MOVIMIENTO DE DATOS

MNEMOTÉCNICO	SINTAXIS	TAMAÑO	FUNCIÓN	XNZVC
EXG	EXG Rm, Rn	L	RM↔RN	-----
LEA	LEA <cea>, An	L	dirección fuente→An	-----
LINK	LINK An, #<d16>	No	An→-(SP); SP→An SP+d16→SP	-----
MOVE	MOVE.t<ea>,<adea>	L,W,B	(fuente)→destino	-**00
MOVE from SR	MOVE SR,<adea>	W	SR→destino	-----
MOVE to CCR	MOVE <adea>,CCR	W	(fuente)→CCR	*****
MOVE to SR	MOVE <adea>,SR	W	IF S=1 (fuente)→SR ELSE EXCEPCIÓN	*****
MOVE USP	MOVE USP,An	L	IF S=1 USP→An ELSE EXCEPCIÓN	*****
MOVE USP	MOVE An,USP	L	IF S=1 An→USP ELSE EXCEPCIÓN	*****
MOVEA	MOVEA.t<ea>,An	L,W	(fuente)→destino	-----
MOVEM	MOVEM.t <lista_reg>,<cea> MOVEM.t <lista_reg>,-(An)	L,W	registros→destino	-----
MOVEM	MOVEM.t <cea>,<lista_reg> MOVEM.t(An)+,<lista_reg>	L,W	(fuente→registros	-----
MOVEP	MOVEP.t Dn,d(Am) MOVEP.t d(Am),Dn	L,W	(fuente)→destino	-----
MOVEQ	MOVEQ #<d8>,Dn	L	dato (8 bits)→destino	-**00
PEA	PEA <cea>	L	dirección→-(SP)	-----
SWAP	SWAP Dn	W	Dn[31:16]↔Dn[15:0]	-**00
UNLK	UNLK An	No	An→SP; (SP)+→An	-----

ARITMÉTICA ENTERA

MNEMOTÉCNICO	SINTAXIS	TAMAÑO	FUNCIÓN	XNZVC
ADD	ADD.t <ea>,Dn ADD.t Dn, <amea>	L,W,B	(fuente)+(destino)→destino	*****
ADDA	ADDA.t <ea>,An	L,W	(fuente)+(destino)→destino	-----
ADDI	ADDI.t # <dato>,<adea>	L,W,B	Dato+(destino)→destino	*****
ADDQ	ADDQ.t# <d3>, <aea>	L,W,B	d3+(destino)→destino	*****
ADDX	ADDX.t Dm,Dn o ADDX.t -(Am),-(An)	L,W,B	(fuente)+(destino)+X→ destino	*****
CLR	CLR.t <adea>	L,W,B	0→destino	-0100
CMP	CMP.t <ea>,Dn	L,W,B	(destino)-(fuente);activa CCR	-****
CMPA	CMPA.t <ea>,An	L,W	(destino)-(fuente);activa CCR	-****
CMPI	CMPI.t #<Dato>,<adea>	L,W,B	(destino)-Dato;activa CCR	-****
CMPM	CMPM.t (Am)+,(An)	L,W,B	(destino)-(fuente);activa CCR	-****
DIVS	DIVS <dea>,Dn	W	(destino)/(fuente)→destino Cociente→destino[15:0], Resto→destino[31:16]	-***0
DIVU	DIVU <dea>,Dn	W	(destino)/(fuente)→destno; Cociente→destino[15:0], Resto→destino[31:16]	-***0
EXT	EXT.t Dn	L,W	Dn(Extend en signo)→Dn	-**00
MULS	MULS <dea>,Dn	W	(fuente)*(destino)→destino	-**00
MULU	MULU <dea>,Dn	W	(fuente)*(destino)→destino	-**00
NEG	NEG.t <adea>	L,W,B	0-(destino)→destino	*****
NEGX	NEGX.t <adea>	L,W,B	0-(destino)-X→destino	*****
SUB	SUB.t <ea>,Dn SUB.t Dn, <amea>	L,W,B	(destino)-(fuente)→destino	*****
SUBA	SUBA.t <ea>,An	L,W	(destino)-(fuente)→destino	-----
SUBI	SUBI.t # <dato>,<adea>	L,W,B	(destino) Dato→destino	*****
SUBQ	SUBQ.t# <d3>, <aea>	L,W,B	(destino)-d3→destino	*****
SUBX	SUBX.t Dm,Dn SUBX.t -(Am),-(An)	L,W,B	(destino)-(fuente)-X → destino	*****
TAS	TAS <adea>	B	Test (destino)→CCR; 1→destino[bit7]	-**00
TST	TST <adea>	L,W,B	Test (destino)→CCR	-**00

