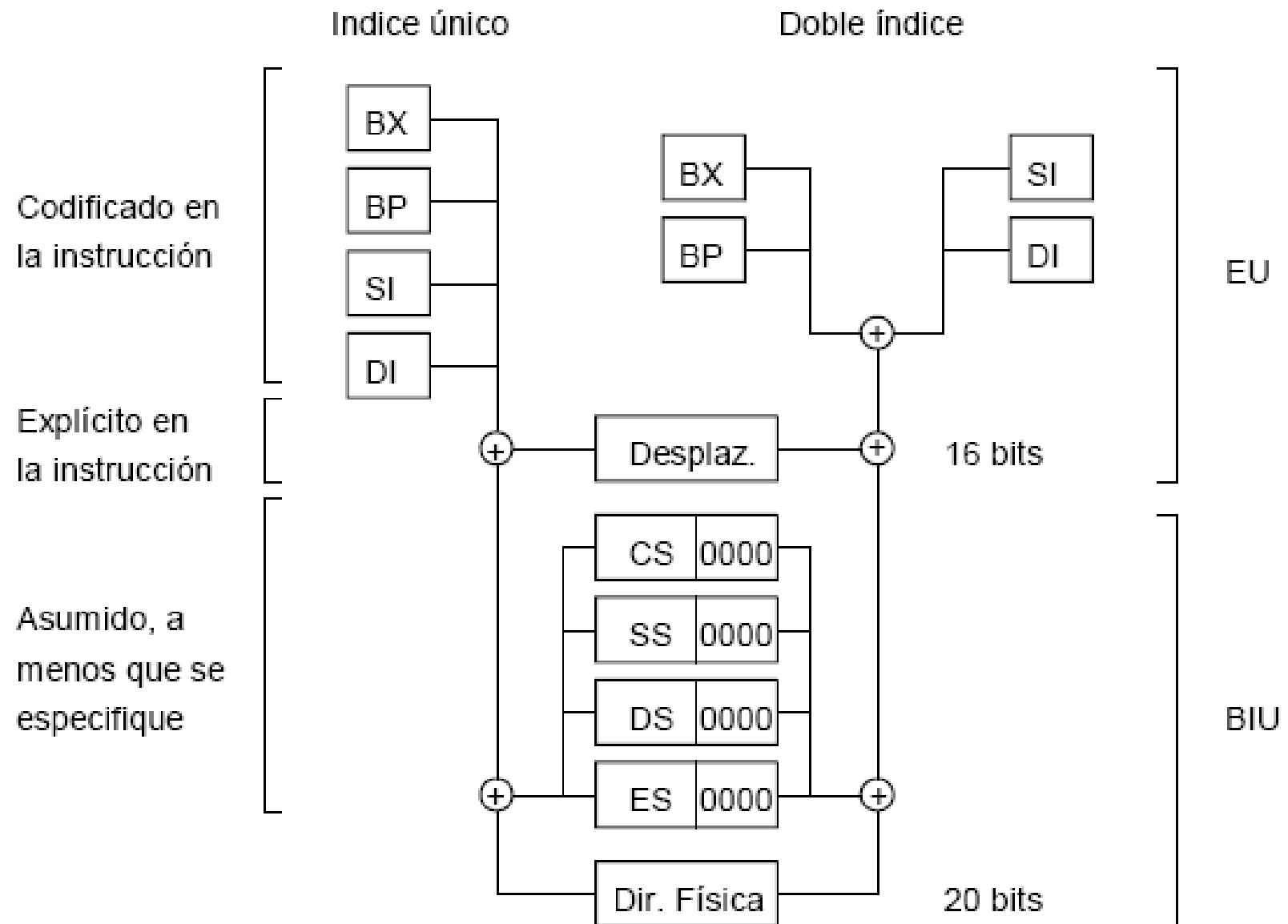


# MODOS DE DIRECCIONAMIENTO

- Tipos de Operandos:
  - La instrucción misma
  - En registros
  - En memoria\*
  - En puertos (ports) de I/O\*

\*requieren ciclos de acceso a memoria
- Dirección efectiva:
  - Dirección que la EU calcula para acceder a un operando de memoria



- ***Direccionamiento Inmediato o Registro***

MOV AX,2CH

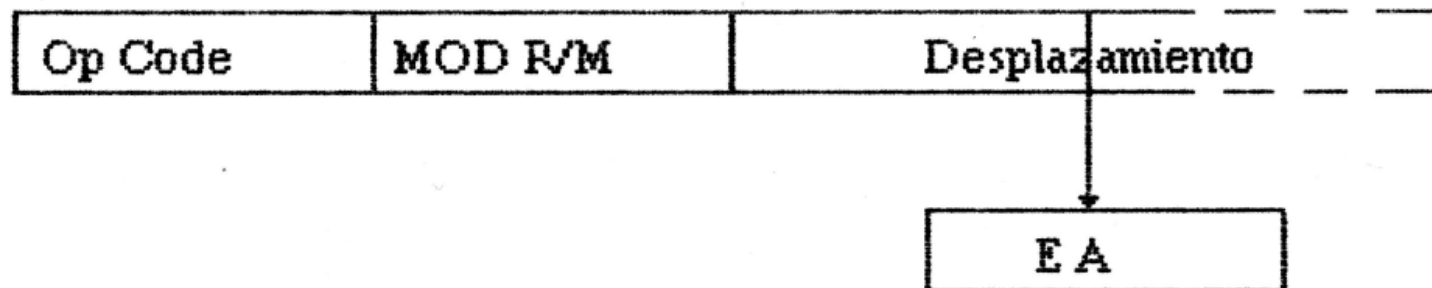
MOV AX,BX

– no hay ciclos de bus

- ***Direccionamiento Directo***

MOV AX,VAR\_1

– VAR\_1 es el desplazamiento dentro del segmento de datos (DS)

