

LOS 92 ZERO PAGE

Page:- 00 Col:- 00

Step		Octal	Step
00		*****	00
01		*****	01
02	HARDWARE INTERRUPT ADDRESS		02
03	JUMP IZ 0004	BACK ADDRESS	03
04	→ INTERRUPT HANDLER		04
05	JUMP Z 0005 TRAP	1/1200	05
06			06
07			07
10	→ DISMISS INTERRUPT handler	1/1227	10
11	→ CONNECTION Pkt	1/1645	11
12			12
13	→ MODEN INPUT SERVICE (1/1645)		13
14	→ SYSTEM TABLE		14
15	IORST (from MAINS OFF rtn)	000017	15
16	LDA Z 0021	212021	16
17	STA Z 0002	252002	17
20	HALT (from MA=SW rtn)	000001	20
21	JUMP IZ 0010 Dismiss Interrupt	026010	21
22	→ PRINT QUEUE TABLE origin		22
23	→ TASK SPOOL TABLE origin		23
24	→ DEVICE TABLE origin		24
25	→ File TABLE TABLE origin		25
26	→ MAINS OFF DEVICE LIST		26
27	Abnormal Interrupt Flag (used by 003333 HALT)		27
30	SPOOLING ACTIVITY COUNTER		30
31	COMMAND ROUTINE ACTIVE (B17)		31
32	→ ON-LINE SECURITY CONTROL BLOCK		32
33	*ENTRY LOCKOUT routine		33
34	INT ON	back address	34
35	INT ON	000006	35
36	→ JUMP Z 0036 loop	000004	36
37	→ FIRST FREE CORE SEGMENT	022036	37
40	CURRENT TASK NUMBER		40
41			41
42			42
43	A reg after Convert to Binary		43
44	B reg after Convert to Binary		44
45	Number of Characters Input on last GET/SPLIT call		45
46	Latest TASK to use COMMAND rtn		46
47	→ TASK CONTROL TABLE origin		47
50	→ SUSPEND STACK 1st Entry		50
51	Maximum TASK NUMBER		51
52	→ DISC CONTROL TABLE origin		52
53	→ DISC LIST origin		53
54	→ first FREE DISC VECTOR		54
55	Number of I/O STATIONS		55
56	→ FREE SUSPEND CHAIN		56
57	MAXIMUM PRINT QUEUE NUMBER (=DELETIONS QUEUE NO)		57
60	MAXIMUM PLAIN PAPER PRINT QUEUE NUMBER		60
61	Print Spool Queue Extension Count		61
62			62
63			63
64			64
65	BASE ADDRESS (+bit17 set)		65
66	→ INPUT or PRINT BUFFER (3600-)	CURRENT TASK	66
67	→ SPOOL BUFFER (3400-)	CURRENT TASK	67
70	→ Next Record Number for SPOOL routine	CURRENT TASK	70
71		CURRENT TASK	71
72	→ MASTER BUFFER (3200-)	CURRENT TASK	72
73	→ FILE TABLE	CURRENT TASK	73
74	→ TASK CONTROL AREA (3720-)	CURRENT TASK	74
75			75
76	→ INHIBIT MASKS		76
77	→ SYSTEM DATE		77

