FOUR-PHASE SYSTEMS, INC.®

SYSTEM IV/70 Machine Instruction Card

OPERATION CODES

00x	HLT	10x	ORM	20x	ANM	30x	XOM
01x	LDA1	11x	INR	21x	DEC	31x	STA1
02x	LD23	12x	ADM	22x	SKN	32x	ST23
03x	LDA	13x	ADA,ADD	23x	SBA,SUB	33x	CPA
04x	LDB	14x	ORA,OR	24x	ANA,AND	34x	XOA,XOR
05x	LD1	15x	AD1	25x	SB1	35x	CP1
06x	LD2	16x	AD2	26x	SB2	36x	CP2
07x	LD3	17x	AD3	27x	SB3	37x	CP3
40x	STZ	50x	SLR	60x	SKZ	70x	XEC
41x	SAM	51x	SLRD	61x	MCC	71x	BRM
42x	STP	52x	SLA	62x	BOF	72x	BRA
43x	STA	53x	SLAD	63x	BZO	73x	BNZ
44x	STB	54x	SRL	64x	BMI	74x	BPL
45x	ST1	55x	SRLD	65x	BCR	75x	BC1
46x	ST2	56x	SRA	66x	BAL	76x	BC2
47x	ST3	57x	SRAD	67x	BGT	77x	BC3
007	LCL	107	ROR	207	RAND	307	RXOR
017	LPL	117	RADD,RCC	217	RSUB	317	RCM2
027	RCL	127	MPY	227	DIV	327	POP
037	RLC	137	MVL	237	MVE	337	UP
047	LCR	147	UFA	247	CDA2	347	IN
057	LPR	157	FAD	257	FSB	357	TRT
067	RCR,RCPY,NOP	167	FMP	267	FDV	367	†
077	RRC	177	MVR	277	IOB	377	BOOT
407	SCL	507	BRD	607	CPN		
417	SPL	517	BRR	617	MVCR		
427	PUSH	527	EXCT	627	DADD		
437	DOWN	537	EXSN	637	CPL		
447	SCR	547	PIA	647	DSUB		
457	SPR	557	PID	657	ODD		
467	TRAP	567	PIR	667	MVCL		
477	ECS	577	IOID	677	IO		

† Unpredictable results will be obtained if this op code is executed.

MNEMONICS

ADA	13x E	DADD	627 A	MVCR	617 A	SCR	447 B
ADD	13x E	DEC	21x I	MVE	237 B	SKN	22x I
ADM	12x E	DIV	227 E	MVL	137 B	SKZ	60x I
AD1	15x E	DOWN	437 C	MVR	177 B	SLA	52x H
AD2	16x E	DSUB	647 A	NOP	067 L	SLAD	53x H
AD3	17x E	ECS	477 N	ODD	657 L	SLR	50x H
ANA	24x K	EXCT	527 N	OR	14x K	SLRD	51x H
AND	24x K	EXSN	537 N	ORA	14x K	SPL	417 B
ANM	20x K	FAD	157 F	ORM	10x K	SPR	457 B
BAL	66x I	FDV	267 F	PIA	547 M	SRA	56x H
BCR	65x I	FMP	167 F	PID	557 M	SRAD	57x H
BC1	75x I	FSB	257 F	PIR	567 M	SRL	54x H
BC2	76x I	HLT	00x L	POP	327 C	SRLD	55x H
BC3	77x I	IN	347 C	PUSH	427 C	STA	43x D
BGT	67x I	INR	11x I	RADD	117 J	STA1	31x D
BMI	64x I	IO	677 N	RAND	207 J	STB	44x D
BNZ	73x I	IOB	277 N	RCC	117 L	STP	42x D
BOF	62x I	IOID	577 M	RCL	027 J	STZ	40x D
BOOT	377 N	LCL	007 B	RCM2	317 J	ST1	45x D
BPL	74x I	LCR	047 B	RCPY	067 J	ST2	46x D
BRA	72x I	LDA	03x D	RCR	067 J	ST3	47x D
BRD	507 I	LDA1	01x D	RLC	037 J	ST23	32x D
BRM	71x I	LDB	04x D	ROR	107 J	SUB	23x E
BRR	517 I	LD1	05x D	RRC	077 J	TRAP	467 L
BZO	63x I	LD2	06x D	RSUB	217 J	TRT	357 B
CDA2	247 J	LD3	07x D	RXOR	307 J	UFA	147 F
CPA	33x G	LD23	02x D	SAM	41x D	UP	337 C
CPL	637 A	LPL	017 B	SBA	23x E	XEC	70x L
CPN	607 A	LPR	057 B	SB1	25x E	XOA	34x K
CP1	35x G	MCC	61x L	SB2	26x E	XOM	30x K
CP2	36x G	MPY	127 E	SB3	27x E	XOR	34x K
CP3	37x G	MVCL	667 A	SCL	407 B		

