

OS - Pgm 1

Page:- Col:- 00

Step	Instruction	Address	Comment	Octal	Step
00			-> Module No. (LIST) Spool Buffer	3571-	00
01			-> "Read" End of Area	2000-	01
02			-> Key Area ("KEY")	3640-	02
03	Offset Addresses			110005-	03
04					04
05					05
06					06
07					07
10	CLA/COMPBA		(Main Program) + LIST		10
11	JSBR	I2 1613	Specify I/O Station / Unit		11
12	JSBR	I2 1652	P4T "Module / Core Dump / Unit"		12
13	P1=0160-				13
14	JSBR	I2 1634	Specify External Units		14
15	P1=011402				15
16	JSBR	I2 1770	GET OCTAL "MODULE"		16
17	P1=0273 1/2-				17
20	CMPA	Z 0344	004000		20
21	NOOP				21
22	SKNGT				22
23	JUMP	I2 1641	Error - invalid module No.		23
24	STA	I 0077	= Module No. / Unit		24
25	STA	I 0000	Link Spool Buffer		25
26	AND				26
27	JUMP	0050	Core Dump between units		27
30	JSBR	I2 1612	Octal -> ASCII core No.		30
31	P1=0214 1/2-				31
32	JSBR	I2 1741	Move Pad (Heading -> Spool Buffer)		32
33	P1=0200-				33
34	P2=3500-				34
35	P3=602 dump				35
36	JSBR	I2 1640	GET "TITLE"		36
37	P1=0074-				37
40	JSBR	I2 1651	Spool & POST		40
41	LDA	I 0077	= Module No. last time		41
42	AND				42
43	JUMP	0025	Out next core dump		43
44	JUMP	0016	Out next module		44
45					45
46					46
47					47
50	JSBR	I2 1770	Get Octal "CORE FLOW" & Core Dump		50
51	P1=0262 1/2-				51
52	JSBR	I2 1725	STA Low Address		52
53	P1=3572-				53
54	JSBR	I2 1605	Address -> ASCII		54
55	P1=0244 1/2-				55
56	JSBR	I2 1770	Get Octal "To"		56
57	P1=0270 1/2-				57
60	JSBR	I2 1725	STA High Address		60
61	P1=3573-				61
62	JSBR	I2 1605	Address -> ASCII		62
63	P1=0252 1/2-				63
64	JSBR	I2 1741	Move Pad (Heading -> Spool Buffer)		64
65	P1=0230-				65
66	P2=3500-				66
67	P3=602 dump				67
70	JUMP	0036	NOOP		70
71					71
72					72
73				b 0014-	73
74				{ 0174-	74
75			Get "TITLE"	22 0441	75
76				{ 3530-	76
77			Module No.		77

Programmer:-

DS-1

Page:- Col:-01-

Step	Instruction	Address	Comment	Octal	Step
00	CMPA	2 0303	"NUL A"		00
01	JUMP	0107	Next		01
02	CMPA	0156	"NUL C"		02
03	JUMP	0131	Chain Link		03
04	CMPA	0157	"NUL N"		04
05	JUMP	0135	Next Link		05
06	JUMP	IL 1641	End		06
07	JSBR	1770	Get Octal "STEP" + AMOD.		07
10	P ₁ =0152-				10
11	ADA	0001	→ Extract Hex		11
12	STA	0077			12
13	JSBR	1770	Get Octal "OCTAL"		13
14	P ₁ =0145E-				14
15	STA	0076	F ₁ =1		15
16	LDA	0140	F ₂ = 1		16
17	CMPA/COMP				17
20	STA	0122			20
21	JSBR	IL 1670	Fetch 2000 Hex		21
22	P ₁ = /				22
23	P ₂ = 3640-				23
24	P ₃ = 2000-		Extract Hex		24
25	NOOP				25
26	JUMP	I 0372			26
27	JSBR	IL 1671	REINIT		27
30	JUMP	1320	Display Auto vert.		30
31	JSBR	1770	Get Octal "STEP" + Chain Link		31
32	P ₁ = 0152-				32
33	ADA	0001	→ Extract Hex		33
34	STA	0077	→ Chain Link Word		34
35	LDA	I 0077	+ Next Link		35
36	STA	I 0002			36
37	JSBR	IL 1670	Fetch		37
40	P ₁ = /		F ₂ = 1 (Plan 1303) (Correctly 0116)		40
41	P ₂ = 3640-				41
42	P ₃ = 2000-		Extract Hex		42
43	NOOP				43
44	JUMP	1320	Display Auto vert.		44
45				SP	45
46				SP 0	46
47				C T	47
50				A 2	50
51				SP NUL	51
52				SP SP	52
53				S T	53
54				E P	54
55				SP NUL	55
56				NUL C	56
57				NUL N	57
60				CR M	60
61				O D	61
62				U L	62
63				E /	63
64				C O	64
65				R E	65
66				S D	66
67				U M	67
70				P SP	70
71				P R	71
72				I N	72
73				T NUL	73
74				CR T	74
75				I T	75
76				L E	76
77				SP NUL	77

Programmer:-

OS - Pgm 1

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	DD				14
15	AT		Octal Mode No.		15
16	A?				16
17				NUL	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	LDA	255			45
46	SIZE				46
47	LDA	256			47
50	Jump	253			50
51					51
52					52
53	STA	1572			53
54	Jump	1305			54
55	STA	1572	214-		55
56	Jump	1305	1535-		56
57				NUL	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 1

Page:- Col:- 03

Step	Instruction	Address	Comment	Octal	Step
00	CHA/COMP SA		(May Plan Dept Q) + PMAP		00
01	JSBR	12 1612	Specify I/O Station Point Q		01
02	JSBR	12 1652	DAT "Program & Overlay Ref."		02
03	Pi = 0340 1/2 -				03
04	JSBR	12 1634	Specify Exported		04
05	Pi = 011402				05
06	CHA				06
07	JSBR	12 1725	STA Min		07
10	Pi = 3401 -				10
11	LDA	Z 0361	177177		11
12	JSBR	12 1725	STA Max		12
13	Pi = 3402 -				13
14	JSBR	Z 1770	Get Octal "PROCESS?"		14
15	Pi = 0352 -				15
16	AND				16
17	JUMP	0332			17
20	JSBR	12 1725	STA Min		20
21	Pi = 3401 -				21
22	JSBR	12 1725	STA Max		22
23	Pi = 3402 -				23
24	JSBR	1770	Get Octal "LMS"		24
25	Pi = 0363 -				25
26	AND				26
27	JUMP	0332			27
30	JSBR	12 1725	STA		30
31	Pi = 3402 -				31
32	JSBR	12 1651	Specify Point		32
33	JUMP	Z 1402	to Max PMA		33
34					34
35					35
36					36
37					37
40					40
41				CR P	41
42				O G	42
43				R A	43
44				H SP	44
45				H A	45
46				P N42	46
47				011402 P0	47
50			SPLIT "PROCESS?"		50
51			(PMAP, ZERO)	0352 -	51
52				CR P	52
53				R O	53
54				C E	54
55				S S	55
56				? N42	56
57				CR F	57
60				I R	60
61				S T	61
62				SP N42	62
63				SP SP	63
64				H A	64
65				S T	65
66				SP N42	66
67					67
70					70
71					71
72	LDA	0076			72
73	STA	I 0077			73
74	JUMP	0127			74
75	JORA	Z 0323	hd 9 (Instruction as job)		75
76	STA	0461			76
77	JUMP	0417			77

Programmer:-

OS Page 1.