Programmer:-

09

Page:- 00 Col:- 01

Step	Instruction	Address	Comment	Octal	Step
00	CHB		*BOOTSTRAP	004400	00
01	DAT03B		Dfuo0 Exclowelab	010670	01
02	DAT03B		Cone Address of	010570	02
03	DAT01B/START		Secta of Read	011470	03
04	JUMP	0104	Jump	020104	04
05					05
06					06
07			UN ALLOCATED	-	07
10			DEVICE 70	-	10
11			DISC CONTROL TABLE	71	11
12			BIT=BUSY, 31-16 -> 120 element.	72	12
13				73	13
14			DEVICE 70	-	14
15			-> DISC LABEL AREA	71	15
16			(12secta)	72	16
17				73	17
20			0.	-	20
21			FILE TABLE TABLE	1.	21
22				2.	22
23				3.	23
24				000000	24
25			Directory System Table	000000	25
26				000000	26
27				000000	27
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
40			-> NEXT QUEUE ELEMENT	-	40
41			ISSUING TASK NUMBER	-	41
42			OPTIONS (BIT=lock) / FILE IDENTIFIER	-	42
43			R/W, No. of Sectors, DISC NUMBER	-	43
44			-> BUFFER or -> SHARED BUFFER CONTROL BLOCK	-	44
45			DRIVE SECTOR NUMBER	-	45
46	Current		-> EXTRACT AREA / HASH TOTAL / Next Record (POST)	-	46
47	DISC Q		Options, File Identifier (Sector 2)	-	47
50	TRANSFER VECTOR		-> RECORD KEY	-	50
51			-> RECORD within Buffer	-	51
52			LOGICAL RECORD LENGTH	-	52
53			OVERLAY Rtn. NUMBER / Spool Record No.	-	53
54			PROGRAM ENTRY POINT	-	54
55			FETCH Return Address (Sector 2) / Spool Record No.	-	55
56			FETCH / Spool / Post. Return Address	-	56
57			LOAD / REWRITE RETURN ADDRESS	-	57
60			GET/SPUT CONTROL WORD	-	60
61			" " Return Address	-	61
62			" " Parameter Counter	-	62
63			GET/FETCH -> Target Key Area (address)	-	63
64			GET Ring -> Min (Sector) Key -> FCB	-	64
65			" " -> Max (Sector)	-	65
66				-	66
67	TRANSIENT			-	67
70	WORK SPACE			-	70
71				-	71
72				-	72
73				-	73
74				-	74
75				-	75
76				-	76
77				-	77

OS - CONSTANT FACTORS 0 to 63

Page:- 00 Col:- 02

Step	Instruction	Address	Comment	Octal	Step
00			NUL	0	00
01			Bit 1 SOH	1	01
02			Bit 2 STX	2	02
03			ETX	3	03
04			Bit 3 EOT	4	04
05			ENQ	5	05
06			ACK	6	06
07			BEL	7	07
10			Bit 4 BS	8	10
11			TAB	9	11
12			LF	10	12
13			VT	11	13
14			FF	12	14
15			CR	13	15
16			SO	14	16
17			SI	15	17
20			Bit 5 DLE	16	20
21			DC1	17	21
22			DC2	18	22
23			DC3	19	23
24			DC4	20	24
25			NAK	21	25
26			SYN	22	26
27			ETB	23	27
30			CAN	24	30
31			EM	25	31
32			SUB	26	32
33			ESC	27	33
34			FS	28	34
35			GS	29	35
36			RS	30	36
37			US	31	37
40			Bit 6 SP	32	40
41			!	33	41
42			"	34	42
43			#	35	43
44			\$	36	44
45			%	37	45
46			&	38	46
47			'	39	47
50			(40	50
51)	41	51
52			*	42	52
53			+	43	53
54			,	44	54
55			-	45	55
56			.	46	56
57			/	47	57
60			φ	48	60
61			1	49	61
62			2	50	62
63			3	51	63
64			4	52	64
65			5	53	65
66			6	54	66
67			7	55	67
70			8	56	70
71			9	57	71
72			:	58	72
73			;	59	73
74			<	60	74
75			=	61	75
76			>	62	76
77			?	63	77

Programmer:-

OS - CONSTANT FACTORS

Page:- 00 Col:- 03

Step	Instruction	Address	Comment	Octal	Step	
00			Initial word 0	000000	00	
01				000000	01	
02			B:17 @ 64	000100	02	
03			A 65	000101	03	
04			F 70	000106	04	
05			K 75	000113	05	
06			P 80	000120	06	
07			U 85	000125	07	
10			Z 90	000132	10	
11				95	000137	11
12				99	000143	12
13				100	000144	13
14				120	000170	14
15				125	000175	15
16			B:18	128	000200	16
17				150	000226	17
20				192	000300	20
21				200	000310	21
22				250	000372	22
23			B:19	256	000400	23
24				300	000454	24
25				320	000500	25
26				384	000600	26
27				400	000620	27
30				448	000700	30
31				500	000764	31
32			B:10	512	001000	32
33				600	001130	33
34				700	001274	34
35				800	001440	35
36				900	001604	36
37				999	001747	37
40				1000	001750	40
41			B:11	1024	002000	41
42				1500	002734	42
43				2000	003720	43
44			B:12	2048	004000	44
45				3000	005670	45
46				4000	007640	46
47			B:13	4096	010000	47
50				5000	011610	50
51				6000	013560	51
52			B:14	8192	020000	52
53				9999	023417	53
54				10000	023420	54
55			B:15	16384	040000	55
56			B:16	32768	100000	56
57				50000	141520	57
60				60000	165140	60
61				65535	177777	61
62				100000	000001	62
63					103240	63
64				1000000	000017	64
65					041100	65
66				10000000	000230	66
67					113200	67
70				9999999	002765	70
71					160377	71
72						72
73						73
74			"SP SP"		020040	74
75				127	000177	75
76				-1	377777	76
77			B:17		200000	77

