



Instrucciones

Proporcionar los fundamentos del
conjunto de instrucciones



Instrucción MOV

[etiqueta:]	MOV	{registro/memoria},{registro/memoria/inmediato}
-------------	-----	---

- Mueve datos inmediatos
 - MOV AX,25
 - MOV BYTEVAL,25
 - MOV WORDVAL [BX],25
- Mueve registros
 - MOV AX,CX
 - MOV DS,AX
 - MOV BYTEVAL,BH
 - MOV [SI],AX



Instrucción MOV

- Mueve memoria directa
 - MOV BH,BYTEVAL
 - MOV AX,WORDVAL [BX]
- Mueve registro de segmento
 - MOV AX,DS
 - MOV WORDVAL,DS



Instrucción MOV

- Las operaciones MOV que NO son permitidas son:
 - De memoria memoria
 - Inmediato a registro de segmento
 - Registro de segmento a registro de segmento.
 - Nota: El destino debe ser de la misma longitud que el fuente. De lo contrario se debe usar MOVSX y MOVZX.



La instrucción LEA

[etiqueta:]	LEA	{registro/memoria}
-------------	-----	--------------------

- Es útil para inicializar un registro con una dirección de desplazamiento.
- “load Offset Address”
- Inicializa un desplazamiento en: BX, DI, SI.
- LEA BX, DATABLK
- MOV SAVBYTE, [BX]



Instrucciones INC y DEC

[etiqueta:]	INC/DEC	{registro/memoria}
-------------	---------	--------------------

- Son instrucciones adecuadas para aumentar y disminuir en 1 los contenidos de registros y localidades de memoria.
- Sólo necesitan de un operando.
- Podrá modificar las banderas OF, SF, y ZF. Para instrucciones de salto condicional.



La instrucción INT

- Una instrucción INT interrumpe el procesamiento y accesa la tabla de servicios de interrupción en memoria, para determinar la dirección de la rutina solicitada.
- La operación transfiere el control al DOS o al BIOS para una acción específica y posteriormente regresa al programa para continuar su ejecución.



La Instrucción INT

■ Pasos

- Decrementa en 2 el apuntador de pila y mete el contenido del registro de banderas.
- Limpia las banderas IF y TF.
- Decrementa en 2 el apuntador de pila y mete en la pila el CS.
- Decrementa en 2 el apuntador de pila y mete el IP.
- Hace que la operación solicitada sea realizada.



Logica y Control de Programas

Ciclos y Saltos



Instrucciones para Lógica y Control

Operaciones de comparación	Operaciones de transferencia	Operaciones Lógicas	Corrimiento y rotación
CMP	CALL	AND	SAR/SHR
TEST	JMP	NOT	SAL/SHL
Jnnn	OR	RCR/ROR	
LOOP	XOR	RCL/ROL	



Direcciones Corta, Cercana y Lejana

- Dirección corta:
 - Desplazamiento de un byte, limitado a una distancia de -128 a 127 bytes.
- Dirección Cercana:
 - Desplazamiento de una palabra, limitado a una distancia de $-32,768$ a $32,767$ bytes, dentro del mismo segmento.
- Dirección Lejana:
 - Puede estar en otro segmento y es alcanzada por una dirección de segmento más un desplazamiento; se usa CALL



Direcciones

	Corta	Cercana	Lejana
	Mismo segmento −128 a 127	Mismo segmento −32,768 a 32,767	Otro segmento
Instrucciones			
JMP	Sí	Sí	Sí
Jnnn	Sí	Sí	No
LOOP	Sí	No	No
CALL	N/A	Sí	Sí



Etiquetas de Instrucciones

- Las instrucciones JMP, Jnnn y LOOP; requieren un operando que se refiere a la etiqueta de una instrucción.

- Ejemplo:

```
        JMP    A90
        ...
A90:    MOV    AH,00
```

- A90: terminada con (:) le da el atributo de cercana. Un erro común es la omisión de los dos puntos.