OPERACIONES DE CONTROL DE PROGRAMA

MNEMOTÉCNICO	SINTAXIS	TAMAÑO	FUNCIÓN	XNZVC
Bcc	Bcc <etiqueta>	Desplaz. 16 bits	IF cc True PC+despl.→PC	-----
Bcc.S	Bcc.S <etiqueta>	Desplaz. 8 bits	IF cc True PC+despl.→PC	-----
BRA	BRA <etiqueta>	Despl. 8 o 16 bits	PC+despl.→PC	-----
BSR	BSR <etiqueta>	Despl. 8 o 16 bits	PC→-(SP); PC+despl.→PC	-----
DBcc	DBccDm, <etiqueta>	Desplaz. 16 bits	IF cc False Dm-1→Dm IF Dm≠-1 PC+despl.→PC ELSE PC+2→PC (instr. sig.)	-----
JMP	JMP <cea>	No	destino→PC	-----
JSR	JSR <cea>	No	PC→-(SP); destino→PC	-----
Scc	Scc <adea>	B	IF cc True $FF→destino ELSE $00→destino	-----
RTR	RTR	No	(SP)+→CCR; (SP)+→PC	*****
RTS	RTS	No	(SP)+→PC	-----

OPERACIONES LÓGICAS

MNEMOTÉCNICO	SINTAXIS	TAMAÑO	FUNCIÓN	XNZVC
AND	AND.t <dea>,Dn AND.t Dn, <amea>	L,W,B	(fuente)<AND>(destino)→ destino	- **00
ANDI	ANDI.t #<dato>,<adea>	L,W,B	dato<AND>(destino)→ destino	-**00
ANDI to CCR	ANDI #<d8>,CCR	B	d8<AND>CCR→CCR	*****
ANDI to SR*	ANDI #<d16>,SR	W	IF S=1 d16 <AND>SR→SR ELSE EXCEPCIÓN	*****
EOR	EOR.t Dn,<adea>	L,W,B	(fuente)<XOR>(destino)→ destino	-**00
EORI	EORI.t #<dato>,<adea>	L,W,B	dato<XOR>(destino)→ destino	-**00
EORI to CCR	EORI #<d8>,CCR	B	d8<XOR>CCR→CCR	*****
EORI to SR*	EORI #<d16>,SR	W	IF S=1 d16<XOR>SR→SR ELSE EXCEPCIÓN	*****
NOT	NOT.t <adea>	L,W,B	<NOT>(destino)→destino	-**00
OR	OR.t <dea>,Dn OR.t Dn,<amea>	L,W,B	(fuente)<OR>(destino)→ destino	-**00
ORI	ORI.T #<dato>,<adea>	L,W,B	dato<OR>(destino)→destino	-**00
ORI to CCR	ORI #<d8>,CCR	B	d8<OR>CCR→CCR	*****
ORI to SR*	ORI #<d16>,SR	W	IF S=1 d16<OR>SR→SR ELSE EXCEPCIÓN	*****

DESPLAZAMIENTOS Y ROTACIONES

MNEMOTÉCNICO	SINTAXIS	TAMAÑO	FUNCIÓN	XNZVC
ASL/ASR	ASL/ASR Dm,Dn ASL/ASR #<d3>,Dn	L,W,B	(destino) desplaz. aritmét. de <num> bits→destino	*****
ASL/ASR	ASL/ASR <amea>	W (conta- dor des- plaz.=1)	(destino) desplaz. aritmét. de 1 bit→destino	*****
LSL/LSR	Igual que ASL/ASR ASL/ASR	Igual que ASL/ASR	(destino) desplaz. lógico de <num> (o 1) bits→ destino	****0*
ROL/ROR	Igual que ASL/ASR	Igual que ASL/ASR	(destino) rotado <num> (o 1) bits→destino	-**0*
ROXL/ROXR	Igual que ASL/ASR	Igual que ASL/ASR	(destino) rotado (con exten.) <num> (o 1) bits→destino	***0*

