UNIVERSIDAD AUTONOMA METROPOLITANA
Casa abierta al tiempo Azcapotzalco

DEPARTAMENTO DE ELECTRONICA

Sistemas Digitales con Microprocesadores

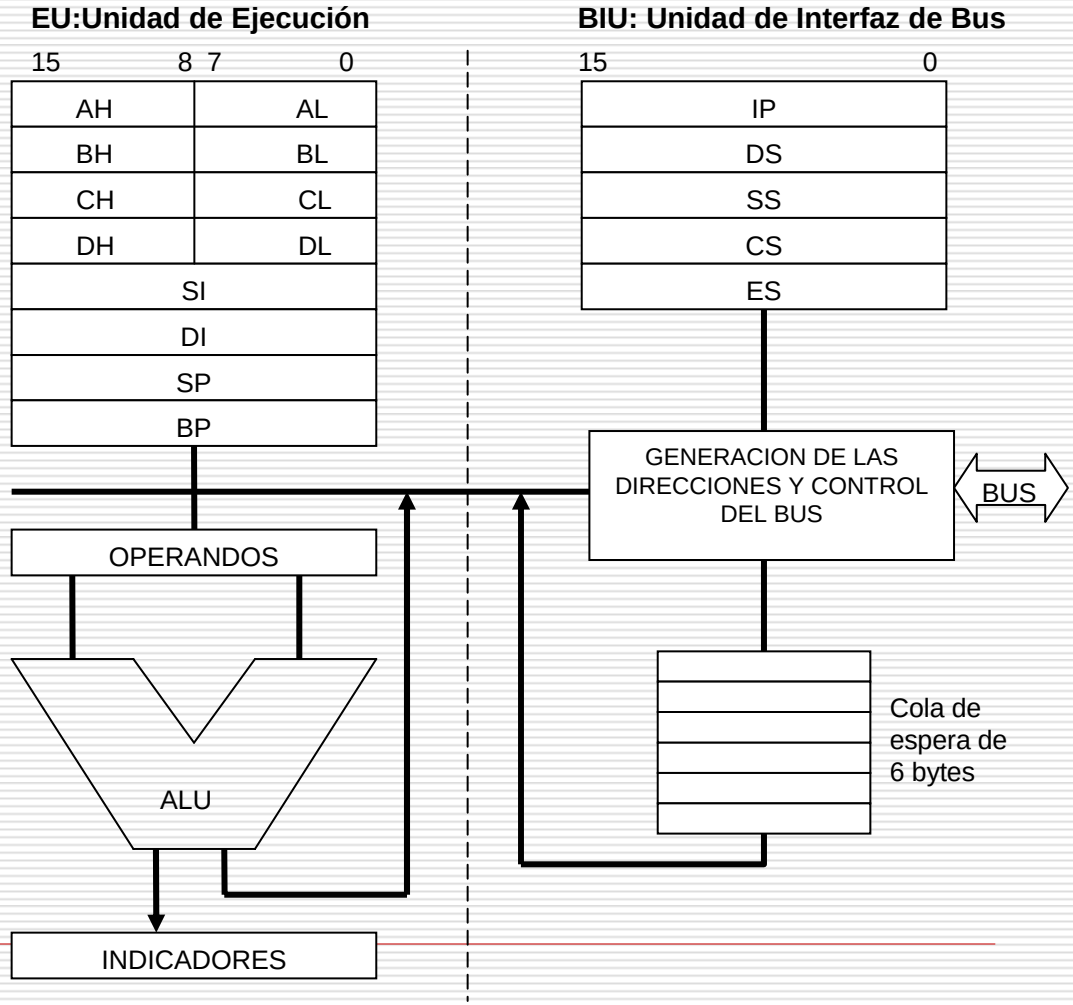
112135

Tema 2. Características del Microprocesador 80X86.

1. Arquitectura básica del microprocesador 8086
   1.1 Arquitectura Interna BIU y EU
2. Especificaciones de hardware
3. Memoria
   3.1 Lineal
   3.2 Segmentada
4. Registros
5. Modos de Direccionamiento
6. Conjunto de instrucciones

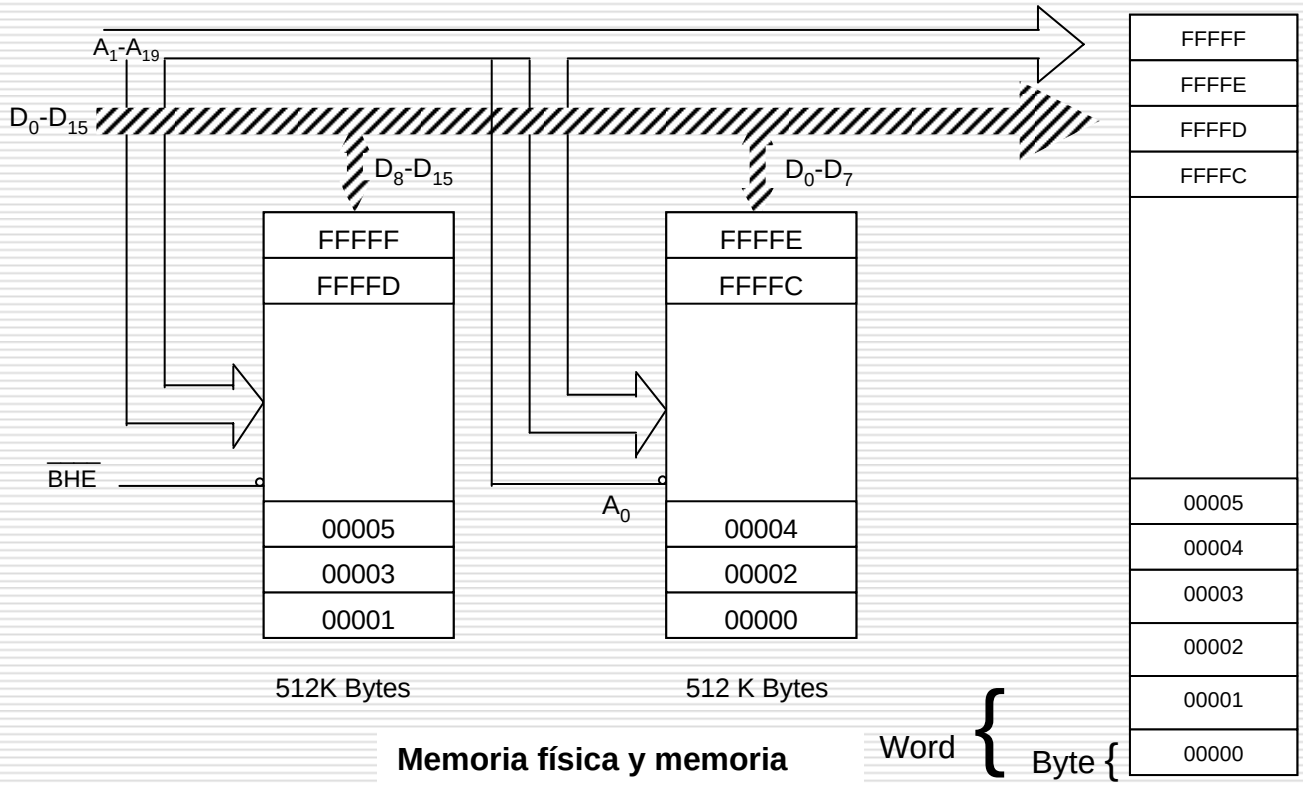
Tema 2. Características del Microprocesador 80X86.
8086 Arquitectura Interna

El microprocesador 8086 se encuentra organizado como dos procesadores separados, la unidad de interfaz de bus (BIU) y la unidad de ejecución (EU).



Tema 2. Características del Microprocesador 80X86.

Memoria Lineal



Memoria física y memoria lógica

Tema 2. Características del Microprocesador 80X86.

