SIC, SIC/XE Instruction Table

Mnemonic	Format	Opcode	Effect	Notes
ADD m	0/3/4	18	A <-- (A) + (m..m+2)	
ADDF m	3/4	58	F <-- (F) + (m..m+5)	X F
ADDR r1,r2	2	90	r2 <-- (r2) + (r1)	X
AND m	0/3/4	40	A <-- (A) & (m..m+2)	
CLEAR r1	2	B4	r1 <-- 0	X
COMP m	0/3/4	28	A : (m..m+2)	
COMPF m	3/4	88	F : (m..m+5)	X F C
COMPR r1,r2	2	A0	(r1) : (r2)	X F C
DIV m	0/3/4	24	A : (A) / (m..m+2)	
DIVF m	3/4	64	F : (F) / (m..m+5)	X F
DIVR r1,r2	2	9C	(r2) <-- (r2) / (r1)	X
FIX	1	C4	A <-- (F) [convert to integer]	X F
FLOAT	1	C0	F <-- (A) [convert to floating]	X F
HIO	1	F4	Halt I/O channel number (A)	P X
J m	0/3/4	3C	PC <-- m	
JEQ m	0/3/4	30	PC <-- m if CC set to =	
JGT m	0/3/4	34	PC <-- m if CC set to >	
JLT m	0/3/4	38	PC <-- m if CC set to <	
JSUB m	0/3/4	48	L <-- (PC); PC <-- m	
LDA m	0/3/4	00	A <-- (m..m+2)	
LDB m	3/4	68	B <-- (m..m+2)	X
LDCH m	0/3/4	50	A [rightmost byte] <-- (m)	
LDF m	3/4	70	F <-- (m..m+5)	X F
LDL m	0/3/4	08	L <-- (m..m+2)	
LDS m	3/4	6C	S <-- (m..m+2)	X
LDT m	3/4	74	T <-- (m..m+2)	X
LDX m	0/3/4	04	X <-- (m..m+2)	
LPS m	3/4	D0	Load processor status from information beginning at address m (see Section 6.2.1)	P X
MUL m	0/3/4	20	A <-- (A) * (m..m+2)	
MULF m	3/4	60	F <-- (F) * (m..m+5)	X F
MULR r1,r2	2	98	r2 <-- (r2) * (r1)	X
NORM	1	C8	F <-- (F) [normalized]	X F
OR m	0/3/4	44	A <-- (A) | (m..m+2)	
RD m	0/3/4	D8	A [rightmost byte] <-- data from device specified by (m)	P
RMO r1,r2	2	AC	r2 <-- (r1)	X
RSUB	0/3/4	4C	PC <-- (L)	
SHIFTL r1,n	2	A4	r1 <-- (r1); left circular shift n bits. {In assembled instruction, r2=n-1}	X
SHIFTR r1,n	2	A8	r1 <-- (r1); right shift n bits with vacated bit positions set equal to leftmost bit of (r1). {In assembled instruction, r2=n-1}	X
SIO	1	F0	Start I/O channel number (A); address of channel program is given by (S)	P X
SSK m	3/4	EC	Protection key for address m <-- (A) (see Section 6.2.4)	P X
STA m	0/3/4	0C	m..m+2 <-- (A)	
STB m	3/4	78	m..m+2 <-- (B)	X
STCH m	0/3/4	54	m <-- (A) [rightmost byte]	
STF m	3/4	80	m..m+5 <-- (F)	X

```
STI m           3/4       D4    Interval timer value <--            P X
                                  (m..m+2) (see Section
                                  6.2.1)
STL m           0/3/4     14    m..m+2 <-- (L)
STS m           3/4       7C    m..m+2 <-- (S)                        X
STSW m          0/3/4     E8    m..m+2 <-- (SW)                     P
STT m           3/4       84    m..m+2 <-- (T)                        X
STX m           0/3/4     10    m..m+2 <-- (X)
SUB m           0/3/4     1C    A <-- (A) - (m..m+2)
SUBF m          3/4       5C    F <-- (F) - (m..m+5)                  X F
SUBR r1,r2       2        94    r2 <-- (r2) - (r1)                    X
SVC n            2        B0    Generate SVC interrupt. {In           X
                                  assembled instruction, r1=n}
TD m            0/3/4     E0    Test device specified by (m)        P     C
TIO              1        F8    Test I/O channel number (A)         P X   C
TIX m           0/3/4     2C    X <-- (X) + 1; (X) : (m..m+2)             C
TIXR r1          2        B8    X <-- (X) + 1; (X) : (r1)             X   C
WD m            0/3/4     DC    Device specified by (m) <-- (A)     P
                                  [rightmost byte]
------------  ------  ------  ---------------------------------  ------
```

Notes column:

P Privileged instruction

X Instruction available only on SIC/XE version

F Floating point instructions

C Condition code CC set to indicate result of operation (one of <, =, or >). That is when such an instruction, for instance “COMP m” executes, the result of (A):(m..m+2)

From *Systems Programming* by Leland Beck, 3rd edition, pages 497-498