Page:- Col:- 04

Step	Instruction	Address	Comment	Octal	Step
00	JSBR	IL 1652	PUT "PROGRAM AISKRY KAZATELAKI"		00
01	P ₁ = 0700 -				01
02	JSBR	IL 1634	Specify Escape Point		02
03	P ₁ = 0406 -				03
04	LDA	Z 0201	Bit 1		04
05	JSBR	IL 1636	INHIBITED		05
06	JSBR	IL 1640	SPLIT "AMEND DIRECTORY?"		06
07	P ₁ = 1610 -				07
10	JUMP	0543	No.		10
11	JSBR	IL 1640	Yes. SPLIT "INPUT?"		11
12	P ₁ = 1602 -				12
13	JUMP	0476	No. - No. of		13
14	LDA	Z 0203	Yes.		14
15	STA	0424			15
16	JUMP	0375	Patch.		16
17	LDA	0415			17
20	STA	0462			20
21	JSBR	IL 1640	GET "NAME"		21
22	P ₁ = 1604 -				22
23	JSBR	IL 1670	FETCH Next directory element		23
24	P ₁ = /				24
25	P ₂ = 1674 -		→ Name		25
26	P ₃ = 1674 -		→ Extract Name		26
27	JUMP	0500	Not found.		27
30	JUMP	0470	Match! → Forward.		30
31	JSBR	IL 1612	Octal → ASCII		31
32	P ₁ = 0747 -				32
33	JSBR	1770	GET "NAME n"		33
34	P ₁ = 0743 -				34
35	LDB	Z 0045	No. of directory input.		35
36	BNB				36
37	JUMP	0451	By-pass - no cancellation.		37
40	AND				40
41	JUMP	0510	Deletion request.		41
42	APOS				42
43	JUMP	IL 1641	Error - out of limit.		43
44	EMPA	Z 0344	004000		44
45	NOOP				45
46	SKNCT				46
47	JUMP	IL 1641	Error - out of limit.		47
50	STA	1676	New Module No.		50
51	LDA	1677	= Entry Point		51
52	JSBR	IL 1605	Address → DSCDI		52
53	P ₁ = 0756 1/2 -				53
54	JSBR	1770	GET "ENTRY x!"		54
55	P ₁ = 0753 -				55
56	A = 0				56
57	STA	1677	New entry point.		57
60	JSBR	IL 1670	FETCH Next directory element		60
61	P ₁ = /				61
62	P ₂ = /		→ Name		62
63	P ₃ = 0		No extraction		63
64	JSBR	IL 1777	Halt - Name directory full.		64
65	JSBR	IL 1672	OVERWRITE		65
66	P ₁ = 1674 -				66
67	JUMP	0406			67
70	JSBR	IL 1652	Put "OVERWRITE" Match from 0430		70
71	P ₁ = 1040 -				71
72	LDA	1676	Module No.		72
73	JUMP	0431			73
74			→ Null Name	1670 -	74
75			→ Name	1674 -	75
76	LDA	Z 0204			76
77	JUMP	0415			77

Programmer:-

OS - Pgm 1.

Page:- Col:- 05

Step	Instruction	Address	Comment	Octal	Step
00	NOOP		NEW NAME		00
01	NOOP		* Hold used name.		01
02	CLA				02
03	STA	1676			03
04	STA	1677			04
05	LDA	0474	→ Hold name.		05
06	STA	0462			06
07	JUMP	0472	Handle with "word" rtn.		07
10	LDA	0461	* Delete entry.		10
11	STA	0513			11
12	JSBR	I2 1670	FETCH BLOCK element		12
13	P ₁ = /		Fetch.		13
14	P ₂ = 1674 -		→ None		14
15	P ₃ = 0		No extraction		15
16	JSBR	I2 1777	PART - name disappeared.		16
17	JSBR	I2 1672	OVERWRITE (clear entry)		17
20	P ₁ = 1670 -				20
21	JSBR	I2 1741	Name/code (name)		21
22	P ₁ = 1674 -				22
23	P ₂ = 0773 -				23
24	P ₃ = 4cler				24
25	JSBR	I2 1652	Put "deleted"		25
26	P ₁ = 0772 -				26
27	JUMP	0406			27
30	JSBR	I2 1670	FETCH BLOCK index + delete index		30
31	P ₁ = 200401				31
32	P ₂ = 1676 -		→ Module No.		32
33	P ₃ = 0		No extraction		33
34	JSBR	I2 1672	OVERWRITE		34
35	P ₁ = 1670 -		Null entry		35
36	JSBR	I2 1652	Put "DELETED"		36
37	P ₁ = 0764 -				37
40	JUMP	0406	Out of cooperation		40
41					41
42					42
43	JSBR	1770	Yes. Get "MODULE"		43
44	P ₁ = 0273 ₂ -				44
45	AND				45
46	JUMP	I2 1641	End		46
47	CHPA	Z 0344	004000		47
50	NOOP				50
51	SKNPT				51
52	JUMP	I2 1641	End		52
53	STA	1676	= Module No.		53
54	JSBR	I2 1670	FETCH Index Record		54
55	P ₁ = "01"				55
56	P ₂ = 1676 -		→ Module No.		56
57	P ₃ = 1664 -		11. 1. 1. 1.		57
60	LDA	I2 0151	= 1st word of record.		60
61	AND				61
62	JUMP	0566	Not loaded		62
63	JSBR	I2 1652	PUT "ADJUMENT"		63
64	P ₁ = 1040 -				64
65	JUMP	0644			65
66	JSBR	I2 1640	GET "SECTORS"		66
67	P ₁ = 1612 -				67
70	LDA	1677	= No. of Sectors		70
71	SWAPA				71
72	IORA	Z 0202	000002		72
73	STA	1664			73
74	JSBR	1770	GET "START LOC'N"		74
75	P ₁ = 1021 ₂ -				75
76	STA	1665			76
77	JUMP	0606			77

Programmer:-

OS - Program

Page:- Col:- 06

Step	Instruction	Address	Comment	Octal	Step
00				1541bb	00
01			(G+D) FIRST ASC NO.	1bb1-	01
02				11b17b-	02
03				11317b-	03
04			MIN	113171-	04
05			MAX		05
06	JSBR	I2 1700	Extract from FCB (→ Sectors In Use Count)		06
07	P ₁ = 4,002				07
10	STA	0642	→ Sectors In Use Count		10
11	SFB	Z 0202	CF2 → No. of Sectors in FCB		11
12	LDA	I 0642	No. of sectors in use		12
13	STA	1666	Start sectors no. - 1		13
14	ADA	1677	No. of sectors required		14
15	CMPA	I2 B	No. of sectors in file		15
16	NOOP				16
17	→ SKNCT				17
20	JUMP	0632	File Field		20
21	→ JUMP	Z 0635	Patch		21
22	JSBR	I2 1670	Fetch block index record		22
23	P ₁ = 200401				23
24	P ₂ = 1676-		→ Module No		24
25	P ₃ = 0		No extraction		25
26	JSBR	I2 1672	OVERWRITE		26
27	P ₁ = 1664-				27
30	NOOP				30
31	JUMP	0406	Out of range operation		31
32	JSBR	I2 1652	PUT "NO SPACE"		32
33	P ₁ = 1047-				33
34	JUMP	0406			34
35	STA	I 0642	Start sectors now in use patch from 0621		35
36	INSZ	1666	Start sector no.		36
37	JSBR	I2 0667	Start control record		37
40	JUMP	0622			40
41					41
42			→ No. of Sectors In Use Count		42
43				277777	43
44	LDA	1666	Start Sector		44
45	JSBR	I2 1612	Octal → ASCII		45
46	P ₁ = 1645-				46
47	JSBR	1770	Get "SECTOR-?"		47
50	P ₁ = 1641-			777777	50
51	P ₁ = LDB	Z 0045	No. of files input		51
52	BNB				52
53	JUMP	0657	bypass - no deletion & amendment		53
54	→ ANB	0.47			54
55	JUMP	0530	Delete Module		55
56	→ STA	1666	New Start Sector No.		56
57	P ₁ = 161 LDA	1665	Core Address		57
60	JSBR	I2 1605	Address → ASCII		60
61	P ₁ = 1656-				61
62	→ JSBR	I2 1770	Get "LDBS AT-?"		62
63	P ₁ = 1651-	1665			63
64	A = φ				64
65	STA	1665	New End Address		65
66	→ JUMP	0622	Updelp		66
67					67
70					70
71					71
72					72
73					73
74					74
75					75
76					76
77					77