Programmer:-

OS - D800 Read/Write OS

Page:- 0 Col:- 04

Step	Instruction	Address	Comment	Octal	Step
00					00
01					01
02	* INTERRUPT			← BA →	02
03	JUMP	0003	loop - Interrupt should be OFF		03
04			→ Entry from Cold Start 6/0000		04
05	JUMP	I 0006	to "OP" routine		05
06			→ "OP" routine 11/0600		06
07			SECTOR No., CONFIGURATION TABLE	000042	07
10			SECTOR No., OS R/W op	000041	10
11			No. of Sectors, Configuration Table	000003	11
12			Recovery Proc { PHASE1 Source/Target		12
13			Table { PHASE2 Source/Target		13
14			{ No. of Sectors, Phase 1		14
15			{ No. of Sectors, Phase 2		15
16					16
17					17
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
30					30
31					31
32					32
33					33
34					34
35					35
36	JSBR	Z 0045	Test Status		36
37	JUMP	0102	Status - Retry		37
40	LDA	Z B	340000 } Same Drive ID		40
41	ANDA	0107			41
42	STA	0167			42
43	ESWRA				43
44	JUMP	0110	Header		44
45	*ENTRY		WAIT for "Done" to Test Status	← BA →	45
46	DONE			012770	46
47	JUMP	0046			47
50	DATIBA/STOP			016370	50
51	NOOP				51
52	AND		Status ?		52
53	INSZ	0045	No.		53
54	JUMP	I 0045	Return.		54
55				"DATOIB/START"	011470
56				"DATOIB/IOAS"	013470
57	*ENTRY		TRANSFER	← BA →	57
60	STA	0174	Care Addresses		60
61	NOOP				61
62	NOOP				62
63	LDA	0174			63
64	DATORA			014570	64
65	DATOIB/START+IOAS				65
66	JSBR	Z 0045	WAIT for "Done" to Test Status		66
67	JUMP	0061	Status - Retry		67
70	LDA	0174	Care Addresses		70
71	JUMP	I 0057	Return		71
72	JSBR	Z 0114			72
73	LDA	0171			73
74	JUMP	I 0004	6/0000 to initialized		74
75	DONE			(from 0105) 012770	75
76	JUMP	0075			76
77	JUMP	0036			77

OS - 2800 Read/Write OS (cont)

Page:- 0 Col:- 05

Step	Instruction	Address	Comment	Octal	Step
00	IORST		*BOOTSTRAP	000017	00
01	INT OFF			000005	01
02	CAB			004400	02
03	DAT02B			010570	03
04	DAT01A/START			011470	04
05	JUMP	0075			05
06					06
07			MASK	340000	07
10	STA	0171	*DYNAMIC ENTRY		10
11	CHA				11
12	JUMP	0072			12
13					13
14	*ENTRY		OS Read/Write	← BA →	14
15	LDB	0167			15
16	STB	0173			16
17	LDB	0055	"DAT01A/START"		17
20	APOS				20
21	LDB	0056	"DAT01B/START"		21
22	STB	0065			22
23	LDA	0163	No. of Sectors	*RETRY	23
24	STA	0177	Sector Count		24
25	SWAP				25
26	RSA				26
27	STA	0176	Word Count		27
30	LDA	0164	Start Core Address 0/0200		30
31	STA	0175			31
32	LDB	0165			32
33	JSBR	0057	TRANSFER	*NEXT SECTOR	33
34	ADA	0164	CFIX		34
35	INCB				35
36	DESZ	0177	Sector Count		36
37	JUMP	0133	auto next sector		37
40	CHA			[next JUMP 0670]	40
41	ADA	I 0175			41
42	INSZ	0175			42
43	DESZ	0176	Word Count	Calculate Header Sum	43
44	JUMP	0141			44
45	LDB	0065			45
46	CHPB	0056	"00AS"		46
47	JUMP	0154	this is a "Write" cell		47
50	CHPA	0170	previous Header Sum		50
51	JUMP	I 0114	OK - return		51
52	HALT		HASH FAIL		52
53	JUMP	0123	Retry		53
54	STA	0170	insert new header sum		54
55	CHA		Address		55
56	CAB		Sector		56
57	JSBR	0057	Write Sector		57
60	LDB	0010			60
61	JSBR	0057	Write Sector 41		61
62	JUMP	I 0114	return.		62
63			No. of Sectors	95	63
64			Start Core Address / Sector Size	000200	64
65			Start Sector No	000050	65
66			MASK	000777	66
67			Point	-	67
70			Header Sum	-	70
71			Base Addr.	-	71
72				-	72
73			Dirip	-	73
74			Core Addr.	-	74
75			Core Addr.	-	75
76			Word Count	-	76
77			Sector Count	-	77

