ASCII a imprimir.

Output: Ninguna

Registros afectados: Ninguno

Ejemplo:

```
mov ah, 2 ; Numero de funcion = 2
```

```
mov dl, 'A' ; Caracter a imprimir en ASCII
```

```
int 21h ; Ejecuta la rutina del S.O.
```