La instrucción JMP

[etiqueta:]	JMP	Dirección corta, cercana o lejana
-------------	-----	-----------------------------------

- Usada comúnmente para la transferencia de control, es un salto incondicional, ya que la operación transfiere el control bajo cualquier circunstancia.
- Ver ejemplo...



La instrucción LOOP

[etiqueta:]	LOOP	Dirección corta
-------------	------	-----------------

- En el ejemplo anterior JMP realizó un ciclo infinito. Para realizar un ciclo un número finito de veces usamos LOOP. Esta instrucción requiere un valor inicial en el registro CX. En cada iteración, LOOP de forma automática disminuye 1 de CX. Si el valor de Cx es 0, el control pasa a la siguiente instrucción. Si el valor de CX no es cero, el control pasa a la dirección del operando.
- Ejemplo...



La instrucción CMP

[etiqueta:]	CMP	{registro/memoria},{registro/memoria/inmediato}
-------------	-----	---

- Es usada para comparar dos campos de datos.
- El resultado de CMP afecta las banderas AF, CF, OF, PF, SF y ZF.
- Ejemplo:

```
        CMP    BX,00
        JZ     B50
        ...
B50:    MOV    AH,00
```




Instrucciones de salto condicional

[etiqueta:]	Jnnn	Dirección corta
-------------	------	-----------------

- Es una variedad de instrucciones que transfieren el control del programa dependiendo la configuración del registro de banderas.
- Por ejemplo, se puede comparar dos campos y después saltar de acuerdo con los valores de las banderas que la comparación establece.



Salto condicional

- Un ejemplo práctico podría ser remplazar la instrucción LOOP con dos enunciados:
 - Uno que decremente CX y otro que realice un salto condicional
 - DEC CX ;Equivalente a LOOP
 - JNZ A20
- JNZ prueba la bandera ZF.
- Claro que para este ejemplo es mejor usar LOOP, aunque próximamente veremos que LOOP tiene usos limitados.



Datos con signo o sin signo

- La distinción determinará cual es la instrucción a utilizar.
 - Sin signo: trata los bits como bits de datos.
 - Con signo: trata el bit de más a la izquierda como un signo.
 - Ejemplo, para AX=11000110 y BX=00010110:
 - `CMP AX,BX`
 - Esta instrucción compara el contenido de AX con el contenido de BX. Para datos con signo, el valor de AX es mayor; sin embargo, para datos con signo el valor de AX es menor a causa del signo negativo.



Saltos con base en datos sin signo

Símbolo	Descripción	Bandera
JE/JZ	Salta si es igual o salta si es cero	ZF
JNE/JNZ	Salta si no es igual o salta si no es cero	ZF
JA/JNBE	Salta si es mayor o salta si no es menor o igual	CF, ZF
JAЕ/JNB	Salta si es mayor o igual o salta si no es menor	CF
JB/JNAE	Salta si es menor o salta si no es mayor o igual	CF
JBE/JNA	Salta si es menor o igual o salta si no es mayor	CF, AF



Saltos con base en datos con signo

Símbolo	Descripción	Bandera
JE/JZ	Salta si es igual o salta si es cero	ZF
JNE/JNZ	Salta si no es igual o salta si no es cero	ZF
JG/JNLE	Salta si es mayor o salta si no es menor o igual	ZF, SF, OF
JGE/JNL	Salta si es mayor o igual o salta si no es menor	SF, OF
JL/JNGEL	Salta si es menor o salta si no es mayor o igual	SF, OF
JLE/JNG	Salta si es menor o igual o salta si no es mayor	ZF, SF, OF



Pruebas aritméticas especiales

Símbolo	Descripción	Bandera
JS	Salta si el signo es negativo	SF
JNS	Salta si el signo es positivo	SF
JC	Salta si hay acarreo (igual que JB)	CF
JNC	Salta si no hay acarreo	CF
OFJO	Salta si hay desbordamiento	OF
JNO	Salta si no hay desbordamiento	OF
JP/JPE	Salta si hay paridad o salta si la paridad es par	PF
JNP/JPO	Salta si no hay paridad o salta si la paridad es impar	PF



Llamada a procedimientos

- Hasta ahora los segmentos de código que hemos usado constan de un solo procedimiento.
- Un segmento de código puede tener un número indefinido de procedimientos.
- Un procedimiento (subrutina) es una sección de código que realiza una tarea definida y clara.



