

$$\text{Mem} = \left[\begin{matrix} \text{SI} \\ \text{BP} \end{matrix} \right] + \left[\begin{matrix} \text{SI} \\ \text{DI} \end{matrix} \right] + \{\text{Desplazamiento}\}$$

Si se utiliza BP el segmento por defecto es SS
No existen las instrucciones Mem, Mem

R16 = AX, BX, CX, DX, BP, SI, DI
R8 = cualquiera de los 8
= dígito hexadecimal
RegSeg = CS, DS, SS, ES

MOV ___ , ___	Mueve fuente a destino
XCHG ___ , ___	Intercambia fuente con destino
PUSH ___	Apila dato
POP ___	Desapila dato
PUSHF	Apila flags
POPF	Modifica flags con la cima de la pila
IN AL , DX	Copia en AL el puerto especificado en DX
OUT DX , AL	Copia en el puerto especificado por DX el valor de AL
INC ___	Incrementa dato
DEC ___	Decrementa dato
ADD ___ , ___	Suma
ADC ___ , ___	Suma con acarreo
SUB ___ , ___	Resta
SBB ___ , ___	Resta con acarreo
CMP ___ , ___	Compara (hace una resta pero no almacena el resultado)
CBW	Expande signo de AL a AH
CWD	Expande signo de AX a DX
NEG ___	Complemento a 2
AND ___ , ___	AND
TEST ___ , ___	AND sin almacenar resultado
OR ___ , ___	OR
XOR ___ , ___	OR exclusivo
NOT ___	Complemento a 1
SHL ___ , 1	1 desplazamiento a la izquierda
SHL ___ , CL	Varios (valor de cl) desplazamientos a la izquierda
SHR ..	Desplaza a la derecha introduciendo 0 (desplazamiento lógico)
SAR ..	Desplaza a la derecha introduciendo repitiendo valor anterior (desplazamiento aritmético)
ROL ..	Rota a la izquierda

ROR ..	Rota a la derecha
RCL ..	Rota a la izda. El bit de carry forma parte
RCR ..	Rota a la derech. El bit de carry forma parte
JMP #####	Salto NEAR
JMP R16	IP ← R16
JMP Mem	IP ← [Mem]
JMP #####:#####	Salto FAR
JMP FAR Mem	IP CS ←← [Mem]
CALL #####	llamada NEAR
CALL R16	Apila IP; IP ← R16
CALL Mem	Apila IP; IP ← [Mem]
CALL #####:#####	llamada FAR
CALL FAR Mem	Apila CS; Apila IP IP CS ←← [Mem]
RET	Desapila en IP Retorno de procdmto NEAR
RETF	Desapila IP; Desapila CS Retorno FAR
RET #####	Retorno NEAR con modificación posterior de SP
RETF #####	Retorno FAR con modificación posterior de SP
J* #####	Saltos condicionales
C/NC	C = 1/0
Z/NZ	Z = 1/0
O/NO	O = 1/0
P/NP	P = 1/0
S/NS	S = 1/0
A/NA	$\overline{C} \cdot \overline{Z} = 1/0$
L/NL	$S \oplus O = 1/0$
G/NG	$(\overline{S \oplus O}) \cdot \overline{Z} = 1/0$
INT ##	Apila flags, CS e IP; Modifica CS e IP con el vector especificado por ##
IRET	Desapila flags, CS e IP
CLI	Pone IF = 0 Inhibiendo interrupciones
STI	Pone IF = 1

NOP	En la fase de ejecución el procesador no hace nada (no operación)
MOV RegSeg,R16	registros de segmento
MOV RegSeg,Mem	
MOV Mem,RegSeg	
MOV R16,RegSeg	
PUSH RegSeg	
POP RegSeg	
LEA R16,Mem	Carga la parte de offset de Mem en R16
LDS R16,Mem	Carga un puntero FAR en DS:R16
LES R16,Mem	Carga un puntero FAR en ES:R16

Tabla de Instrucciones intel 8086

Adjuntar esta tabla, junto con los apuntes anteriores, a las transparencias del tema de implementación de la asignatura:

Sistemas Operativos

Murcia abril de 2005