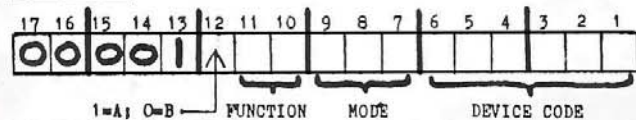


PRIVILEGED INSTRUCTIONS

000001 HALT	000010 Skip if MAINS OFF
000002 MASK	000011 Skip if MAINS ON
000003 Acknowledge Interrupt	000012 Skip if PARITY
000004 Interrupt ON	000013 Skip if PROTECTION
000005 Interrupt OFF	000014 Skip if BOUNDARY
000006 Skip if Interrupt ON	000015 Skip if MA=SW
000007 Skip if Interrupt OFF	000016 Skip if CONT. INTERRUPT
	000017 I,O,RESET

INPUT/OUTPUT

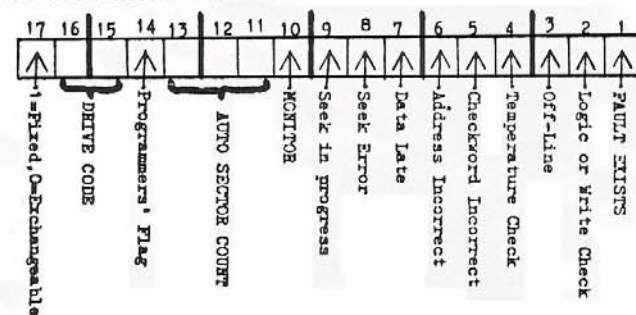


A	B	FUNCTION	SKIP IF	MODE
4	0	No operation	BUSY	1 DAT11
5	1	Set BUSY, Clear DONE	NOT BUSY	2 DAT12
6	2	Clear BUSY, Clear DONE	DONE	3 DAT13
7	3	IOPLS	NOT DONE	4 DAT01
				5 DAT02
				6 DAT03
				7 SKIP

In	out	STANDARD DEVICE CODES
20	60	Alpha-numeric Keyboard
50	40	Visual Display Unit
-	30	Printer
-	34	Serial Printer (BCL)
66	67	IBM I/O Writer
11	33	Paper Tape (8 track)
70	70	Disc

Devices raise an INTERRUPT REQUEST whilst the DONE flag is set, unless disabled by the current MASK.

DISC STATUS REGISTER (R3)



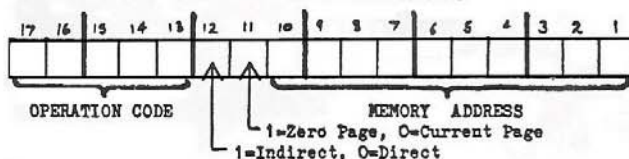
L.O.S. DISC FAULT MESSAGE

"UNIT xxxxxx SECTOR xxxxxx R STATUS xxxxxx"

The UNIT field identifies the device (bits 6:1), drive (16:15) and cartridge (17). The SECTOR field identifies the base sector of the transfer (not necessarily the faulty sector since up to 8 sectors may be involved in the transfer). The system continually retries the transfer (re-issuing the message approx. every 20 secs.) until the fault clears, or until the unit is taken off-line.

TYPE	SECTORS	Last Sect	RPM	Av. Access
DB18	6496 of 128 words	014537	2,400	52msec
D1600	12992 of 128 words	031277	2,400	49msec

MEMORY REFERENCE INSTRUCTIONS



OCTAL	MNEMONIC		μ sec
02	JUMP	Jump unconditionally	1.2
03	JSBR	Jump to Subroutine	1.2
04	INSZ	Increment, Skip if result is Zero	2.6
05	DESZ	Decrement, Skip if result is Zero	2.6
06	ANDA	AND into A	2.4
07	IORA	INCLUSIVE OR into A	2.4
10	XORA	EXCLUSIVE OR into A	2.4
11	ADA	ADD to A	2.4
12	ADB	ADD to B	2.4
13	SFA	SUBTRACT from A	2.4
14	SFB	SUBTRACT from B	2.4
15	ADAC	ADD to A with carry flag	2.4
16	ADBC	ADD to B with carry flag	2.4
17	SFAC	SUBTRACT from A with carry flag	2.4
20	SFBC	SUBTRACT from B with carry flag	2.4
21	LDA	LOAD into A	2.4
22	LDB	LOAD into B	2.4
23	CMPA	COMPARE with A, Skip if Unequal *	2.4
24	CMPB	COMPARE with B, Skip if Unequal *	2.4
25	STA	STORE A	2.4
26	STB	STORE B	2.4

* all 17 bits participate in the comparison. The GT flag is set if the register (excluding B17) is greater than the operand, otherwise the GT flag is cleared.