OPERACIONES EN BCD

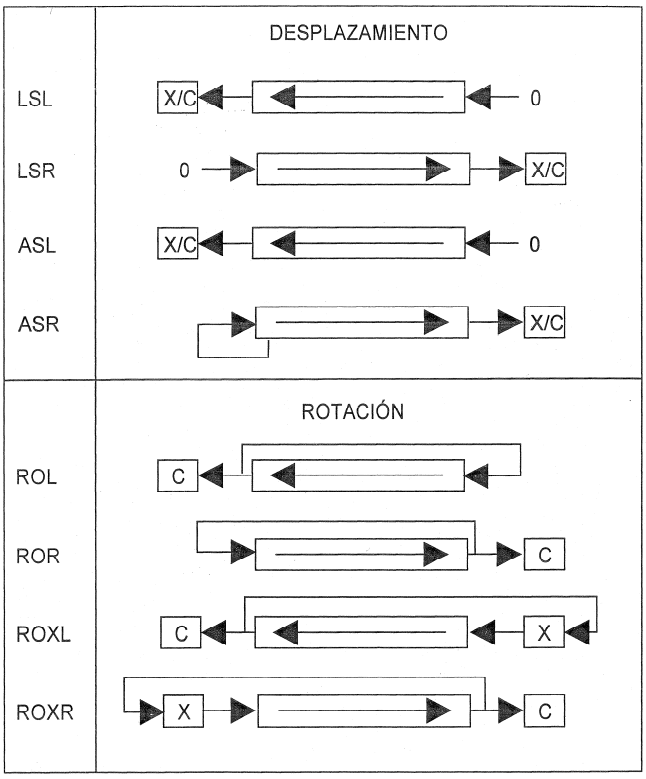
MNEMOTÉCNICO	SINTAXIS	TAMAÑO	FUNCIÓN	XNZVC
ABCD	ABCD Dm,Dn ABCD -(Am),-(An)	B	$(fuente_{10})+(destino_{10})+X\rightarrow destino_{10}$	*U*U*
NBCD	NBCD <adea>	B	$0-(destino_{10})-X\rightarrow destino_{10}$	*U*U*
SBCD	SBCD Dm,Dn SBCD -(Am),-(An)	B	$(fuente_{10})-(destino_{10})-X\rightarrow destino_{10}$	*U*U*

MANIPULACIÓN DE BITS

MNEMOTÉCNICO	SINTAXIS	TAMAÑO	FUNCIÓN	XNZVC
BCHG	BCHG Dm,Dn o BCHG #<d8>,Dn	L (bit mod 32)	<NOT>(destino[bit])→flag Z <NOT>(destino[bit])→ destino[bit]	– –*– –
BCHG	BCHG Dm,<amea> o BCHG #<d8>,<amea>	B (bit mod 8)	Idem	– –*– –
BCLR	Igual que BCHG		<NOT>(destino)[bit])→flag Z 0→destino[bit]	– –*– –
BSET	Igual que BCHG		<NOT>(destino)[bit])→flag Z 1→destino[bit]	– –*– –
BTST	BTST Dm,Dn o BTST #<d8>,Dn	L (bit mod 32)	<NOT>(destino)[bit])→flag Z	– –*– –
BTST	BTST DM,<mea> o BTST #<d8>,<mea>	B (bit mod 8)	Idem	– –*– –

CÓDIGO DE CONDICIÓN	
*	Afectado.
–	No afectado.
0	Puesto a 0 (clear).
1	Puesto a 1 (set).
U	No definido.

ea Cualquier dirección efectiva.
dea Dirección efectiva de datos.
mea Dirección efectiva de memoria.
cea Dirección efectiva de control.
aea Dirección efectiva alterable (datos o memoria).
adea Dirección efectiva alterable de datos.
amea Dirección efectiva alterable de memoria.
acea Dirección efectiva alterable de control.