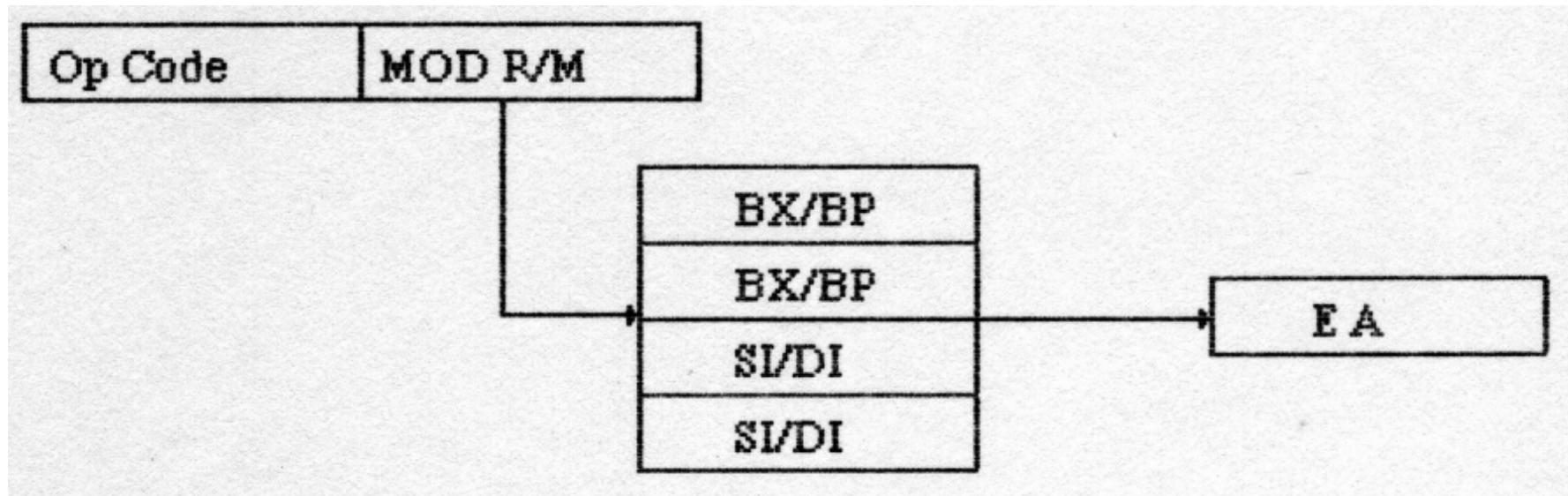


- ***Direccionamiento Indirecto por Registro***

- La EA del operando proviene directamente de uno de los registros base (BX o BP) o índice (DI o SI).

```
MOV AX,[BX]
```

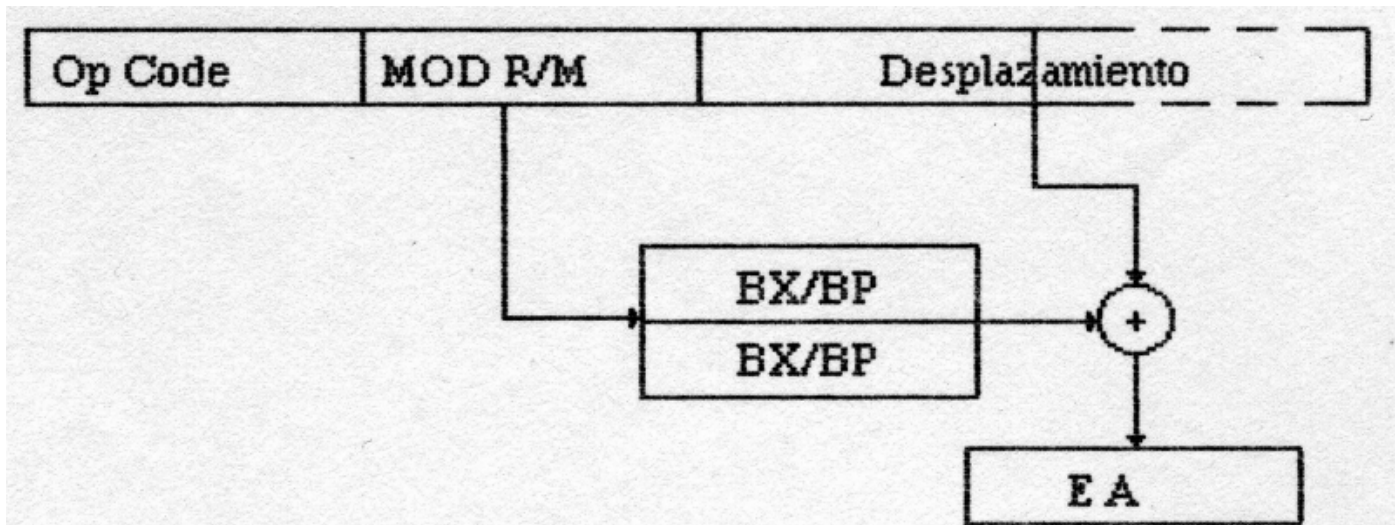
- el lugar de memoria apuntado por BX se carga en AX



- ***Direccionamiento Basado***

- La dirección efectiva se calcula como la suma del desplazamiento, más el contenido de uno de los registros base BX (segmento DS) o BP (segmento SS)
- Usos: Manejo de Estructuras

`MOV AX,[BX + VAR_1] ; AX << (BX+VAR_1)`



- ***Direccionamiento Indexado***

- La dirección efectiva se calcula como la suma del desplazamiento, más el contenido de uno de los registros índices DI (segmento ES) o SI (segmento DS)
- Usos: Manejo de Arreglos

