



**Universidad de Costa Rica**  
**Escuela de Ciencias de la Computación e Informática**  
**CI-1220 Ensambladores y microprocesadores**

**II Semestre de 2010**

**Resumen del conjunto de instrucciones**  
**(Pentium )**

Instrucción	Codificación	Ejemplo	Ciclos	
			80486	Pentium
AAA	00110111	AAA	3	3
AAD	11010101 00001010	AAD	14	10
AAM	11010100 00001010	AAM	15	18
AAS	00111111	AAS	3	3
ADC reg,reg	000100dw   TD/R Reg Ind	ADC AX,BX / ADC AL,BL / ADC CX,SI	1	1 o 3
ADC mem,reg	000100dw   TD/R Reg Ind   Desp	ADC DATOY,AL / ADC DATO2[DI],CL	3	1 o 3
ADC reg,mem	000100dw   TD/R Reg Ind   Desp	ADC BL,DATO1 / ADC SI,LIST1	2	1 o 2
ADC reg,imm	100000sw   TD/R 010 Ind   Desp   Dato	ADC CX,3     10000001 11010001 03 ADC DL,34H   10000000 11010010 34	1	1 o 3
ADC mem,imm	100000sw TD/R 010 Ind Desp Dato	ADC DATO3[DI],3   10000001 10010101 03 ADC DATO4,33     10000001 00010110 21	3	1 o 3
ADC acc,imm	0001010w Dato	ADC AX,3 / ADC AL,1AH	1	1
ADD reg,reg	00000dw  TD/R Reg Ind  Desp	ADD AX,BX 00000001 11000011 ADD AL,BL 00000000 11000011	1	1 o 3
ADD mem, reg	0000000w TD/R Reg Ind Desp	ADD DATOY, AL / ADD [EDX+4*ECX], EBX	3	1 o 3
ADD reg, mem	0000001w TD/R Reg Ind Desp	ADD BL, DATO2 / ADD CX, [EDI]	2	1 o 2
ADD reg, imm	100000sw TD/R 000 Ind Dato	ADD CX, 3 / ADD CX, 1834H / ADD EDX, 1345H	1	1 o 3
ADD mem, imm	100000sw TD/R 000 Ind   Desp   Dato	ADD DATO4, 33 / ADD LIST, 'A'	3	1 o 3
ADD acc, imm	100000sw   TD/R 000 Ind   Dato	ADD AX, DATA	1	1
AND reg, reg	001000dw   TD/R Reg Ind	AND CX, DX	1	1 o 3
AND mem, reg	001000dw   TD/R Reg Ind   Desp	AND BIT, AL / AND DATA[SI], DI	3	1 o 3
AND reg, mem	001000dw   TD/R Reg Ind   Desp	AND BL, LIST / AND ESI, [ECX+43H]	2	1 o 2