CONDICIÓN	INSTRUCCIÓN QUE LA EVALÚA	CÓDIGO PARA BIFURCAR SI SE CUMPLE ESTA CONDICIÓN	
		CON SIGNO	SIN SIGNO
A > B	CMP B,A (1)	GT	HI
A >= B	CMP B,A (1)	GE	CC
A < B	CMP B,A (1)	LT	CS
A <= B	CMP B,A (1)	LE	LS
A = B	CMP B,A	EQ	EQ
A <> B	CMP B,A	NE	NE
A es positivo	TST A (2)	PL	—
A es negativo	(2)	MI	—
Desbordamiento	(3)	VS	CS
No desbordamiento	(3)	VC	CC

(1) Si se usa CMP, el orden de los términos en la condición tiene que ser el contrario al de los operandos de la instrucción de comparación. Se pueden usar otras instrucciones para activar los flags de condición.

(2) La variable lógica "A es positivo" ("A es negativo") es activada por muchas instrucciones. Por ejemplo, cuando se suman dos números, se activa la variable si el resultado es mayor o igual que cero (menor que cero respectivamente).

(3) La variable lógica desbordamiento es activada por muchas instrucciones, como las instrucciones aritméticas, cuando el resultado de la operación excede el rango de números representable para ese tamaño y esa aritmética (con o sin signo).

OPERACIONES DE CONTROL DEL SISTEMA

MNEMOTÉCNICO	SINTAXIS	TAMAÑO	FUNCIÓN	XNZVC
RESET*	RESET	No	IF S=1 activa RESET ELSE EXCEPCIÓN	– – – – –
RTE	RTE	No	IF S=1 (SP)+→SP; (SP)+→PC ELSE EXCEPCIÓN	*****
STOP*	STOP #<d16>	No	IF S=1 d16→SR; PC+4→PC; Pausa hasta excepción ELSE EXCEPCIÓN	*****
CHK	CHK <dea>,Dn	W	IF Dn<0 OR Dn>(fuente) EXCEPCIÓN	–*UUU
ILLEGAL	ILLEGAL	No	PC→–(SSP); SR→–(SSP) Vector #4→PC	– – – – –
TRAP	TRAP #<d4>	No	PC→–(SSP); SR→–(SSP) Vector #4→PC	– – – – –
TRAPV	TRAPV	No	IF V=1 EXCEPCIÓN	– – – – –
NOP	NOP	No	PC+2→PC	– – – – –

Clasificación de los modos de direccionamiento en categorías

MODO DE DIRECCIONAMIENTO	CATEGORÍAS DE EA							
	EA	DEA	MEA	CEA	AEA	ADEA	AMEA	ACEA
Dn	X	X			X	X		
An	X				X			
(An)	X	X	X	X	X	X	X	X
(An)+	X	X	X		X	X	X	
–(An)	X	X	X		X	X	X	
d_{16}(An)	X	X	X	X	X	X	X	X
d_8(An,Ri.X)	X	X	X	X	X	X	X	X
XXXX	X	X	X	X	X	X	X	X
XXXXXX	X	X	X	X	X	X	X	X
d_{16}(PC)	X	X	X	X				
d_8(PC,Ri.X)	X	X	X	X				
#XXXXXXXX	X	X	X					

Aritmética con signo

CONDICIÓN	SIGNIFICADO	CÁLCULO
GT	"Greater than" - mayor que	Z + (N xor V) = 0
LT	"Less than" - menor que	N xor V = 1
GE	"Greater or equal" - mayor o igual	N xor V = 0
LE	"Less or equal" - menor o igual	Z + (N xor V) = 1
VS	"Overflow" - desbordamiento	V = 1
VC	"No overflow" - no desbordamiento	V = 0
PL	"Plus" - más	N = 0
MI	"Minus" - menos	N = 1

Aritmética sin signo

CONDICIÓN	SIGNIFICADO	CÁLCULO
HI	"Higher" - mayor	Z + C = 0
CS	"Carry set" - menor	C = 1
CC	"Carry clear" - mayor o igual	C = 0
LS	"Low or same" - menor o igual	Z + C = 1

Aritmética con signo o sin signo

CONDICIÓN	SIGNIFICADO	CÁLCULO
EQ	"Equal" - igual	Z = 1
NE	"No equal" - no igual	Z = 0
T	"True" - siempre cierto	1
F	"False" - siempre falso	0