Each level of Indirect addressing adds 1.2 μsec.

MASKS	
0/0201	Bit 1
0/0202	Bit 2
0/0204	Bit 3
0/0210	Bit 4
0/0220	Bit 5
0/0240	Bit 6
0/0302	Bit 7
0/0316	Bit 8
0/0323	Bit 9
0/0332	Bit 10
0/0341	Bit 11
0/0344	Bit 12
0/0347	Bit 13
0/0352	Bit 14
0/0355	Bit 15
0/0356	Bit 16
0/0377	Bit 17
0/1752	000377
0/1753	177400
0/0375	000177
0/0320	000300
0/0325	000500
0/0326	000600
0/0376	377777
0/0374	"SP SP"

STANDARD CORE LOCATIONS	
0/0040	=TASK NUMBER
0/0077	→SYSTEM DATE
0/0072	→MASTER BUFFER (3200-)
0/0067	→SPOOL BUFFER (3400-)
0/0066	→INPUT or PRINT BUFFER (3600-)
0/0073	→FILE TABLE
0/0074	→TASK CONTROL AREA origin (3720-)
0/0057	=Max. Plain Paper Queue Number
0/0060	=Max. (DELETIONS) Queue Number
0/1300	=PASSWORD 1
0/1305	=PASSWORD 2 (EL Package)
0/1312	=PASSWORD 3
0/1317	=PASSWORD 4
3742-	=Printer Completion request word (B17=0 cancel, B16=1=Task No of station to be flashed at end)
AFTER "GET":	
0/0043	=A register contents
0/0044	→Next Input Field in buffer
0/0045	=No. of characters input
AFTER "FETCH":	
0/0151	→Record (1st word) within buffer
0/0152	=Logical Record Length (words)
BEFORE "SPOOL":	
0/0070	→Next Spool Record Number
BEFORE "UNSPPOOL":	
3577-	=Next Spool Record Number in chain

REGISTER INSTRUCTIONS

	9	8	7	6	5	4	3	2	1
B 002	CLC	LEFT or RIGHT	SHIFT	ROTATE	with CARRY	DEC	INC	SKIP if MSB=β	SKIP if LSB=β
A 003									
B 004	CLEAR	One's	CLC	COMP	SKIP	SWAP	CLEAR	COMP	ESVR
A 005		COMP					SIGN	SIGN	
B 006	TRUE or FALSE	SKIP if NEG	SKIP if NOT β	SKIP if CARRY	CLC	SKIP if GT	CLGT	CLEAR	One's COMP
A 007									

MICRO-PROGRAMMING RULES

- Instructions will be executed in order from the left, except CLEAR SIGN which on some machines precedes CLEAR.
- If skips are combined, the skip will occur only if all conditions are fulfilled, except if AMSB and LSB (or BMSB and BLSB) are combined the skip will occur if either or both conditions are met.
- The effect of combining SHIPT and ROTATE or INC and DEC is undefined.

003001	ALSB	002001	BLSB	Skip if Bit 1 = β
003002	AMSB	002002	BMSB	Skip if Bit 16 = β
007600	ANEG	006600	ENEG	Skip if Bit 17 = 1
007500	ANβ	006500	BNβ	Skip if not zero
007200	APOS	006200	BPOS	Skip if Bit 17 = β
007100	Aβ	006100	Bβ	Skip if zero
007002	CLA	006002	CLB	Clear
002400	CLC			Clear Carry Flag
006004	CLGT			Clear Greater Than Flag
005004	CLSA	004004	CLSB	Clear Bit 17
007001	COMP	006001	COMPB	Complement
004040	COMP			Complement Carry Flag
005002	COMP	004002	COMP	Complement Bit 17
003010	DECA	002010	DECB	Decrement by one
005001	ESVRA	004001	ESVRB	Enter Switch Register
003004	INCA	002004	INCB	Increment by one
003240	LRA	002240	LRB	Rotate Left by 1 (excl. B17)
003260	LRAC	002260	LRBC	Rotate Left by 1 with Carry
001600+n	LRAn	000600+n	LRBn	Rotate Left by n (excl. B17)
003300	LRA	002300	LSB	Left Shift by 1 (excl. B17)
001700+n	LSAn	000700+n	LSBn	Left Shift by n (excl. B17)
000000	NOP			No Operation
003040	RRA	002040	RRB	Rotate Right by 1 (excl. B17)
003060	RRAC	002060	RRBC	Rotate Right by 1 with Carry
001400+n	RRAn	000400+n	RRBn	Rotate Right by n (excl. B17)
003100	RSA	002100	RSB	Right Shift by 1 (excl. B17)
001500+n	RSAn	000500+n	RSBn	Right Shift by n (excl. B17)
001520+n	RSAn	000520+n	RSBn	Right Shift by n Arithmetic
001001	SETGT			Set Greater Than Flag
005006	SETSA	004006	SETSB	Set Bit 17
006440	SKC			Skip if Carry Flag = 1
006410	SKGT			Skip if Greater Than Flag = 1
004020	SKIP			Skip unconditionally
006040	SKNC			Skip if Carry Flag = β
006010	SKNGT			Skip if Greater Than Flag = β
005010	SWAP	004010	SWAPB	Swap Bytes