Programmer:-

OS Pgm 1.

Page:- Col:- 07

Step	Instruction	Address	Comment	Octal	Step
00					00
01			CR P		01
02			R O		02
03			G R		03
04			A M		04
05			L		05
06			I B		06
07			R A		07
10			R Y		10
11			M		11
12			A I		12
13			N T		13
14			E N		14
15			A N		15
16			C E		16
17			MUL CR		17
20			A M		20
21			E N		21
22			D SP		22
23			D I		23
24			A E		24
25			C T		25
26			O R		26
27			T ?		27
30			MUL SP		30
31			SP "		31
32			I N		32
33			P U		33
34			T "		34
35			? MUL		35
36					36
37					37
40			SP SP		40
41			N A		41
42			H E		42
43			SP MUL		43
44			CR M		44
45			O D		45
46			U L		46
47	LDA	1574	E SP		47
50	LORA	2 352			50
51	JUMP	757			51
52			? MUL		52
53			CR E		53
54			N T		54
55			R Y		55
56			SP		56
57	STA	1574			57
60	LDA	2 352			60
61	STA	1563			61
62	JUMP	245			62
63			? MUL		63
64			BEL SP		64
65			SP SP		65
66			D E		66
67			K E		67
70			T E		70
71			D MUL		71
72			CR BEL		72
73					73
74					74
75					75
76			SP D		76
77			E L		77
			E T		

Programmer:-

OS Page 1.

Step	Instruction	Address	Comment	Octal	Step
00			E D		00
01			NUL CR		01
02					02
03					03
04					04
05					05
06					06
07					07
10			CR S		10
11			E C		11
12			T O		12
13			R S		13
14			(D		14
15			E C		15
16			I H		16
17			A L		17
20) SP		20
21			NUL CR		21
22			S T		22
23			A R		23
24			T SP		24
25			L O		25
26			C I		26
27			N SP		27
30			NUL CR		30
31			A U		31
32			T O		32
33			- R		33
34			E S		34
35			O L		35
36			V E		36
37			? NUL		37
40			BER SP		40
41			SP A		41
42			M E		42
43			N D		43
44			M E		44
45			W T		45
46			NUL		46
47			SP SP		47
50			BER N		50
51			O SP		51
52			S P		52
53			A C		53
54			E NUL		54
55			CR Z		55
56			E R		56
57			O I		57
60			S E		60
61			SP S		61
62			E C		62
63			T O		63
64			R S		64
65			NUL CR		65
66			D I		66
67			S C		67
70			SP N		70
71			O SP		71
72			NUL		72
73					73
74					74
75					75
76					76
77					77

Programmer:-

OS - Pgm 1

Page:- Col:- 11

Step	Instruction	Address	Comment	Octal	Step
00	JSBR	12 1652	PUT title		
01	P ₁ =1055-		↓ ZERO		00
02	LDA	Z 0201	Bd 1		01
03	JSBR	12 1636	Indicated ?		02
04	JSBR	12 1634	Specify Program Point		03
05	P ₁ =0/1400				04
06	JSBR	1 1770	Get "DISC No"		05
07	P ₁ =1065-				06
10	IOAA	1172	200400 "Write 1 sector" Jump	1142	07
11	STA	1175			10
12	JSBR	1770	Get "START SECTOR (OCT)"		11
13	P ₁ =1626-				12
14	STA	1177			13
15	JSBR	12 1640	Get "SECTORS (DEC)"		14
16	P ₁ =1621-				15
17	JSBR	12 1640	SPALT "PROCESS!"		16
20	P ₁ =0350-				17
21	JUMP	1106	No		20
22	JSBR	12 1710	Yes CHECK CORE (Master Buffer)		21
23	P ₁ =3700-	1710	Master Buffer		22
24	P ₂ =128 words				23
25	P ₁ =11 LDA	Z 0072	→ Master Buffer		24
26	STA	1176			25
27	JSBR	12 1707	Duplicate (Read @ data)		26
30	P ₁ =1174-				27
31	P ₂ =0/0142				30
32	P ₃ =4 words				31
33	JSBR	12 1623	LOADQ (Write 1 sector)		32
34	INSZ	Z 0145	Sector No.		33
35	DESZ	1173	Sector Count		34
36	JUMP	1133	Next Sector		35
37	JSBR	12 1652	PUT "PER"		36
40	P ₁ =0/0207-				37
41	JUMP	Z 1106			40
42	AND A	Z 1752			41
43	A=B				42
44	Jump	I 3			43
45	JSBR	1770			44
46	1365 -				45
47	Jump	1155			46
50	JSBR	12 1634	Copy Error Point (Jump 1202)		47
51	P ₁ =0/1400				50
52	LDA	Z 0201	Bd 1		51
53	JSBR	12 1636	Indicated ?		52
54	JUMP	1204			53
55	IOAA	1172			54
56	Jump	1111			55
57	Com. A	1223			56
60	Jump	747			57
61	Jump	1223			60
62	Jump	117			61
63			SP 0		62
64			V E		63
65			R W		64
66			R I		65
67			T E		66
70			? WPL		67
71			SP 0	300000	70
72			1162-		71
73			WPL	200400	72
74			Sector Count	/	73
75			Options	000000	74
76			W, Sectors Via No.	/	75
77			→ Buffer	/	76
77			Sector Number	/	77

