

Instrucciones x86

INSTRUCCION	Descripcion	Complementos	Tipo
MOVx op1, op2	op2 <- op1	LongWord, Word, Byte	Movimiento
MOV\$xy op1, op2	op2 <-Extsign(op1)	BaW, BaL, WaL	Movimiento
MOVZ\$xy op1, op2	op2 <-Extzero(op1)	BaW, BaL, WaL	Movimiento
PUSHL op1	%esp <= %esp - 4, M[%esp] <= op1		Empilar
POPL op1	op1 <= M[%esp], %esp <= %esp + 4		Desempilar
LEAL op1, op2	op2 <= &op1	op1: @M	Aritmetica
ADDx op1, op2	op2 <= op2+op1	L, W, B	Aritmetica
SUBx op1, op2	op2 <= op2-op1	L, W, B	Aritmetica
ADCx op1, op2	op2 <= op2 + op1 + CF	L, W, B	Aritmetica
SBBx op1, op2	op2 <= op2 - op1 - CF	L, W, B	Aritmetica
INCx op1	op1 += 1	L, W, B	Aritmetica
DECx	op1 -= 1	L, W, B	Aritmetica
NEGx	op1 = - op1	L, W, B	Aritmetica
IMUL op1, op2	op2 <= op2 * op1	op2: %	Aritmetica
IMUL inm, op1, op2	op2 <= op1 * inm	inm: \$,	Multiplicacion
IMULL op1	%ext <= op1 * %eax	op1: @M o %	Multiplicacion Enteros
MULL op1	%ext <= op1 * %eax	op1: @M o %	Multiplicacion Naturales
CLTD	%ext : ExtSign(%eax)		Extension Signo
IDIVL op1	%eax <= %ext / op1, %edx <= %ext % op1	op1: @M o %	Division Enteros
DIVL op1	%eax <= %ext / op1, %edx <= %ext % op1	op1: @M o %	Division Naturales
ANDx op1,op2	op2 <= op1 & op1	L, W, B	Lógicas
ORx op1, op2	op2 <= op2 op1	L, W, B	Lógicas

INSTRUCCION	Descripcion	Complementos	Tipo
XORx op1, op2	$op2 \leftarrow op2 \wedge op1$	L, W, B	Lógicas
NOTx op1	$op1 \leftarrow ! op1$	L, W, B; k: \$	Lógicas
SALx k, op1	$op1 \leftarrow op1 \ll k$	L, W, B; k: \$	Lógicas
SHLx k, op1	$op1 \leftarrow op1 \ll k$	L, W, B; k: \$	Lógicas
SARx k, op1	$op1 \leftarrow op1 \gg k$	L, W, B; k: \$	Lógicas
SHRx k, op1	$op1 \leftarrow op1 \gg k$	L, W, B; k: \$	Lógicas
CMPx op1, op2	$op2 - op1$	L, W, B, mod. flags	Lógicas
TESTx op1, op2	$op2 \& op1$	L, W, B i mod. flags	$op1 == op2 ?$
JMP etiq	$\%eip \leftarrow @etiq$	&etiq	Salta Incondicional
JMP op	$\%eip \leftarrow op$	op es @	Salta Incondicional
Jcc etiq	$\%eip \leftarrow etiq \text{ (if)}$	cc: E, NE, G, GE, L, LE	Salta condicional Enteros
Jcc etiq	$\%eip \leftarrow etiq \text{ (if)}$	cc: A, AE, B, BE	Salta condicional Naturales
Jcc etiq	$\%eip \leftarrow etiq \text{ (if)}$	cc: Z, NZ, C, NC, O	Salta condicional flags
CALL etiq	$\%esp \leftarrow \%esp - 4; M[\%esp] \leftarrow EIP; \%eip \leftarrow \&etiq$	Guarda @ret i PC = & etiq	Lllamar f(x)
CALL op	$\%esp \leftarrow \%esp - 4; M[\%esp] \leftarrow EIP; \%eip \leftarrow \&etiq$	Guarda @ret i PC = & op	Lllamar f(x)
RET	$\%eip \leftarrow M[\%esp]; \%esp \leftarrow \%esp + 4$		Retorna

$\%ext = \%edx : \%eax$

Tabla de Flags	
JE	Jump equal
JNE	Jump not equal
JS	Jump Negative
JNS	Jump not Negative
JG	Jump Greater (signed)
JGE	Jump Greater or equal (signed)
JL	Jump Less (signed)
JLE	Jump Less or equal (signed)
JA	Jump Greater (not signed)
JAЕ	Jump Greater or equal (not signed)
JB	Jump Less (not signed)
JBE	Jump Less or equal (not signed)