AND reg, imm	100000sw	TD/R 100 Ind	Dato	AND BP, 1	1	1 o 3
AND mem, imm	100000sw	TD/R 100 Ind	Desp   Dato	AND DATA, 33	3	1 o 3
AND acc, imm	1000000w	Dato		AND AX, 3	1	1
ARPL {reg/mem}, reg	01100011	TD/R Reg Ind	Desp	ARPL AX, BX / ARPL [EBX], AX	9	7
BOUND reg, mem	01100010	TD/R Reg Ind	Desp	BOUND AX, LIST / BOUND SI, [BX+2]	7	8
BSF reg, reg	10111100	TD/R Reg Ind	Desp	BSF EAX, EDX / BSF EBX, EAX	6 - 42	6 - 42
BSF reg, mem	0F 10111100	TD/R Reg Ind	Desp	BSF EBP, [EDX + 4 * ECX] / BSF CX, DATO3 [DI]	7 - 43	6 - 43
BSR reg, reg	0F 10111101	TD/R Reg Ind	Desp	BSR EAX, EDX / BSR EBX, EAX	6 - 103	7 - 71
BSR reg, mem	0F 10111101	TD/R Reg Ind	Desp	BSR EBP, [EDX + 4 * ECX] / BSR CX, DATO3 [DI]	7 - 104	7 - 72
BSWAP reg32	0F   11001 Reg			BSWAP EDX / BSWAP ESI	1	1
BT/BTC/BTR/BTS	ttt = 100 = BT; ttt = 111 = BTC; ttt = 110 = BTR; ttt = 101 = BTS					
BT {reg/mem}, imm8	0F   10111010	TD/R ttt Ind	Desp   Dato	BT BP, 10H / BT DATA[DI], 3	3	4
BT reg, reg	0F   10ttt011	TD/R Reg Ind		BT AX, CX	3	4 o 9
BT mem, reg	0F   10ttt011	TD/R Reg Ind	Desp	BT DATA[DI], DX	4	4 o 9
BTC/BTR/BTS reg, imm8	0F   10111010	TD/R ttt Ind	Dato	BTC AX, 2/ BTR BP, 10H	6	7 o 8
BTC/BTR/BTS mem, imm8	0F   10111010	TD/R ttt Ind	Desp   Dato	BTC DATO1, 2/ BTC DATO2[DI], 3	8	8
BTC/BTR/BTS reg, reg	0F 10ttt011	TD/R Reg Ind		BTC AX, CX/ BTR SI, CX	6	7 o 13
BTC/BTR/BTS mem, reg	0F 10ttt011	TD/R Reg Ind	Desp	BTC DATA4, AX / BTS [EBX], DX	13	7 o 13
CALL etiqueta (cercano)	11101000	Desp16		CALL HOME/ CALL ET	3	1
CALL etiqueta (lejano)	10011010	Desp16 Seg		CALL WHAT/ CALL WHERE	18	4
CALL reg (cercano)	11111111	TD/R 010 Ind		CALL SI/ CALL AX	5	2
CALL {reg/mem}	11111111	TD/R Reg Ind	Desp	CALL Vect Call Vec[SI][DI]	5	2
CBW (sin operando)	10011000	AL-> AX		CBW	3	3
CDQ (sin operando)	10011001	EAX -> EDX:EAX		CDQ	2	2
CLC	11111000			CLC	2	2
CLD	11111100			CLD	2	2
CLI	11111010			CLI	5	7
CLTS	0F   00000110			CLTS	7	10
CMC	10011000			CMC	2	2
CMOV reg, {mem/reg}	0F   0100cccc	TD/R Reg Ind	Desp	CMOVNZ AX,FROG	1	1
CMP reg, reg	001110dw	TD/R Reg Ind		CMP AX,BX	1	1 o 2
CMP mem, reg	001110dw	TD/R Reg Ind	Desp	CMP DATA,BX / CMP ESI, [ECX + 200H]	2	1 o 2
CMP reg, imm	100000sw	TD/R 111 Ind	Dato	CMP EDX, 1834H / CMP DI, 1AH	1	1 o 2
CMP mem, imm	100000sw	TD/R 111 Ind	Dato	CMP BYTE PTR [EDI], 1AH / CMP DATA, 3	2	1 o 2
CMP acc, imm	000111w	Dato		CMP AX, 3 / CMP AL, 'S' / CMP EAX, 1234H	1	1
CMPS	1010011w			CMPSB / CMPSW / CMPSB DATA1, DATA	8	5
CMPXCHG reg, reg	0F   1011000w	11 RegFuente RegDestino		CMPXCHG EAX, ECX / CMPXCHG BX, DX	6	6
CMPXCHG mem, reg	0F   1011000w	TD/R Reg Ind	Desp	CMPXCHG DATA, ECX / CMPXCHG DT[SI], DX	7	6
CMPXCHG8B mem64	0F   11000111	TD/R Red Ind	Desp	CMPXCHG8B DATA	-	10

