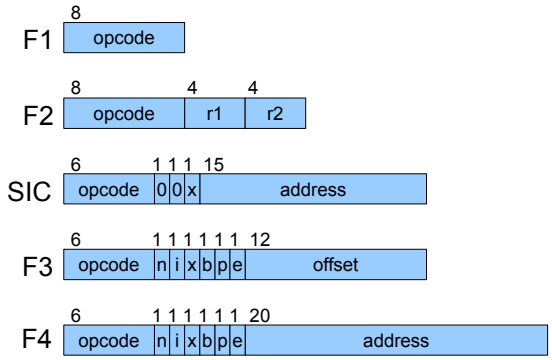
Mnemonic	Opcode	Format	Effect
ADD m	18	S/3/4	A ← (A) + (m..m+2)
ADDF m	58	3/4	F ← (F) + (m..m+5)
ADDR r1, r2	90	2	r2 ← (r2) + (r1)
AND m	40	S/3/4	A ← (A) & (m..m+2)
CLEAR r1	B4	2	r ← 0
COMP m	28	S/3/4	(A) : (m..m+2)
COMPF	88	3/4	(F) : (m..m+5)
COMPR	A0	2	(r1) : (r2)
DIV	24	S/3/4	A ← (A) / (m..m+2)
DIVF	64	3/4	F ← (F) / (m..m+2)
DIVR r1, r2	9C	2	r2 ← (r2) / (r1)
FIX	C4	1	A ← int(F)
FLOAT	C0	1	F ← float(A)
HIO	F4	1	haltio(A)
J m	3C	S/3/4	PC ← m
JEQ m	30	S/3/4	PC ← m if CC is = Text
JGT m	34	S/3/4	PC ← m if CC is >
JLT m	38	S/3/4	PC ← m if CC is <
JSUB m	48	S/3/4	L ← (PC); PC ← m
LDA m	00	S/3/4	A ← (m..m+2)
LDB m	68	3/4	B ← (m..m+2)
LDCH m	50	S/3/4	A.low ← (m)
LDF m	70	3/4	F ← (m..m+5)
LDL m	08	S/3/4	L ← (m..m+2)
LDS m	6C	S/3/4	S ← (m..m+2)
LDT m	74	S/3/4	T ← (m..m+2)
LDX m	04	S/3/4	X ← (m..m+2)
LPS m	D0	3/4	PS ← (m..2)
MUL m	20	S/3/4	A ← (A) * (m..m+2)
MULF m	60	3/4	F ← (F) * (m..m+5)
MULR r1, r2	98	2	r2 ← (r2) * (r1)
NORM	C8	1	F ← norm(F)
OR m	44	S/3/4	A ← (A) | (m..m+2)
RD m	D8	S/3/4	A.low ← readdev(m)
RMO r1, r2	AC	2	(r2) ← (r1)
RSUB	4C	S/3/4	PC ← (L)
SHIFTL r1, n	A4	2	(r1) ← (r1) << n
SHIFTR r2, n	A8	2	(r1) ← (r1) >> n
SIO	F0	1	startio(A, S)
SSK m	EC	3/4	...
STA m	0C	S/3/4	m..m+2 ← (A)
STB m	78	3/4	m..m+2 ← (B)
STCH m	54	S/3/4	m ← (A.low)
STF m	80	3/4	m..m+5 ← (F)
STI	D4	3/4	timer ← (m..m+2)
STL m	14	S/3/4	m..m+2 ← (L)
STS m	7C	3/4	m..m+2 ← (S)
STSW m	E8	S/3/4	m..m+2 ← (SW)
STT m	84	3/4	m..m+2 ← (T)
STX m	10	S/3/4	m..m+2 ← (X)
SUB m	1C	S/3/4	A ← (A) - (m..m+2)
SUBF m	5C	3/4	F ← (F) - (m..m+5)
SUBR r1, r2	94	2	r2 ← (r2) - (r1)
SVC n	B0	2	interrupt(n)
TD m	E0	S/3/4	testdev(m)
TIO	F8	1	testio(A)
TIX m	2C	S/3/4	X ← (X) + 1; (X) : (m..m+2)
TIXR r1	B8	2	X ← (X) + 1; (X) : (r)
WD m	DC	S/3/4	writedev(m, A.low)