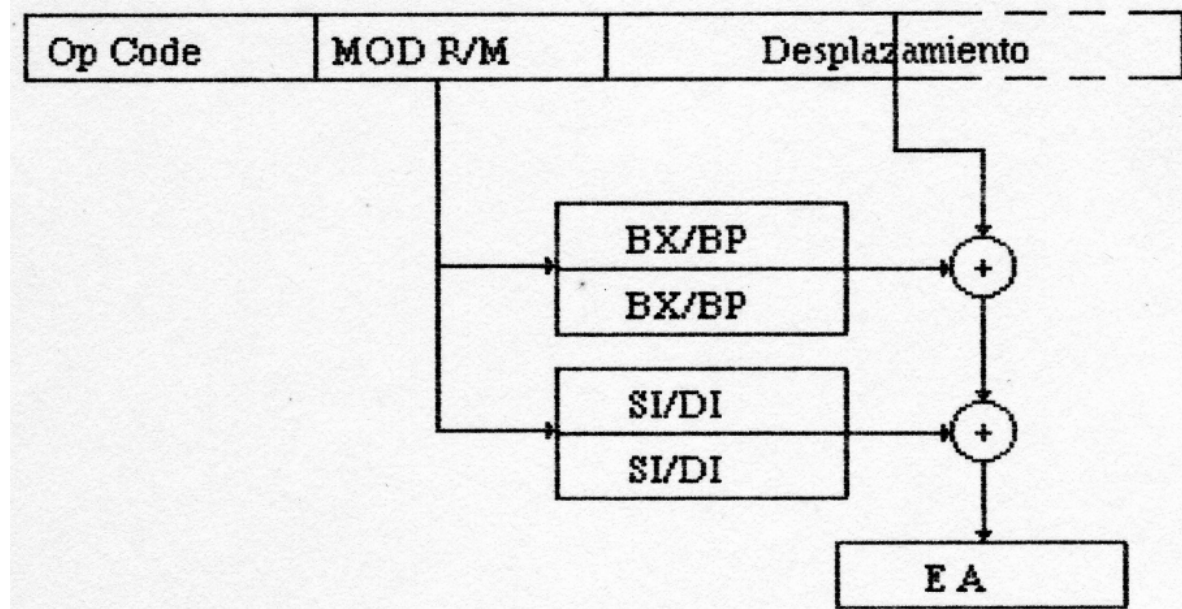
```
MOV AX, [SI + desp_1]
```

- desp\_1 es el offset dentro del segmento, donde comienza el arreglo y los elementos del mismo se acceden variando SI.

- **Direccionamiento Basado-Indexado**

- La dirección efectiva se calcula como la suma de un desplazamiento más el contenido del registro índice (DI o SI), más el contenido de un registro base (BX o BP).
- Usos: Arreglos de dos dimensiones como matrices

**MOV AX, [BX + SI + col\_1]**



- ***Direccionamiento de Cadenas ("strings")***
  - Las instrucciones de manejo de cadenas ("strings"): MOVS, CMPS, SCAS, LODS, STOS, REP, etc.)
  - Usan implícitamente los registros índices fuentes (SI) y destino (DI)
  - Las instrucciones de repetición, automáticamente ajustan a los registros DI y SI.
  - La longitud del arreglo lo determina el contenido de CX.
  - La dirección (creciente o decreciente) la determina el "flag" DF.

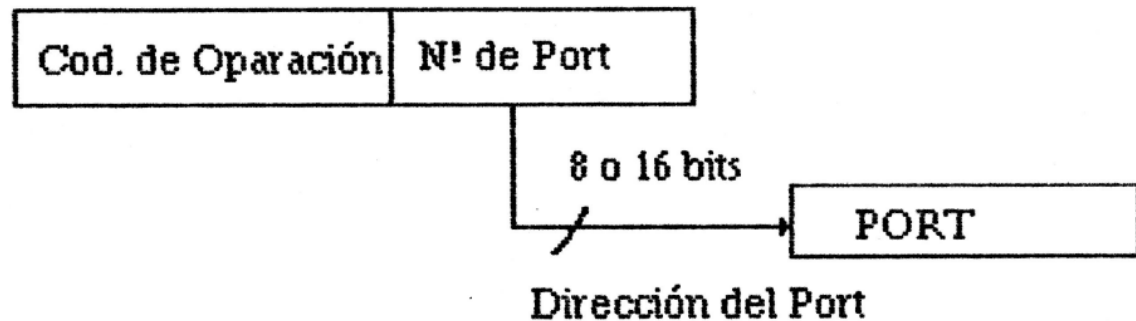


- **Direccionamiento de Dispositivos de I/O**

- Existen dos modos de direccionamiento:

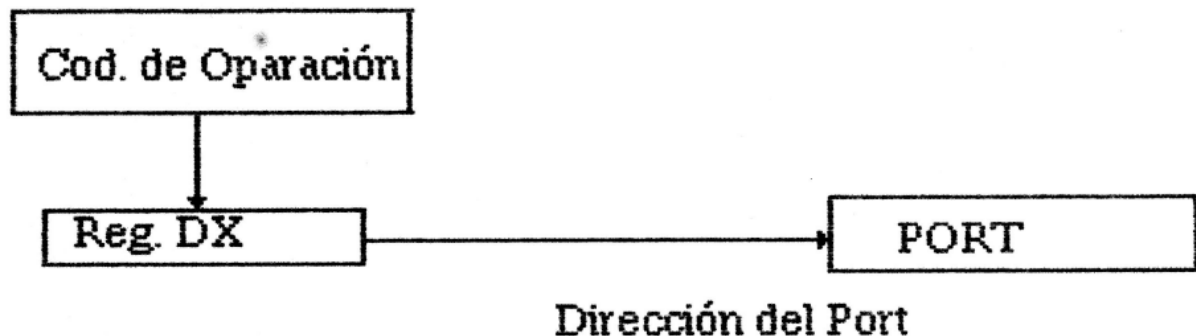
- Direccionamiento del port en forma directa (0 a 255)