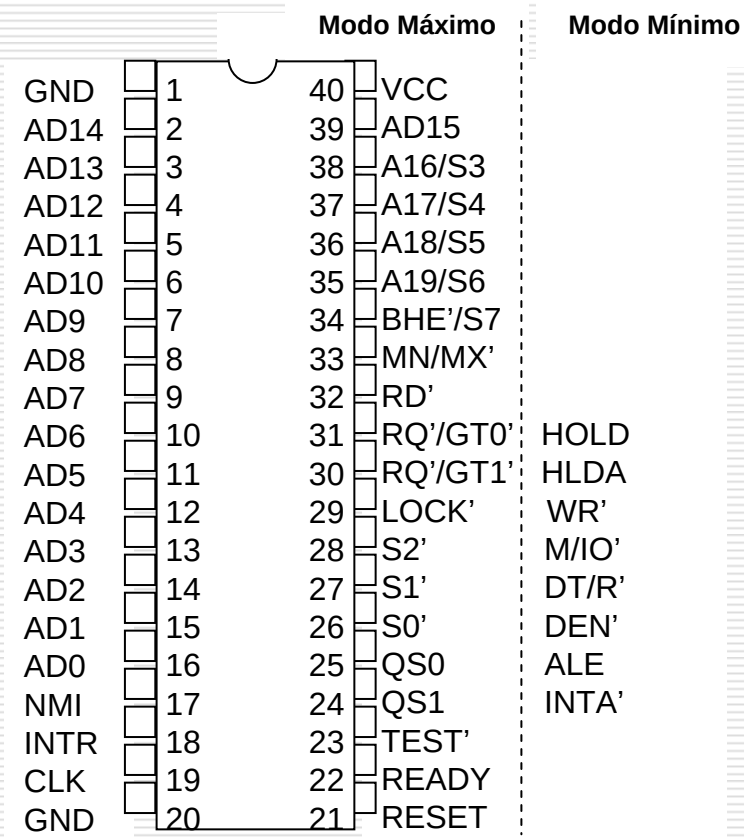
Memoria Lineal

- MEMORIA: El espacio de direcciones de un sistema basado en un microprocesador, se referencia como memoria física o memoria lógica. En la mayoría de los casos la estructura de la memoria lógica es diferente de la estructura de memoria física. La memoria lógica es el sistema de memoria como lo ve el programador, mientras que la memoria física es la estructura de hardware actual del sistema de memoria.
- La **memoria lógica** del 8086 empieza en la localidad de memoria 00000H y se extiende hasta la localidad FFFFFH. Este rango de direcciones especifica el mega byte de memoria disponible.
- **Memoria Física**: Cuando el microprocesador direcciona una palabra de 16 bits de memoria se acceden dos bytes consecutivos. Por ejemplo la palabra de la localidad 00122H se encuentra almacenada en el byte 00122H y 00123H con el byte menos significativo almacenado en la dirección 00122H. Si una doble palabra de 32 bits se almacena en la localidad 00120H esto implica que se almacena en los bytes 00120H, 00121H, 00122H y 00123H con el byte menos significativo almacenado en el byte 00120H y el byte mas significativo en la localidad 00123H.
- La memoria física en el 8086 es de 16 bits de ancho . Se encuentra compuesta por dos bancos de memoria cada uno de 512K bytes. La señal BHE' activa el banco alto (de direcciones nones) y la señal Ao activa el banco bajo (de direcciones pares).

Tema 2. Características del Microprocesador 80X86.

Especificaciones de hardware

- El 8086 es un microprocesador de 16 bits con una capacidad de direccionamiento de memoria de 1 MB (2^{20}) y un espacio separado de puertos de E/S con una capacidad de 64 KB (2^{16}). El CPU se comunica con su ambiente externo a través del bus multiplexado de direcciones, datos y status y un bus de control. Para transferir datos o buscar instrucciones, el CPU ejecuta un ciclo de bus.



Configuración de terminales del 8086

Tema 2. Características del Microprocesador 80X86.
8086 Arquitectura Interna

- La BIU proporciona las funciones de hardware, incluyendo la generación de direcciones de memoria y E/S para la transferencia de datos entre el procesador y el mundo exterior. Lee las instrucciones de la memoria y las almacena en una FIFO (cola de instrucciones) de 6 bytes, hasta que la EU las capte para ejecutarlas. Así la BIU se encarga de transferir los datos entre la memoria (o los puertos) y la CPU, y mientras tanto la EU está procesando una instrucción. La BIU siempre mantiene llena la cola de espera.

- La EU recibe los códigos de instrucción y datos de la BIU, ejecuta esas instrucciones, y almacena los resultados en los registros generales. A través de regresar los datos a la BIU, los datos pueden almacenarse en una localidad de memoria o escritos a un dispositivo de salida. La EU no tiene conexión directa con el sistema de buses. Recibe y transmite todos sus datos a través de la BIU.

Tema 2. Características del Microprocesador 80X86.

Registros del 8086

Registros de Datos. Incluye el acumulador AX y registros BX, CX y DX. Cada registro es de 16 bits pero pueden accederse como registros tamaño byte o palabra. Esto es, BX es el registro base de 16 bits mientras que BH hace referencia al byte de mayor orden del registro base. Los registros de datos normalmente se utilizan para almacenar resultados temporales de instrucciones .

- **AX.**- **Registro acumulador,** dividido en AH y AL (8 bits cada uno). Al usarlo se genera una instrucción que ocupa un byte menos que si se utilizara otro registro de uso general. Su parte más baja, AL, también tiene esa propiedad. EL registro AL es el equivalente al acumulador de los procesadores anteriores (8080 y 8085). Además hay instrucciones como **DAA; DAS; AAA; AAS; AAM; AAD; LAHF; SAHF; CBW; CWD; IN y OUT** que trabajan con AX o con uno de sus dos bytes (AH o AL). También se utiliza este registro junto con DX) en **multiplicaciones** y **divisiones**.
- **BX**.- **Registro base**, dividido en BH y BL.

  Es el registro base y se utiliza para **direccionamiento indirecto**.
- **CX**.- **Registro contador**, dividido en CH y CL.

  Se utiliza como contador en bucles (instrucción **LOOP**), en **operaciones con cadenas** (usando el prefijo **REP**) y en **desplazamientos y rotaciones** (usando el registro CL).
- **DX.**- **Registro de datos**, dividido en DH y DL.

  Se utiliza junto con el registro AX en **multiplicaciones y divisiones**, en la instrucción **CWD y en IN y OUT** para direccionamiento indirecto de puertos (el registro DX indica el número de puerto de entrada/ salida).

Tema 2. Características del Microprocesador 80X86.

Registros del 8086

- **Registros Apuntadores e Indices**.Son únicamente de 16 bits de ancho y no pueden ser accedidos como byte bajo y alto. Se utilizan como apuntadores a memoria. Por ejemplo, la instrucción MOV AH, [SI] se interpreta con palabras como “Mueve el byte cuya dirección está contenida en el registro SI hacia el registro AH”. SI entonces se interpreta como apuntador a la localidad de memoria deseada. Los corchetes alrededor de SI indican una dirección de memoria; esto es, son utilizados para indicar cual es el valor de dirección al que se hace referencia en la instrucción, valor apuntado por SI.

- **SP**.- **Apuntador de pila** (no se puede subdividir).

  Aunque es un registro de uso general, debe utilizarse solo como apuntador de pila, la cual sirve para almacenar las direcciones de retorno de subrutinas y los datos temporales (mediante las instrucciones PUSH y POP). Al introducir (push) un valor en la pila este registro se decrementa en dos, mientras que al extraer (pop) un valor de la pila este registro se incrementa en dos.

Tema 2. Características del Microprocesador 80X86.

Registros del 8086

- **BP.- Apuntador base** (no se puede subdividir).
  Generalmente se utiliza para realizar direccionamiento indirecto dentro de la pila.
- **SI**.- **Apuntador índice** (no se puede subdividir).
  Sirve como apuntador fuente para las operaciones con cadenas. También sirve para realizar direccionamiento indirecto.
- **DI**.- **Apuntador destino** (no se puede subdividir).
  Sirve como apuntador destino para las operaciones con cadenas. También sirve para realizar direccionamiento indirecto.

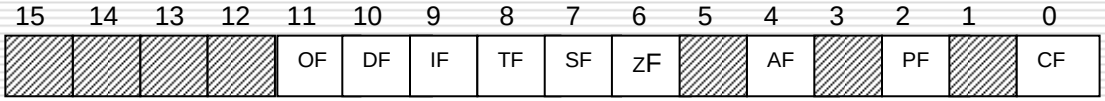
  Cualquiera de estos registros puede utilizarse como fuente o destino en operaciones aritméticas y lógicas.

- El registro IP se incluye en el grupo de apuntadores e índices, pero este registro tiene solo una función –apuntar a la siguiente instrucción a ser buscada por la BIU. El registro IP es físicamente parte de la BIU y no bajo el control directo del programador, como en el caso de los otros registros apuntadores.

Tema 2. Características del Microprocesador 80X86.