Programmer:-

OS-Program 1 READ

Page:- 0 Col:- 12-

Step	Instruction	Address	Comment	Octal	Step
00	JSBR	12	1652	1st till *READ	00
01	R=1254-				01
02	JUMP		1150	fetch	02
03				1127-	03
04	JSBR		1770	Get Octal "Disc No"	04
05	R=10652-				05
06	ANDA	Z	1752	Leftmost Byte	06
07	AND				07
10	JUMP		1300		10
11	JORA	Z	0323	000400 Read 1 sector	11
12	STA		1251		12
13	LDA		1251	↓ NEXT SECTOR	13
14	CLSA			erase "Read"	14
15	STA		1251		15
16	JSBR		1770	Get Octal "Sector (Oct)"	16
17	R=12622-				17
20	APOS				20
21	JUMP		1233	1231	21
22	STA		1253		22
23	JSBR	12	1615	Transfer (Read into Master Buffer)	23
24	R=1250-				24
25	LDA	Z	0144	→ Buffer	25
26	KDB	Z	0316	Record length 128 words	26
27	JSBR		1325		27
30	JUMP		1213	data next	30
31	CLSA				31
32	A=B				32
33	CLSA			STA 1253	33
34	A=B			JSBR 12 164B	34
35	STA		1253	Master Sector No. 1177-	35
36	JSBR	12	1640	SPLIT "OVERWRITE" Jump 1213	36
37	R=1170-			LDA 1251	37
40	JUMP		1213	No. CLSA, come-A	40
41	LDA		1251	Yes. STA 1251	41
42	CLSA/COM/SA			indicate "WRITE" Jump 1223	42
43	STA		1251	LDA 1374	43
44	JUMP		1223	STA 1574	44
45				JSBR 12 1701	45
46					46
47				Jump 1157	47
50				000000	50
51				-	51
52				→ Buffer 2000-	52
53				sector	53
54				CR R	54
55				E A	55
56				D SP	56
57				S E	57
60				C T	60
61				O R	61
62				NUL CR	62
63				S E	63
64				C T	64
65				O R	65
66				(O	66
67				C T	67
70) NUL	70
71				SP H	71
72				T SP	72
73					73
74					74
75					75
76					76
77				NUL	77

Programmer:-

OS-Pgm 1 Read File Record.

Page:- 0 Col:- 13-

Step	Instruction	Address	Comment	Octal	Step
00	JSBR	1770	Get Octal "FILE ID"		00
01	P=1500-				01
02	ANDR	Z 4752	000377		02
03	STAR	0140	STAR 1246		03
04	JORA	1374	045400 Jump 1243		04
05	STAR	1574	J500 IL 104h		05
06	JSBR	IL 1640	SPLIT "TEST?" 1571-		06
07	P=1571-				07
10	JUMP	1314	No. L0A 1574		10
11	LDA	1574	Yes L0A 1563		11
12	XORR	Z 0355	Bit 15 } Assume "No Test" option STA is 74		12
13	STAR	1574	Jump 1556		13
14	JSBR	IL 1640	GET "KEY"		14
15	P=1574-				15
16	A=0				16
17	JROR	0100			17
20	LDA	Z 0151	→ Record in Buffer		20
21	LDB	Z 0152	= Record 3 words		21
22	JSBR	1325	Display		22
23	JUMP	1314	Only next key.		23
24	L0				24
25	*ENTRY		DISPLAY Record.	← BA →	25
26	STB	1375	Word Count		26
27	JSBR	IL 1605	Address → ASCII		27
30	P=1520-				30
31	JSBR	IL 1652	GET "BUFFER."		31
32	P=1514-				32
33	LDA	Z 0215	"NUL CR"		33
34	STAR	0400			34
35	STAR	0441			35
36	LDA	Z 0001	→ Extract 1/100		36
37	STAR	1377	← Current Word		37
40	JSBR	IL 1731	Space File	↓ NEXT LINE	40
41	P=0401-				41
42	P=32 words				42
43	LDA	Z 0210	CF8		43
44	STAR	1376	line counter		44
45	LDA	1341	→ buffer		45
46	STAR	1351			46
47	LDA	I 1377	= Next Word	* NEXT Word	47
50	JSBR	IL 1612	Octal → ASCII		50
51	P=				51
52	LDA	1351			52
53	LDA	Z 0204			53
54	STAR	1351			54
55	INSZ	1377	← Current Word		55
56	DESZ	1375	Word Count		56
57	JUMP	1363			57
60	JSBR	IL 1652	PUT last line		60
61	P=0400-				61
62	JUMP	I 1325	Return. →		62
63	DESZ	1376	Word Count		63
64	JUMP	1347	Close next word		64
65	JSBR	IL 1652	PUT Line		65
66	P=0400-				66
67	JUMP	1340	close next line		67
70					70
71					71
72					72
73					73
74				WASH 045400	74
75			Word Count		75
76			Line Count		76
77			→ Current Word		77

OS-1

Step	Instruction	Address	Comment	Octal	Step
00	JSR	1652	PUT "RESTRICT TASK" * R		00
01	Pi=1525½-				01
02	JSR	1652	PUT "BT OWN RCBY"		02
03	Pi=3106½-				03
04	NOOP				04
05	JSR	1634	Specify Escape Point		05
06	Pi=011402				06
07	JSR	1770	Get Octal "TASK NO"		07
10	Pi=1534½-				10
11	CMPA	Z 0051	Task No.		11
12	NOOP				12
13	SKNCT				13
14	JUMP	I2 1641	Exit		14
15	APOS				15
16	JUMP	I2 1641			16
17	AND				17
20	JUMP	I2 1641			20
21	STA	1477	TASK NO.		21
22	JSR	1770	Get Octal "ADDRESS"		22
23	Pi=1541½-				23
24	LDB	1477	Task No.		24
25	ADB	Z 0047	+Task Control Table origin		25
26	LDB	I2 B	→ TCA		26
27	ADB	Z 0252	→ default restart address		27
30	AND				30
31	LDA	I2 B	Use default restart address		31
32	SFB	Z 0206	→ Escape Point address		32
33	AND				33
34	LDA	I2 B			34
35	NOOP				35
36	SFB	Z 0256	→ Base Address.		36
37	APOS				37
40	ADA	I2 B			40
41	ADB	Z 0214			41
42	STB	Z 0177	→ Restart Word		42
43	LDB	I2 0177	= Restart Word		43
44	CMPB	Z 0376	37777		44
45	JUMP	1560	Task is halted.		45
46	STA	Z 0176	= Restart Address * Sample Data Q.		46
47	LDA	1477	Task No.		47
50	LDB	Z 0052	Dir Control Table origin		50
51	ADB	Z 0267	→ Unallocated Data Q		51
52	STB	Z 0175	Save addresses		52
53	LDB	I2 B			53
54	B=0		End of chain!		54
55	JUMP	1461	No.		55
56	JSR	I2 1652	Put "NOT HALTED!"		56
57	Pi=1547-				57
60	JUMP	1407	try again		60
61	INCB				61
62	CMPA	I2 B	this Task!		62
63	JUMP	1466	Yes - remove from Q and stack restart.		63
64	DECB				64
65	JUMP	1452	try next entry in Q.		65
66	STA	Z 0177	Task No.		66
67	DECB		→ Vector		67
70	LDA	I2 B	→ Next Vector		70
71	STA	I2 0175			71
72	LDA	Z 0054	→ 1st free Q element		72
73	STA	I2 B			73
74	STB	Z 0054			74
75	JSR	I2 1616	STACK restart address		75
76	JUMP	Z 1402	to PROXIM?		76
77			TASK NO.		77

Programmer:-

05-19-1.