OS- 6M → MHI conversion table

Page:- 0 Col:- 06

Step	Instruction	Address	Comment	Octal	Step
00			{ 5/0100		00
01	JORST			000017	01
02			{ 5/0101		02
03	INT OFF			000005	03
04			{ 5/0102		04
05	CHB			004400	05
06			{ 5/0103		06
07	DAT02B			010570	07
10			{ 5/0104		10
11	DAT01A/START			011470	11
12			{ 5/0105		12
13	JUMP	0105		020105	13
14			{ 11/0143		14
15	JUMP	1041	End		15
16			{ 5/1776		16
17			Reverse To Fixed	040070	17
20			{ 5/1102		20
21	LDA	Z 0011	No. of sectors		21
22			{ 5/1103		22
23	STA	Z 0177	Counter		23
24			{ 5/1106		24
25	JUMP	1200	Transfer to program table		25
26			{ 5/1154		26
27	Noop				27
30			{ 5/1157		30
31	LDB	Z A			31
32			{ 5/1161		32
33	ADB	1134	Label Sector No.		33
34			{ 5/0001		34
35			2/0720 → Interrupt Handler		35
36			{ 10/11610		36
37			Φ Reverse from Free Collection	000000	37
40			{ 2/11003		40
41			2/0716		41
42			{ 0/0140		42
43	CHA				43
44					44
45				000000	45
46					46
47					47
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
60					60
61					61
62					62
63					63
64					64
65					65
66					66
67			Table pointer	0/6600	67
70	LDB	I 0667			70
71	BNØ				71
72	JUMP	I 0114	Return.		72
73	INSZ	0667			73
74	LDA	I 0667			74
75	STA	I2 B			75
76	INSZ	0667			76
77	JUMP	0670	automatic.		77

OS - Dummy Disc Label Buffer

(Sector)

Page:- 0 Col:- 10

Step	Instruction	Address	Comment	Octal	Step
00			DISC NUMBER	000300	00
01			Non-Security	000000	01
02			Fixed Cartridge, Disc ϕ	200000	02
03					03
04					04
05					05
06					06
07					07
10					10
11					11
12					12
13					13
14					14
15					15
16					16
17					17
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
60					60
61					61
62					62
63					63
64					64
65					65
66					66
67					67
70					70
71					71
72					72
73					73
74					74
75					75
76					76
77					77

OS - Dummy Disc Label Buffer (continued)

Page:- 0 Col:- 11

Step	Instruction	Address	Comment	Octal	Step
00					00
01					01
02					02
03					03
04					04
05					05
06					06
07					07
10					10
11					11
12					12
13					13
14					14
15					15
16					16
17					17
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
60					60
61					61
62					62
63					63
64					64
65					65
66					66
67					67
70					70
71					71
72					72
73					73
74					74
75					75
76					76
77					77

Programmer:-

OS - FILE TABLE ϕ

Page:- 00 Col:- 12

Step	Instruction	Address	Comment	Octal	Step
00					00
01					01
02					02
03					03
04					04
05					05
06					06
07					07
10					10
11					11
12					12
13					13
14					14
15					15
16					16
17					17
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
30					30
31					31
32					32
33					33
34					34
35					35
36					36
37					37
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
60					60
61					61
62					62
63					63
64					64
65					65
66					66
67					67
70					70
71					71
72					72
73					73
74					74
75					75
76					76
77					77