**IN AL,PORT**

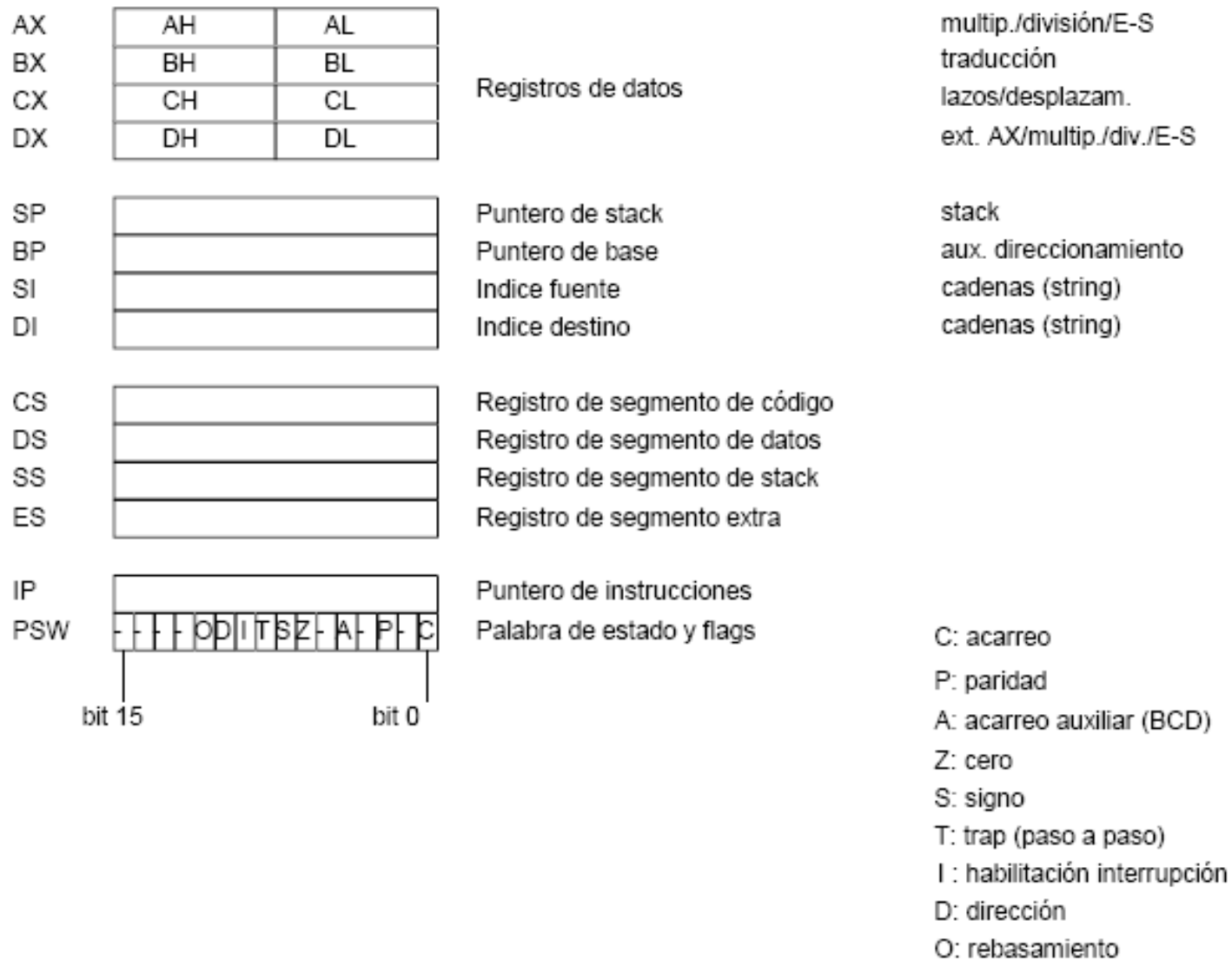


- Direccionamiento del port en forma indirecta

**IN AL,DX**



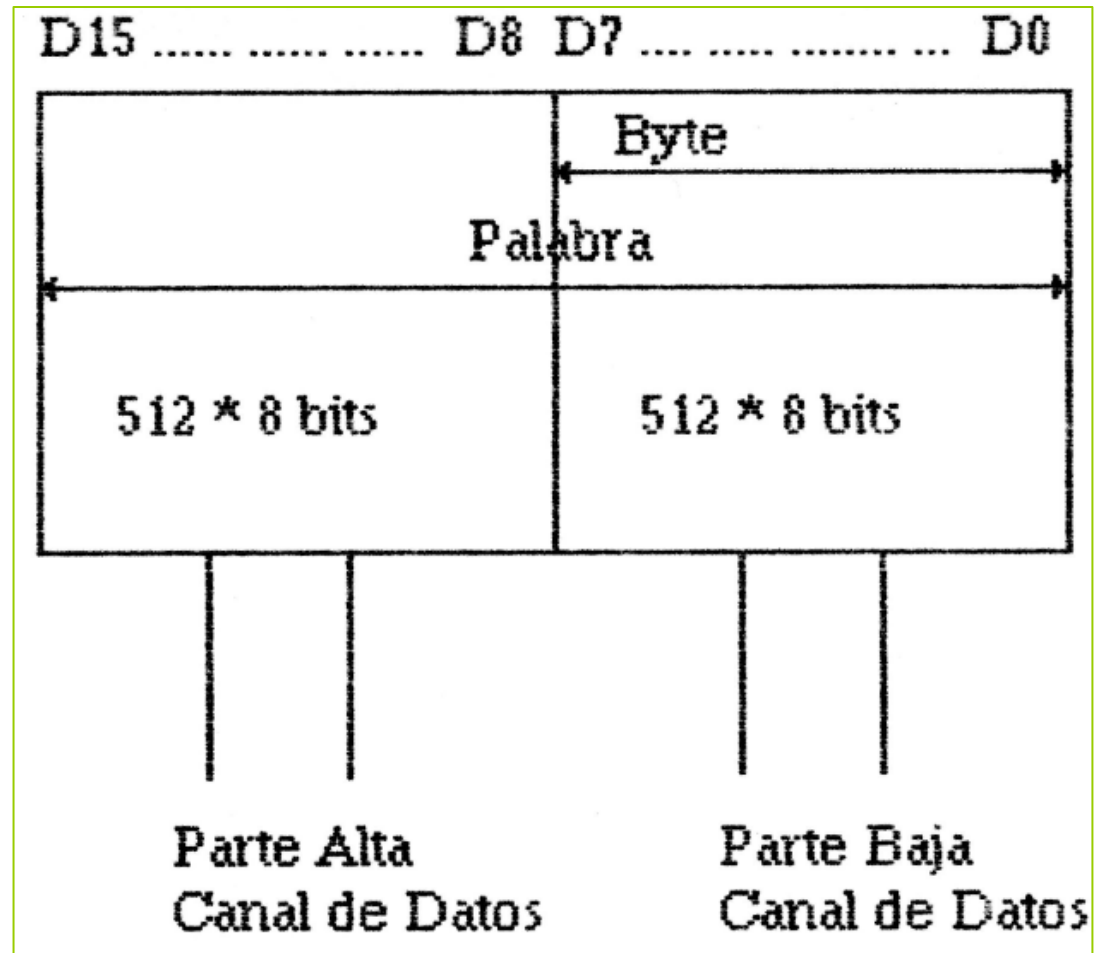
# Modelo de Programación



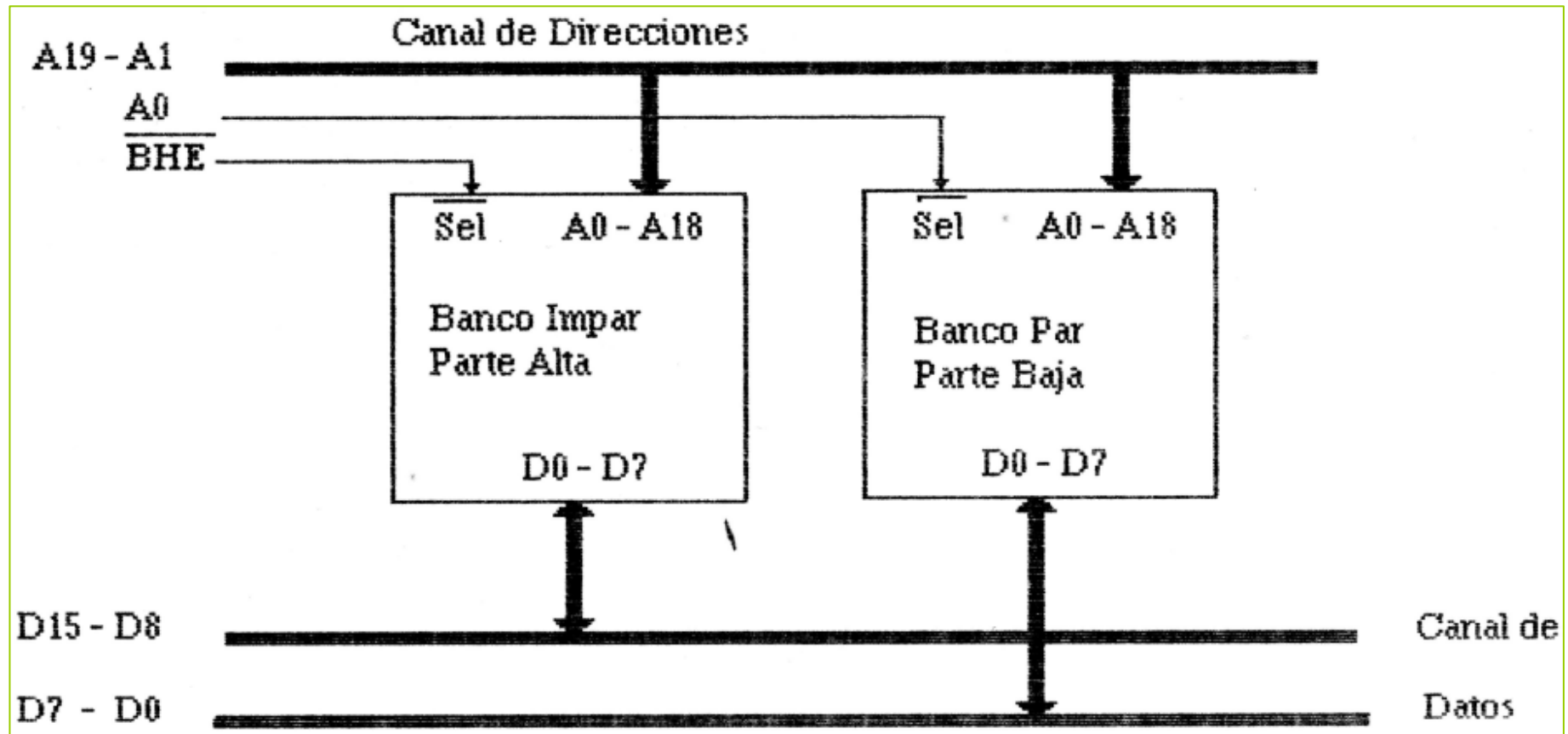
# Organización de Memoria

- El 8086 tiene la capacidad de direccionar un Mbyte de memoria física (20 líneas de direcciones A0 - A19).
- La memoria se organiza como 2 bancos de 512 Kbyte cada uno, correspondiendo uno a las direcciones pares y el otro a las impares.
- El banco par está asociada con la parte baja del canal de datos (D0...D7) y el impar con la parte alta del canal de datos (D8...D15).