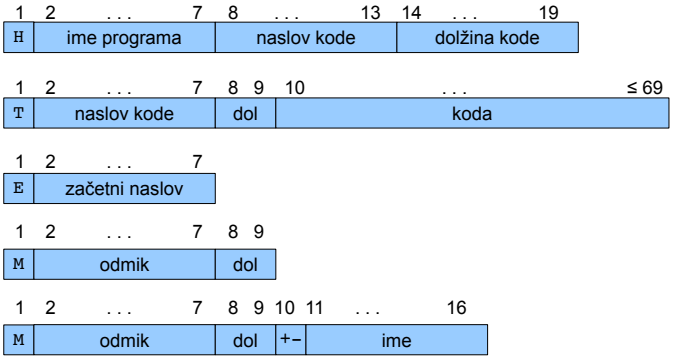
Formati ukazov



b	p	vrsta
1	0	bazno
0	1	pc-relativno
0	0	neposredno
1	1	nedovoljeno

n	i	vrsta
1	0	posredno
0	1	takojšnje
1	1	enostavno
0	0	SIC format

Objektna datoteka



Številke registrov

A	X	L	B	S	T	F
0	1	2	3	4	5	6

	SIC	SIC/XE
Pomnilnik	32 kB (2^{15})	1 MB (2^{20})
Dolžina besede	3 B	=
Cilj naslova	LSB	=
Vrste podatkov	24 bit ints, ASCII	+ 48 bit floats
Registri	A, X, L, PC, SW	+ B, S, T, F
Formati ukazov	SIC (opcode,x,addr)	+ F1 (opcode), F2 (opcode,r1,r2), F3 (opcode,nixbpe,offset), F4 (opcode,nixbpe,addr)
Način izračuna TA	TA = addr TA = addr + (X)	TA = addr TA = addr + (X) TA = (B) + addr TA = (B) + addr + (X) TA = (PC) + addr TA = (PC) + addr + (X)
Načini uporabe TA	(TA)	ni=01: TA … takojšnje ni=10: ((TA)) … posredno ni=11: (TA) ...preprosto ni=00: ukaz v SIC formatu

Način naslavljanja	Biti `nixbpe`	Zbirniški ukaz	UN	Operand	Opombe
Preprost	`110000`	`op c`	odmik	(UN)	`D`
	`110001`	`+op m`	naslov	(UN)	`4D`
	`110010`	`op m`	(PC) + odmik	(UN)	`A`
	`110100`	`op m`	(B) + odmik	(UN)	`A`
	`111000`	`op c,X`	odmik + (X)	(UN)	`D`
	`111001`	`+op c,X`	naslov + (X)	(UN)	`4D`
	`111010`	`op m,X`	(PC) + odmik + (X)	(UN)	`A`
	`111100`	`op m,X`	(B) + odmik + (X)	(UN)	`A`
	`000---`	`op m`	b/p/e/odmik	(UN)	`D S`
	`001---`	`op m,X`	b/p/e/odmik + (X)	(UN)	`D S`
Posredno	`100000`	`op @c`	odmik	((UN))	`D`
	`100001`	`+op @m`	naslov	((UN))	`4D`
	`100010`	`op @m`	(PC) + odmik	((UN))	`A`
	`100100`	`op @m`	(B) + odmik	((UN))	`A`
Takojšnje	`010000`	`op #c`	odmik	UN	`D`
	`010001`	`+op #m`	naslov	UN	`4D`
	`010010`	`op #m`	(PC) + odmik	UN	`A`
	`010100`	`op #m`	(B) + odmik	UN	`A`

Legenda:
c … konstanta med 0 in 4095
m … naslov (ali konstanta) večja od 4095
4 … format 4
D … neposredno naslavljanje
A … zbirnik izbere med PC-relativnim in bazno-relativnim naslavljanjem
S … kompatibilno z standardnim SIC