Ventajas en el uso de procedimientos

- Reduce la cantidad de código.
- Fortalece la mejor organización del programa.
- Facilita la depuración del programa.
- Ayuda en el mantenimiento progresivo del programa.



Operaciones CALL y RET

- La instrucción CALL transfiere el control del programa a un procedimiento, y la instrucción RET regresa del procedimiento llamado al procedimiento original que hizo la llamada.

[etiqueta:]	CALL	Procedimiento
[etiqueta:]	RET	[inmediato]



Llamada y regreso cercano

- Una llamada a un procedimiento en el mismo segmento es cercana y realiza lo siguiente:
 - Disminuye el SP en 2 (una palabra)
 - Mete el IP en la Pila
 - Inserta la dirección del desplazamiento del procedimiento llamado en el IP.
- RET
 - Saca el antiguo valor de IP de la pila y lo envía al IP
 - Incrementa el SP en 2.



Llamada y regreso lejano

- Una llamada (CALL) lejana llama a un procedimiento etiquetado con FAR, que puede estar en un segmento de código separado. Un CALL lejana mete a la Pila al CS y al IP, y el RET los saca de la Pila.
- Ver ejemplo de un CALL y RET cercano



Operaciones Booleanas

AND, OR, XOR, TEST, Y NOT



Operaciones Booleanas

- Usadas para poner bits en 0 o en 1, en operaciones para manejar datos ASCII y en operaciones especiales de manejo de bits.

[etiqueta:]	Operación	{registro/memoria},{registro/memoria/inmediato}
-------------	-----------	---



Operaciones Booleanas

- Las operaciones compara los bits de los dos operandos referenciados y de acuerdo con esto establece las banderas CF, OF, PF, SF y ZF
- AND, OR, XOR se comportan como ya es conocido.
- TEST establece las banderas igual que lo hace AND, pero no cambia los bits de los operandos.
- Ver ejemplos del libro (pg. 125)
- La instrucción NOT sólo invierte los bits en un byte o palabra en un registro o en memoria.



Cambio de minúsculas a mayúsculas

- La mayúsculas de A a la Z son de 41H hasta 5AH y las minúsculas son de la 61H a la 7AH. La única diferencia es que el bit 5 de una mayúscula está en cero y para minúsculas está en 1.



Pantalla y teclado

Desplegar información en pantalla
y recibir información desde el
teclado



Introducción

- Manejaremos mayormente la instrucción INT para procesos de I/O.
- INT 10H (del BIOS) para pantalla.
- INT 21H (del DOS) para mostrar salidas en pantalla y aceptar datos del teclado.
- Para identificar el tipo de operación que la interrupción va a realizar se verifica el contenido de AH.



Operaciones

FUNCIONES DE LA INT 10H	FUNCIONES DE LA INT 21H
02H FIJA EL CURSO	02H DESPLIEGA EN LA PANTALLA
06H RECORRE LA PANTALLA	09H DESPLIEGA EN PANTALLA
	0AH ENTRADA DESDE EL TECLADO
	3FH ENTRADA DESDE EL TECLADO
	40H DESPLIEGA EN PANTALLA