Programmer:-

OS-

Page:- 00 Col:- 14

Step	Instruction	Address	Comment	Octal	Step
00	JSBR	I2 1652	PUT "CAN" (Clear Screen)		00
01	P _i = 7/1700				01
02	JUMP	I2 1403	to I/O STATION CONTROL PROGRAM		02
03			→ I/O STATION CONTROL PGM 4/1440		03
04			→ PRINTER CONTROL PGM 4/1500		04
05	JUMP	I2 1641	TO ERROR HANDLER		05
06	DESZ	Z 0030	Specifying Activity counter (Printer Idle)		06
07	NOOP				07
10	→ JUMP	I2 1404	to PRINTER CONTROL PGM		10
11			→ PRINTER COMPLETION 3/0600		11
12			→ SKIP if I/O STATION An. 1/0371		12
13			→ Skip if I/O Print Q with Request MTA. 2/0235		13
14			→ Resolve Absolute Byte 2/0335		14
15			→ Load Absolute Byte 2/0345		15
16			→ Store Absolute Byte 2/0360		16
17			→ Load Absolute Byte (interrupt off) 4/0461		17
20			INVALID FILE 12/1170		20
21			"BINARY" DIRECT 12/0725		21
22			"ORDINARY" DIRECT 12/0157		22
23			PROGRAM DIRECTORY 12/1000		23
24			PROGRAM OVERLAY 12/0676		24
25			PROGRAM 12/0700		25
26			HYBRID 13/0755		26
27	FILE ACCESS		INDEXED SEQUENTIAL 7/0210		27
30	ROUTINE				30
31	POINTERS				31
32					32
33					33
34					34
35					35
36					36
37					37
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
60					60
61					61
62					62
63					63
64					64
65					65
66					66
67					67
70					70
71					71
72					72
73					73
74					74
75					75
76					76
77					77

Programmer:-

OS-

Page:- 00 Col:- 15

Step	Instruction	Address	Comment	Octal	Step
00					00
01					01
02					02
03					03
04					04
05					05
06					06
07					07
10					10
11					11
12					12
13					13
14					14
15					15
16					16
17					17
20					20
21					21
22					22
23					23
24					24
25					25
26					26
27					27
30					30
31			fire core 129		31
32					32
33					33
34					34
35					35
36					36
37					37
40					40
41					41
42					42
43					43
44					44
45					45
46					46
47					47
50					50
51					51
52					52
53					53
54					54
55					55
56					56
57					57
60					60
61					61
62					62
63					63
64					64
65					65
66					66
67					67
70					70
71					71
72					72
73					73
74					74
75					75
76					76
77					77

Programmer:-

OS

Page:- 00 Col:- 16

Step	Instruction	Address	Comment	Octal	Step
00					00
01					01
02					02
03			STORE ABSOLUTE BYTE (ind off) 4/1420		03
04			DECODE (returns 12cler) 12/0160		04
05			ADDRESS → ASCII (returns 9cler) 13/0670		05
06			CONVERT INPUT to Binary & Test Limits 3/1300		06
07			PACK DATE 3/1332		07
10			SPECIFY PRINTER PROGRAM RE-ENTRY 4/1765		10
11			SPECIFY " OVERLAY TERMINATION 4/1755		11
12			OCTAL → ASCII CONVERSION 3/1357		12
13			SPECIFY I/O STATION PRINT @ 2/0221		13
14			ASSIGN PRINT QUEUE 2/0461		14
15			TRANSFER 12/1600		15
16			STACK 13/0631		16
17			COMMAND ROUTINE 2/1500		17
20			ESTABLISH TASK 13/0241		20
21			STIN 13/0400		21
22			STOUT 12/1532		22
23			LOAD@ 12/1651		23
24			TASK SCHEDULER 13/1102		24
25			SUSPEND 13/0226		25
26			SUSPEND 2 13/0231		26
27			RESOLVE OFFSET BLOCK 2/0200		27
30	*ENTRY		RESOLVE OFFSET ADDRESS		30
31	(APOS			← BA →	31
32	ADA	Z 0065	+Base + 167 [2cler carry]	007200	32
33	→ JUMP	I 1630	Return	112065	33
34				025630	34
35			SPECIFY ESCAPE POINT 3/1422		35
36			GET PASSWORD 3/0700		36
37			INHIBIT 3/0761		37
40			METACODE / ASCII CONVERSION 4/1066		40
41			GET / SPLIT / GET & FETCH 13/0033		41
42			ERROR HANDLER 13/0200		42
43			TASK INITIATOR 2/0700		43
44			SPECIFY DEFAULT RESTART ADDR. 3/1151		44
45			PRINTER OUTPUT 4/0200		45
46			SPOOL 12/0266		46
47			UNSPool 12/0066		47
50			POST to PRINT QUEUE 12/0200		50
51			STRIP (UNSPool) / Q not exact 12/0300		51
52			SPOOL & POST / Q original 12/0121		52
53			PUT 13/0000		53
54			FLASH ALL STATIONS 13/1600		54
55			FLASH SINGLE STATION 13/1642		55
56			NAME & ADDRESS PROCESSOR 3/1600		56
57					57
60					60
61			ISAM head / Relate Headers 7/0546		61
62			→ Restart "FETCH" (not ch GET) 13/0742		62
63			ISAM Search in Progress Flag		63
64			ISAM Find Entry in Primary Index 7/0300	000000	64
65			ISAM Find Entry in Secondary Index 7/0400		65
66			SECURITY in Progress FLAG (P/O)	000000	66
67			STOW CONTROL RECORD 12/1267		67
70			FETCH 12/1100		70
71			REWRITE 12/1506		71
72			OVERWRITE 12/1500		72
73			WRITE 12/1536		73
74			RESCHEDULE DISC QUEUE 12/1345		74
75			ASSIGN FILE TABLE 4/1700		75
76			HALT (RECORD KEY ERROR) 12/1250		76
77			BUFFER PROCESSOR (not ch Fetch) 12/10750		77