Assembler Format ①	Octal Code	Condition Codes	Name	Timing ②
A DECIMAL OPTION				
1 CPL sbs,sbd,L	637	ZMC	Compare block Logic	14+6Q
1 CPN sbs,sbd,L1,L2	607	OZMC	Compare block Numeric	16+6P
1 DADD sbs,sbd,L1,L2	627	OZMC	Decimal Addition	18+8P(Rec,+6+6P)
1 DSUB sbs,sbd,L1,L2	647	OZMC	Decimal Subtraction	18+8P(Rec,+6+6P)
1 MVCR sbs,sbd,L	617	— —	Move Character Right	20+4Q
1 MVCL sbs,sbd,L	667	— —	Move Character Left	20+4Q
B WORD- AND CHARACTER-MANIPULATION				
1 LCL e	007	Z	Load Character Left	22,32,32
1 LCR e	047	Z	Load Character Right	16,32,36
1 LPL e	017	Z	Load Parallel Left	42,42,26
1 LPR e	057	Z	Load Parallel Right	20,38,42
1 MVE c	237	— —	Move block	6+4W
1 MVL b,c	137	— —	Move block Left	2+28W
1 MVR b,c	177	— —	Move block Right	2+28W
1 SCL e	407	Z	Store Character Left	24,38,38
1 SCR e	447	Z	Store Character Right	20,38,42
1 SPL e	417	Z	Store Parallel Left	14
1 SPR e	457	Z	Store Parallel Right	14
1 TRT e	357	— —	Translate bytes	144 max
C LIST PROCESSING				
1 DOWN e	437	— —	Down list	12
1 IN e	347	— —	Insert into list	20, 16ns
1 POP e	327	— —	Pop up list	16
1 PUSH e	427	— —	Push down list	22, 18ns
1 UP e	337	— —	Up list	12, 10ns
D LOAD/STORE				
1 LDA* e,x	03x	— —	Load RA	6
1 LDA1* e,x	01x	— —	Load RA & X1	12
1 LDB* e,x	04x	— —	Load RB	6
1 LD1* e,x	05x	— —	Load X1	6
1 LD2* e,x	06x	— —	Load X2	6
1 LD3* e,x	07x	— —	Load X3	6
1 LD23* e,x	02x	— —	Load X2 & X3	12
1 SAM* e,x	41x	— —	Store RA address	12
1 STA* e,x	43x	— —	Store RA	8
1 STA1* e,x	31x	— —	Store RA & X1	12
1 STB* e,x	44x	— —	Store RB	8
1 STP* e,x	42x	— —	Store RP	8
1 STZ* e,x	40x	— —	Store Zero	8
1 ST1* e,x	45x	— —	Store X1	8
1 ST2* e,x	46x	— —	Store X2	8
1 ST3* e,x	47x	— —	Store X3	8
1 ST23* e,x	32x	— —	Store X2 & X3	12
E FIXED POINT				
1 ADA* e,x	13x	OZMC	Add to RA	8
1 ADM* e,x	12x	OZMC	Add to Memory	10
1 AD1* e,x	15x	OZMC	Add to X1	8
1 AD2* e,x	16x	OZMC	Add to X2	8
1 AD3* e,x	17x	OZMC	Add to X3	8
1 DIV c	227	ZMC	Divide	18+8N
1 MPY c	127	ZMC	Multiply	24+8N
1 SBA* e,x	23x	OZMC	Subtract from RA	8
1 SB1* e,x	25x	OZMC	Subtract from X1	8
1 SB2* e,x	26x	OZMC	Subtract from X2	8
1 SB3* e,x	27x	OZMC	Subtract from X3	8
F FLOATING POINT				
1 FAD	157	ZMC	Floating Add	$46+4N_1+8N_2$
1 FDV	267	ZMC	Floating Divide	228
1 FMP	167	ZMC	Floating Multiply	$220+8N_3$
1 FSB	257	ZMC	Floating Subtract	$52+4N_1+8N_2$
1 UFA	147	ZMC	Unnormalized Floating Add	90
G COMPARISON				
1 CPA* e,x	33x	OZMC	Compare RA	8
1 CP1* e,x	35x	OZMC	Compare X1	8
1 CP2* e,x	36x	OZMC	Compare X2	8
1 CP3* e,x	37x	OZMC	Compare X3	8