Registros del 8086

- **Registro de Banderas**
- **Registro de indicadores (banderas) 16 bits**



- **CF**.- (bandera de acarreo). Si vale 1, indica que hubo acarreo (en caso de suma) o préstamo (en caso de resta) desde el bit de orden más significativo del resultado. Este indicador se utiliza por instrucciones que suman o restan números que ocupan varios bytes. Las instrucciones de rotación pueden aislar un bit de la memoria o de un registro poniéndolo en el acarreo.
- **PF**.- (bandera de paridad). Si vale uno, el resultado tiene paridad par, es decir, un número par de bits en 1. Este indicador se puede utilizar para detectar errores de transmisión.

Tema 2. Características del Microprocesador 80X86.

Registros del 8086

- **AF**.- (bandera auxiliar de acarreo). Si vale 1, indica que hubo arrastre o préstamo de nibble (cuatro bits) menos significativo al nibble más significativo. Este indicador se usa con las instrucciones de ajuste decimal.
- **ZF**.- (bandera de cero). Si este indicador vale 1, el resultado de la operación es cero.
- **SF**.- (bandera de signo). Refleja el bit más significativo del resultado. Como los números negativos se representan en la notación de complemento a dos, ese bit representa el signo; 0 si es positivo, si es negativo.
- **OF**.- (bandera de sobreflujo o desbordamiento). Si vale 1, hubo un desborde en una operación aritmética con signo, esto es, un digito significativo debido a que el tamaño del resultado es mayor que el tamaño del destino.

Tema 2. Características del Microprocesador 80X86.

Registros del 8086

- **TF**.- (bandera de trap) Cuando TF vale 1, la CPU automáticamente genera una interrupción interna después de cada instrucción (el control se pasa a una dirección especial previamente definida por el programador), permitiendo inspeccionar los resultados del programa a medida que se ejecuta instrucción por instrucción. Normalmente, se pasa el control a un programa que despliega todos los registros y banderas del CPU. Esto se utiliza para depuración.
- **IF**.- (bandera de interrupción) Cuando IF se activa, la entrada de solicitud de interrupción (enmascarable) externa INTR del 8086 se habilita, esto es, ocurre una interrupción por hardware, el control será transferido a una rutina de servicio de interrupción (ISR). Cuando esta rutina haya terminado, se ejecuta una instrucción IRET (retorno de interrupción y el control será transferido se regreso a la instrucción en el programa principal que se estaba ejecutando cuando la interrupción ocurrió. Las interrupciones internas y la no enmascarable siempre se reconocen independientemente del valor de IF.
- **DF**.- (bandera de dirección). Esta bandera se utiliza con las instrucciones de cadena (string). Cuando DF se activa, el apuntador de memoria de cadena se decrementará automáticamente; si se pone a cero, el apuntador se incrementará.

Tema 2. Características del Microprocesador 80X86.

Registros del 8086

Registros de Segmento.

- El último grupo de registros es el llamado registros de segmento. Estos registros son utilizados por la BIU para determinar la dirección de memoria de salida para el procesador cuando se va a leer o escribir a la unidad de memoria o E/S.

Los registros de segmento se llaman:

- **CS:** Registro de segmento de código.
- **DS:** Registro de segmento de datos.
- **ES:** Registro de segmento extra.
- **SS:** Registro de segmento de pila.

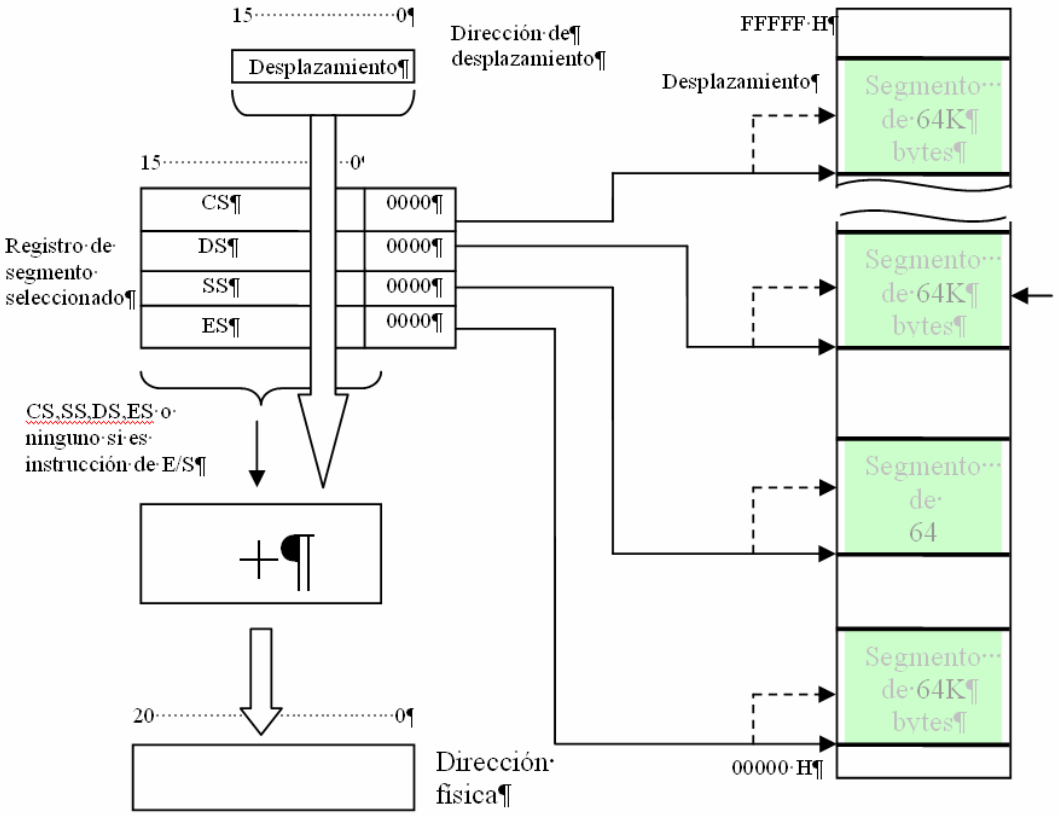
Estos Registros proporcionan la dirección de inicio de cada segmento.

Tema 2. Características del Microprocesador 80X86.

Memoria Segmentada

- La dirección de inicio de cada segmento se obtiene multiplicando por 10H (añadiendo cuatro ceros al final) el contenido del registro de segmento correspondiente y sumando el desplazamiento, lo cual nos da una dirección de 20 bits (5 cifras hexadecimales).

Dir. real = Reg. de segmento *10H + offset



Modelo de Segmentación de Memoria

Tema 2. Características del Microprocesador 80X86.

Memoria Segmentada

- Direcciones del 8086, segmentos por default y desplazamientos

Registro de Segmento	Desplazamiento	Uso
CS	IP	Dirección de instrucción
SS	SP o BP	Dirección de pila
DS	BX, DI (si no es instrucción de cadena), SI, o un número de 16 bits	Dirección de dato
ES	DI para instrucciones de cadena	Dirección de destino de cadenas

Tema 2. Características del Microprocesador 80X86.

Registros del 8086

		Prefijo para cambiar de segmento Si se utiliza otro registro, el ensamblador genera un byte de prefijo correspondiente al segmento antes de la instrucción:			
Mnemónico general Op-code Operando	**Código Objeto**	**Mnemónico**	**Segmento de memoria**	**Operación simbólica**	**Descripción**
CS:	**2E** A1 00 10 **2E** 89 4E 00	MOV AX, CS:MEMWCS[a] MOV CS:[BP],CX	Código Código	AX←CS:[1001H:1000H] CS:[BP]←CX	El segmento default para el operando de memoria fuente o destino es cambiado por el segmento de código.
ES:	**26** A1 00 10 **26** 89 4E 00	MOV AX, ES:MEMWES[a] MOV ES:[BP],CX	Extra Extra	AX←ES:[1001H:1000H] ES:[BP+1:BP]← CX	El segmento default para el operando de memoria fuente o destino es cambiado por el segmento extra.
DS:	**3E** 89 4E 00	MOV DS:[BP],CX	Datos	DS:[BP+1:BP]←CX	El segmento default para el operando de memoria fuente o destino es cambiado por el segmento de datos.
SS:	**36** A1 00 10 **36** 89 0F	MOV AX,SS:MEMWSS[a] MOV SS:[BP],CX	Pila Pila	AX←SS:[1001H:1000H] SS:[BP+1:BP]←CX	El segmento default para el operando de memoria fuente o destino es cambiado por el segmento de pila.

