

Programa en Assembler:

```
ORG 1000H
NUM0 DB 0CAH
NUM1 DB 0
NUM2 DW ?
NUM3 DW 0ABCDH
NUM4 DW ?
```

```
ORG 2000H
MOV BL, NUM0
MOV BH, 0FFH
MOV CH, BL
MOV AX, BX
MOV NUM1, AL
MOV NUM2, 1234H
MOV BX, OFFSET NUM3
MOV DL, [BX]
MOV AX, [BX]
MOV BX, 1006H
MOV WORD PTR [BX], 1006H
HLT
END
```

Lenguaje Máquina:

```
8A 1E 00 10
B7 FF
8A EB
8B C3
88 06 01 10
C7 06 02 10 34 12
BB 04 10
8A 17
8B 07
BB 06 10
C7 07 06 10
F4
```

A) Detalle de cada instrucción:

MOV BL, NUM0: se transfiere un byte de datos desde una dirección de memoria, dada por la etiqueta del segundo operando (num0: 1000h), hacia un registro, dado por el primer operando (BL), reemplazándose su valor anterior. En estos casos, la copia del byte es un modo de direccionamiento **directo de memoria**.

MOV BH, 0FFH: se transfiere el byte dado como segundo operando (FFh) hacia el registro dado por el primer operando (BH), reemplazándose su valor anterior. El modo de direccionamiento es **inmediato**.

MOV CH, BL: se transfiere un byte desde un registro, definido en el segundo operando (BL), para asignarlo en otro definido en el primer operando (CH). El modo de direccionamiento es **directo de registro**.

MOV AX, BX: se transfiere un Word formado por la “parte alta” y “parte baja” del registro dado en el segundo operando (BX: BH y BL), hacia la “parte alta” y “parte baja” de otro dado en el primer operando (AX: AH y AL), respetándose el orden original. El modo de direccionamiento es **directo de registro**.

MOV NUM1, AL: se transfiere un byte desde un registro, dado por el segundo operando (AL), hacia una dirección de memoria, dada por la etiqueta del primer operando (num1: 1001h). MDD **directo de registro**.

MOV NUM2, 1234H: se transfiere un Word dado como segundo operando (1234h) hacia una dirección de memoria, dada por la etiqueta del 1º operando (num2: 1002h), asignando el byte menos significativo en dicha dirección (1002h = 34h), y el byte restante en la siguiente dirección (1003h = 12h). MDD **inmediato**.

MOV BX, OFFSET NUM3: se transfiere un Word definido por la dirección de la etiqueta dada como segundo operando (num3 está ubicado en la dirección 1004h, éste es el valor a copiar), hacia un registro, dado por el primer operando (BX), asignándose el byte más significativo en la “parte alta” (BH = 10h), y el otro byte restante en la “parte baja” (BL = 04h) de dicho registro. Modo de direccionamiento **inmediato**.

MOV DL, [BX]: se transfiere y asigna un byte en un registro, dado por el primer operando (DL), dicho byte se encuentra contenido en la dirección de memoria dada por el valor del registro BX. Este es el caso del modo de direccionamiento **indirecto con registro**, donde la indirección siempre es definida por [BX].

MOV AX, [BX]: se transfiere y asigna un Word en un registro, dado por el primer operando (AX). El byte menos significativo de dicho Word se encuentra contenido en la dirección de memoria dada por el valor del registro BX; y el más significativo se encuentra en la dirección siguiente. En la “parte baja” del registro destino (AL) se asignará el byte por indirección, y en la “parte alta” (AH) el byte de la siguiente dirección. Este caso también se incluye dentro del modo de direccionamiento **indirecto con registro** (sólo [BX]).

MOV BX, 1006H: se transfiere un Word dado como segundo operando (1006h) en un registro, dado por el primer operando (BX), asignándose en la “parte alta” del mismo el byte más significativo (BH = 10h), y en la “parte baja” el otro byte, menos significativo (BL = 06h). El modo de direccionamiento es **inmediato**.

MOV WORD PTR [BX], 1006H: se transfiere un Word dado como segundo operando (1006h) en una dirección de memoria, dado por el Word definido en el registro BX, asignando en la dirección “apuntada” el byte menos significativo del 2º operando (06h), y el más significativo (10h) en la dirección siguiente. Como el valor a copiar se encuentra explícito en la instrucción, el modo de direccionamiento es **inmediato**.

B) Tabla de instrucciones:

Instrucción	Modo de direccionamiento	Valor final de destino
MOV BL, NUM0	Directo de memoria	BL = [1000h] = CAh
MOV BH, OFFH	Inmediato	BH = FFh
MOV CH, BL	Directo de registro	CH = CAh
MOV AX, BX	Directo de registro	AX = FFCAh
MOV NUM1, AL	Directo de registro	[1001h] = CAh
MOV NUM2, 1234h	Inmediato	[1002h] = 34h; [1003h] = 12h
MOV BX, OFFSET NUM3	Inmediato	BX = 1004h
MOV DL, [BX]	Indirecto con registro	DL = CDh
MOV AX, [BX]	Indirecto con registro	AX = ABCDh
MOV BX, 1006H	Inmediato	BX = 1006h
MOV WORD PTR [BX], 1006H	Inmediato	[1006h] = 06h; [1007h] = 10h

C) Registros temporales:

El registro **ri** aparece en el simulador cuando se realiza una transferencia de datos desde / hacia alguna dirección de memoria, como en el caso de la primera instrucción: **MOV BL, NUM0**. En este caso, se guarda un Word con la dirección del operando fuente (num0: 1000h). Otro caso similar sucede en la instrucción: **MOV NUM1, AL**. Se guarda un Word con la dirección, pero ahora es el operando destino (num1: 1001h).

En cada instrucción del lenguaje máquina, los dos primeros bytes indican al procesador qué tipo de acción se debe realizar. Para los casos mencionados, el segundo byte indica cuál es el registro fuente o destino, y los otros bytes restantes forman un Word que representa la dirección de memoria en Little Endian.

Como **IP** apunta siempre a la siguiente instrucción a ejecutar, y el operando referido a la dirección de memoria ocupa más de un byte, es necesario contar con este registro **ri** para direccionar el **bus de datos** en los **MOV registro-memoria**, **MOV memoria-registro** y también en el **MDD indirecto con registro [BX]**.

Los registros **ri** e **id** aparecen en el simulador cuando se realiza un MDD inmediato en memoria, donde **ri** contiene la dirección de memoria (primer operando), e **id** el valor a asignar si se trata de un Word, también representado en Little Endian. En las instrucciones **MOV NUM2, 1234H** y **MOV WORD PTR [BX], 1006H** aparecen ambos (siempre primero **ri**), para almacenar la dirección de memoria y el Word a asignar.

Si la dirección de memoria podría representarse con un único byte, el registro **ri** no sería necesario, así como también el registro **id** no es utilizado cuando se asigna sólo un byte: **MOV BYTE PTR [BX], OP2**.