Assembler Format (1)	Octal Code	Condition Codes	Name	Timing (2)
		H SHIFT ACCUMULATOR		
l SLA* e,x	52x	O	Left Arithmetic single	9+3K (+1)
l SLAD* e,x	53x	O	Left Arithmetic Double	8+4K
l SLR* e,x	50x	— —	Left Rotate single	6+2K
l SLRD* e,x	51x	— —	Left Rotate Double	9+5K (+1)
l SRA* e,x	56x	— —	Right Arithmetic single	6+2K
l SRAD* e,x	57x	— —	Right Arithmetic Double	8+4K
l SRL* e,x	54x	— —	Right Logical single	6+2K
l SRLD* e,x	55x	— —	Right Logical Double	8+4K
		I BRANCH/SKIP		
l BAL* e,x	66x	— —	Branch & Link using X2	6
l BCR* e,x	65x	— —	Branch if Carry	6
l BC1* e,x	75x	— —	Branch & Count X1	10
l BC2* e,x	76x	— —	Branch & Count X2	10
l BC3* e,x	77x	— —	Branch & Count X3	10
l BGT* e,x	67x	— —	Branch if logically Greater	6
l BMI* e,x	64x	— —	Branch on Minus	6
l BNZ* e,x	73x	— —	Branch on Nonzero	6
l BOF* e,x	62x	O	Branch on Overflow	6
l BPL* e,x	74x	— —	Branch on not minus	6
l BRA* e,x	72x	— —	Branch unconditional	6
l BRD e	507	OZMC	Branch Return Debreak	8
l BRM* e,x	71x	— —	Branch & Mark	10+I
l BRR e	517	OZMC	Branch Return	8
l BZO* e,x	63x	— —	Branch on Zero	6
l DEC* e,x	21x	— —	Decrement memory, skip if zero	14
l INR* e,x	11x	— —	Increment memory, skip if zero	10+I
l SKN* e,x	22x	— —	Test memory, Skip if Negative	8
l SKZ* e,x	60x	— —	Test memory, Skip if Zero	10
		J REGISTER-TO-REGISTER		
l CDA2	247	— —	Copy Double	8
l RADD s,d,b	117	OZMC	Register Add	8
l RAND s,d,b	207	ZM	AND source to dest	8
l RCL s,d,b,c	027	— —	Copy then Rotate Left	6+2K
l RCM2 s,d	317	OZMC	2's complement	10
l RCPY s,d,b	067	— —	Copy source to dest	6
l RCR s,d,b,c	067	— —	Copy then Rotate Right	6+2K
l RLC s,d,b,c	037	— —	Rotate Left, then Copy	8+2K
l ROR s,d,b	107	ZM	OR source to dest	8
l RRC s,d,b,c	077	— —	Rotate Right, then Copy	8+2K
l RSUB s,d,b	217	OZMC	Register Subtract	8
l RXOR s,d,b	307	ZM	XOR source to dest	8
		K LOGICAL		
l ANA* e,x	24x	ZM	AND to RA	8
l ANM* e,x	20x	ZM	AND to Memory	10
l ORA* e,x	14x	ZM	OR to RA	8
l ORM* e,x	10x	ZM	OR to Memory	10
l XOA* e,x	34x	ZM	XOR to RA	8
l XOM* e,x	30x	ZM	XOR to Memory	10
		L CONTROL		
l HLT e	00x	— —	Halt	6
l MCC* e,x	61x	ZM	Memory set CC	8
l NOP	067	— —	No operation	6
l ODD s,d,b,c	657	— —	Compute odd parity	10+3K (+1)
l RCC s	117	ZM	Register set CC	8
l TRAP e	467	— —	Trap to 41_8	6
l XEC* e,x	70x	— —	Execute [EA]	2
		M INTERRUPT		
l IOID e	577	— —	Indirect Interrupt	10
l PIA e	547	— —	Priority Interrupt Arm	6
l PID e	557	— —	Priority Interrupt Disarm	6
l PIR e	567	— —	Priority Interrupt Reset	8+6T or 6
		N INPUT/OUTPUT		
l BOOT s,d	377	Z	Bootstrap load	NA
l ECS d,b	477	— —	Enter Console Keys	8
l EXCT e	527	— —	External Command	6
l EXSN e	537	— —	External Sense	6
l IO e	677	— —	Input/Output Words	28+I+6(W-1)
l IOB e	277	Z	Input/Output Bytes	86 in, 122 out

① Assembly language form of instruction. Mnemonic is in capital letters, label and operand generally in lower case. * with the mnemonic means indirect addressing may be used. Other entries:

l = label. Ok with any machine instruction.

e = expression. Must be reducible to an address or a count. Not optional.

e,x = expression plus indexing. The expression is the same as above, but indexing may also be applied. Indexing is always optional.

s,d = source and destination registers. Never options.

b = byte control. If no byte control is given by the programmer where indicated, assembler supplies 7 (all bytes).

c = count. The count is treated modulo 64.

sbs,sbd = Starting Bytes, Source & Destination. 0, 1, or 2, indicating the starting bytes for a Decimal Option instruction.

L = Length in bytes -1 of memory block.

L1 = Difference in bytes in lengths of memory blocks.

L2 = Length in bytes-1 of the source block.

② Timing in machine cycles; cycle time is 2 μsec. Indexing adds 4 cycles; indirect addressing 2 cycles. Other symbols:

K = shift count (range = 0-63)

W = # words = count field + 1 (1-64)

ns = no skip

N = count (range = 0-23)

N_1 = prealign (range 0-63, average 8-9)

N_2 = normalize (range = 0-23, avg. = 5-6)

N_3 = 0, 1, or 2

I = 0 normal, 4 if fetched at interrupt

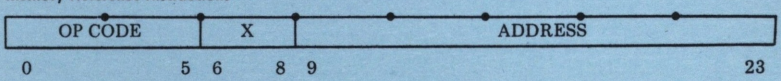
(+1) = Add one if number of cycles is odd

P = Length of Destination in words, Max 22

Q = Length of Block in words, Maximum 86

INSTRUCTION FORMATS
(Machine Language Form)

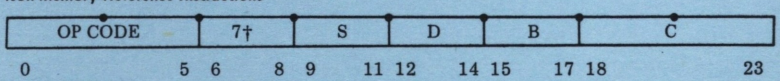
Memory Reference Instructions



X = Modification field

0, 7 = no address modification
1 = indirect address
2 = index X1 and direct
3 = index X1 and indirect
4 = index X2 and direct
5 = index X2 and indirect
6 = index X3 and direct

Non-Memory Reference Instructions



Any or all of S, D, B, C may be optional, required, or unused. See the individual instruction for this information.

S = Source register D = Destination register

S or D	0	1	2	3	4	5	6	7
REGISTER	R0	R1	RP	RA	RB	X1	X2	X3

B = Byte Control

	Byte 0	Byte 1	Byte 2

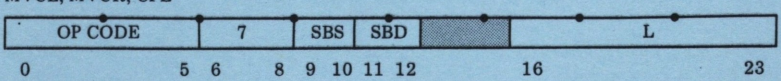
B	0	1	2	3	4	5	6	7
BYTES AFFECTED	None	2	1	1,2	0	0,2	0,1	0,1,2

C = count; range = $0\text{-}63_8$

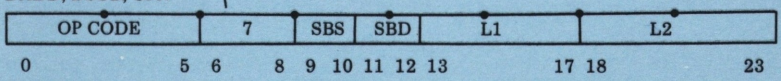
†For Shift Accumulator instructions, Mod Field $<7_8$.