CPUID	0F   10100010	CPUID	-	14
CWD	10011000	CWD	3	2
CWDE	10011000	CWDE	3	3
DAA	00100111	DAA	2	3
DAS	00101111	DAS	2	3
DEC reg8	1111111w   TD/R 001 Ind	DEC BL	1	1 o 3
DEC {reg16/reg32}	01001 Reg	DEC CX / DEC DI / DEC ECX	1	1
DEC mem	1111111w   TD/R 001 Ind   Desp	DEC LIST / DEC WORD PTR [EBX]	3	1 o 3
DIV {reg/mem}	1111011w   TD/R 110 Ind   Desp	DIV BL / DIV ECX / DIV LIST / DIV DATO6[DI]	40	17 – 41
ENTER imm, 0	11001000   Dato   00	ENTER 4, 0 / ENTER 1024, 0	14	11
ENTER imm, 1	11001000   Dato   01	ENTER 10, 1	17	15
ENTER imm, imm	11001000   Dato   Dato8	ENTER 100, 3	17	15 + 2n
HLT	11110100	HLT	4	variable
IDIV reg	1111011w   TD/R 111 Ind	IDIV ECX / IDIV DH	43	22-46
IDIV mem	1111011w   TD/R 111 Ind   Desp	IDIV DATA / IDIV WORD PTR [EBX]	44	22-46
IMUL {reg/mem}	1111011w   TD/R 101 Ind   Desp	IMUL BL / IMUL CX / IMUL BYTE PTR[BX]	42	10-11
IMUL reg, imm	011010s1   TD/R Reg Ind   Dato	IMUL DI, 100 / IMUL EDX, 20	42	10
IMUL reg, {reg/mem}, imm	011010s1   TD/R Reg Ind   Desp   Dato	IMUL DX, AX, 2 / IMUL CX, DATOY,99	42	10
IMUL reg, {reg/mem}	0F   10101111   TD/R Reg Ind   Desp	IMUL CX,DX / IMUL DI,BX / IMUL DX, DATOY	42	10
IN acc, puerto	1110010w   #Puerto	IN AL,12H / IN EAX, 10H	14	7
IN acc, DX	1110110w	IN AL, DX / IN EAX, DX	14	7
INC reg8	1111111w   TD/R 000 Ind	INC BL / INC DH	1	1 o 3
INC {reg16/reg32}	1111111w   TD/R 000 Ind	INC CX / INC ESP	1	1
INC mem	1111111w   TD/R 000 Ind   Desp	INC DATA	3	1 o 3
INS	0110110w	INSB / INSW / INSD	17	9
INT tipo	11001101   Tipo	INT 12H	30	16-82
INT 3	11001100	INT 3	26	13-56
INTO	11001110	INTO	28	13-56
INVD	0F   00001000	INVD	4	15
IRET / IRE	11001101   Dato	IRET \ IRETD	15	8-27
JCond etiqueta (8 bits)	0111cccc   Desp8	JZ ADENTRO	3/1	1
JCond etiqueta (16 bits)	0F   0111cccc   Desp16 Seg	JGE DESBORDE	3/1	1
JCXZ etiqueta JECXZ etiqueta	11100011	JCXZ BELOW / JECXZ EQUAL	8/5	6/5
JMP etiqueta (corto)	11101011   Desp8	JMP SHORT_UP	3	1
JMP etiqueta (cercano)	11101001   Desp16	JMP UNDER	3	1
JMP etiqueta (lejano)	11101010   Desp16 Seg	JMP AGAIN	17	3
JMP reg (cercano)	11111111   TD/R 100 Ind	JMP AX	3	2
JMP mem (cercano)	11111111   TD/R 100 Ind   Desp	JMP FROG	5	4
JMP mem (lejano)	11111111   TD/R 101 Ind   Desp   Seg	JMP UP	13	4