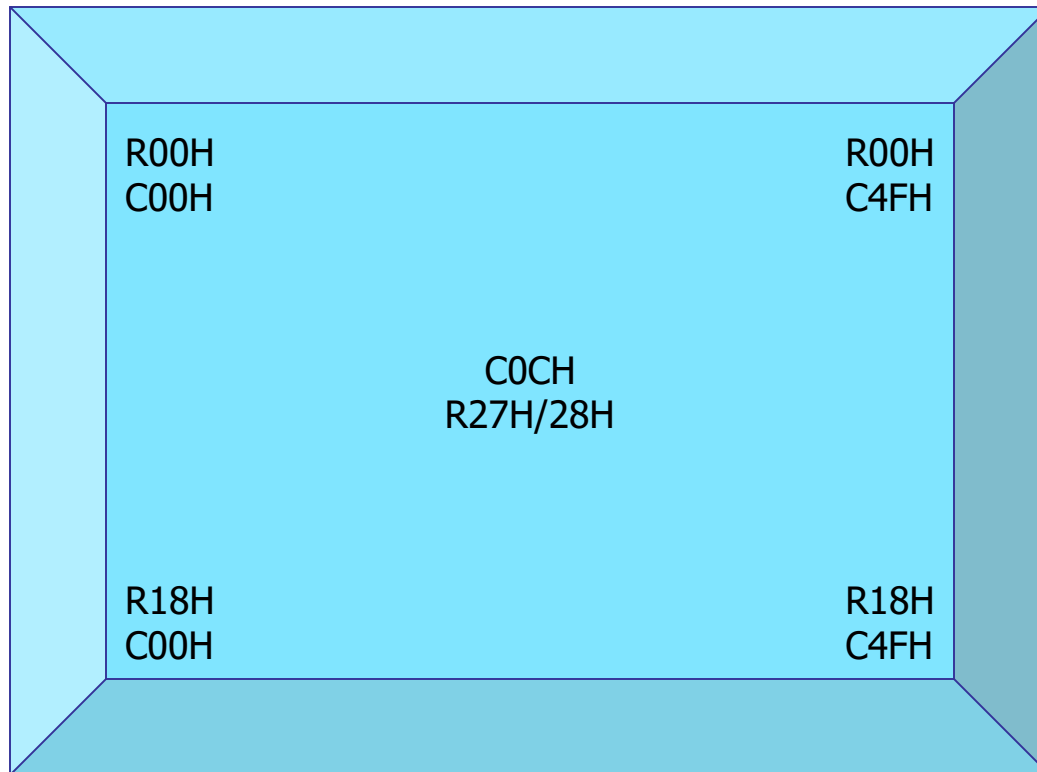


La Pantalla

- La pantalla es una malla de posiciones direccionables, en cualquiera de las cuales se puede colocar el cursor.
- Un monitor común:
 - 25 renglones y 80 columnas



Pantalla





Pantalla

- Área de despliegue de video (bufer).
- Técnicamente los programas podrán pasar los datos al área de despliegue, no es lo recomendado, ya que las direcciones de esta área podrán variar.
- La práctica recomendada es utilizar las instrucciones de interrupción adecuada: Las funciones de la INT 10H para despliegue, ubicar el curso en cualquier posición y limpiar la pantalla, y las funciones de INT 21H para diferentes tipos de despliegue.



Colocación del cursor

- INT 10H y la función 02H en el AH indica la operación de colocar el cursor.
- Se carga el número de página en BH, por lo común es 0.
- En DX se carga el renglón y la columna.
- Ejemplo:

- Colocar cursor en renglón 05, columna 12:

```
MOV    AH,02H
MOV    BH,00
MOV    DH,05
MOV    DL,12  ; podrá ser mov dx,050CH
INT     10H
```



Limpiar Pantalla

- INT 10H y función 06H maneja el borrado o recorrido de la pantalla.
- Podrá limpiar todo o parte del despliegue, iniciando en cualquier parte y terminando de igual forma.
 - AH = función 06H
 - AL = 00H para la pantalla completa
 - BH = número del atributo
 - CX = renglón:columnas iniciales
 - DX = renglón:columnas finales



Limpiar Pantalla

- Este ejemplo tiene el atributo 71H que establece toda la pantalla con fondo blanco (7) con primer plano azul(1):

```
MOV     AX,0600H
MOV     BH,71H
MOV     CX,0000H
MOV     DX,184FH
INT 10H
```

- [Ver ejemplo...](#)



Despliegue en Pantalla

- INT 21H función 09H se usa para desplegar de forma simple.
- Requiere de la definición de una cadena de despliegue en el área de datos. La cadena es seguida inmediatamente por un delimitador de signo de pesos (\$), el cual utiliza la operación para finalizar el despliegue.
- Ejemplo:
 - `NAMPRMP DB 'Nombre del cliente','$' ;cadena de despliegue`



Despliegue en Pantalla

- Ejemplo:
 - Coloque la función 09H en el registro AH, utilice LEA para cargar la dirección de la cadena en DX, y ejecute INT 21H.
 - Ver ejemplo del cambio de a mayúsculas.
- La operación INT no cambia el contenido de los registros.
- Si al final de la cadena se omite el signo de pesos, la operación despliega caracteres de la memoria hasta que encuentre un signo así, si existe alguno.