# Organización de Memoria



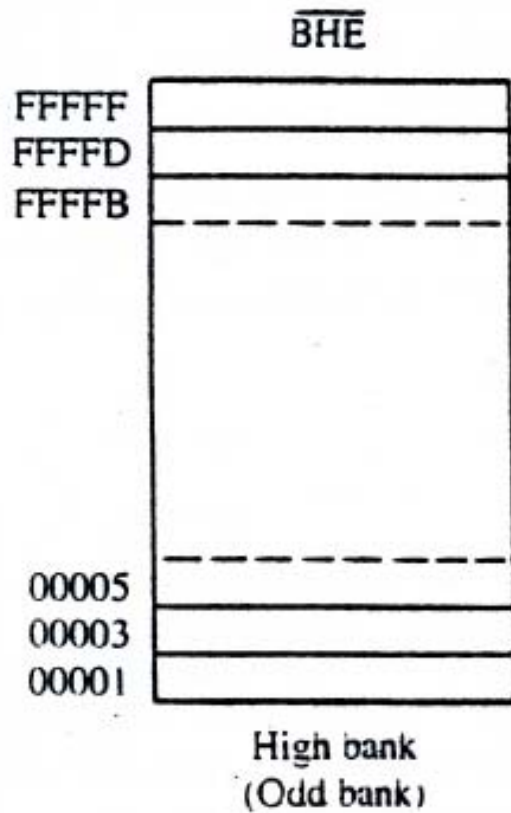
# Organización de Memoria



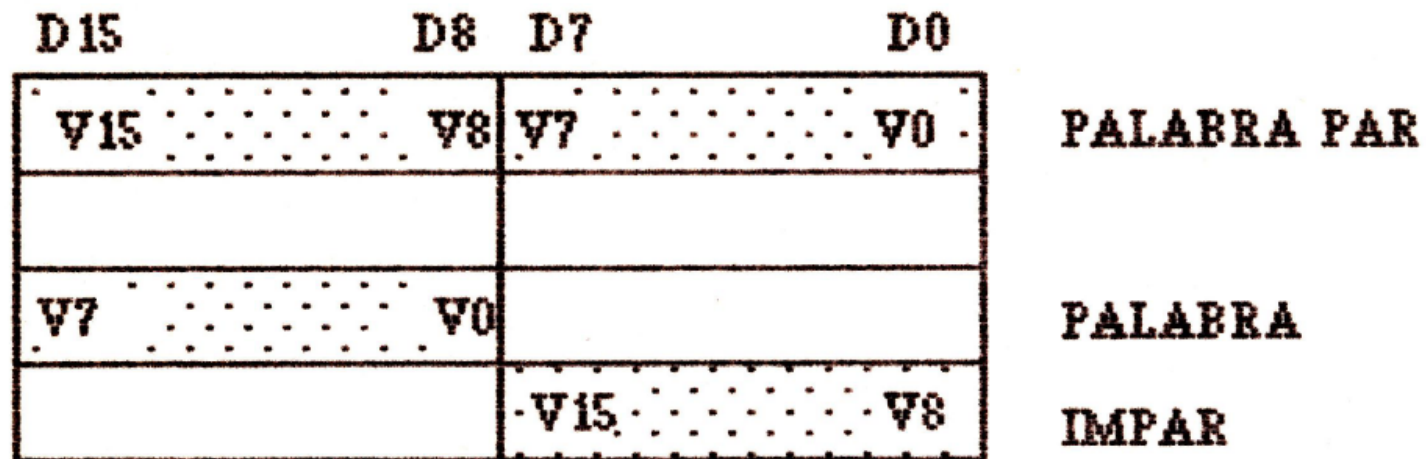
# Organización de Memoria

<b>BHE</b>	<b>A0</b>	<b>Byte Transferido</b>
0	0	palabra (16 bits)
0	1	byte alto (8bits)
1	0	byte bajo (8bits)
1	1	ningún acceso