Decimal Option Instructions

MVCL, MVCR, CPL



DADD, DSUB, CPN



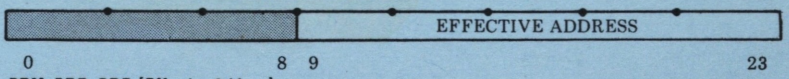
SBS = Starting byte source SBD = Starting byte destination

SBS or SBD	0	1	2	3
Starting Byte	leftmost byte	middle byte	rightmost byte	illegal

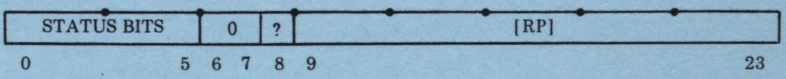
L1 = Difference in bytes between lengths of source and destination quantities
L2 = Length in bytes of source, -1

SPECIAL WORD FORMATS

Memory Reference Indirect [Reference or Indexed Address]



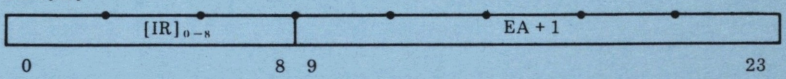
BRM, BRR, BRD [Effective Address]



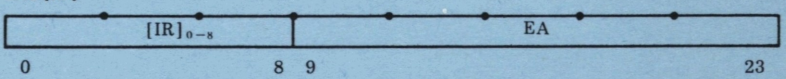
Status Bits

Bit	0	1	2	3	4	5
Indication	Halt	Parity	Overflow	Zero	Minus	Carry

BRM [RP]

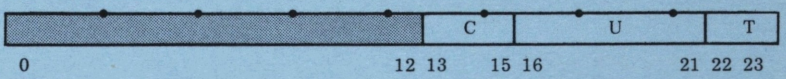


BAL [RP]

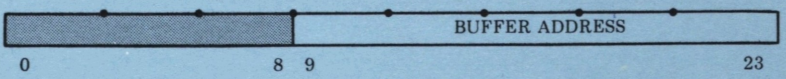


IO, IOB [Effective Address]

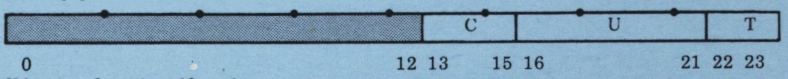
Word 0 (even address)



Word 1 (next odd address)



BOOT [D]



[Memory Location 1] = 1

Word 0 and [D] Format

Bit	Meaning	Bit	Meaning
0 - 12	(not used)	22 - 23	T = type of I/O operation:
13 - 15	C = device channel number		00 = data out
16 - 21	U = device unit number		01 = data in
			10 = control
			11 = status

PIA, PID, PIR [Effective Address]

Bit	Interrupt Level Affected	Bit	Interrupt Level Affected
0 - 15	(not used)	20	3
16	7	21	2
17	6	22	1
18	5	23	0
19	4		

EXCT, EXSN [Effective Address]

Bit	0 - 19	20	21	22	23
Sense Line Bit	(not used)	0	1	2	3
Command Line Bit	(not used)	0	1	2	3

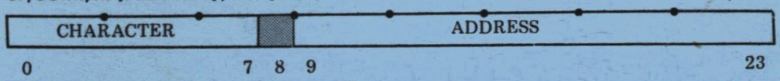
EXCT [EA] FUNCTIONS

Function	Bits Set in EA			
	20	21	22	23
Enable 5-10-31 Intensity†	1	0	0	1
Enable 300 Intensity	1	0	1	0
Disable Variable Intensity†‡	1	0	1	1
Enable MM Halt on Parity Error	1	1	0	0
Disable MM and Reset Sense Bit†‡	1	1	0	1
Select Odd Parity Test‡	1	1	1	0
Force Parity Bit†				
† Interrupt on Level 7	X	X	X	1
‡ Also effected by system reset				

EXSN [EA] FUNCTIONS

Function	Bits Set in EA			
	20	21	22	23
Sense Parity Error Bit	X	X	X	1

UP, DOWN, IN [FREE Table]; TRT [Table]



Character Manipulation and Input Pack Instructions

B = Byte Control C = Count

BOOTSTRAP WORDS [CONSOLE KEYS]

UNIT	WORD	UNIT	WORD
Card Reader	37705101	4450/8250 Diskette	37707175
Buffered Card Reader	37703115	8507 Tape (7-track)	37705261
8230 Disc	37705121	8512 Tape (800 bpi)	37705221
8240 Disc	37705201	8513 Tape (1600 bpi)	37705241

CHARACTER SET

Video Attribute Characters
"300" System: bits 0 and 1 must be 1; bits 4 and 5 control the attributes. Bits 4 and 5 = 00 or 01: normal; = 10: intensify; = 11: blank. Also 05, 0205: cursor with blank; 010, 0210: cursor with normal; 031, 0231: cursor with intensify.
"5-10-31" System: 05, 0205: blank; 010, 0210: normal; 031, 0231: intensify.