Maximum value of n is 15 (decimal). Execution Time 1.4 μsec, except SETGT (1.2 μsec) and multiple shifts/rotates (1.6 μsec if n ≤ 1; 3.2 μsec if n ≤ 6; 4.8 μsec n ≤ 11; 6.4 μsec n ≤ 15).

TRANSLATION TABLE

DEC	OCT	BYTE	ASCII	PRINT POS'N
0	000	000000	MUL	3600-
1	001	000400	SOH	3601-
2	002	001000	STX	3602-
3	003	001400	ETX	3603-
4	004	002000	EOT	3604-
5	005	002400	ENQ	3605-
6	006	003000	ACK	3606-
7	007	003400	BEL	3607-
8	010	004000	BS	3608-
9	011	004400	HT	3609-
10	012	005000	LF	3610-
11	013	005400	VT	3611-
12	014	006000	FF	3612-
13	015	006400	CR	3613-
14	016	007000	SO	3614-
15	017	007400	SI	3615-
16	020	010000	DLR	3616-
17	021	010400	DC1	3617-
18	022	011000	DC2	3618-
19	023	011400	DC3	3619-
20	024	012000	DC4	3620-
21	025	012400	NAK	3621-
22	026	013000	SYN	3622-
23	027	013400	ETB	3623-
24	030	014000	CAN	3624-
25	031	014400	EM	3625-
26	032	015000	SUB	3626-
27	033	015400	ESC	3627-
28	034	016000	FS	3628-
29	035	016400	GS	3629-
30	036	017000	RS	3630-
31	037	017400	US	3631-
32	040	020000	SP	3632-
33	041	020400	!	3633-
34	042	021000	"	3634-
35	043	021400	#	3635-
36	044	022000	\$	3636-
37	045	022400	%	3637-
38	046	023000	&	3638-
39	047	023400	'	3639-
40	050	024000	(3640-
41	051	024400)	3641-
42	052	025000	*	3642-
43	053	025400	+	3643-
44	054	026000	,	3644-
45	055	026400	-	3645-
46	056	027000	.	3646-
47	057	027400	/	3647-
48	060	030000	0	3648-
49	061	030400	1	3649-
50	062	031000	2	3650-
51	063	031400	3	3651-
52	064	032000	4	3652-
53	065	032400	5	3653-
54	066	033000	6	3654-
55	067	033400	7	3655-
56	070	034000	8	3656-
57	071	034400	9	3657-
58	072	035000	:	3658-
59	073	035400	;	3659-
60	074	036000	<	3660-
61	075	036400	=	3661-
62	076	037000	>	3662-
63	077	037400	?	3663-

