

Ensamblaje

Ensamblaje

- Cada instrucción tiene un código de máquina correspondiente.

– Ejemplo:

• MOV AH, 00 10110 100 00000000

| |

cod w reg = AH

w indica ancho: 0 – un byte, 1 – dos bytes

- Byte del modo direccionamiento consiste en:
 - oo: modo de dos bits; 00, 01 y 10 se refieren a localidades de memoria
 - rrr: una referencia de tres bits, un registro
 - mmm: una referencia de tres bits a un registro o a memoria
 - d: califica a rrr si es fuente o destino
 - w: 0 para uso de un byte y 1 para uso de palabra
 - s: signo (utilizado solo en operaciones aritméticas)

- Ejemplo:

ADD BX, AX

00000011 11 011 000

|| | | |

cod dw oo rrr mmm

d=1, w=1

Sumar AX (000) a BX (011)

- Registros

rrr	w=0	w=1 (16bits)	w=1 (32 bits)
000	AL	AX	EAX
001	CL	CX	ECX
010	DL	DX	EDX
011	BL	BX	EBX
100	AH	SP	ESP
101	CH	BP	EBP
110	DH	SI	ESI
111	BH	DI	EDI

- Asignación del campo de registro para los campos de segmento

rrr	Registro de segmento
000	ES
001	CS
010	SS
011	DS
100	FS
101	GS

- Descripción del campo registro o memoria (mmm) de 16 bits

mmm	Función
000	DS:[BX+S]
001	DS:[BX+DI]
010	SS:[BP+S]
011	SS:[BP+DI]
100	DS:[SI]
101	DS:[DI]
110	SS:[BP]
111	DS:[BX]

- Bits modificadores (oo)

oo	Función
00	Si mmm = 110, desplazamiento sigue al código de operación, si no, no se usa desplazamiento
01	Desplazamiento de 8 bits
10	Desplazamiento de 16 bits
11	mmm especifica un registro en vez de dirección

- Instrucciones dos bytes

ADD AX, BX

```

00000011 11 000 011
  ||      | |   |
cod  dw  oo rrr mmm

```

- Instrucciones de tres bytes

MOV PALABRAMem, AX

```

10100011 dddddddd dddddddd
  ||      |
dw      despl.

```

- Instrucciones de cuatro bytes

LEA DX, mem

```

10001101 00 010 110 d--d d--d
  |      | |   |
LEA      oo rrr mmm despl.

```

- AAA

– 00110111

- AAD

– 11010101 00001010

- ADD

– 000000dw oorrmmm disp

ADD AX, BX

00000011 11 000 011