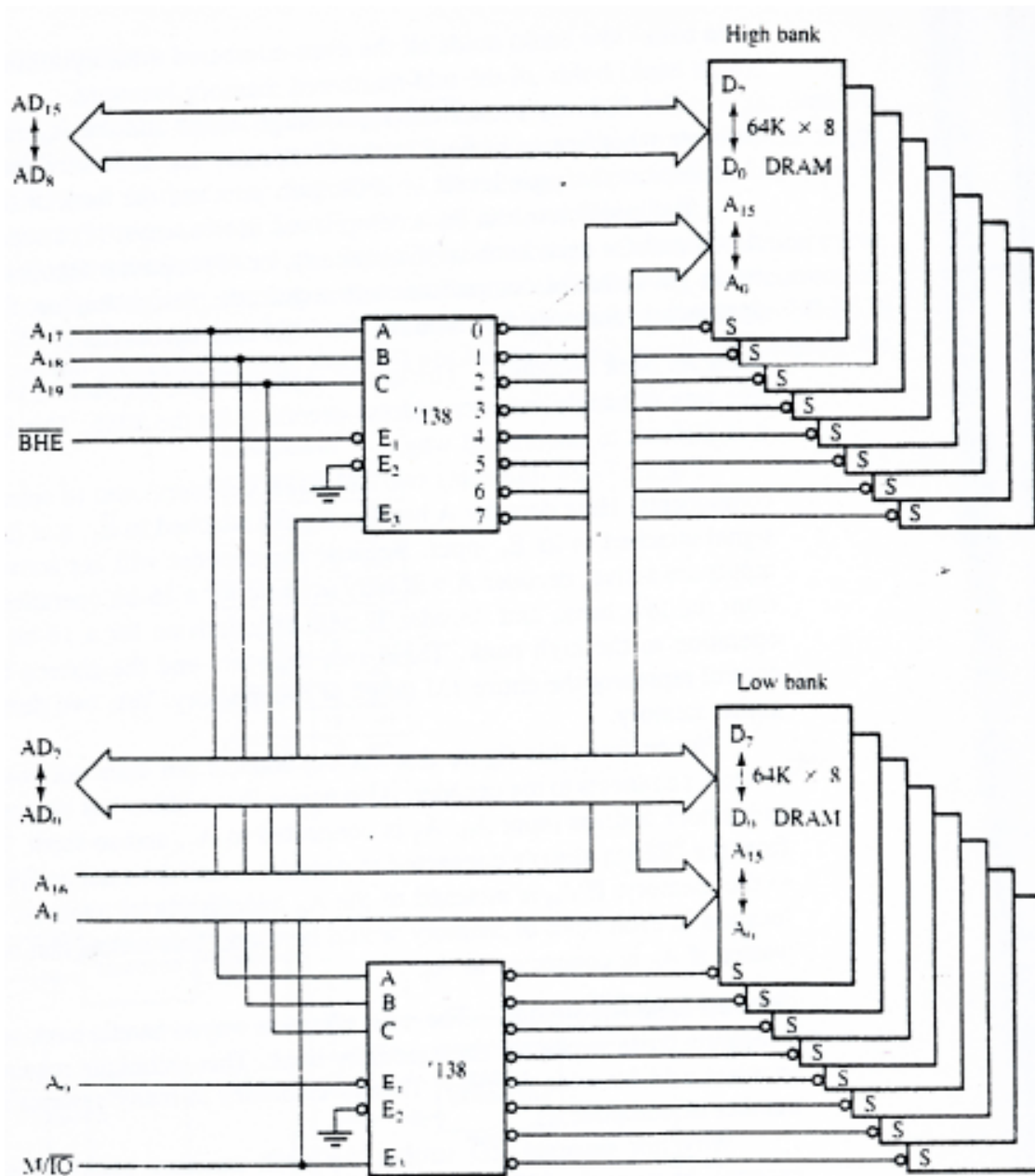
# Organización de Memoria



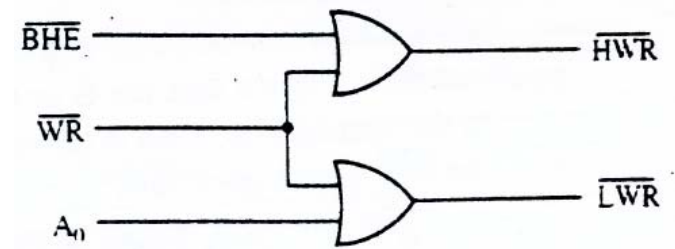
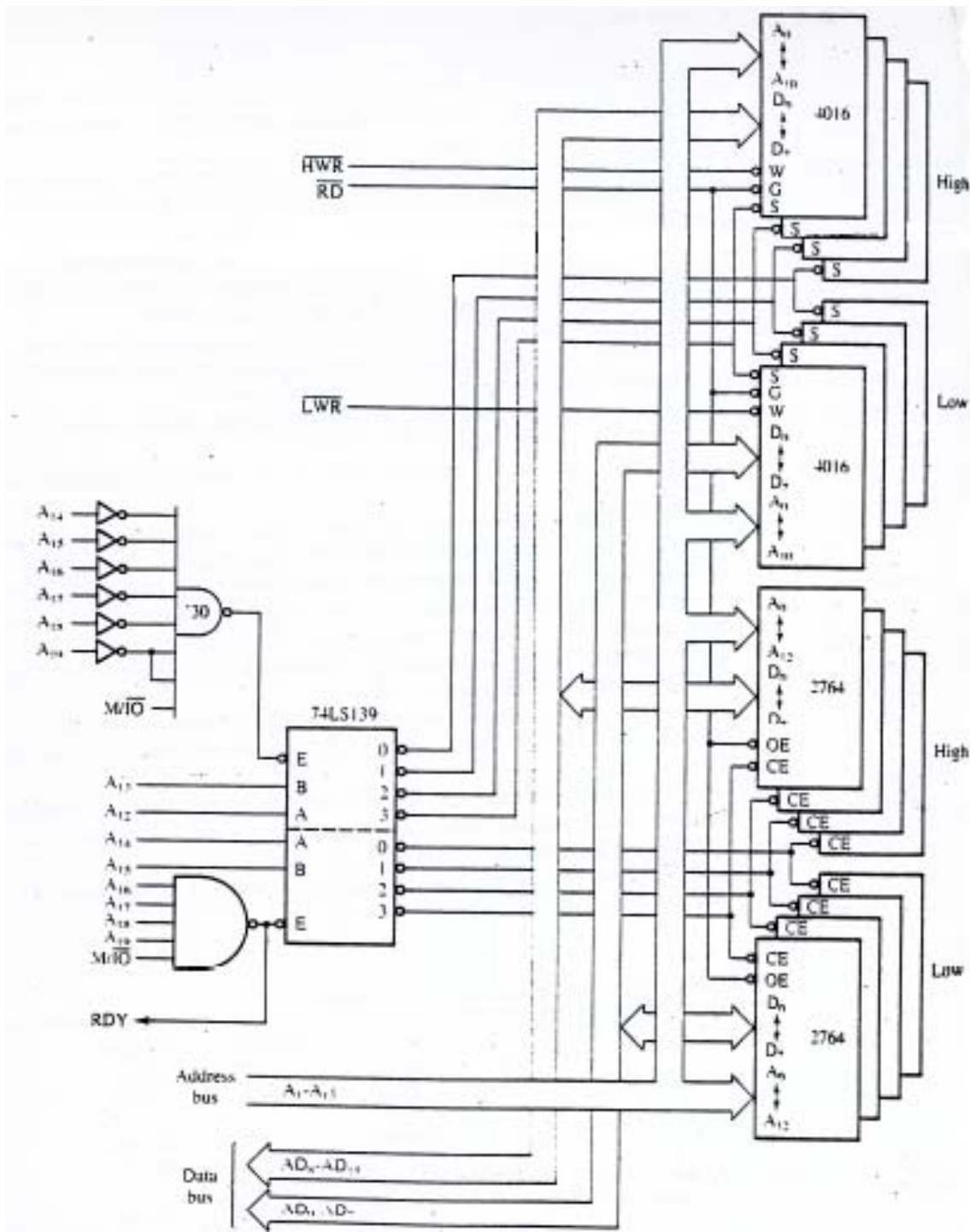
- El 8086 puede acceder a datos de 8 o 16 bits (byte o word).
- Una palabra (16bits) almacena los 2 bytes en direcciones consecutivas de memoria.
- Un byte reside en una dirección par y el otro en una impar.