[a]El ensamblador automáticamente genera el cambio de segmento si las palabra de memoria han sido previamente definidas para asignarles otro segmento. En la tabla se asume que cada palabra empieza en la dirección 1000H en el segmento definido.

Tema 2. Características del Microprocesador 80X86.

Modos de Direccionamiento

- Los ***Modos de Direccionamiento*** son las diversas formas con las que se puede indicar a un μP donde debe encontrar o depositar un dato, en una instrucción. Identifica los operandos de la operación a realizar, fuente y destino de los datos sobre los que se operará.
- Los operandos se pueden especificar por: un registro de la CPU, una localidad de memoria, un puerto de Entrada/Salida o un dato inmediato.
- Existen dos grupos principales según los operandos se encuentren en registros o en memoria: modos de *Registro*, y modos de *Memoria*.
- La dirección física de memoria o registro donde se encuentra realmente el operando se llama *Dirección Efectiva*, y se representa *< ea >*. Según el modo de direccionamiento utilizado, la CPU tendrá que realizar unos cálculos distintos hasta obtener el valor de dicha dirección efectiva.
- Las instrucciones pueden contener hasta 2 operandos que se denominan *Operando Fuente*, el del lado derecho, y *Operando Destino*, el del lado izquierdo. En estos casos, cada operando tendrá su propio modo de direccionamiento.

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

- **Direccionamiento Inmediato.**
- Transfiere un byte o palabra de datos inmediato hacia el operando destino. Este modo es usado para inicializar registros o localidades de memoria y para operara sobre ellos con valores constantes de datos.
- Ej: MOV AX,0ABCDH
  MOV BL,12H
- Las instrucciones que usan el modo de direccionamiento inmediato obtienen el dato como parte de la instrucción.
  B8 00 10 MOV AX,1000H
- Este modo no opera con registros de segmento, por lo que no se puede cargar un registro de segmento de manera inmediata.

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

Direccionamiento por Registro.

- Transfiere un byte o palabra desde un registro fuente hasta un registro destino.

- Ej. MOV AX,CX
- INC BX
- El operando no requiere ninguna referencia de memoria.

Nota: Los dos operandos no pueden ser registros de segmento.

- El registro de segmento de código (CS) nunca puede utilizarse como destino.
- No se permite el acceso entre registros de segmento ni de distintos tamaños.

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

- **Direccionamiento Directo.**
- Transfiere un byte o palabra contenido en una localidad de memoria en el segmento DS a un registro de 8 o 16 bits. La localidad de memoria puede ser el operando fuente o destino.
- Ej: **MOV AL,[1234H]**
- En el modo de direccionamiento directo, la dirección de memoria se proporciona directamente como parte de la instrucción (Puede ser a través de etiquetas, en las cuales el programador no necesita conocer la dirección numérica)
- **MOV AH, MEMBDS**
- 8A 00 10 **MOV AH,[MEMBDS]** AH←[1000H]

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

Direccionamiento Indirecto.

- El modo de direccionamiento directo se usa para acceder localidades de memoria de manera no frecuente. Sin embargo cuando una localidad de memoria debe ser leída o escrita varias veces dentro de un programa, la búsqueda repetida de la dirección lógica hace este modo ineficiente. El modo de **direccionamiento indirecto** resuelve este problema almacenando esta dirección de memoria en un registro base (BX,BP)o un registro índice (SI o DI)

- MOV [DI],BH
- MOV [BP],DL

8B 04	MOV AX, [SI]	AL←[SI]; AH←[SI+1]
FF 25	JMP [DI]	IP←[DI+1:DI]
FE 46 00	INC BYTE PTR[BP]	[BP]←[BP] + 1
FF 0F	DEC WORD PTR[BX]b	[BX+1:BX]←[BX+1:BX]-1

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

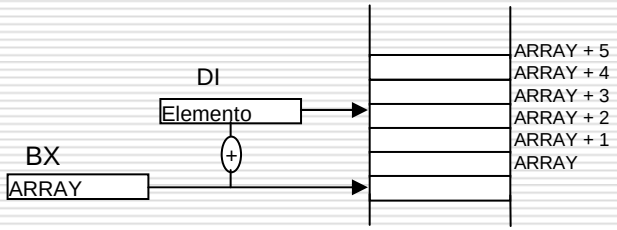
Direccionamiento Base mas Indice.

- Transfiere un byte o palabra entre un registro y una localidad de memoria direccionada por la suma de un registro base más un registro índice.
- Este modo es usado para el acceso a tablas.
- Ej. MOV AX,[BX+DI]
- En muchos casos, el registro base retiene la dirección de inicio de un arreglo de memoria, y el registro índice retiene la posición relativa de un dato en un arreglo.

```
8B 00   MOV AX,[BX+SI]          AH←[BX+SI+1], AL←[BX+SI];
FF 21   JMP [BX+DI]             IP←[BX+DI+1:BX+DI];
FE 02   INC BYTE PTR[BP+SI]     [BP+SI]←[BP+SI]+1  ;
FF 0B   DEC WORD PTR[BP+DI]     [BP+DI+1:BP+DI]←[BP+DI+1:BP+DI]-1
```

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

- Ej. Si se desea direccionar los elementos en un arreglo de datos localizados en el segmento de datos en la localidad ARRAY. Se requiere cargar BX con la dirección ARRAY y DI con el número del elemento del arreglo que se desea acceder.
- MOV BX, OFFSET ARRAY
- MOV DI,3
- MOV AL,[BX+DI]



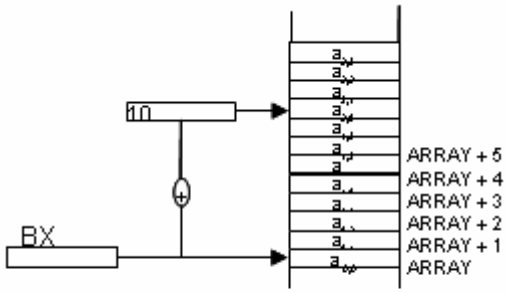
MOV AL,[BX+DI]

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

Direccionamiento Relativo a Registro

- Transfiere un byte o palabra entre un registro y una localidad de memoria direccionada por **un registro** base o un registro índice **más un desplazamiento**.
- Si la localidad de memoria se direcciona por la **suma de un registro base y un desplazamiento**, también se conoce como **direccionamiento basado**.
- Ej. MOV AX,[BX+10H]
- Si la localidad de memoria se direcciona por la **suma de un registro índice y un desplazamiento**, también se conoce como **direccionamiento indexado.**
- Ej. MOV AX,[SI+500H]

  - MOV AX,[BX+4]
  - MOV AX,ARRAY [SI]
  - MOV LIST[BP],CL
  - MOV ARRAY[DI],AL



MOV AL,[BX+10]

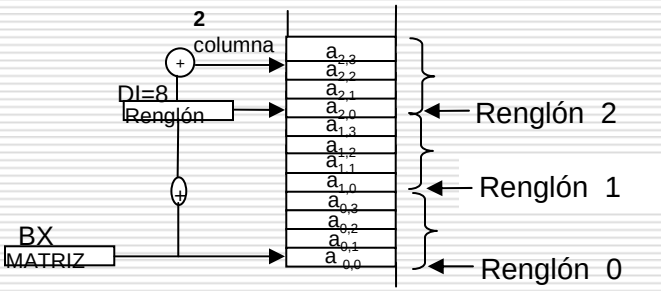
Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

- **Direccionamiento Relativo Base más índice.**
- Transfiere un byte o palabra entre un registro y la localidad de memoria direccionada por un registro base más un registro índice más un desplazamiento.

- Ej. MOV AX, [BX+SI+100H]
- MOV AX, ARRAY[BX+DI]
- MOV LIST[BP+DI],CL

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

Este tipo de direccionamiento es usado comúnmente para direccionar arreglos de datos en memoria de dos dimensiones (matrices).