Tarea

- Hacer un programa que acepte un mensaje en minúsculas y convierta el mensaje en mayúsculas, el mensaje será limitado por la tecla <enter>. Y desplegralo en el centro de la pantalla.



Entrada de Teclado

- Función 0AH de INT 21H, acepta datos desde el teclado.
- 1. La interrupción necesita conocer la longitud máxima de los datos de entrada.
- 2. La operación envía a la lista de parámetros el número de bytes que realmente se introdujeron.



Entrada del teclado

- LABEL, es una directiva con el tipo de atributo BYTE, que sólo provoca alineación en un límite de byte.
 - 1er. Byte, contiene el límite del número máximo de caracteres, entre 00H y FFH.
 - 2do. Byte, almacena el número real de caracteres introducidos.
 - 3er. Byte, inicia un campo que contiene los caracteres tecleados.



Código

- NAMEPAR LABEL BYTE
- MAXLEN DB 20
- ACTLEN DB ?
- NAMEFLD DB 20 DUP('')
- Para solicitar una entrada:
 - MOV AH,0AH
 - LEA DX,NAMEPAR
 - INT 21H
- Ver ejemplo del libro pg. 142



Aceptar y desplega nombres

- El siguiente programa pide al usuario que introduzca un nombre y después lo despliega en el centro de la pantalla y emite un sonido la bocina.



Uso de caracteres de control para desplegar

- Una manera de hacer más eficaz el despliegue es usar los caracteres de control: Retorno de carro, Avance de línea y el Tabulador.
- En ASCII o en Hexadecimal.

CARACTERES DE CONTROL	ASCII	HEX	EFFECTO EN EL CURSOR
Retorno de carro	13	0DH	Restablece a la posición de la extrema izquierda
Avance de línea	10	0AH	Avanza a la línea siguiente
Tabulador	09	09H	Avanza a la siguiente marca de tabulador



Caracteres de control

- Siempre que despliegues salidas o aceptes entradas, puedes usar estos caracteres de control para el manejo del cursor.
- Ejemplo:

```
MENSAJE      DB      09,'Reporte anual',13,10,'$'  
  
.....  
  
MOV          AH,09H  
  
LEA          DX,MENSAJE  
  
INT          21H
```



Caracteres de Control

- El uso de EQU para definir los caracteres de control pueden hacer que un programa sea más legible.

CR EQU 13 ;o EQU 0DH

LF EQU 10

TAB EQU 09

MENSAJE DB TAB,'Reporte Anual',CR,LF,'\$'



Función 02H para despliegue en pantalla

- A diferencia de la función 09H, la función 02H puede ser útil para desplegar un solo carácter.
- El carácter debe ser cargado en DL.

```
MOV    AH,02H
```

```
MOV    DL,char
```

```
INT     21H
```



Función 02H

- Ejemplo

CONAME DB 'Servicios de Software',13,10

.....

MOV AH,02H

MOV CX,23

LEA DI,CONAME

A30: MOV DL,[DI]

INT 21H

INC DI

LOOP A30



Manejadores de Archivos

- Un manejador de archivo sólo es un número que hace referencia a un dispositivo específico.
- Manejadores de archivo preestablecidos.

MANEJADOR	DISPOSITIVOS
00	Entrada, por lo regular del teclado (CON)
01	Salida, por lo regular la pantalla (CON)
02	Error en la entrada, pantalla (CON)
03	Dispositivo Auxiliar (AUX)
04	Impresora (LPT1 o PRN)



Despliegue con manejadores

- La función 40H de la INT 21H utiliza los manejadores de archivo para solicitar las operaciones de despliegue.
 - AX = Función 40H
 - BX = Manejador de archivo 01
 - CX = Número de caracteres a desplegar
 - DX = Dirección del área de despliegue
- Una operación exitosa regresa al AX el número de bytes escritos y pone en cero la bandera de acarreo.
- Una operación fallida pone en uno la bandera de acarreo y regresa un código de error en el AX (05H= acceso denegado...).