Octal Code	Keyboard Character	Control Character Interpretation (ASCII)		Display Character
000	=c	NUL	Null	•
001	Ac	SOH	Start of Heading (CC)	△ (1)
002	Bc	STX	Start of Text (CC)	b (1)
003	Cc	ETX	End of Text (CC)	¢
004	Dc	EOT	End of Transmission (CC)	◢ (NL)
005	Ec	ENQ	Enquiry (CC)	♯ (1) (2)
006	Fc	ACK	Acknowledge (CC)	♯ (1)
007	Gc	BEL	Bell	¤ (1)
010	Hc	BS	Backspace (FE)	⇤ (1) (2)
011	Ic	HT	Horizontal Tabulation (FE)	←
012	Jc	LF	Line Feed (FE)	\ (45°)
013	Kc	VT	Vertical Tabulation (FE)	/ (45°)
014	Lc	FF	Form Feed (FE)	£
015	Mc	CR	Carriage Return (FE)	■ (Check)
016	Nc	SO	Shift Out	¬
017	Oc	SI	Shift In	| (Left)
020	Pc	DLE	Data Link Escape (CC)	| (Right)
021	Qc	DC1	Device Control 1	[
022	Rc	DC2	Device Control 2	\ (Check)
023	Sc	DC3	Device Control 3	– (EOM)
024	Tc	DC4	Device Control 4	√ (1)
025	Uc	NAK	Negative Acknowledge (CC)	– (Over)
026	Vc	SYN	Synchronous Idle (CC)	^
027	Wc	ETB	End of Transmission Block (CC)	]
030	Xc	CAN	Cancel	□
031	Yc	EM	End of Medium	↑ (1) (2)
032	Zc	SUB	Substitute	■
033	+c	ESC	Escape	°
034	,c (comma)	FS	File Separator (IS)	◄ (Stop)
035	-c (minus)	GS	Group Separator (IS)	▮
036	.c	RS	Record Separator (IS)	► (MI)
037	/c	US	Unit Separator (IS)	\ (Slash)

c CONTROL key pressed at same time (CC) Communication Control
(FE) Format Effector (IS) Information Separator

(1) These symbols are currently displayed but not supported. Other symbols may be substituted on later models.

(2) Blank for 7002 Processor

Octal Code	Keyboard Character
040	space bar
041	!
042	"
043	#
044	$
045	%
046	&
047	' (7^{S})
050	(
051	)
052	*
053	+
054	, (Comma)
055	-
056	.
057	/ (Slash)
060	0 (zero)
061	1
062	2
063	3
064	4
065	5
066	6
067	7
070	8
071	9
072	:
073	;
074	<
075	=
076	>
077	?

Octal Code	Keyboard Character
100	@ (=^{S})
101	A^{S}
102	B^{S}
103	C^{S}
104	D^{S}
105	E^{S}
106	F^{S}
107	G^{S}
110	H^{S}
111	I^{S}
112	J^{S}
113	K^{S}
114	L^{S}
115	M^{S}
116	N^{S}
117	O^{S}
120	P^{S}
121	Q^{S}
122	R^{S}
123	S^{S}
124	T^{S}
125	U^{S}
126	V^{S}
127	W^{S}
130	X^{S}
131	Y^{S}
132	Z^{S}
133	÷
134	× (mult)
135	| (-^{S}) (centered)
136	EXP ↑ (÷^{S})
137	— (×^{S}) (under)

Octal Code	Keyboard Character
140	` (0^{S}) (zero)
141	A
142	B
143	C
144	D
145	E
146	F
147	G
150	H
151	I
152	J
153	K
154	L
155	M
156	N
157	O
160	P
161	Q
162	R
163	S
164	T
165	U
166	V
167	W
170	X
171	Y
172	Z
173	÷^{C} {
174	×^{C} (mult) ¦
175	*^{C} }
176	↑^{C} ~
177	←^{C}

Octal Code	Keyboard Character	Interpretation
200	↑	Cursor Up
201	←	Cursor Left
202	→	Cursor Right
203	↓	Cursor Down
204	EOM	End of Message
205	ATTEN	Attention
206	ROLL (↓^{S})	Roll Down
207	ERASE (HOME^{S})	Erase Screen
210	HOME	Cursor Home
211	TAB	Horizontal Tab
212	ROLL (↑^{S})	Roll Up
213	TAB^{S}	Vertical Tab
214	EOM^{S}	Shifted EOM
215	CURSOR RETURN	Cursor Return
216	CURSOR RETURN^{S}	Shifted Cursor Return
217	INSERT (→^{S})	Insert
220	DELETE (←^{S})	Delete
221	F1	Function Key 1
222	F2	Function Key 2
223	F3	Function Key 3
224	F4	Function Key 4
225	F5	Function Key 5

Octal Code	Keyboard Character	Interpretation
226	F6	Function Key 6
227	F7	Function Key 7
230	F8	Function Key 8
231	F9	Function Key 9
232	F10	Function Key 10
233	F11	Function Key 11
234	→^{C}	Control →
235	TOTAL	Total
236	↓^{C}	Control Roll Down
237	EOM^{C}	Control EOM
241	— — —	Lightpen
260	0^{C}	Control 0
261	1^{C}	Control 1
262	2^{C}	Control 2
263	3^{C}	Control 3
264	4^{C}	Control 4
265	5^{C}	Control 5
266	6^{C}	Control 6
267	7^{C}	Control 7
270	8^{C}	Control 8
271	9^{C}	Control 9
375	HOME^{C}	Control Home
376	CURSOR RETURN^{C}	Control Cursor Return
377	TAB^{C}	Control TAB

s=SHIFT key pressed at same time c=CONTROL key pressed at same time

FOUR-PHASE SYSTEMS, INC.©

ASSEMBLER DIRECTIVES (1)

DATA DEFINERS (2)

DCA	Define Constant: delimited ASCII character string.
DCD	Define Constant: double-precision floating point expression.
DCN	Define Constant: numeric (decimal or octal)
DCS	Define Constant: single-precision floating point expression.
MZE	Prefix 5-digit operand with 777 (negative zero).
PZE	Prefix 5-digit operand with 000 (positive zero).
RPZE	Prefix S-D-B-C operand with 007.