MOV AL,[BX+DI+2]

Selecciona elemento $a_{2,2}$

MOV BX, OFFSET MATRIZ

MOV DI,8

MOV AL,[BX+DI+2]

$$\begin{pmatrix} a_{0,0} & a_{0,1} & a_{0,2} & a_{0,3} \\ a_{1,0} & a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,0} & a_{2,1} & a_{2,2} & a_{2,3} \end{pmatrix}$$

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

Modo de direccionamiento de string

- En computación un string (cadena) es una secuencia de bytes o palabras almacenadas en memoria. Una tabla de datos es un ejemplo de string. Debido a su importancia el 8086 tiene algunas instrucciones diseñadas específicamente para manejar strings (cadenas) de caracteres.
- Estas instrucciones tienen un modo de direccionamiento especial y usan a DS:SI para apuntar al string fuente y a ES:DI para apuntar al string destino.
- MOVSB mueve el byte del dato fuente a la localidad destino. SI y DI se incrementan o decrementan automáticamente dependiendo del valor de la bandera D.

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

		Instrucciones de Cadena (string)			
Mnemónico general Op-code Operando	**Código Objeto**	**Mnemónico**	**Segmento de memoria**	**Operación simbólica**	**Descripción**
STOSB	AA	STOSB	Extra	ES:[DI]←AL Si DF=0, DI←DI+1 Si DF=1, DI←DI-1	Transfiere un byte o palabra del registro AL o AX a la cadena direccionada por DI en el segmento extra; Si DF=0, incrementa DI, de lo contrario decrementa DI; las banderas no son afectadas.
STOSW	AB	STOSW	Extra	ES:[DI] ← AL ES:[DI+1] ←AH Si DF=0, DI←DI+2 Si DF=1, DI←DI-2	
LODSB	AC	LODSB	Datos	AL← DS:[SI] Si DF=0,SI←SI+1 Si DF=1, SI←SI-1	Transfiere un byte o palabra de la cadena direccionada por SI en el segmento de datos al registro AL o AX; Si DF=0, incrementa SI, de lo contrario decrementa SI; las banderas no son afectadas.
LODSW	AD	LODSW	Datos	AL←DS:[SI] AH←DS:[SI+1] Si DF=0, SI←SI+2 Si DF=1, SI←SI-2	

Tema 2. Características del Microprocesador 80X86.
Modos de Direccionamiento

Direccionamiento	Cod. Ob.	Mnemónico	Segmento	Operación simbólica
Inmediato	B8 00 10	MOV AX,1000H	Código	AH←10H; AL ←00
Registro	8B D1	MOV DX,CX	Dentro del CPU	DX←CX
Directo	8A 00 10	MOV AH,[MEMBDS]	Datos	AH←[1000H]
Indirecto a registro	8B 04 FF 25 FE 46 00 FF 0F	MOV AX, [SI] JMP [DI] INC BYTE PTR[BP] DEC WORD PTR[BX]	Datos Datos Stack Datos	AL←[SI]; AH←[SI+1] IP←[DI+1:DI] [BP]←[BP] + 1 [BX+1:BX]←[BX+1:BX]-1
Indexado	8B 44 06 FF 65 06	MOV AX,[SI+6] JMP [DI+6]	Datos Datos	AL←[SI+6]; AH←[SI+7] IP ← [DI+7:DI+6]
Basado	8B 46 02 FF 67 02	MOV AX,[BP+2] JMP [BX+2]^{c}	Stack Datos	AL←[BP+2]; AH←[BP+3] IP←[BX+3:BX+2]
Base mas índice	8B 00 FF 21 FE 02 FF 0B	MOV AX,[BX+SI] JMP [BX+DI] INC BYTE PTR[BP+SI] DEC WORD PTR[BP+DI]	Datos Datos Stack Stack	AL←[BX+SI]; AH←[BX+SI+1] IP←[BX+DI+1:BX+DI] [BP+SI]←[BP+SI]+1 [BP+DI+1:BP+DI]←[BP+DI+1:BP+DI]-1
Relativo base mas índice	8B 40 05 FF 61 05 FE 42 05 FF 4B 05	MOV AX,[BX+SI+5] JMP [BX+DI+5] INC BYTE PTR[BP+SI+5] DEC WORD PTR[BP+DI+5]	Datos Datos Stack Stack	AL←[BX+SI+5]; AH←[BX+SI+6] IP←[BX+DI+6:BX+DI+5] [BP+SI+5]←[BP+SI+5]+1 [BP+DI+6:BP+DI+5]←[BP+DI+6:BP+DI+5]-1
String	A4	MOVSB	Extra, Datos	ES:[DI]←DS:[SI] Si DF=0, entonces SI←SI+1; DI←DI+1 Si DF=1, entonces SI←SI-1; DI←DI-1

Tema 2. Características del Microprocesador 80X86.

Conjunto de Instrucciones del 8086

Mnemónico	Código Objeto	Mnemónico	Segmento de memoria	Operación simbólica
MOV dest,fuente	8B C3	MOV AX,BX	Dentro del CPU	AX←BX
	8A E3	MOV AH,BL	Dentro del CPU	AH←BL
	A1 00 10	MOV AX,MEMWDS	Datos	AL←[1000H], AH←[1001H]
	A0 02 10	MOV AL,MEMBDS	Datos	AL←[1002H]
	89 1E 00 10	MOV MEMWDS,BX	Datos	[1000H]←BL, [1001H]←BH
	88 1E 02 10	MOV MEMBDS,BL	Datos	[1002H]←BL
	C7 06 00 10 34 12	MOV MEMWDS,1234H	Datos	[1000H]←34H, [1001H]←12H
	C6 06 02 10 34	MOV MEMBDE,34H	Datos	[1002H]←34H
	B0 10	MOV AL,10H	Código	AL←10H
	B8 00 10	MOV AX,1000H	Código	Al←00H, AH←10H
	8E D8	MOV DS,AX	Dentro del CPU	DS←AX
	8C C2	MOV DX,ES	Dentro del CPU	DX←ES
	8E 06 00 10	MOV ES,MEMWDS	Datos	ES←[1001H:1000H]
	8C 0E 00 10	MOV MEMWDS,CS	Datos	[1001H:1000H]←CS

Tema 2. Características del Microprocesador 80X86.

Instrucciones de Transferencia de datos especiales

Mnemónico	Código Objeto	Mnemónico	Segmento de memoria	Operación simbólica
XCHG dest,fuente	93 86 C7 87 14	XCHG AX,BX XCHG AL,BH XCHG [SI],DX	Dentro del CPU Dentro del CPU Datos	AX ⇄ BX AL ⇄ BH [SI] ⇄ DL; [SI+1] ⇄ DH
LAHF	9F	LAHF	Dentro del CPU	AH ← Banderas
SAHF	9E	SAHF	Dentro del CPU	Banderas ← AH
IN acumulador,puerto	E4 26 E5 26 EC ED	IN AL,26H IN AX,26H IN AL,DX IN AX,DX	[b] [b] [b] [b]	AL← puerto 26H AL←puerto 26H, AH←puerto 27H AL←puerto DX AL← puerto DX; AH←puerto DX+1
OUT puerto,acumulador	E6 26 E7 26 EE EF	OUT 26H,AL OUT 26H,AX OUT DX,AL OUT DX,AX	[b] [b] [b] [b]	puerto 26H←AL puerto 26H←AL; puerto 27H←AH puerto DX←AL puerto DX←AL; puerto DX+1 ←AH
LEA dest,fuente[c]	8D 1E 00 10	LEA BX,MEMBDS	Datos	BL←00; BH ←10H
LDS dest,fuente[d] LES dest,fuente[d]	C5 1C C4 1C	LDS BX,DWORD PTR[SI] LES BX,DWORD PTR[SI]	Datos Datos	BL ←[SI]; BH←[SI+1]; DS←[SI+3:SI+2] BL ←[SI]; BH←[SI+1]; ES←[SI+3:SI+2]
XLAT	D7	XLAT	Datos	AL←[BX+AL];

Tema 2. Características del Microprocesador 80X86.

Instrucciones de cadena

Mnemónico	Código Objeto	Mnemónico	Segmento	Operación simbólica
STOSB	AA	STOSB	Extra	ES:[DI]←AL Si DF=0, DI←DI+1 Si DF=1, DI←DI-1
STOSW	AB	STOSW	Extra	ES:[DI] ← AL; ES:[DI+1] ←AH Si DF=0, DI←DI+2 Si DF=1, DI←DI-2
LODSB	AC	LODSB	Datos	AL← DS:[SI] Si DF=0,SI←SI+1 Si DF=1, SI←SI-1
LODSW	AD	LODSW	Datos	AL←DS:[SI]; AH←DS:[SI+1] Si DF=0, SI←SI+2 Si DF=1, SI←SI-2
MOVSB	A4	MOVSB	Datos, Extra	ES:[DI] ← DS:[SI] Si DF=0,DI←DI+1, SI←SI+1 Si DF=1, DI←DI-1, SI←SI-1
MOVSW	A5	MOVSW	Datos, Extra	ES:[DI+1 :DI] ← DS:[SI+1: SI] Si DF=0,DI←DI+2, SI←SI+2 Si DF=1, DI←DI-2, SI←SI-2

Tema 2. Características del Microprocesador 80X86.