- Si el byte correspondiente a la parte baja de la palabra (v0...v7) reside en una dirección par: **palabra alineada**
- Si el byte correspondiente a la parte baja de la palabra (v0...v7) reside en una dirección impar: **palabra no alineada**







1 Mbyte RAM Dinámica



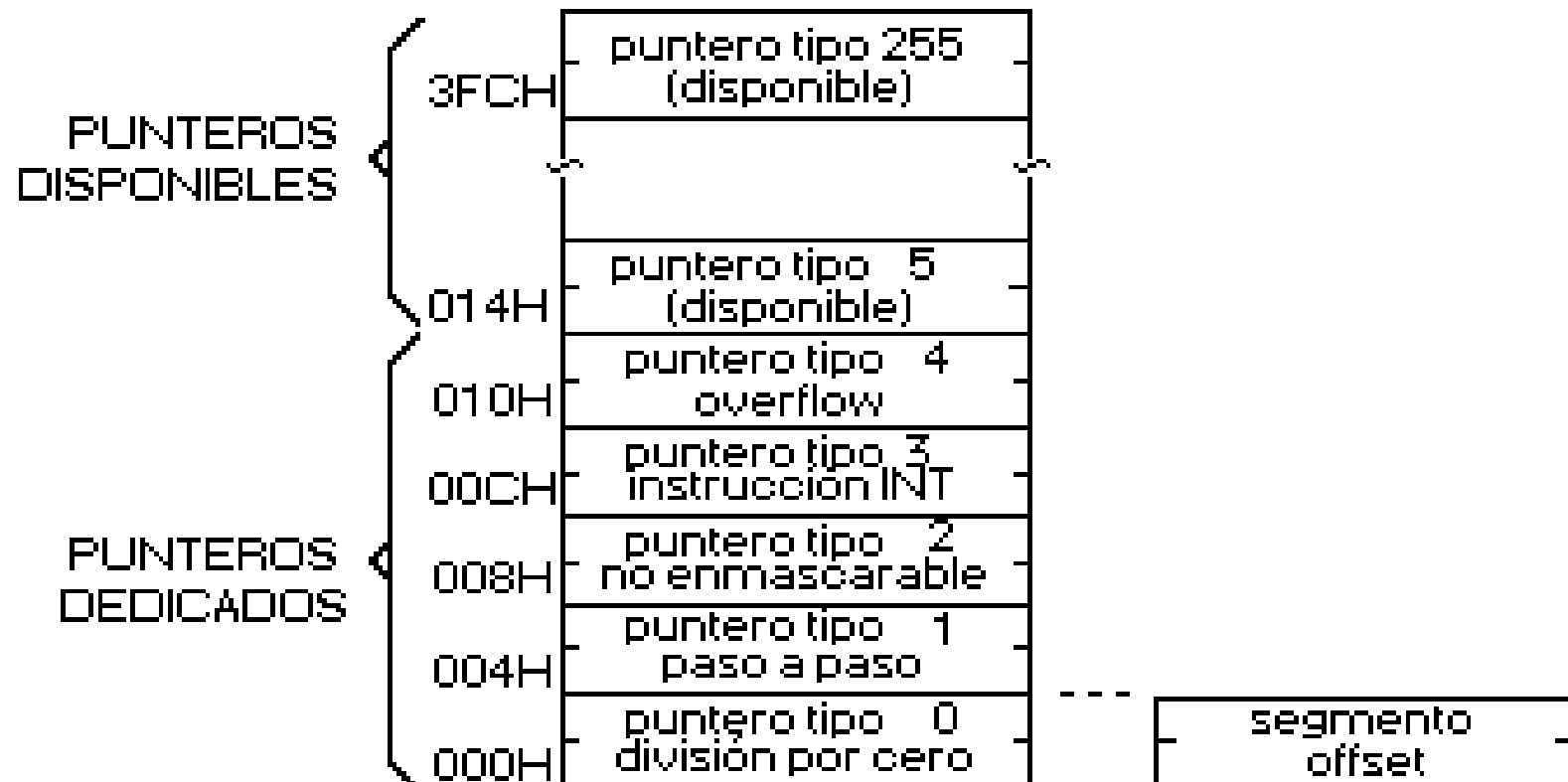
Memoria	Dirección
EPROM	F0000H - FFFFFH
RAM	00000H - 03FFFH