SYMBOL DEFINERS (3)

EQU	Define label as equal to 5-digit operand.
SET	Same as EQU, but allow redefinition by subsequent SET.

STORAGE ALLOCATORS

BES	Allocate /e/ locations starting with $; put label at $ + /e/. (oprnd= e)
BSS	Allocate /e/ locations starting with $; put label at $. (operand= e)

CONDITIONAL ASSEMBLY DIRECTIVES (4)

IFGT	Skip to statement labeled 'name' if /e/ > 0. (operand= e,name)
IFLT	Skip to statement labeled 'name' if /e/ < 0. (operand= e,name)
IFNZ	Skip to statement labeled 'name' if /e/ ≠ 0. (operand= e,name)
IFZO	Skip to statement labeled 'name' if /e/ = 0. (operand= e,name)
SKIP	Skip next /e/ statements if /e/ > 0. (operand= e)

LINKAGE CONTROLS

ENTRY	Enter symbol stated in operand as a virtual link.
EOP	End; link to library via label field entry if given.

MISCELLANEOUS

DUP	Duplicate, /e/ times, statements to 'name'-1. (operand= e,name)
END	End; LOADOV will use operand of last END statement for /E location.
FORCE	Put next instruction at even $ if oprnd even, or at odd $ if oprnd odd.
INCLD	Open named source file to RANGE statements. (operand= name,drive#)
INEND	Close file opened by last INCLD statement.
LTRL	Define literal key. (operand= key)
ORG	Assign label to $, then set $ equal to operand.
RANGE	INCLD range of SNEDIT-numbered records given. (oprnd = first,last)

LISTING CONTROLS

EJECT	Eject page.
NPRNT	Suppress listing except for errors, virtuals, and XREF.
PRNT	Enable listing.
SPACE	Skip 'n' lines. (operand= n)
TITLE	Entitle each page with operand.

Notes:

(1) $ = current location count; /e/ = expression as evaluated by assembler.

(2) If label field is used, label is assigned to $.

(3) Label field entry is mandatory for these directives.

(4) A "label" referenced by the operand of a conditional directive can be any combination of from one to six bytes; normal label rules do not apply.

Statement (record) Format:

1 6	8 13	15 (first blank) 80	
LABEL	DIRECTIVE	OPERAND	COMMENTS

7071/7072 Channel Adapters (5)	0	Data Transfer Required (2)
	1	Attn or DE accepted
	2	— —
	3	System Reset (2)
	4	— —
	5	Printer Status Request (1)
	6-7	Type: 0 = New Command (2) 2 = End Operation 1 = Continue (2) 3 = End Operation
	8	Keyboard Lock
	9-11	New Command Code: 0 = No Command 4 = — — 1 = Write Buffer 5 = Read Manual Input 2 = Write Line 6 = Short Read 3 = Erase Buffer 7 = Read Full Buffer
	12-15	Line Address (New Command)
	14-15	Byte Boundary: 0 = Word Full 2 = Left Two Bytes 1 = Left Byte
	16-19	2848 Address
	20-23	Address within 2848
7200 Series Keyboards	0-15	— —
	16	Data Ready
	17	— —
	18	Data Lost (4)
	19-23	Unit Address
8001/8003 Card Readers (unbuffered) (5)	0-20	— —
	21	Card In Reader (3)
	22	Busy (3)
	23	Not Ready (3)
8001/8003 Card Readers (buffered)	0-18	— —
	19	Motion Check (6)
	20	Reader Error (6)
	21	Hopper Check (6)
	22	Buffer Not Full (3) (6)
	23	Not Ready (3)
8121/8122 Character Printer Unit (7)	0-14	— —
	15	Printer Restart
	16	System Not Ready (3)
	17	Paper Out
	18	Check Condition (2)
	19	Print Head Not Ready
	20	Paper In Motion
	21	Printing
	22	Busy
	23	Not Ready
8121/8122 Character Printer Controller	0-17	— —
	18	Receiving or Scanning
	19	Sending Data
8131 Character Printer	0-20	— —
	21	Not Ready (3)
	22	Output Needed (2)
	23	— —
81LP Line Printers	0-20	— —
	21	Paper Out (4)
	22	Busy (3)
	23	Not Ready
8100 KB/Printer	0-20	— —
	21	Baud Rate Error
	22	Output Needed (2)
	23	Input Ready (2)
8230/8231 Cartridge Disc Unit	0-16	— —
	17	Seek Incomplete Error (2) (6) (8)
	18	Head Range Error (2) (8)
	19	Header Error
	20	Too Late
	21	CRC Error
	22	Busy
	23	Not Ready (3)