Page:- Col:- 15-

Step	Instruction	Address	Comment	Octal	Step
00					00
01			CR F		01
02			I L		02
03			E SP		03
04			I D		04
05			SP MUL		05
06			SP SP		06
07			T E		07
10			S T		10
11			? MUL		11
12			CR H		12
13			E Y		13
14			SP MUL		14
15			SP SP		15
16			B Y		16
17			F F		17
20			E R		20
21			SP		21
22					22
23					23
24					24
25			MUL CR		25
26			R E		26
27			S T		27
30			A R		30
31			T SP		31
32			T A		32
33			S H		33
34			MUL CR		34
35			T A		35
36			S H		36
37			SP N		37
40			O SP		40
41			MUL SP		41
42			SP A		42
43			D D		43
44			R E		44
45			S S		45
46			SP MUL		46
47			SP SP		47
50			REN N		50
51			O T		51
52			SP H		52
53			A L		53
54			T E		54
55			D !		55
56			MUL		56
57			0/1402		57
60	STA	IZ	0177		60
61	JUMP	Z	1402	to "Proc. Adv."	61
62					62
63				20377	63
64				(6+9) LAST REC no.	64
65				14313b	65
66				22b-	66
67				113171-	67
70				113171-	70
71			SILVER "TEST"	300000	71
72				1505-	72
73				Po 1300-	73
74					74
75			GET FE+cm Key	1511-	75
76				-No 3640-	76
77				2000-	77

Programmer:-

OS-1 part 1

Page:- Col:- 16-

Step	Instruction	Address	Comment	Octal	Step
00	LDA	1274		300000	00
01	Jump	1045			01
02			Split "INPUT"?	300000	02
03					03
04				220004	04
05			Get "NAME"	0737-	05
06				1674-	06
07				001400	07
10			Split "New Director?"	300400	10
11				0716½-	11
12				040100	12
13				1010-	13
14			GET "SECTORS" (P10)	1677-	14
15			Min CFI 0/0201		15
16			Max CFB 0/0215		16
17			Split "NO-RESERVE?"	300000	17
20				1030½-	20
21				040100	21
22			GET "SECTORS" (20)	1010-	22
23				1173-	23
24			Min CFI 0/0201		24
25			Max 50000 0/0357		25
26				CR S	26
27				T A	27
30				R T	30
31				SP S	31
32				E C	32
33				T O	33
34				R (34
35				O C	35
36				T A	36
37				L)	37
40				L SP NULL	40
41				CR S	41
42				E C	42
43				T O	43
44				R SP	44
45	BDDA	1362		-	45
46	STA	106		-	46
47	Jump	1314		-	47
50				? NULL	50
51				CR L	51
52				O A	52
53				D S	53
54				SP A	54
55				T SP	55
56					56
57					57
60					60
61					61
62					62
63					63
64			NSect "02"		64
65			Core Address		65
66			Start Sector No		66
67			Mode		67
70				000000	70
71			Empty entry	000000	71
72				000000	72
73				000000	73
74			Name		74
75					75
76			Method No.		76
77			No. of Sectors Entry #.		77

Programmer:-

OS - Page 1

Page:- Col:- 17-

Step	Instruction	Address	Comment	Octal	Step
00	*ENTRY				00
01	LDA	Z 0066	→ Input Buffer	← BA →	01
02	ADA	Z 0066	x2		02
03	STA	Z 0177	Source x2		03
04	CHA				04
05	STA	Z 0176	Octal Word		05
06	STA	Z 0175	"/" indicator		06
07	LDB	Z 0177	Source x2		07
10	INSZ	Z 0177	*NEXT CHAR.		10
11	JUMP	1715			11
12					12
13					13
14					14
15	JSBR	I2 1415	Load Absolute Byte.		15
16	A = φ				16
17	JUMP	1722			17
20	LDA	Z 0176	= Octal Word		20
21	JUMP	I 1700	Return.		21
22	CMPA	Z 0257	"NUL 1"		22
23	JUMP	1755			23
24	LDB	Z 0176	Octal Word So far		24
25	SKGT				25
26	JUMP	1760	Not octal - test for "-"		26
27	CMPA	Z 0267	"NUL 7"		27
30	NOOP				30
31	SKNCT				31
32	JUMP	I2 1641	Not octal - error		32
33	SFA	Z 0260			33
34	DESZ	Z 0175	"/" input previously?		34
35	JUMP	1743	No.		35
36	AND/CLC				36
37	JUMP	1745	φ input		37
40	CMPA	Z 0201	CFI		40
41	JUMP	1745	1 input		41
42	JUMP	I2 1641	Not φ or 1 - error		42
43	CAC/LSB				43
44	LSB				44
45	BRS/SKNC				45
46	JUMP	I2 1641	Overflow: Error.		46
47	LSB				47
50	ADB	Z A			50
51	SKNC				51
52	CXSB/COMSB				52
53	STB	Z 0176	Octal Word		53
54	JUMP	1707	Out of range character		54
55	LDA	Z 0201	to "/" input		55
56	STA	Z 0175	Set indicator		56
57	JUMP	1707	Out of range char.		57
60	CMPA	Z 0255	"NUL -"		60
61	CXSB/COMSB/STB				61
62	JUMP	I2 1641	Error - not octal		62
63	STB	Z 0176	Octal Word.		63
64	JUMP	1720			64
65					65
66			GET Octal Input	{ 200020	66
67					67
70	*ENTRY		Get Octal input	← BA →	70
71	LDA	I 1770			71
72	INSZ	1770			72
73	STA	1767			73
74	JSBR	I2 1640	GET		74
75	K=1766-				75
76	JSBR	1700	Octal → Binary		76
77	JUMP	I 1770	Return.		77

Programmer:-

OS - Part 1

Page:- 1 Col:- 00- 20-

Step	Instruction	Address	Comment	Octal	Step
00	JSBR	0100	Initialise		00
01	JUMP	1200	*"OP" Entry		01
02	JSBR	I2 1635	GET PASSWORD		02
03	Pi=011324		*"HALT" Entry		03
04	HALT	I1			04
05	JUMP	0002			05
06	LDA	167			06
07	STA	11			07
10	JSBR	I2 17bb			10
11		I1 16			11
12	STA	17b	MIN		12
13	INCB				13
14	LDA	I2 B	MAX		14
15	STA	171			15
16	INCB				16
17	LDA	I2 B			17
20	STA	45	RBL LENGTH		20
21	NOOP				21
22	JSBR	I2 164D			22
23	GOB -				23
24	AN=b				24
25	JUMP	37			25
26	COMPA	2 3b3	"a"		26
27	Jump	33			27
30	Jump	I2 1641			30
31	JSBR	I2 164b			31
32	GOB -				32
33	LDA	11			33
34	ANDA	2 1752			34
35	SEPA				35
36	STA	4b			36
37	JGCL	I2 167b	FETCH + LOCIL		37
40	FE				40
41	11317b -				41
42	13 34bb -				42
43	JSBR	I2 171b			43
44	P1 - 34bb -				44
45					45
46	JSBR	I2 1671			46
47	LDA	17b			47
50	COMPA	171			50
51	Jump	54			51
52	INCB	17b			52
53	Jump	37			53
54	LDA	1752			54
55	Jump	1735			55
56	JSBR	I2 1721	LDA Febltable		56
57	Pi=3775 -				57
60	APCS				60
61	INCA				61
62	LDA				62
63	ANDA	2 0202			63
64	ANDA	2 0014	→ System Table		64
65	LDA	I2 .A	= System III		65
66	JSBR	I2 1725	STA		66
67	Pi=0000 -				67
70	JUMP	I 0100	Return		70
71			"JUMP 0045"	020045	71
72			Key	0	72
73				0	73
74				0	74
75			Data	0	75
76				0	76
77				0	77