LAHF	10011111	LAHF	3	2
LAR reg, {reg/mem}	0F   00000010   TD/R Reg Ind   Desp	LAR CX, DX / LAR AX, LIST	11	8
LDS	11000101   TD/R Reg Ind   Desp	LDS SI, DATA	6	4
LEA reg, mem	10111 Reg   Desp	LEA DI, DATA 3	2	1
LEAVE	11001001	LEAVE	5	3
LES reg, mem	11000100   TD/R Reg Ind   Desp	LES SI, LIST2	6	4
LFS registro, memoria	0F   10110100   TD/R Reg Ind   Desp	LFS DI, DATA3	6	4
LGDT mem64	0F   00000001   TD/R 010 Ind   Desp	LGDT DESCRIP \ LGDT TABLED	11	6
LGS reg, mem	0F   10110101   TD/R Reg Ind   Desp	LGS DI, DATA3 \ LGS CX, PNTR	6	4
LIDT mem64	0F   00000001   TD/R 011 Ind   Desp	LIDT DATA3 \ LIDT LIST2	11	6
LLDT reg	0F   00000000   TD/R 010 Ind   Desp	LLDT BX \ LLDT CX	11	9
LMSW reg16	0F   00000001   TD/R 100 Ind	LMSW BX	2	8
LMSW mem	0F   00000001   TD/R 100 Ind   Desp	LMSW TOAD	3	8
LOCK:instrucción	11110000	LOCK:XCH AX,BX	1	1
LODS	1010110w	LODSB / LODSW / LODSD	5	2
LOOP/LOOPD etiqueta	11100010   Desp	LOOP NEXT, LOOP BACK	7/6	5/6
LOOPE/LOOPED etiqueta	11100001   Desp	LOOPED AGAIN / LOOPED WOW	9/6	7/8
LOOPNE/LOOPNED etiqueta	11100000   Desp	LOOPNE FORWARD LOOPNZ TRY AGAIN / LOOPNZD WOO	9/6	7/8
LSL reg, {reg/mem}	0F   00000011   TD/R Reg Ind   Desp	LSL AX, BX / LSL AX, DATA1	10	8
LSS reg, mem	0F   10110010   TD/R Reg Ind   Desp	LSS DI, DATA1	6	4
LTR {reg/mem16}	0F   00000000   TD/R 001 Ind   Desp	LTR AX / LTR TASK	20	10
<b>MOV reg, reg</b>	100010dw   TD/R Reg Ind   Desp	MOV CL, CH / MOV [DI], DH / MOV CL, DATA	1	1
MOV mem, reg				
MOV reg, mem				
MOV mem, imm	1100011w   TD/R 000 Ind   Desp   Dato	MOV VAR2, 23H / MOV DWORD PTR[ECX], 1	1	1
MOV reg, imm	1011w   Reg   Dato	MOV BX, 22H / MOV DI, 100	1	1
MOV mem, acc	101000dw   Dato	MOV VAR1, AL / MOV AX, VAR2 MOV VAR3, EAX	1	1
MOV acc, mem				
MOV seg, reg	100011d0   TD/R sss Ind	MOV SS, AX / MOV DS, DX / MOV ES, CX MOV BX, DS / MOV CX, FS / MOV CX, ES	1	1
MOV reg, seg				
MOV seg, mem	100011d0   TD/R sss Ind   Desp	MOV SS, DATAS / MOV ES, TEMP1	1	2 ó 3
MOV mem, seg	100011d0   TD/R sss Ind   Desp	MOV DATA2, CS / MOV TEMP, DS	1	1
MOV reg, cr	0F   001000d0   11 Reg Ind	MOV EBX, CR0 / MOV ECX, CR2	4	4
MOV cr, reg	0F   001000d0   11 Reg Ind	MOV CR1, EBX / MOV CR3, EDX	4	12 – 46
MOV reg, dr	0F   001000d1   11 Reg Ind	MOV ECX, DR7 / MOV EBX, DR1	10	11
MOV dr, reg	0F   001000d1   11 Reg Ind	MOV DR0, EAX / MOV DR3, EDX	11	11
MOV reg, tr	0F   001001d0   11 Reg Ind	MOV EBX, TR6 / MOV ECX, TR7	4	11
MOV tr, reg	0F   001001d0   11 Reg Ind	MOV TR6, EAX / MOV TR7, EBX	6	11
MOVS	1010010w	MOVSB / MOVSW / MOVS DATA1, DATA2	7	4
MOVSB reg, {reg/mem}	0F   1011111w   TD/R Reg Ind   Desp	MOVSB BX, AL	3	3
MOVZX reg, {reg/mem}	0F   1011011w   TD/R Reg Ind   Desp	MOVZX EAX, NUMB	3	3