Función 40H

- Ejemplo

CONAME DB 'Servicios de Software',0DH,0AH

.....

MOV AH,40H

MOV BX,01

MOV CX,23

LEA DX,CONAME

INT 21H



Manejador para entrada del teclado

- La función 3FH de la INT 21H, usa manejadores de archivo para solicitar entrada del teclado, aunque es una operación un poco ineficaz.
 - AH = Función 3FH
 - BX = Manejador de archivo 00
 - CX = Número máximo de caracteres que se aceptan.
 - DX = Dirección del área de datos para introducir los caracteres.

Procesamiento avanzado de pantalla



Recorrido, video inverso,
intermitencia y gráficas a color.



Servicios ofrecidos

- 00H Establece el modo de video.
- 01H Establece el tamaño del cursor.
- 02H Establece la posición del cursor.
- 03H Lee la posición del cursor.
- 04H Lee la posición de la pluma óptica.
- 05H Selecciona la página activa.
- 06H Recorre la pantalla hacia arriba.
- 07H Recorre la pantalla hacia abajo.
- 08H Lee el atributo o carácter en la posición del cursor.
- 09H Despliega el atributo o carácter en la posición del cursor.
- 0AH Despliega el carácter en la posición del cursor.
- 0BH Establece la paleta de colores.
- 0CH Escribe el pixel punto.
- 0DH Lee el pixel punto.
- 0EH Escribe en teletipo.
- 0FH Obtiene el modo actual de video.
- 11H Genera carácter.
- 12H Selecciona rutina alterna de pantalla.
- 13H Despliega cadena de caracteres.
- 1BH Regresa la información de funcionalidad o de estado.
- 1CH Guarda o restaura el estado de video.



Adaptadores de video

- MDA Adaptador de pantalla monocromática.
- HGC Tarjeta de gráficos Hércules.
- CGA Adaptador de gráficos a colores.
- EGA Adaptador de gráficos mejorado.
- MCGA Adaptador de gráficos en multicolores.
- VGS Matriz de gráficos de video.



Adaptadores de video

- El controlador de video
 - Es el “caballo de batalla”, genera las señales de restreo del monitor para el modo selccionado, texto o gráfico.
- El video de BIOS
 - Actua como una interfaz con el adapatador de video, contiene rutinas, como para establecer el cursor y desplegar caracteres.
- El área de despliegue de video
 - Contiene la información que el monitor va a mostrar. Las interrupciones que manejan el despliegue de la pantalla de forma direta transfieren a esta área sus datos.



Especificaciones del modo de video

- La función 00H puede designar el modo de video, puede conmutar entre texto y gráfico. También limpia la pantalla.
- Ejemplo, el modo 3 representa modo texto, color y resolución de la pantalla, dependiendo el tipo de monitor.
- Para designar un modo nuevo, solicite la INT 10H, con la función 00H en el registro AH y el modo en el AL.



Modo de Texto

- El modo de texto se utiliza para el despliegue normal en la pantalla de caracteres ASCII.
- Proporciona acceso a todo el conjunto de caracteres ASCII extendido
- Figura 10-1 muestra los modos de texto comunes.



Modo de Texto

- Byte de atributo
 - Un byte de atributo en modo de texto determina las características de cada carácter mostrado.
 - Cuando un programa establece un atributo, permanece hasta que otra operación lo cambie.
 - Formato:

	Fondo				Frente			
BL	R	G	B	I	R	G	B	
7	6	5	4	3	2	1	0	



Modo texto

- Despliegue monocromático
 - Ver tabla.
- Despliegue a color
 - El fondo puede mostrar uno de ocho colores y los caracteres pueden mostrar uno de 16 colores.
 - La intermitencia e intensidad sólo se aplican al primer plano.
 - Ver ejemplo pg.158