TRANSLATION TABLE

DEC	OCT	BYTE	ASCII	PRINT POS'N
64	100	040000	A	3664-
65	101	040400	B	3665-
66	102	041000	C	3666-
67	103	041400	D	3667-
68	104	042000	E	3668-
69	105	042400	F	3669-
70	106	043000	G	3670-
71	107	043400	H	3671-
72	110	044000	I	3672-
73	111	044400	J	3673-
74	112	045000	K	3674-
75	113	045400	L	3675-
76	114	046000	M	3676-
77	115	046400	N	3677-
78	116	047000	O	3678-
79	117	047400	P	3679-
80	120	050000	Q	3680-
81	121	050400	R	3681-
82	122	051000	S	3682-
83	123	051400	T	3683-
84	124	052000	U	3684-
85	125	052400	V	3685-
86	126	053000	W	3686-
87	127	053400	X	3687-
88	130	054000	Y	3688-
89	131	054400	Z	3689-
90	132	055000	[3690-
91	133	055400	\	3691-
92	134	056000]	3692-
93	135	056400	^	3693-
94	136	057000	_	3694-
95	137	057400	`	3695-
96	140	060000	a	3696-
97	141	060400	b	3697-
98	142	061000	c	3698-
99	143	061400	d	3699-
100	144	062000	e	3700-
101	145	062400	f	3701-
102	146	063000	g	3702-
103	147	063400	h	3703-
104	150	064000	i	3704-
105	151	064400	j	3705-
106	152	065000	k	3706-
107	153	065400	l	3707-
108	154	066000	m	3708-
109	155	066400	n	3709-
110	156	067000	o	3710-
111	157	067400	p	3711-
112	160	070000	q	3712-
113	161	070400	r	3713-
114	162	071000	s	3714-
115	163	071400	t	3715-
116	164	072000	u	3716-
117	165	072400	v	3717-
118	166	073000	w	3718-
119	167	073400	x	3719-
120	170	074000	y	3720-
121	171	074400	z	3721-
122	172	075000	{	3722-
123	173	075400		3723-
124	174	076000	!	3724-
125	175	076400	@	3725-
126	176	077000	~	3726-
127	177	077400	DEL	3727-

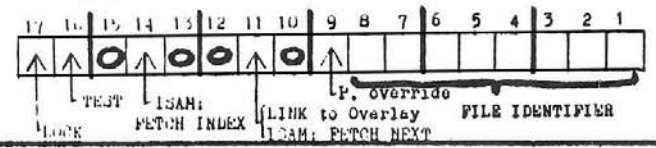
RESIDENT SERVICES

Service Code	Service Name	Service Description
037711	JSHK 12 1711	GENERAL LIBRARY
037713	JSHK 12 1713	Adv [SSTC]
037714	JSHK 12 1714	Add to Double Word [TS]
037715	JSHK 12 1715	Calculate Check Digit [r]
037716	JSHK 12 1716	Clear Block of Core [TC]
037717	JSHK 12 1717	Complement Double Word [T]
037718	JSHK 12 1718	Compute [S'S/SaT]
037719	JSHK 12 1719	Conversion: ASCII -> Binary [STC]
037720	JSHK 12 1720	Conversion: ASCII -> Metacode [STC]
037721	JSHK 12 1721	Conversion: Binary -> ASCII [CST]
037722	JSHK 12 1722	Conversion: Metacode -> ASCII [TSC]
037723	JSHK 12 1723	Conversion: Octal Address -> 9 chars [T]
037724	JSHK 12 1724	Conversion: Octal Word -> 6 chars [T]
037725	JSHK 12 1725	Convert to -ve if +ve [TC]
037726	JSHK 12 1726	Convert to +ve if -ve [TC]
037727	JSHK 12 1727	Divide with Remainder [SST]
037728	JSHK 12 1728	Divide and Round [SST]
037729	JSHK 12 1729	Divide by 10 with Remainder [r]
037730	JSHK 12 1730	Divide by 10 Rounded [r]
037731	JSHK 12 1731	Flash [S]
037732	JSHK 12 1732	Flash Single Station [S]
037733	JSHK 12 1733	Halt [r]
037734	JSHK 12 1734	Left Shift Block of Core [TC]
037735	JSHK 12 1735	Load Character [Sr]
037736	JSHK 12 1736	Load Word [Sr]
037737	JSHK 12 1737	Move & Pad Characters [STC]
037738	JSHK 12 1738	Move Words [STC]
037739	JSHK 12 1739	Multiply [SST]
037740	JSHK 12 1740	Multiply by 10 [TCr]
037741	JSHK 12 1741	Name & Address Processor [PVV]-> PSTVV
037742	JSHK 12 1742	Resolve Offset Address [r]
037743	JSHK 12 1743	Resolve Offset Block [T]
037744	JSHK 12 1744	Right Shift Block of Core [TC]
037745	JSHK 12 1745	Skip if Blocks Equal [SSC]
037746	JSHK 12 1746	Skip if Block Not Zero [SC]
037747	JSHK 12 1747	Space-Fill Block of Core [TC]
037748	JSHK 12 1748	Store Character [Tr]
037749	JSHK 12 1749	Store Word [Tr]
037750	JSHK 12 1750	Subtract [SSTC]
037751	JSHK 12 1751	Subtract from Double Word [TS]
037752	JSHK 12 1752	Suspend *
037753	JSHK 12 1753	Swap Blocks of Core [TTC]
037754	JSHK 12 1754	Unpack Date [Tr]
037755	JSHK 12 1755	INPUT/OUTPUT STATION LIBRARY
037756	JSHK 12 1756	Convert Input & Test Limits [VV]
037757	JSHK 12 1757	to Error Handler *
037758	JSHK 12 1758	Get [P]-> CST. *
037759	JSHK 12 1759	Get Password [S] *
037760	JSHK 12 1760	Inhibit [r]
037761	JSHK 12 1761	Pack Date [r]
037762	JSHK 12 1762	Put [S] *
037763	JSHK 12 1763	to "PROGRAM?" prompt (clears screen) *
037764	JSHK 12 1764	to "PROGRAM?" prompt (without clear) *
037765	JSHK 12 1765	Specify I/O Station Escape Point [V]
037766	JSHK 12 1766	Specify I/O Station Print Queue [r] *
037767	JSHK 12 1767	Specify I/O Station Print Q + Reprint [r] *
037768	JSHK 12 1768	Split [P]-> CS *
037769	JSHK 12 1769	FILE ACCESS LIBRARY
037770	JSHK 12 1770	Delete IS Record [CK] *
037771	JSHK 12 1771	Extract From File Control Block [Cr]
037772	JSHK 12 1772	Fetch [CKT] *
037773	JSHK 12 1773	Load IS Record [CSK] *
037774	JSHK 12 1774	Overwrite [S] *
037775	JSHK 12 1775	Rewrite *
037776	JSHK 12 1776	Stop System Control Record *
037777	JSHK 12 1777	SPOOLING LIBRARY
037778	JSHK 12 1778	Assign Print Queue [r]
037779	JSHK 12 1779	Post to Print Queue [K] *
037780	JSHK 12 1780	Spool *
037781	JSHK 12 1781	Spool & Post to Print Queue *
037782	JSHK 12 1782	Unspool [K] *
037783	JSHK 12 1783	PRINTER LIBRARY
037784	JSHK 12 1784	Nominate Re-entry Point [V]
037785	JSHK 12 1785	Nominate Termination Subroutine [V]
037786	JSHK 12 1786	Print Line [r] *
037787	JSHK 12 1787	to Task Control Program *
037788	JSHK 12 1788	Skip if Original