MUL {reg/mem}	1111011w   TD/R 100 Ind   Desp	MUL BL / MUL DATA9	42	11
NEG {reg/mem}	1111011w   TD/R 011 Ind   Desp	NEG BL / NEG DATA	1	1 o 3
NOP	10010000	NOP	3	1
NOT {reg/mem}	1111011w   TD/R 010 Ind   Desp	NEG DATA9 \ NEG WORD PTR [ESI]	3	1 o 3
OR reg, reg	000010dw   TD/R Reg Ind	OR AX, BX – 00001011 11000011	1	1 o 2
OR mem, reg	000010dw   TD/R Reg Ind   Desp	OR DATAY, AL – 00001000 00000110 disp	3	1 o 3
OR reg, mem	000010dw   TD/R Reg Ind   Desp	OR BL, DATA1 – 00001010 00011110 disp	2	1 o 3
OR reg, imm	100000sw   TD/R 001 Ind   Desp   Dato	OR CX, 3 – 10000011 11001001	1	1 o 3
OR mem, imm	100000 sw   TD/R 001 Ind   Desp   Dato	OR DATAS, 3	3	1 o 3
OR acc, imm	0000110w   Dato	OR AX, 3	1	1
OUT puerto, acc	1110011w   #Puerto	OUT 12H, AL	10	12-26
OUT DX, acc	1110111w	OUT DX, AL	10	12-26
OUTS	0110111w	OUTB / OUTW / OUTD	10	13-27
POP reg	01011 Reg	POP EDI / POP BX	1	1
POP mem	10001111   TD/R 000 Ind   Desp	POP DATA	4	3
POP seg	00sss111	POP ES / POP DS	3	3
POP {FS/GS}	0F   10sss001	POP FS / POP GS	3	3
POPA	01100001	POPA	9	5
POPAD	66H   01100001	POPAD	9	5
POPF	10010000	POPF	6	4 ó 6
POPFD	66H   10010000	POPFD	6	4 ó 6
PUSH reg	01010   Reg	PUSH CX / PUSH EDI	1	1
PUSH mem	11111111   TD/R 110 Ind   Desp	PUSH DATA1	4	1 o 2
PUSH seg	00ss110	PUSH ES / PUSH CS / PUSH DS	3	1
PUSH {FS/GS}	0F   10sss000	PUSH FS / PUSH GS	3	1
PUSH imm	011010s0   Dato	PUSH 2000H / PUSH 53220 / PUSH ‘,’	1	1
PUSHA	01100000	PUSHA	11	5
PUSHAD	66H   01100000	PUSHAD	11	5
PUSHF	10011100	PUSHF	3	3 o 4
PUSHFD	66H   10011100	PUSHFD	3	3 o 4
ROL/ROR/RCL/RCR	ttt = 000 = ROL; ttt = 001 = ROR; ttt = 010 = RCL; ttt = 011 = RCR			
ROL/ROR/RCL/RCR reg, 1	1101000w   TD/R ttt Ind	RCL CL,1 / ROL SI, 1	3	1 o 3
ROL/ROR/RCL/RCR mem, 1	1101000w   TD/R ttt Ind   Desp	RCL CL,1 / ROL SI, 1	4	1 o 3
ROL/ROR reg, CL	1101001w   TD/R ttt Ind	ROR CL,1 / ROL SI, 1	3	4
RCL/RCR reg, CL	1101001w   TD/R ttt Ind	RCL CL,1 / RCR SI, 1	3	7-27
ROL/ROR mem, CL	1101001w   TD/R ttt Ind   Desp	ROR DATA, CL / ROL DATA[SI], CL	4	4
RCL/RCR mem, CL	1101001w   TD/R ttt Ind   Desp	RCL DATA, CL / RCR DATA[SI], CL	9	9-26
ROL/ROR reg, imm	1100000w   TD/R ttt Ind   Dato	ROL CH,4 / ROR ESI,14	2	1 o 3
RCL/RCR reg, imm	1100000w   TD/R ttt Ind   Dato	ROL CH,4 / ROR ESI,14	8	8-27
ROL/ROR mem, imm	1100000w   TD/R ttt Ind   Desp   Dato	ROL DATA, 5 / ROR WORD PTR [ESI], 2	4	1 o 3