8240/8241 Disc Storage Unit	0	Interface Not Available (3) (4)
	1-2	Drive That Completed Seek Last
	3-4	Reserved Bits; Should be 0 (4)
	5	Read Only
	6	Defect Detected
	7-15	Current Cylinder Address
	16	File Unsafe
	17	Seek Incomplete Error (2) (4) (8)
	18	Head Range Error
	19	Header Error
	20	Too Late
	21	CRC Error
	22	Busy
	23	Not Ready (3)
8250/4450 Diskette Unit	0-18	— —
	19	Busy
	20	Not Ready
	21	Transfer Pending
	22	Operation Check
	23	Track 0
8411 Asynchronous Data Set Controller	0-11	— —
	12	Long Space (2)
	13-15	— —
	16	Ring Indicator (2)
	17	Data Set Ready (1)
	18	Clear to Send (2)
	19	Carrier Detect (1)
	20	Parity Error
	21	Baud Rate Error
	22	Output Needed (2)
	23	Input Ready (2)
8435 Synchronous Data Set Controller (7)	0-8	— —
	9	Output Needed (2)
	10	Data Set Ready (1)
	11	Clear to Send
	12	Sync Received
	13	Data Lost
	14	Carrier Detect (1)
	15	Input Ready in 16-23 (2)
	16-23	(Data Byte)
8436-1 Synchronous Data Set Controller (buffered)	0-7	— —
	8	Transmitted Data Lost
	9	Output Needed (2)
	10	Data Set Ready (1)
	11	Clear to Send (1)
	12	Sync Received
	13	Received Data Lost
	14	Carrier Detect (1)
	15	Input Ready (2)
8436-(2-5) Synchronous Data Set Controller (buffered, PROM)	0-6	— —
	7	Ring Indicator (1)
	8	Transmitted Data Lost
	9	Output Needed (2)
	10	Data Set Ready (1)
	11	Clear to Send
	12	Sync Received
	13	— —
	14	Carrier Detect (1)
	15	Input Ready (2)
8507/8511/8512 Magnetic Tape Units (5)	0-5	— —
	6	800 BPI Selected (8707 only)
	7	556 BPI Selected (8507 only)
	8	200 BPI Selected (8507 only)
	9	Even Parity Selected (8507 only)
	10	Short
	11-12	Byte Boundary: (8507) 0 = all four types; 1 = right byte; 2 = right two bytes; 3 = right three bytes. (8511/8512) 0 = all three bytes; 1 = right byte; 2 = right two bytes; 3 = all three bytes
	13	Rewinding

STATUS WORDS (Continued)

8507/8511/8512 Magnetic Tape Units (5)	14	Stopping Motion (3) (6)
	15	Data Transfer Required (2)
	16	File Mark (6)
	17	Too Late (2)
	18	EOT (6)
	19	BOT (6)
	20	Write Protect
	21	Parity Error
	22	Busy (4)
	23	Not Ready
8513/8504 Magnetic Tape Units (5)	0-5	— —
	6	Operation Complete (2) (6)
	7	1600 BPI ID
	8	Corrected Parity Error (6)
	9	Reject (2) (6)
	10	Short (6)
	11-12	Byte Boundary (same as 8511)
	13	Rewinding (6)
	14	— —
	15	Data Transfer Required (2) (6)
	16	File Mark (6)
	17	Too Late (6)
	18	EOT (6)
	19	BOT (6)
	20	Write Protect
	21	Uncorrectable Error (6)
	22	Busy
	23	Not Ready

Status Words Footnotes:

(1) Interrupt on change of status.
(2) Interrupt on true.
(3) Interrupt on false.
(4) Special considerations; see PUP Manual.
(5) Controller's IOID depends on transaction.
(6) May interact with other status bits.
(7) Status obtained by Data-In IO.
(8) Reset by Control-Restore.

CONTROL WORDS

7071/7072 Channel Adapters		
Initialization	0-1	Must be 01
	2-3	— —
	4-11	Printer Address
	12-19	Upper Control Unit Address
	20-23	Lower Control Unit Address
Load Byte	0-13	Must be 0 . . . 0
	14-15	Starting Byte 0 = Left 1 = Middle 2 = Right
	16-23	Byte Count
Load Bit Register	0-1	Must be 00
	2	Printer Not Available
	3-7	Must be 0
	8	No Conversion
	9	Printer Intervention Required
	10	Must be 1
	11	Printer Busy
	12-23	— —
Load Address Register	0-6	Must be 0 . . . 0
	7	Device End Only
	8-11	Must be 0 . . . 0
	12	Attention
	13	Channel End/Device End
	14-15	— —
	16-23	Control Unit Address

7200 Series Keyboards (Alarms) (1)	0-12	— —
	13-15	Must be 011
	16-21	Unit Address
	22-23	Must be 10
8001/8003 Card Readers (unbuffered)	0-21	— —
	22	Convert Hollerith to ASCII
	23	Select Unpacked Mode
8001/8003 Card Readers (buffered)	0-20	— —
	21	Clear Controller
	22	Convert Hollerith to ASCII
	23	Select Unpacked Mode
8121/8122 Character Printer		
General Command (2)	0	Must be 1
	1	"Program 1" light (1 = on)
	2	"Program 2" light (1 = on)
	3	"Program 3" light (1 = on)
	4-11	— —
	12	Must be 1
	13-17	— —
	18	Set "Program" lights as per 1-3
	19-20	— —
	21-22	11 = Set ribbon up (print) 01 - Set ribbon down (nonprint)
	23	Restore Print Head
Print Command (10 char/inch) (2)	0	Must be 0
	1-11	— —
	12	Must be 0
	17-23	ASCII Character Code
Print Command (char/inch and direction specified) (2)	0	Must be 0
	1	0 = move head right K/60 inches 1 = move head left K/60 inches
	2-11	Binary value of "K"
	12	Must be 1
	13-16	— —
	17-23	ASCII Character Code
Head Position and Paper Command (2)	0	Must be 1
	1	0 = head right K/60 inches 1 = head left K/60 inches
	2-11	Binary value of "K"
	12	Must be 0
	13	0 = paper up L/48 inches 1 = paper down L/48 inches
	14-23	Binary value of "L"
8121/8122 Printer Controller	0-21	— —
	22	Stop Scan
	23	Continue Scan
81LP Line Printers	0-22	— —
	23	Select Character Mode
8230/8231 Cartridge Disc Unit	0	0 = Read, 1 = Write
	1	Seek
	2	Brute Force
	3	Restore
	4	Header Only
	5-7	— —
	8-11	Sector Count (3)
	12-19	Cylinder Address
	20-23	Track and Sector Address
8240/8241 Disc Storage System		
Word 1	0	— —
	1-6	Sector Count (3)
	7-15	Cylinder Address
	16-20	Initial Head Address
	21-23	Initial Sector Address