Instrucciones de cadena

Mnemónico	Código Objeto	Mnemónico	Segmento	Operación simbólica
SCASB	AE	SCASB	Extra	AL – ES:[DI]; actualiza banderas Si DF=0, DI← DI + 1 Si DF=1, DI← DI - 1
SCASW	AF	SCASW	Extra	AX – ES:[DI+1:DI]; actualiza banderas Si DF=0, DI← DI + 2 Si DF=1, DI← DI - 2
CMPSB	A6	CMPSB	Extra, Datos	DS:[SI] - ES:[DI]; actualiza banderas Si DF=0 DI←DI+1, SI←SI+1 Si DF=1 DI←DI-1, SI←SI-1
CMPSW	A7	CMPSW	Extra, Datos	DS:[SI+1:SI] - ES:[DI+1:DI]; actualiza banderas Si DF=0 DI←DI+2, SI←SI+2 Si DF=1 DI←DI-2, SI←SI-2
Prefijo REP				
REP	F3 AA F3 AB F3 A4 F3 A5	REP STOSB REP STOSW REP MOVSB REP MOVSW	Extra Extra Extra, datos Extra, datos	STOSB; CX← CX – 1 Repite hasta que CX=0 STOSW; CX← CX – 1 Repite hasta que CX=0 MOVSB; CX← CX – 1Repite hasta que CX=0 MOVSW; CX←CX-1Repite hasta que CX=0

Tema 2. Características del Microprocesador 80X86.
Instrucciones de cadena (Prefijo de repetición)

Mnemónico	Código Objeto	Mnemónico	Segmento	Operación simbólica
REP	F3 AA	REP STOSB	Extra	STOSB; CX← CX – 1 Repite hasta que CX=0
	F3 AB	REP STOSW	Extra	STOSW; CX← CX – 1 Repite hasta que CX=0
	F3 A4	REP MOVSB	Extra, datos	MOVSB; CX← CX – 1Repite hasta que CX=0
	F3 A5	REP MOVSW	Extra, datos	MOVSW; CX←CX-1Repite hasta que CX=0
REPE/REPZ	F3 AE	REPZ SCASB	Extra	SCASB; CX ← CX – 1 Repite si ZF = 1 y CX ≠ 0
	F3 AF	REPZ SCASW	Extra	Similar a la anterior excepto SCASW
	F3 A6	REPZ CMPSB	Extra, datos	Similar a la anterior excepto CMPSB
	F3 A7	REPZ CMPSW	Extra, datos	Similar a la anterior excepto CMPSW
REPNE/REPNZ	F2 AE	REPNE SCASB	Extra	SCASB; CX ← CX – 1 Repite si ZF = 0 y CX ≠ 0
	F2 AF	REPNE SCASW	Extra	Similar a la anterior excepto SCASW
	F2 A6	REPNE CMPSB	Extra, datos	Similar a la anterior excepto CMPSB
	F2 A7	REPNE CMPSW	Extra, datos	Similar a la anterior excepto CMPSW

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Corrimiento y Rotación

A←shl A, (S)←0(S)

Mnemónico	Código Objeto	Mnemónico	Segmento de memoria	Operación simbólica
SAL/SHL[b] dest,contador	D1 E0 D3 E0	SAL AX,1 SAL AX,CL	Dentro del CPU Dentro del CPU	
SAR dest, contador	D0 F8 D2 F8	SAR AL,1 SAR AL,CL	Dentro del CPU Dentro del CPU	
SHR dest, contador	D1 2C D2 2C	SHR WORD PTR[SI],1 SHR BYTE PTR[SI],CL	Datos Datos	
RCL dest, contador	D1 D3 D3 D3	RCL BX,1 RCL BX,CL	Dentro del CPU Dentro del CPU	
RCR dest, contador	D0 DB D2 DB	RCR BL,1 RCR BL,CL	Dentro del CPU Dentro del CPU	
ROL dest, contador	D1 04 D2 04	ROL WORD PTR[SI],1 ROL BYTE PTR[SI],CL	Datos Datos	
ROR dest, contador	D1 0E 00 10 D2 0E 04 10	ROR MEMWDS,1 ROR MEMBDS,CL	Datos Datos	

Tema 2. Características del Microprocesador 80X86.
Instrucciones Lógicas

Mnemónico [BX]←	**Código Objeto**	**Mnemónico[a]**	**Segmento de memoria**	**Operación simbólica**
NOT dest	F7 D3 F6 14	NOT BX NOT BYTE PTR[SI]	Dentro del CPU Datos	BX←BX' [SI] ← [SI]'
AND dest, fuente	23 CA 22 3C 25 00 80	AND CX, DX AND BH,BYTE PTR [SI] AND AX, 8000H	Dentro del CPU Datos Código	CX ← CX ^ DX BH ← BH ^ [SI] AX ← AX ^ 8000H
OR dest, fuente	0B CA 0A 3C 0D 00 80	OR CX,DX OR BH, BYTE PTR[SI] OR AX,8000H	Dentro del CPU Datos Código	CX ← CX + DX BH ← BH + [SI] AX ← AX + 8000H
XOR dest, fuente	33 CA 32 3C 35 00 80	XOR CX,DX XOR BH, BYTE PTR[SI] XOR AX,8000H	Dentro del CPU Datos Código	CX ← CX ⊕ DX BH ← BH ⊕ [SI] AX ← AX ⊕ 8000H
TEST dest, fuente	85 D1 84 3C A9 00 80	TEST CX,DX TEST BH, BYTE PTR[SI] TEST AX,8000H	Dentro del CPU Datos Código	CX ^ DX; actualiza banderas BH ^ [SI]; actualiza banderas AX ^ 8000H; actualiza banderas

Tema 2. Características del Microprocesador 80X86.

Instrucciones de Adición y Substracción

Mnemónico	Código Objeto	Mnemónico[a]	Segmento de memoria	Operación simbólica
ADD dest, fuente	03 F2	ADD SI, DX	Dentro del CPU	SI ← SI + DX
	00 2F	ADD BYTE PTR[BX],CH	Datos	[BX]←[BX] + CH
	81 C7 00 80	ADD DI,8000H	Dentro del CPU	DI← DI + 8000H
	81 06 00 10 00 80	ADD MEMWDS,8000H[a]	Datos	[1001H:1000H]← [1001H:1000H]+8000H
ADC dest, fuente	13 F2	ADC SI, DX	Dentro del CPU	SI ← SI + DX + CF
	10 2F	ADC BYTE PTR [BX],CH	Datos	[BX]←[BX]+CH + CF
	81 D7 00 80	ADC DI, 8000H	Dentro del CPU	DI← DI + 8000H + CF
	81 16 00 10 00 80	ADC MEMWDS,8000H[a]	Datos	[1001H:1000H]← [1001H:1000H]+8000H+CF
SUB dest, fuente	2B F2	SUB SI, DX	Dentro del CPU	SI ← SI - DX
	28 2F	SUB BYTE PTR[BX],CH	Datos	[BX]←[BX] - CH
	81 EF 00 80	SUB DI,8000H	Dentro del CPU	DI← DI - 8000H
	81 2E 00 10 00 80	SUB MEMWDS,8000H[a]	Datos	[1001H:1000H]← [1001H:1000H]-8000H
SBB dest, fuente	1B F2	SBB SI, DX	Dentro del CPU	SI ← SI - DX - CF
	18 2F	SBB BYTE PTR [BX],CH	Datos	[BX]←[BX]- CH - CF
	81 DF 00 80	SBB DI, 8000H	Dentro del CPU	DI← DI - 8000H - CF
	81 1E 00 10 00 80	SBB MEMWDS,8000H	Datos	[1001H:1000H]← [1001H:1000H]-8000H-CF

		Ejemplo codificado		
Mnemónico general **Op-code Operando**	**Código Objeto**	**Mnemónico[a]**	**Segmento de memoria**	**Operación simbólica**
INC dest[b]	FE C3 FF 05 FE 06 04 10	INC BL INC WORD PTR[DI] INC MEMBDS[c]	Dentro del CPU Datos Datos	BL←BL+1 [DI+1:DI]←[DI+1:DI]+1 [1004H]← [1004H]+1
DEC dest[b]	FE CB FF 0D FE 0E 04 10	DEC BL DEC WORD PTR[DI] DEC MEMBDS[c]	Dentro del CPU Datos Datos	BL←BL-1 [DI+1:DI]←[DI+1:DI]-1 [1004H]← [1004H]-1
NEG dest[b]	F6 DB F7 1D F6 1E 04 10	NEG BL NEG WORD PTR[DI] NEG MEMBDS[c]	Dentro del CPU Datos Datos	BL←0 - BL [DI+1:DI]← 0 - [DI+1:DI] [1004H]← 0 - [1004H]

CMP dest, fuente	3A C4
	39 0D
	81 3E 00 10 00 80
	81 FF 00 80
		CMP AL,AH
		CMP [DI],CX
		CMP MEMWDS,800H[a]
		CMP DI,8000H
			Dentro del CPU
			Datos
			Datos
			Dentro del CPU
				AL – AH; actualiza banderas
				[DI+1:DI]-CX; actualiza banderas
				[1001H:1000H]-8000H;actualiza banderas
				DI-8000H;actualiza banderas

Tema 2. Características del Microprocesador 80X86.