ROL/ROR mem, imm	1100000w   TD/R ttt Ind   Desp   Dato	ROL DATA, 5 / ROR WORD PTR [ESI], 2	9	8-27
RDMSR	0F   00110010	RDMSR	-	20-24
REP MOVSw	F3   1010010w	REP MOVSB / REP MOVSW / REP MOVSD	12 + 3n	13 + n
REP STOSw	F3   1010101w	REP STOSW	9 + 14n	9 + n
REP INSw	F3   0110110w	REP ISNB	17 + 5n	25 + 3n
REP OUTSw	F3   0110111w	REP OUTSB	17 + 5n	25 + 4n
REPE CMPSw	F3   1010011w	REPE CMPSB / REPE CMPSW REPE CMPSD	7 + 7n	9 + 4n
REPNE CMPSw	F2   1010011w	REPNE CMPSB / REPNE CMPSW REPNE CMPSD	7 + 7n	8 + 4n
REPE SCASw	F3   1010111w	REPE SCASB / REPE SCASW / REPE SCASD	7 + 5n	9 + 4n
REPNE SCASw	F2   1010111w	REPNE SCASB / REPNE SCASW / REPNE SCASD	7 + 5n	9 + 4n
RET (cercano)	11000011	RET	5	2
RET imm (cercano)	11000010   Dato	RET 4 / RET 100H	5	3
RET (lejano)	11001011	RET	13	4-23
RET imm (lejano)	11001010   Dato	RET 100H	5	4-23
RSM	0F   10101010	RSM	-	83
SAHF	10011110	SAHF	2	2
SAL/SAR/SHL/SHR	ttt = 100 = SHL/SAL; ttt = 101 = SHR; ttt = 111 = SAR			
SAL/SAR/SHL/SHR reg, 1	1101000w   TD/R ttt Ind	SAL CL, 1 / SAR SI, 1	3	1 o 3
SAL/SAR/SHL/SHR mem, 1	1101000w   TD/R ttt Ind   Desp	SHL BYTE PTR [SI], 1 / SAR NUMB, 1	4	1 o 3
SAL/SAR/SHL/SHR reg, CL	1101001w   TD/R ttt Ind	SAL CL, 1 / SAR SI, 1	3	4
SAL/SAR/SHL/SHR mem, CL	1101001w   TD/R ttt Ind   Desp	SAL DATA, CL / SHR TEMP, CL	4	4
SAL/SAR/SHL/SHR reg, imm	1100000w   TD/R ttt Ind   Dato	SAL CH, 4 / SHR ESI, 23	2	1 o 3
SAL/SAR/SHL/SHR mem, imm	1100000w   TD/R ttt Ind   Desp   Dato	SAR NUM, 3 / SHL WORD PTR [ESI], 15	4	1 o 3
SBB reg, reg	000110dw   TD/R Reg Ind   Desp	SBB CL, DL / SBB AX, DX / SBB EAX, EBX	1	1 o 2
SBB mem, reg	000110dw   TD/R Reg Ind   Desp	SBB DATAJ, CL / SBB NUM, EBX / SBB [EAX], CX	1	1 o 3
SBB reg, mem	000110dw   TD/R Reg Ind   Desp	SBB CX, DYTES / SBB DX, [EBX+EDI]	1	1 o 2
SBB reg, imm	100000sw   TD/R 011 Ind   Desp   Dato	SBB CX, 3 / SBB DL, 34H / SBB EDX, 1345H	1	1 o 3
SBB mem, imm	100000sw   TD/R 011 Ind	SBB CX, 3	1	1 o 3
SBB acc, imm	0001110w   Dato	SBB AX, 3	1	1
SCASw	1010111w	SCASB / SCASW / SCASD	6	4
SETCond reg8	0F   1001cccc   TD/R 000 Ind	SETA BL	3	1 o 2
SETCond mem8	0F   1001cccc   TD/R 000 Ind	SETA DATAK / SETAE LESS_OR_SO	3	1
SGDT mem	0F   00000001   TD/R 000 Ind   Desp	SGDT MEMORY / SGDT GLOBAL	10	4
SIDT mem	0F   00000001   TD/R 001 Ind   Desp	SIDT DATAS / SIDT INTERRUPT	10	4
SLDT reg	0F   00000000   TD/R 000 Ind   Desp	SLDT CX / SLDT DX	2	2
SLDT mem	0F   00000000   TD/R 000 Ind   Desp	SLDT NUMBS / SLDT LOCALS	3	2
SHLD reg, reg, imm	0F   10100100   TD/R Reg Ind   Desp   Dato	SHLD AX, CX, 10 / SHLD DX, BX, 8	2	4