Programmer:-

OS - Page 1

Page:- 01 Col:- 01- 21-

Step	Instruction	Address	Comment	Octal	Step
00	*ENTRY		INITIALISE	← BA →	00
01	JSR	I2 1627	Resolve Object Addresses		01
02	P ₁ = S.1/0760				02
03	JSR	I2 1627	" " " "		03
04	P ₁ = S.0/0100				04
05	LDA	Z 0201			05
06	P ₁ = 0 JSR	I2 1626	Indirect?		06
07	JSR	I2 1707	Duplicate		07
10	P ₁ = 2700-				10
11	P ₁ = 3200-				11
12	P ₂ = 3200 words			000500	12
13	LDA	0766			13
14	STA	1342			14
15	JSR	I2 1643	Specify default start address		15
16	P ₁ = 3200-				16
17	LDA	Z 0047	→ Task Control Table		17
20	ADA	Z 0040	+ Control words no.		20
21	LDA	I2 A	→ Control Block, then words		21
22	LDA	I2 A	= Device Code, control words		22
23	STA	Z 0177	Save		23
24	ANDA	Z 1752	Bottom Byte (Leaves out device code)		24
25	JORA	0703			25
26	STA	1515			26
27	LDA	Z 0177			27
30	SWAP				30
31	ANDA	Z 1752	Bottom Byte (Leaves input device code)		31
32	STA	Z 0176	Save.		32
33	ADA	0704	} Setup queue etc.		33
34	STA	1505			34
35	ADA	0705			35
36	STA	1604			36
37	ADA	0706			37
40	STA	1511			40
41	ADA	0707			41
42	STA	1506		42	
43	LDA	Z 0176	= Input Device Code		43
44	ADA	Z 0024	2/1000		44
45	STA	1677	Set up Completion Address		45
46	LDA	I 1677	= Old Service Pointer	} Swap service pointers	46
47	LDA	0763	= New " " "		47
50	STA	I 1677			50
51	STB	1673			51
52	LDA	Z 0176	= Input Device Code		52
53	CMPA	Z 0227			53
54	NOOP				54
55	SKNLT				55
56	JUMP	I 0166			56
57	CXA		+ANK.		57
60	STA	1471	} eliminate decode.		60
61	STA	1472			61
62	JSR	I2 1707	Duplicate received parameter set		62
63	P ₁ = 2177-				63
64	P ₂ = 3344-				64
65	P ₃ = 6666				65
66	JUMP	0056			66
67					67
70					70
71					71
72					72
73					73
74					74
75					75
76					76
77					77

Programmer:-

OS - Pgm 1

Page:- S.1 Col:- 02 → 06

AS for S.1/P2 → 16

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 1

Page:- 31 Col:- 07 27-

Step	Instruction	Address	Comment	Octal	Step
00			Escape to "PROGRAMS"	01400 P6	00
01				200407	01
02			GET "PASSWORD"	302751	02
03				015400	03
04				010200	04
05			MASKS f. of code.	000600	05
06				002000	06
07				003100	07
10				Pa 3004-	10
11			GET "TEXT"	200577	11
12				2722 1/2-	12
13					13
14					14
15					15
16					16
17					17
20					20
21					21
22					22
23				CR	23
24				T E	24
25				X T	25
26				SP NUL	26
27				BEL SP	27
30				N O	30
31				SP S	31
32				P A	32
33				C E	33
34				NUL CR	34
35				S E	35
36				W D	36
37				SP NUL	37
40				E S	40
41				S A	41
42				G E	42
43				NUL CR	43
44				T A	44
45				S K	45
46				SP NUL	46
47				NUL S	47
50				NUL P	50
51				NUL CR	51
52				P A	52
53				S S	53
54				W O	54
55				R D	55
56				! NUL	56
57				! NUL	57
60			→ Overlay Start	00000-	60
61			→ "OP!" Code	3200-	61
62			→ DECODE	2600-	62
63			→ Keyboard Service Rtn.	3504-	63
64			→ Key Area for Processing	2072-	64
65			→ Data Area	2074-	65
66			→ Breakpoint Handler Entry point	0400-	66
67					67
70					70
71					71
72					72
73					73
74					74
75					75
76			→ Octal "GET" Rtn.	1770-	76
77					77

Programmer:-

OS-1

Step	Instruction	Address	Comment	Octal	Step	
00	JSBR	IL 1652	P4T "Send Message"		00	
01	P=2733-				01	
02	JSBR	IL 1627	Reverse Off of Block.		02	
03	P=2776-				03	
04	JSBR	I 0776	Get offset "TASK"		04	
05	P=2742-				05	
06	APOS				06	
07	JUMP	IL 1641	End		07	
10	CHPA	Z 0055	No. of I/O Status.		10	
11	NOOP				11	
12	SHNGT				12	
13	JUMP	IL 1641	End		13	
14	STA	1074			14	
15	JSBR	IL 1640	GET "TEXT"		15	
16	P=2714-				16	
17	LDA	Z 0066	→ Target Buffer		17	
20	ADA	Z 0066	x2		20	
21	ADA	Z 0045	No. of chars input.		21	
22	STA	Z 0177	Format X2.		22	
23	LDA	Z 0217	"NUL SI"		23	
24	JSBR	1100	} The format ID.		24	
25	LDA	Z 0240		"NUL SP"		25
26	JSBR	1100				26
27	CLA			"NUL NUL"		27
30	JSBR	1100			30	
31	LDA	Z 0040	Task No.		31	
32	JSBR	IL 1612	Get → DSPJT		32	
33	P=3070-				33	
34	JSBR	IL 1707	Default		34	
35	P=3064-				35	
36	P=3570-				36	
37	P=8conds.				37	
40	LDA	1074	Target Task No.		40	
41	JSBR	IL 1654	Fetch Single Task		41	
42	P=3570-				42	
43	JUMP	1004	auto vert		43	
44					44	
45					45	
46					46	
47					47	
50					50	
51					51	
52				1137 -	52	
53					53	
54					54	
55					55	
56					56	
57					57	
60					60	
61					61	
62					62	
63					63	
64				CR BEL	64	
65				SO F	65	
66				R O	66	
67				M SP	67	
70					70	
71				Task No. {	71	
72					72	
73				: SP	73	
74				TASK NO.	74	
75	JSBR	IL 1634	Specify escape point. +SEND-		75	
76	P=011402				76	
77	JUMP	1000			77	

Programmer:-

OS - Page 1

Page:- S.1 Col:- 11 31-

Step	Instruction	Address	Comment	Octal	Step
00	XEWKLY			2-15A	00
01	LDB	Z 0177	Target xL		01
02	INSZ	Z 0177			02
03	JSBR	I2 1416	Store Address Byte		03
04	JRMP	I 1100	Return		04
05					05
06					06
07					07
10					10
11					11
12					12
13					13
14					14
15					15
16			Max Check Digit { 999999999	035632	16
17				144777	17
20	CHA/COMPST		* Check Digit List		20
21	JSBR	I2 1613	Specify Point Q		21
22	JSBR	I2 1652	Get "Check Digit List"		22
23	Pi = 3140-				23
24	JSBR	I2 1640	GET "FIRST No."		24
25	Pi = 3173-				25
26	JSBR	I2 1640	GET "LAST No."		26
27	Pi = 3165-				27
30	NOOP				30
31	NOOP				31
32	JSBR	I2 1651	SPOOL & POST		32
33	JUMP				33
34					34
35	JSBR	E 1630			35
36	Sum	I2 A			36
37				1363 -	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				040600	66
67			Get "LAST No"	3156i-	67
70				3403-	70
71				3401-	71
72				3116-	72
73				011400 P0	73
74				040600	74
75			Get "FIRST No"	3151-	75
76				3401-	76
77				Min 0/0200	77
				Max 3116-	