Instrucciones de Multiplicación y División.

Mnemónico	Código Objeto	Mnemónico	Segmento de memoria	Operación simbólica
MUL fuente	F6 E3 F7 E1 F6 27 F7 26 00 10	MUL BL MUL CX MUL BYTE PTR[BX] MUL MEMWDS	Dentro del CPU Dentro del CPU Datos Datos	AX ← AL * BL DX:AX←AX * CX AX← AL * [BX] DX:AX←AX* [1001H:1000H]
IMUL fuente	F6 EB F7 E9 F6 2F F7 2E 00 10	IMUL BL IMUL CX IMUL BYTE PTR[BX] IMUL MEMWDS	Dentro del CPU Dentro del CPU Datos Datos	AX ← AL * BL (con signo) DX:AX←AX * CX (con signo) AX← AL * [BX] (con signo) DX:AX←AX* [1001H:1000H] (con signo)
DIV	F6 F3 F7 F1 F6 37 F7 36 00 10	DIV BL DIV CX DIV BYTE PTR[BX] DIV MEMWDS	Dentro del CPU Dentro del CPU Datos Datos	AX ← AX / BL DX:AX←DXAX / CX AX← AX / [BX] DX:AX←DXAX / [1001H:1000H]
IDIV	F6 FB F7 F9 F6 3F F7 3E 00 10	IDIV BL IDIV CX IDIV BYTE PTR[BX] IDIV MEMWDS	Dentro del CPU Dentro del CPU Datos Datos	AX ← AX / BL (con signo) DX:AX←DXAX / CX (con signo) AX← AX / [BX] (con signo) DX:AX←DXAX /[1001H:1000H] (con signo)

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Ajuste.

Mnemónico	Código Objeto	Mnemó-nico	Segmento de memoria	Operación simbólica
DAA	27	DAA	Dentro del CPU	Si AL ∧ 0F > 9 o AF = 1, entonces AL←AL+ 6; AF ← 1 Si AL > 9F o CF = 1, entonces AL ← AL + 60H; CF ← 1
DAS	2F	DAS	Dentro del CPU	Si AL ∧ 0F > 9 o AF = 1, entonces AL←AL-6; AF ← 1 Si AL > 9F o CF = 1, entonces AL ← AL - 60H; CF ← 1
AAA	37	AAA	Dentro del CPU	Si AL ∧ 0F > 9 o AF = 1, entonces AL←AL+ 6; AH ← AH + 1; AF← 1; CF ←AF; AL ← AL ∧ 0F
AAS	3F	AAS	Dentro del CPU	Si AL ∧ 0F > 9 o AF = 1, entonces AL←AL- 6; AH ← AH - 1; AF← 1; CF ←AF; AL ← AL ∧ 0F
AAM	D4 0A	AAM	Dentro del CPU	AH ← AL / 0AH AL ← Residuo
AAD	D5 0A	AAD	Dentro del CPU	AL ← (AH * 0AH) + AL AH ← 0
CBW	98	CBW	Dentro del CPU	Si AL < 80H, entonces AH ← 0 Si AL > 7F, entonces AH ← FFH
CWD	99	CWD	Dentro del CPU	Si AX < 8000H, entonces DX← 0 Si AX > 7FFFFH, entonces DX ← FFFFH

Instrucciones de Ajuste.

- DAA (decimal adjust after addition)

```
mov al,35h
add al,48h;  AL=7DH
daa          ;  AL=83H
```

- DAS (decimal adjust after substraction)

```
mov bl,48h
mov al,85h   ;
sub al, bl   ;AL=3DH
das          ;AL=37H
```

Instrucciones de Ajuste.

- **AAA(ASCII adjust after addition)**

```
mov ah,0
mov al,'8'    ; AX=0038H
add al,'2'    ;AX=006AH
aaa           ;AX=0100H
or ax,3030h ; AX=3130H='10'
```

- **AAS (ASCII adjust after substraction)** Ajusta el resultado binario obtenido de una instrucción SUB o SBB. Provoca que el resultado en AL sea consistente con una representación ascii. Por ejemplo, el siguiente fragmento de código resta el ascii 9 del ascii 8, en donde después de la instrucción SUB, AX es igual a 00FFh (-1) y la instrucción AAS convierte AX a FF09, el complemento a 10 de -1.

```
Datos segment
        val1 db '8'
        val2 db '9'
Datos ends
Codigo segment 'code'
            ⋮
        mov ah,0
        mov al,val1      ;AX=0038H
        sub  al,val2     ;AX=00FFH
        aas              ;AX=FF09H
```

Instrucciones de Ajuste.

AAM (ASCII adjust after multiplication)

```
Datos segment
    ascVal db 05h,06h
Datos ends
Codigo segment
        :
    mov bl,ascVal ;primer operando
    mov al,ascVal ;segundo operando
    mul bl        ;AX=001EH
    aam           ;AX=0300H
    or ax,3030h ;AX=3330H='30'
```

AAD (ASCII adjust before division)

```
Datos segment
    cociente  db ?
    residuo   db ?
Datos ends
Codigo segment
        :
    mov ax,0307H  ;dividendo (BCD )
    aad           ;AX=0025H (AH*10+AL)
    mov bl,05     ;divisor
    div bl        ;AX=0207H
    mov cociente,al
    mov residuo,ah
```

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Salto.

Mnemónico	Código Objeto	Mnemónico	Segmento	Operación simbólica
JMP near etiqueta	E9 -- -- FF 26 00 10 FF 27 FF E0	JMP MEMN JMP [MEMWDS] JMP [BX] JMP AX	Código Datos Datos Dentro del CPU	IP← MEMN IP←[MEMWDS+1:MEMWDS] IP←[BX+1:BX] IP←AX
JMP SHORT etiqueta	EB --	JMP SHORT MEMS	Código	IP←MEMS
JMP far etiqueta	EA 03 00 D3 9E FF 2E 05 10 FF 2F	JMP FAR PTR MEMF JMP [MEMWWDS] JMP DWORD PTR[BX]	Código Datos Datos	IP←0003H; CS-←9ED3H IP←[1006H:1005H]; CS←[1008H:1007H] IP←[BX+1:BX]; CS←[BX+3:BX+2]
Jcond^{h} short etiqueta	73 --	JNC MEMS	Código	Si CF=0, entonces IP←MEMS
JCXZ short etiqueta	E3 --	JCXZ MEMS	Código	Si CX=0, entonces IP←MEMS

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Salto Condicional.

Saltos Condicionales para números con signo y sin signo		
Mnemónico	**Significado**	**Condición**
JE / JZ	Salta si es igual o salta si es cero	ZF = 1
JNE / JNZ	Salta si no es igual o salta si no es cero	ZF = 0
JC	Salta si hay acarreo	CF = 1
JNC	Salta si no hay acarreo	CF = 0
JS	Salta si el signo es negativo	SF = 1
JNS	Salta si el signo es positivo	SF = 0
JO	Salta si hay desbordamiento (sobreflujo)	OF = 1
JNO	Salta si no hay desbordamiento (sobreflujo)	OF = 0
JP / JPE	Salta si hay paridad o salta si la paridad es par	PF = 1
JNP / JPO	Salta si no hay paridad o salta si la paridad es impar	PF = 0

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Salto Condicional.