SHLD mem, reg, imm	0F   10100100   TD/R Reg Ind   Desp   Dato	SHLD DATOQ, CX, 10	3	4
SHRD reg, reg, imm	0F   10101100   TD/R Reg Ind   Desp   Dato	SHRD CX,DX,2	2	4
SHRD mem, reg, imm	0F   10101100   TD/R Reg Ind   Desp   Dato	SHRD DATOZ,DX,4	2	4
SHRD reg, reg, CL	0F   10100101   TD/R Reg Ind   Desp	SHRD BX,DX,CL	3	4 o 5
SHRD mem, reg, CL	0F   10100101   TD/R Reg Ind   Desp	SHRD DATOZ,DX,CL	3	4 o 5
SMSW reg	0F   TD/R 100 Ind   Desp	SMSW AX / SMSW DX / SMSW BP	2	4
SMSW mem	0F   TD/R 100 Ind   Desp	SMSW DATOQ	3	4
STC	11111001		2	2
STD	11111101		2	2
STI	11111011		2	7
STOSx	1010101w	STOSB / STOSW / STOSD / STOS DATO_LIST	5	3
STR reg	0F   00000000   TD/R 001 Ind   Desp	STR AX	2	2
STR mem	0F   00000000   TD/R 001 Ind   Desp	STR DATO3	2	2
SUB reg, reg	000101dw   TD/R Reg Ind   Desp	SUB CL, DL	1	1 ó 2
SUB mem, reg	000101dw   TD/R Reg Ind   Desp	SUB DATOJ, CL	3	1 ó 3
SUB acc, imm	0010110w   Dato	SUB AI,3 SUB AX,1AH SUB EAX,34H	1	1
SUB reg, mem	0010110w   Dato	SUB CL,DATOL / SUB CX,BYTES	2	1 o 2
SUB reg, imm	100000sw   TD/R 101 Ind   Desp   Dato	SUB CX,3 / SUB DI,1AH / SUB EDX,1345H	1	1 o 3
SUB mem, imm	100000sw   TD/R 101 Ind   Desp   Dato	SUB DATOS,3 / SUB BYTE PTR[EDI],1AH	3	1 o 3
TEST reg, imn	1111011sw   TD/R 000 Ind   Desp   Dato	TEST BX, 3 / TEST DI, 1AH / TEST DH, 44H	1	1 ó 2
TEST mem, imm	1111011sw   TD/R 000 Ind   Desp   Dato	TEST DATOS, 3 / TEST BYTE PT [EDI], 1AH	2	1 ó 2
TEST reg, reg	1000001w   TD/R Reg Ind   Desp	TEST CL, DL / TEST BX, DX / TEST EBP, EBX	1	1 ó 2
TEST mem, reg	1000001w   TD/R Reg Ind   Desp	TEST DATOJ, CL / TEST BYTES, CX	2	1 ó 2
TEST reg, mem	1000001w   TD/R Reg Ind   Desp	TEST DATOJ, CL / TEST BYTES, CX	2	1 ó 2
TEST acc, imm	1010100W   Dato	TEST AL,3 / TEST AX,1AH / TEST EAX,34H	1	1
VERR reg	0F   00000000   TD/R 100 Ind   Desp	VERR CX / VERR DX / VERR DI	11	7
VERR mem	0F   00000000   TD/R 100 Ind   Desp	VERR DATOJ / VERR TESTTB	11	7
VERW reg	0F   00000000   TD/R 101 Ind   Desp	VERW CX / VERW DX / VERW DI	11	7
VERW mem	0F   00000000   TD/R 100 Ind   Desp	VERW DATOJ / VERW TESTB	11	7
WAIT	10011011	WAIT / FWAIT	6	1
WBINVD	0F   00001001	WBINVD	5	2000+
WRMSR	0F   00110000	WRMSR	-	30-45
XADD reg, reg	0F   1100000w   11 Reg Reg	XADD EBX, ECX	3	3 o 4
XADD mem, reg	0F   1100000w   11 Reg Reg	XADD DATO5, ECX	4	3 o 4
XCHG reg, reg	100011w   TD/R Reg Ind	XCHG CL,DL	3	3
XCHG mem, reg	100011w   TD/R Reg Ind	XCHG DATAJ, CL / XCHG BYTES, CX	5	3
XCHG reg, mem	100011w   TD/R Reg Ind	XCHG DATAJ, CL / XCHG BYTES, CX	5	3
XCHG acc, reg	10010 Reg	XCHG AX, DI / XCHG EDX, EAX / XCHG SI, AX	3	2
XCHG reg, acc	10010 Reg	XCHG AX, DI / XCHG EDX, EAX / XCHG SI, AX	3	2
XLAT	10110111		4	4