# Interrupciones

- Existen 3 fuentes de interrupciones:
  - por dispositivos externos a la CPU
  - por instrucciones que al ser ejecutadas disparan una secuencia de interrupción
  - interrupciones disparadas por la EU bajo ciertas condiciones
- A cada interrupción se le asigna un código **tipo** que le permite a la CPU identificarla.
- La cantidad máxima de tipos: 256 (0 a 255).
- Con el tipo de interrupción se accede a un lugar de memoria especialmente reservado para la tabla de vectores de interrupción.
- El vector está formado por el **segmento** y el **offset** donde el mismo comienza

# Interrupciones

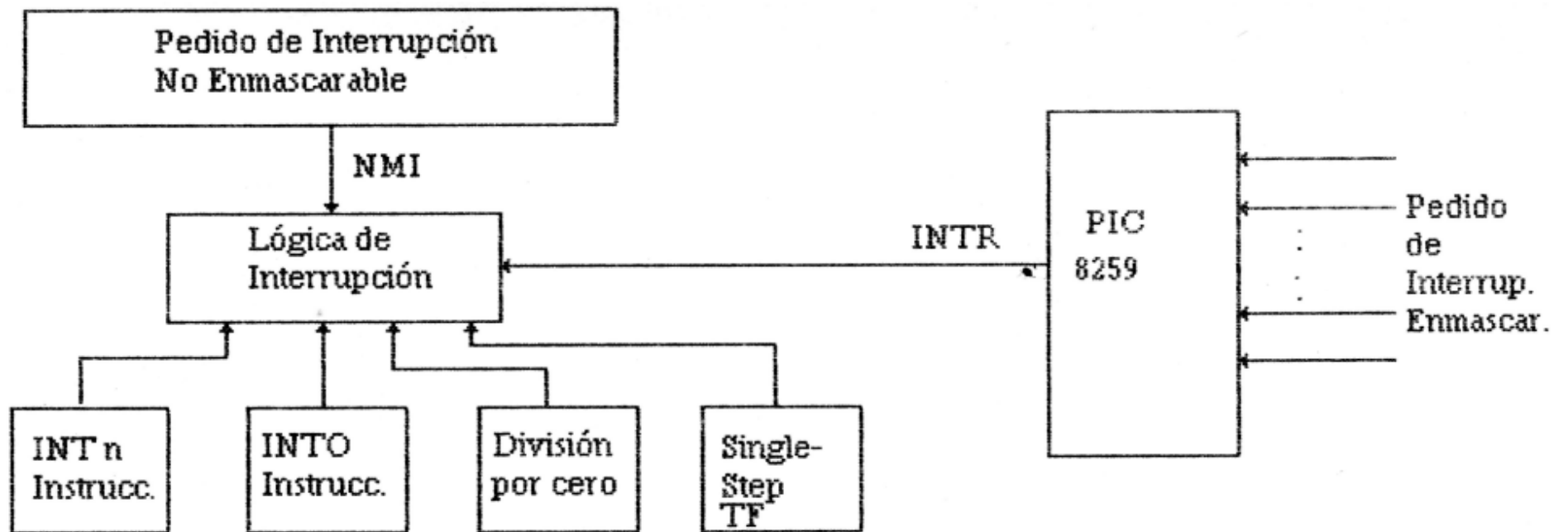
La CPU calcula el punto de entrada correcto a la tabla simplemente multiplicando por 4 al tipo de interrupción.



# Interrupciones

- En respuesta a una interrupción:
  - Se almacena en la pila los contenidos de los registros flags, CS e IP
  - Se pone en cero a los flags IF e TF
  - Carga a los registros CS e IP con el valor hallado en la tabla de punteros de interrupción

# Fuentes de Interrupción



# Fuentes de Interrupción Externas

- Dispone de 2 entradas: **INTR** (Interrupt Request) y la línea **NMI** (non-maskable interrupt)
- **INTR**: bit IF de habilitación, le permite enmascarar (IF=0) o desenmascarar (IF=1) dicha interrupción. (CLI o STI)
- La CPU reconoce el pedido emitiendo 2 ciclos **INTA**
- **NMI**: es una interrupción no enmascarable.
- Tiene asignado internamente el tipo 2 de interrupción.

# Fuentes de Interrupción Internas

- Una interrupción por software (**INT "tipo"**) genera una interrupción al finalizar su ejecución.
- El "tipo" codificado en la instrucción, es utilizado por la CPU para acceder a la tabla de vectores de interrupción.
- Los valores de "tipo" están comprendidos entre 0 y 255.
- Permite simular por software una interrupción externa.
- Las interrupciones por software no pueden ser deshabilitada y las mismas tienen mayor prioridad que las externas.
- El procedimiento de interrupción (igual que las externas), debe retornar con una instrucción de retorno de interrupción IRET.



# Interrupciones especiales

- **INTO (interrupt on overflow):** genera una interrupción tipo 4 si se produjo un overflow en una operación aritmética
- **Error en División:** genera en forma automática una interrupción tipo 0, después de la ejecución de una instrucción de división, cuyo cociente exceda la capacidad del operando destino.
- **Paso a Paso:** genera en forma automática una interrupción tipo 1, si el flag TF está en 1, después de cada instrucción. De gran utilidad para la depuración de programas.