Saltos Condicionales para números sin signo		
Mnemónico	**Significado**	**Condición**
JA / JNBE	Salta si es mayor o salta si no es menor igual	CF = 0 y ZF = 0
JAE / JNB	Salta si es mayor o igual o salta si no es menor	CF = 0
JB / JNAE	Salta si es menor o salta si no es mayor igual	CF = 1
JBE / JNA	Salta si es menor o igual o salta si no es mayor	CF = 1 o Z = 1
Saltos Condicionales para números con signo		
Mnemónico	**Significado**	**Condición**
JG / JNLE	Salta se es mayor o salta si no es menor igual	ZF = 0 y SF = OF
JGE / JNL	Salta si es mayor o igual o salta si no es menor	SF = OF
JL / JNGE	Salta si es menor o salta si no es mayor igual	(SF □ OF)=1
JLE / JNG	Salta si es menor o igual o salta si no es mayor	((SF □ OF) + ZF) = 1

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Salto Condicional.

Mnemónico	**Código Objeto**	**Mnemónico[a]**	**Segmento**	**Operación simbólica**
LOOP etiqueta corta	E2 --[a]	LOOP MEMS[b]	Código	CX ← CX –1 Si CX ≠ 0, entonces IP ← MEMS
LOOPE / LOOPZ etiqueta corta	E1 --[a]	LOOPZ MEMS[b]	Código	CX ← CX –1 Si (CX≠0)^(ZF=1), entonces IP ← MEMS
LOOPNE / LOOPNZ etiqueta corta	E0 --[a]	LOOPNZ MEMS[b]	Código	CX ← CX –1 Si (CX≠0)^(ZF=0), entonces IP ← MEMS

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Llamado a Subrutina.

Mnemónico general Op-code Operando	Código Objeto	Mnemónico	Segmento	Operación simbólica
CALL etiqueta cercana	E8 -- --	CALL MEMN	Código	SP← SP –2; [SP+1:SP]← IP; IP← MEMN
	FF 16 00 10	CALL [MEMWDS]	Datos	SP← SP –2; [SP+1:SP]← IP; IP←[1001H:1000H]
	FF15	CALL [DI]	Datos	SP← SP –2; [SP+1:SP]← IP; IP←[DI+1:DI]
	FF D7	CALL DI	Dentro del CPU	SP← SP –2; [SP+1:SP]← IP; IP←DI

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Llamado a Subrutina (cont.)

Mnemónico	Código Objeto	Mnemónico	Segmento	Operación simbólica
CALL etiqueta lejana	9A 00 10 D3 09	CALL FAR PTR MEMF	Código	SP← SP –2; [SP+1:SP]← CS; CS←09D3H; SP← SP –2; [SP+1:SP]← IP; IP← 1000H
	FF 1E 00 10	CALL [MEMWWDS]	Datos	Similar al anterior excepto: CS ←[1003H:1002H]; IP ← [1001H:1000H]
	FF 1D	CALL DWORD PTR[DI]	Datos	Similar al anterior excepto: CS←[DI+3:DI+2]; IP←[DI+1:DI]

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Retorno de Subrutina.

Mnemónico	Código	Mnemónico	Segmento	Operación simbólica
RET n (cercano)[f]	C3	RET	Pila	IP ← [SP+1:SP]; SP← SP +2;
	C2 08 00	RET 8	Pila	IP←[SP+1:SP]; SP← SP + 2 + 8
RET n (lejano)[f]	CB	RET	Pila	IP←[SP+1:SP]; SP← SP + 2; CS ←[SP+1:SP]; SP←SP+2
	CA 08 00	RET 8	Pila	IP ← [SP+1:SP]; SP←SP + 2; CS←[SP+1:SP]; SP←SP + 2 + 8

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Push y Pop.

Mnemónico general	Código Objeto	Mnemónico	Segmento	Operación simbólica
PUSH fuente	51	PUSH CX	Pila	SP←SP-2; [SP+1]←CH; [SP]←CL
	1E	PUSH DS	Pila	SP←SP-2; [SP+1:SP]←DS
	FF 75 02	PUSH[DI+2]	Pila, datos	SP←SP-2; [SP+1]←[DI+3]; [SP]←[DI+2]
POP dest	59	POP CX	Pila	CL←[SP]; CH←[SP+1]; SP←SP+2
	1F	POP DS	Pila	DS←[SP+1:SP]; SP←SP+2
	8F 45 02	POP[DI+2]	Datos, pila	[DI+3]←[SP+1]; [DI+2]←[SP]; SP←SP+2
PUSHF	9C	PUSHF	Pila	SP←SP-2; [SP+1:SP]← Banderas
POPF	9D	POPF	Pila	Banderas←[SP+1:SP]; SP←SP+2

Tema 2. Características del Microprocesador 80X86.
Instrucciones de Interrupciones.

Mnemónico general	Código Objeto	Mnemónico	Segmento de memoria	Operación simbólica
INT tipo	CD 23	INT 23H	Pila e interrupción saltando a la tabla de vectores (00000 a 003FFH)	SP← SP –2; [SP+1:SP]← Banderas IF←0; TF←0; SP← SP –2; [SP+1:SP]← CS; CS←[0008FH:0008EH]; SP← SP –2; [SP+1:SP]← IP; IP←[0008DH:0008CH]
INTO	CE	INTO	Pila e interrupción saltando a la tabla de vectores (00000 a 003FFH)	Si OF=1, entonces SP← SP –2 [SP+1:SP]← Banderas; IF=0; TF=0; SP← SP –2; [SP+1:SP]← CS; CS←[00013H:00012H]; SP← SP –2; [SP+1:SP]←IP; IP ←[00011H:00010H]
IRET	CF	IRET	Pila	IP ←[SP+1:SP]; SP ← SP + 2; CS ←[SP+1:SP]; SP ← SP + 2; Banderas←[SP+1:SP]; SP ← SP+ 2

Tema 2. Características del Microprocesador 80X86.

Instrucciones de Control del Procesador.

Mnemónico general	Código objeto	Mnemónico	Segmento de memoria	Operación simbólica
STC	F9	STC	Dentro del CPU	CF ← 1
CLC	F8	CLC	Dentro del CPU	CF ← 0
CMC	F5	CMC	Dentro del CPU	CF ← $\overline{CF}$
STD	FD	STD	Dentro del CPU	DF ←1
CLD	FC	CLD	Dentro del CPU	DF ← 0
STI	FB	STI	Dentro del CPU	IF ← 1
CLI	FA	CLI	Dentro del CPU	IF ← 0
HLT	F4	HLT	Dentro del CPU	Ninguna
WAIT	9B	WAIT	Dentro del CPU	Ninguna
LOCK Instrucción	F0 A1 00 10	LOCK MOV AX,MEMWDS	Datos	Ninguna
NOP	90	NOP	Dentro del CPU	Ninguna
ESC número,fuente	DE 0E 00 10	ESC 31H,MEMWDS	Datos	Bus de datos ←[MEMWDS]

Tema 2. Características del Microprocesador 80X86.

Instrucciones de Control del Procesador.

Mnemónico general		**Segmento de memoria**	**Operación**
WAIT	WAIT	Ninguna	Pone en estado de espera ($\overline{TEST}$=1) al procesador hasta que el coprocesador termina su ejecución.
LOCK Instrucción	LOCK MOV AX,MEMWDS	Ninguna	Bloquea el bus. Evita que el 8087 u otros coprocesadores cambien datos al mismo tiempo que el procesador; coloca la línea de salida $\overline{LOCK}$= 0. Lock es un prefijo de un byte que se utiliza para prevenir a los coprocesadores de acceder al bus hasta que Se complete la instrucción siguiente a lock.
NOP	NOP	Ninguna	No operación.
ESC número,fuente	ESC 31H,MEMWDS	Bus de datos ←[MEMWDS]	Coloca el contenido del operando fuente de memoria en el bus de datos y ejecuta un NOP. El primer operando identifica una instrucción escape particular para que sea Ejecuta por el coprocesador.

Interrupciones
Tabla de Vectores de Interrupción

- La Tabla de vectores de interrupción se ubica en los primeros 1024 bytes de memoria (00000H- 003FFH) contiene 256 vectores de interrupción diferentes de 4 bytes.
- Un vector de interrupción contiene la dirección (segmento de código y desplazamiento IP) del procedimiento de interrupción.
  - Los dos bytes menos significativos contienen el IP y los dos bytes más significativos contienen el CS.