XOR reg, reg	000110dw	TD/R Reg Ind	XOR CL,DL / XOR AX,DX / XOR CH,CL	1	1 o 2
XOR mem, reg	000110dw	TD/R Reg Ind	XOR DATOJ, CL / XOR BYTES, CX	3	1 o o3
XOR reg, mem	000110dw	TD/R Reg Ind	XOR CL, DATOL / XOR CX, BYTES	2	1 o 2
XOR reg, imm	100000sw	TD/R 110 Ind   Desp   Dato	XOR CX, 3 / XOR DI, 1AH XOR DL, 34H XOR EDX, 1345H XOR CX, 1834H	1	1 o 3
XOR mem, imm	100000sw	TD/R 110 Ind   Desp   Dato	XOR DATOS, 3 / XOR BYTE PTR[EDI], 1AH	3	1 o 3
XOR acc, imm	0010101w	Dato	XOR AL, 3 / XOR AX, 1AH / XOR EAX, 34H	1	1



**Instrucciones de ensamblaje:**

- La siguiente corresponde a la notación utilizada:

Operandos	
Campo	Descripción
acc	Registro acumulador (AL, AX, EAX)
cr/tr/dr	Registro de control/tarea/depuración
imm	Valor inmediato
mem	Referencia a lugar en memoria
puerto	Valor numérico correspondiente a un número de puerto
reg	Registro de 8, 16 o 32 bits (reg8/reg16/reg32 hacen referencia únicamente a registros de 8, 16 y 32 bits respectivamente)
seg	Registro de segmento

Argumentos	
Campo	Descripción
#Puerto	Valor hexadecimal correspondiente a un número de puerto
cccc	Código de 4 bits para condición *
Dato	Valor hexadecimal de un inmediato
Desp	Desplazamiento (Desp8/Desp16 indican desplazamientos de 8 y 16 bits respectivamente)
Ind	3 bits de tipo de direccionamiento *
Reg	3 bits de código de registro (reg8, reg16 y reg32 indican de forma explícita únicamente registro de 8, 16 y 32 bits) *
sss	Código de 3 bits de registro de segmento *
TD/R	2 bits de modo de direccionamiento *
ttt	Código de 3 bits para tipo (ver tabla)

- En las instrucciones donde no se indique el bit “d” (dirección), el campo “Reg” es por defecto el destino.
- PRECAUCIÓN: Corregir Instrucciones Jcond / JMP Adelante

\* Códigos se encuentran en el documento “Referencias para el ensamblaje de instrucciones Intel iA16-iA32”