Programmer:-

OS - Program

Page:- 5.1 Col:- 12 32-

Step	Instruction	Address	Comment	Octal	Step
00	COXA		GET "01"		
01	JUMP	1365	* OP entry		00
02	ROOP				01
03	CMPB	1343	"NUL S" (from 1337)		02
04	JUMP	1260	breakpoint handler		03
05	CMPB	Z 0304	"NUL F"		04
06	JUMP	1264	File - Control Block		05
07	CMPB	1662	"NUL R"		06
10	JUMP	1474	Read Program		07
11	CMPB	1661	"NUL W"		10
12	JUMP	1240	Write Program		11
13	JUMP	1272	"NUL F"		12
14					13
15					14
16					15
17					16
20					17
21					20
22					21
23					22
24	*ENTRY		GET Octal Input		23
25	LDA	I 1224	EP		24
26	STA	1457	→ ASCII prompt		25
27	INPZ	1224			26
30	CHA				27
31	STA	1672	Clear Octal Word		30
32	STA	1674	Indicate Octal Word		31
33	JSBR	I2 1640	GET Octal Input		32
34	P1=111456-				33
35	LDA	1672	Octal Word Input		34
36	LDB	1666	Termination Character		35
37	JUMP	I 1224	status		36
40	CMPA	1664	= Overlay No. Read? *WRITE OVERLAY		37
41	STIP				40
42	JUMP	1272	Error		41
43	JSBR	I2 1670	STOP Overlay		42
44	P1=load, Write, Priority, "02"				43
45	P2=S.111664		→ Program No.		44
46	LDA	Z 0146	? Same Year Month total.		45
47	STA	1672	S		46
50	JSBR	I2 1670	FETCH & LOCK Overlay Indec		47
51	P1=load, Priority, "01"				50
52	P2=S.111664		→ Program No.		51
53	P2=0/0170		Overlay (Queue)		52
54	LDA	1672	? Target need month total.		53
55	STA	Z 0173	S		54
56	JSBR	I2 1676	REWRITE Overlay Indec		55
57	JUMP	1301			56
60	JSBR	I2 1670	FETCH Break Point Handler		57
61	K=000002				60
62	K=0/0202				61
63	JUMP	I 1342	0400-		62
64	STA	I 1266			63
65	JSBR	I2 1700	Fetch from FCBS * FILE CONTROL Block		64
66	P1=				65
67	LDA	Z 1B			66
70	JUMP	1300			67
71					70
72	CHA/COMPA				71
73	STA	1675	Indicate Error Mode *ERROR		72
74	JUMP	I2 1641	Error		73
75	LDA	1673			74
76	STA	I 1677	Return Original Service routine. *RETURN TO PROGRAM!		75
77	JUMP	Z 1402	to "PROCLIF?"		76
					77

Programmer:-

OS - Paper 1

Page:- S.1 Col:-13 33-

Step	Instruction	Address	Comment	Octal	Step
00	STA	1665	Set Location Pointer * Amend Instruction	1665	00
01	JSBR	1460	Set up ASCII * Get Octal Target	031460	01
02	JSBR	1224	GET Octal Target	001224	02
03	P=1/1642-	1227		001227	03
04	STUMP	1311		001311	04
05	LDA	1341		001341	05
06	JSBR				06
07	JUMP	1000		001000	07
10	JUMP	1275		001275	10
11	CMPB Z	0233	"NUL ESC"	000233	11
12	JUMP	1345	"Escape to 'PROGRAM'"	001345	12
13	CMPB Z	0303	"NUL A"	000303	13
14	JUMP	1300	Reset Address Pointer	001300	14
15	CMPB Z	0351	"NUL GS" (Enter)	000351	15
16	JUMP	1357	Amend control.	001357	16
17	CMPB Z	0212	"NUL LF"	000212	17
20	JUMP	1357	Amend control.	001357	20
21	CMPB Z	0231	"NUL EM" (Reset)	000231	21
22	JUMP	1301	Re-output	001301	22
23	CMPB Z	1347	"NUL NHT" (->)	001347	23
24	JUMP	1403	Forward Space	001403	24
25	CMPB Z	1346	"NUL BS" (-)	001346	25
26	JUMP	1413	Back Space	001413	26
27	CMPB Z	1345	"NUL SH" (↑)	001345	27
30	JUMP	1423	Literal Mode	001423	30
31	CMPB Z	0240	"NUL SP"	000240	31
32	JUMP	1434	Text output	001434	32
33	CMPB Z	1647	"NUL I"	001647	33
34	JUMP	1442	Set Address Pointer Indirectly	001442	34
35	CMPB Z	1370	"NUL O"	001370	35
36	JUMP	1365	Offset Address Pointer	001365	36
37	JUMP	1203		001203	37
40					40
41			Breakpoint in db		41
42			breakpoint table		42
43					43
44			NUL S		44
45			NUL ETX	000003	45
46			NUL SUB	000032	46
47			NUL BS	000010	47
50			NUL NHT	000025	50
51			NUL FS	000034	51
52			NUL GS	000035	52
52	CMPA Z	0256	"NUL ."	000256	52
53	SKIP		(Scan 1624)		53
54	JUMP	1611	Not octal	001611	54
55	ADB Z	0356	Bit 16	000356	55
56	JUMP	1603		001603	56
57	STA I	1665	+AMEND CORE	001665	57
60	JSBR	1460	Setup ASCII	001460	60
61	INSZ	1665		001665	61
62	JSBR IZ	1652	PAT Read ASCII	001652	62
63	P=S.1/1646 1/2				63
64	JUMP	1301		001301	64
65	CXSA/COMPX	1614	OFFSET POINTER	001614	65
66	JSBR Z	1630	Resolve Offset	001630	66
67	JUMP	1300	Reset Address Pointer	001300	67
70			NUL O		70
71	STA	1664	Page No * READ OVERLAY	001664	71
72	JSBR IZ	1670	FETCH Overlay	001670	72
73	P=Text, "02"				73
74	P2=S.1/1664		→ Program No:		74
75	JUMP IZ	1641	-Error - not found	001641	75
76	LDA Z	0144	= Start Core Address (Plasplate)	000144	76
77	JUMP	1300	Reset Address Pointer	001300	77