Programmer:-

OS

Step	Instruction	Address	Comment	Octal	Step
00			EXTRACT WORD FROM FIRE CONTROL BLOCK	2/0473	00
01			CONVERT to + if -	1/0121	01
02			CONVERT to - if +	1/0134	02
03			SAVE REGISTERS	2/0262	03
04			GET MAIN	3/1462	04
05			PREATTENT INTERCEPT	3/1513	05
06			SWAP BLOCKS	1/0174	06
07			DUPLICATE BLOCK	1/0212	07
10			CLEAR BLOCK	1/0226	10
11			ADD	1/0236	11
12			SUBTRACT	1/0266	12
13			ADD into DOUBLE WORD	1/1023	13
14			SUBTRACT from DOUBLE WORD	1/0700	14
15			CORE LENGTH (WORDS)	/	15
16			Start/stop Mask	/	16
17			DEVICE CODE	/	17
20			DEVICE STATUS	/	20
21			LOAD A FROM OFFSET	1/0737	21
22					22
23			SKIP if BLOCKS EQUAL	1/0722	23
24			SKIP if BLOCK NOT ZERO	1/0104	24
25			STORE A AT OFFSET	1/0746	25
26			DIVIDE & ROUND	1/0611	26
27			RIGHT SHIFT BLOCK	1/0044	27
30			LEFT SHIFT BLOCK	1/0057	30
31			SPACE FILL BLOCK	1/0666	31
32			DIVIDE with REMAINDER	1/0245	32
33					33
34			MULTIPLY	1/0257	34
35				1/1344	35
36			Cancel Special Mode	1/1354	36
37			Set Special Mode	1/1309	37
40			CHECK DIGIT CALCULATION	3/0772	40
41			MOVE & PAD	3/1032	41
42					42
43				1/1157	43
44			DIVIDE A by 10 ROUNDED	1/1112	44
45			DIVIDE A by 10 UNROUNDED	1/1100	45
46					46
47			COMPUTE	1/0634	47
50			COMPLEMENT DOUBLE WORD	3/1437	50
51			UNPACK DATE	3/1201	51
52			BOTTOM BYTE MASK	000377	52
53			TOP BYTE MASK	177400	53
54					54
55			SKIP if OUT	2/0231	55
56			SKIP if IN	2/0400	56
57			TEST if DEVICE CLEAR	1/1371	57
60			(Current Mask) (not used)		60
61			MULTIPLY by 10	1/1603	61
62			ASCII → BINARY CONVERSION	1/1417	62
63			INPUT	1/1305	63
64					64
65			BINARY → ASCII CONVERSION	1/1712	65
66			REMOVE INDIRECT	1/1047	66
67				1/1701	67
70				1/1603	70
71			LOAD BYTE	3/1166	71
72			CONVERT to UPPER CASE	2/0662	72
73					73
74					74
75			STORE BYTE	3/1157	75
76			SKIP if ORIGINAL	4/1653	76
77			HALT	3/1107	77

Programmer:-