Word 2	0	0 = Read, 1 = Write
	1	Seek
	2	Brute Force
	3	Restore
	4	Header Only
	5	Resume Scan
	6	Enable Defect Detect
	7	Read CRC
	8	Short Read
	9-16	Short Read Count (3)
	17-23	— —
8250/4450 Diskette Unit	0-16	— —
	17	Write to track > 43
	18	Read
	19	Write
	20	Step +1 track
	21	Step -1 track
	22	Reset Op Check
	23	Restore Head
8411 Asynchronous Data Set Controller	0-11	— —
	12	Force Long Space
	13-20	— —
	21	Suppress Parity
	22	Set Data Set Ready
	23	Set Request to Send
8435 Synchronous Data Set Controller	0-20	— —
	21	Reset Transmitter
	22	Set Request to Send
	23	Reset Receiver
8436-(1 . . .) Buffered Synchronous Data Set Controllers	0-18	— —
	19	Set Data Terminal Ready
	20	Reset Data Terminal Ready
	21	Reset Transmitter
	22	Set Request to Send
	23	Reset Receiver
8507/8511/8512 Magnetic Tape Units	0	Write
	1	Read
	2	Backward
	3	Skip
	4	Set Lower Density (8507 only)
	5	Erase
	6	Rewind
	7	File Mark
	8	Set Word Mode
	9	Reset
	10	Set Even Parity (8507 only)
	11	— —
	12-23	Byte Count
8513/8504 Magnetic Tape Units	0	Write
	1	Read
	2	Backward
	3	Skip
	4	— —
	5	Erase
	6	Rewind
	7	File Mark
	8	Enable Read Timeout
	9	Reset
	10-23	Byte Count

Control Words Footnotes:

(1) CUT word holds controls; CUT+1 not used.

(2) Commands sent as Data-Out IO words.

(3) Count = (n-1), where n is the number of sectors to be transferred.

CONDITION TESTS

Type → / Test ↓	Arithmetic (Sign Bit); No Overflow Possible	Logical (No Sign)	CPL (A = dest, B = source)	Arithmetic (Sign Bit); Overflow Possible	Decimal Arithmetic (A = dest, B = source)
[A] < [B] ?	LDA A CPA B BMI YES NO EQU $	LDA A CPA B BCR YES NO EQU $	LD23 @A CPL BCR YES NO EQU $	LDA A BOF $+1 CPA B BOF NO BMI YES NO EQU $ BMI NO BRA YES	LD23 @A CPN BMI YES NO EQU $
[A] ≤ [B] ?	LDA A CPA B BMI YES BZO YES NO EQU $	LDA A CPA B BGT NO BRA YES	LD23 @A CPL BGT NO BRA YES	LDA A BOF $+1 CPA B BZO YES BOF NO BMI YES NO EQU $ BMI NO BRA YES	LD23 @A CPN BMI YES BOF NO BZO YES NO EQU $
[A] = [B] ?	LDA A CPA B BZO YES NO EQU $	LDA A CPA B BZO YES NO EQU $	LD23 @A CPL BZO YES NO EQU $	LDA A CPA B BZO YES NO EQU $	LD23 @A CPN BOF NO BZO YES NO EQU $
[A] ≠ [B] ?	LDA A CPA B BNZ YES NO EQU $	LDA A CPA B BNZ YES NO EQU $	LD23 @A CPL BNZ YES NO EQU $	LDA A CPA B BNZ YES NO EQU $	LD23 @A CPN BOF YES BNZ YES NO EQU $
[A] ≥ [B] ?	LDA A CPA B BPL YES NO EQU $	LDA A CPA B BCR NO BRA YES	LD23 @A CPL BCR NO BRA YES	LDA A BOF $+1 CPA B BOF NO BPL YES NO EQU $ BPL NO BRA YES	LD23 @A CPN BPL YES NO EQU $
[A] > [B] ?	LDA A CPA B BMI NO BNZ YES NO EQU $	LDA A CPA B BGT YES NO EQU $	LD23 @A CPL BGT YES NO EQU $	LDA A BOF $+1 CPA B BZO NO BOF NO BPL YES NO EQU $ BPL NO BRA YES	LD23 @A CPN BMI NO BOF YES BNZ YES NO EQU $

BIT TESTS USING CC ZERO

TEST [LOC] Bit 'n' = 1?		TEST [LOC] Bit 'n' = 0?	
LDA MASK ANA LOC BNZ YES NO EQU $	LDA MASK ANA LOC BZO NO YES EQU $	LDA MASK ANA LOC BZO YES NO EQU $	LDA MASK ANA LOC BNZ NO YES EQU $
Where [MASK] is bit 'n' = 1, others = 0.			