Programmer:-

OS - Page 1

Page:- S.1 Col:- 14 34-

Step	Instruction	Address	Comment	Octal	Step
00	AND	1665	(Jan 1477)	16 1665	00
01	JUMP	I2 1644	Exit	04 1665	01
02	JUMP	1371	Octal	02 1301	02
03	AND		*FORWARDS	007500	03
04	INSZ	1672	Counter=1	04 1672	04
05	SKIP			004020	05
06	JSBR	1447	Display	03 1447	06
07	INSZ	1665	MA	04 1665	07
10	DESZ	1672	Counter	05 1672	10
11	JUMP	1406		02 1406	11
12	JUMP	1301		02 1301	12
13	AND		*BACKWARDS	007500	13
14	INSZ	1672	Counter=1	04 1672	14
15	SKIP			004020	15
16	JSBR	1447	Display	03 1447	16
17	DESZ	1665	MA	05 1665	17
20	DESZ	1672	Counter	05 1672	20
21	JUMP	1416		02 1416	21
22	JUMP	1301		02 1301	22
23	AND		*LITERAL	007500	23
24	LDA	1665	MA	21 1665	24
25	LRA			003240	25
26	STA	1674	=In x2	25 1674	26
27	JSBR	I2 1640	GET Label	037640	27
30	P=S.1/1454			A	30
31	LDA	1674	In x2	21 1674	31
32	RSA		(ignore odd byte)	003100	32
33	JUMP	1300		02 1300	33
34	AND		*TEXT	007500	34
35	LDA	1665	MA	21 1665	35
36	STA	1440		25 1440	36
37	JSBR	I2 1652	Put Text	037652	37
40	P=0			-	40
41	JUMP	1301	Re-output octal.	02 1301	41
42	AND		*INDIRECT	007500	42
43	LDA	1665	MA	21 1665	43
44	LDA	I2 A		21 6000	44
45	JUMP	1300	Reset MA	02 1300	45
46					46
47	*ENTRY		Put MA + Content	← BA →	47
50	JSBR	1460		03 1460	50
51	JSBR	I2 1652	Put	037652	51
52	P=S.1/1642			A	52
53	JUMP	I 1447		02 5447	53
54			GET Label	200000	54
55				S.1/1633	55
56			GET Octal	200000	56
57				S.1/1633	57
60	*ENTRY		Setup ASCII	← BA →	60
61	LDA	1665	Location Printer	1665	61
62	JSBR	I2 1605	Address → ASCII	1605	62
63	P=S.1/1642				63
64	LDA	I 1665	Location Contents	21 5665	64
65	STA	1446	(Save to avoid double "Printer" error)	25 1446	65
66	JSBR	I2 1612	Octal → ASCII	037612	66
67	P=S.1/1647				67
70	LDA	1446		21 1446	70
71	JSBR	I2 1604	DECODE	037604	71
72	P=S.1/1652				72
73	JUMP	I 1460		02 5460	73
74	LDB	1341			74
75	B=0		*READ octal		75
76	JSBR	1015			76
77	JUMP	1400			77

Programmer:-

OS - Pgm 1.

Page:- S.1 Col:- 15 35-

Step	Instruction	Address	Comment	Octal	Step
00			FLASHPOINT DEVICE CODE		00
01			In x2		01
02			Count		02
03			Status / Actual EOF		03
04	*ENTRY		INPUT SERVICE Rtn.		04
05	DAT123		(0102H)		05
06	DAT1A1STOP		(0161dd)		06
07	3110				07
10	JUMP	1515	OK	006500	10
11	IOPAS			021515	11
12	SWAPB/CASB/COM/SB		*STATUS (0130dd)	004016	12
13	STB	1666	Termination character	261666	13
14	JUMP	1612	Complete	021612	14
15	DAT1A1START		FLASHPOINT (0154dd)	001612	15
16	JSBR	1772	Convert to error code		16
17	CMPB	1675	Error Mode?	211675	17
20	JUMP	1527	No.	021527	20
21	CMPA	1344	Cancel EOF?		21
22	SHIP			004020	22
23	JUMP	1612	No-output "ERROR" (not acknowledged)	021612	23
24	STB	1675	Clear "Error Mode" indicator	261675	24
25	STB	1672	Clear Octal Mode	261672	25
26	JUMP	1620	Complete (re-output present)	021620	26
27	CMPB	1674	Octal Mode?	241674	27
30	JUMP	1550	Yes	021550	30
31	CMPA	0233	"NL ESC" *LITERAL MODE		31
32	JUMP	1612	Complete	021612	32
33	LDB	1674	In x2	221674	33
34	CACIRSB			002500	34
35	SKC			007440	35
36	SWAPA			005010	36
37	STH	1672	Complete	251672	37
40	LDA	1752	Bottom bit	213752	40
41	SKNC			007040	41
42	SWAPA			005010	42
43	ANDH	12 B		066001	43
44	JORH	1672		071672	44
45	STH	12 B		256001	45
46	JNSZ	1674	In x2	041674	46
47	JUMP	1604	Start	021604	47
50	CMPA	1344	Cancel EOF? *OCTAL MODE	221571	50
51	JUMP	1525	Yes - re-output present	021525	51
52	CMPA	0257	"NL /"	232257	52
53	JUMP	1606		021606	53
54	LDB	1672	Octal card so far.	221672	54
55	SHGT			006410	55
56	JUMP	1622		021622	56
57	CMPA	0267	"NL 7"	232267	57
60	NOOP			000000	60
61	SHNGT			006010	61
62	JUMP	1611	Not Octal	021611	62
63	SFA	0260	"NL 0"	132260	63
64	DESZ	1667	"1" input processed?	051667	64
65	JUMP	1573	No.	021573	65
66	AND/CAC			007520	66
67	JUMP	1575	0 input	021575	67
70	CMPA	0201	CFI	232201	70
71	JUMP	1575	1 input	021575	71
72	JUMP	1611	Not 0 & 1	021611	72
73	CAC/ASB			002700	73
74	LSB			002300	74
75	BPOS/SKNC			006240	75
76	JUMP	1512	Overflow! (force 0)	021512	76
77	LSB			002300	77

Programmer:-

OS - Pgm 1.

Page:- 511 Col:- 16- 36-

Step	Instruction	Address	Comment	Octal	Step
00	ADB	Z A		122000	00
01	SYNC			000000	01
02	CHSB/COMPSE				02
03	STB	1672	Octal Word.		03
04	START			(0110dd)	04
05	JUMP	IZ 0010	Divides (input in progress)		05
06	LDA	Z 0201	CFI	"/" INPUT	06
07	STA	1667	Set Indicator		07
10	JUMP	1604	Start.		10
11	STA	1666	Terminating character *NOT OCTAL		11
12	LDA	Z 0202	"NUL STX" (EOF Recount) *COMPLETE		12
13	STA	1503	"Actual" EOF for Supervisor to detect.		13
14	CHA				14
15	STA	1501	Indicates Completion		15
16	JUMP	IZ 0010	Divides (Completed)		16
17					17
20	LDA	Z 0203	"NUL ETX" (EOF Reject)		20
21	JUMP	1613			21
22	CMPA	Z 1350	"NUL FS" Error?		22
23	CHSB/COMPSE/STSR				23
24	JUMP	1352	Test for "." (2 nd byte)		24
25	STB	1672	Octal Word.		25
26	LDA	Z 1351	"NUL GS" (Error +)		26
27	JUMP	1611	Not octal		27
30					30
31					31
32			SP NUL		32
33			LF		33
34			L I		34
35			T :		35
36			NUL		36
37					37
40					40
41			NUL I	000111	41
42			CR		42
43			CR		43
44			Address (ASCII)		44
45					45
46					46
47					47
50			Octal (ASCII)		50
51					51
52			SP NUL		52
53					53
54					54
55			handle		55
56					56
57					57
60			SI - SET		60
61			NUL W		61
62			NUL R		62
63			"Preceded Write" Mode	100300	63
64			Program No. last Read.		64
65			LOCATION POINTER [MEMORY ADDRESS]		65
66			Terminating Character		66
67			"/" Mode Indicator		67
70					70
71					71
72			OCTAL WORD / Octal word		72
73			→ Hardware Error Bit. = B5041		73
74			Octal Mode Indicator (001 & 011)		74
75			Error Mode Indicator (IF 0)		75
76					76
77			Service Pointer, Jump to error current mode		77

Programmer:-