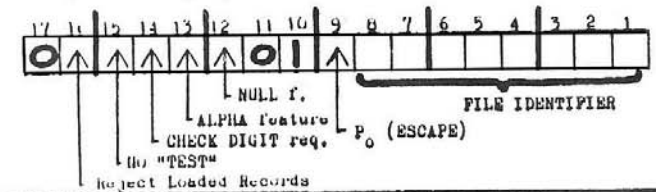
S -> Source; T -> Target; K -> Key; P -> Parameter Block
 C = Control word; V = Value; r = register; * breaks file LOCK

CONTROL WORD FORMATS

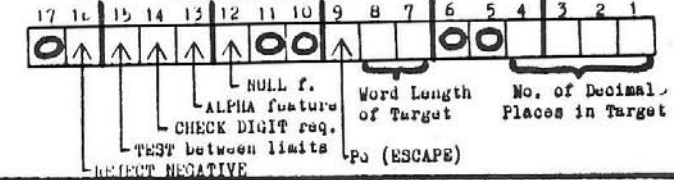
FETCH



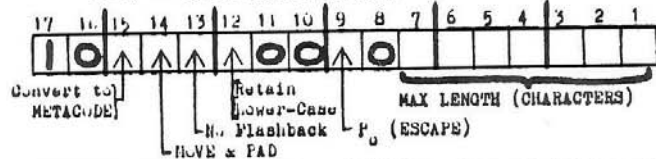
GET - FETCH mode



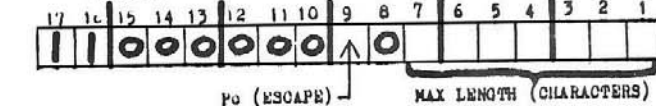
GET - BINARY mode



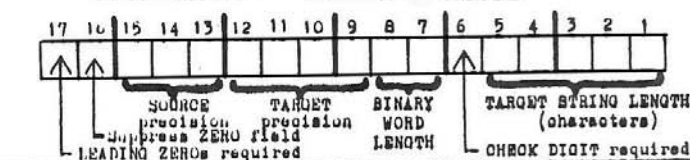
GET - CHARACTER mode



SPLIT



CONVERSION - BINARY -> ASCII



EXTRACT FROM FILE CONTROL BLOCK

