SINESS COMPUTERS LTD. **MOLECULAR 18**

13 sheets

Sheet:-

Date:-

OS- Pgm 1

Page:- Col:- 00

Step	Instruction	Address		Comment	Octal	Step
00				→ Module No. (LIST) Spool Buffer 3571-		00
01				→ "Read" Extract Area 2000-		01
02				→ Key Area ("HEAD") 3640-		02
03	Offset Addresses			11000½-		03
04						04
05						05
06						06
07						07
10	CLA/COMPBA			(Key Plain Paper Q) +LIST		10
11	JSBR	I2	1613	Specify I/O Station Print Q		11
12	JSBR	I2	1652	PUT "Module / Core Dump Print"		12
13	P_1= 0160-					13
14	JSBR	I2	1634	Specify Escape Point		14
15	P_1= 011402					15
16	JSBR	I2	1770	GET OCTAL "MODULE"		16
17	P_1= 0273½ -					17
20	CMPA	Z	0344	004000		20
21	NOOP					21
22	SKNGT					22
23	JUMP	I2	1641	Error - incorrect module No-		23
24	STA	I	0077	= Module No.		24
25	STA	I	0000	Init Spool Buffer		25
26	ANØ					26
27	JUMP		0050	Core Dump between limits		27
30	JSBR	I2	1612	Octal → ASCII Analog No-		30
31	P_1= 0214½ -					31
32	JSBR	I2	1741	Move & Pad (Heading → Spool Buffer)		32
33	P_1= 0200 -					33
34	P_2= 3500 -					34
35	P_3= 60_8 chars					35
36	JSBR	I2	1640	GET "TITLE"		36
37	P_1= 0074-					37
40	JSBR	I2	1651	Spool & POST..		40
41	LDA	I	0077	= Module No. last time		41
42	ANØ					42
43	JUMP		0025	Onto next core dump		43
44	JUMP		0016	Onto next Module?		44
45						45
46						46
47						47
50	JSBR		1770	Get Octal "CORE FROM" & Core Dump		50
51	P_1= 0262½-					51
52	JSBR	I2	1725	STA Low Address		52
53	P_1=3572-					53
54	JSBR	I2	1605	Address → ASCII		54
55	P_1= 0244½-					55
56	JSBR	I2	1770	Get Octal "To"		56
57	P_1 =0270½-					57
60	JSBR	I2	1725	STA High Address		60
61	P_1=3573-					61
62	JSBR	I2	1605	Address → ASCII		62
63	P_1=0252½-					63
64	JSBR	I2	1741	Move & Pad (Heading → Spool Buffer)		64
65	P_1= 0230-					65
66	P_2= 3500 -					66
67	P_3= 60_8 chars					67
70	JUMP		0036	Loop.		70
71						71
72						72
73				P_1 0014-		73
74					220441	74
75				Get "TITLE" 0174-		75
76				3530-		76
77				Module No.	—	77

Programmer:-

BUSINESS COMPUTERS LTD.

MOLECULAR 18

Sheet:-

Date:-

DS-1

Page:- Col:-01-

Step	Instruction		Address	Comment	Octal	Step
00	CMPA	2	0303	"NUL A"		00
01	JUMP		0107	Kneud		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	I2	1641	Exit.		06
07	JSBR		1770	Get Octal "STEP" + Address.		07
10	P_1=0152—					10
11	ADA		0001	→ Extract Word.		11
12	STA		0077			12
13	JSBR		1770	Get Octal "OCTAL"		13
14	P_1=0145—					14
15	STA		0076	Field.		15
16	LDA		0140	Field.		16
17	CLSA/COMPSA					17
20	STA		0122	[illegible]		20
21	JSBR	I2	1670	FETCH BLOCK Head.		21
22	P_1= —					22
23	P_2=3640—			[illegible]		23
24	P_3=2000—			Extract Wrd		24
25	NOOP					25
26	JUMP	I	0372			26
27	JSBR	I2	1671	REWRITE		27
30	JUMP		1320	Display & auto next.		30
31	JSBR		1770	get Octal "STEP" + Chain Link		31
32	P_1=0152—					32
33	ADA		0001	→ Extract Wrd.		33
34	STA		0077	→ Chain Link Word.		34
35	LDA	I	0077	+ Next Link		35
36	STA	I	0002			36
37	JSBR	I2	1670	FETCH		37
40	P_1= /			Field. (from 1303) (Cread by 0116)		40
41	P_2=3640—					41
42	P_4=2000—			Extract Wrd		42
43	NOOP					43
44	JUMP		1320	Display & auto next.		44
45				SP		45
46				SP O		46
47				C T		47
50				A L		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				SP 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:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS - Ryan 1

Page:- Col:- 02

Step	Instruction		Address	Comment	Octal	Step
00				R		00
01				O		01
02				S .		02
03				H		03
04						04
05				O V		05
06				E R		06
07				L A		07
10				Y		10
11				M O		11
12				D U		12
13				L E		13
14	DD					14
15	AT			Overlay Module 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				L .		30
31				O :		31
32				S :		32
33						33
34						34
35				C O		35
36				R E		36
37				D		37
40				U M		40
41				P -		41
42				F R		42
43				O M		43
44						44
45	LDA		255			45
46	SKEY					46
47	LDA		256			47
50	Jump		252	SP		50
51				T O		51
52						52
53	STA		1572			53
54	Jump		1305			54
55	STA		1372	214 -		55
56	Jump		1305	1535 - NUL		56
57				NUL		57
60						60
61						61
62				NUL CR		62
63				C O		63
64				R E		64
65				SP F		65
66				R O		66
67				M SP		67
70				NUL SP		70
71				SP T		71
72				O SP		72
73				NUL CR		73
74				M O		74
75				D U		75
76				L E		76
77				SP NUL		77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS- Page 1

Page:- Col:- 03

Step	Instruction		Address	Comment	Octal	Step
00	CLA/COMPSA			(Many Plain Paper Q) + PMAP		00
01	JSBR	I2	1612	Specify I/O Station Output Q		01
02	JSBR	I2	1652	PAT "Program & Overlay Ref."		02
03	P1= 0340½ -					03
04	JSBR	I2	1634	Specify Exceptional.		04
05	P1=0/1402					05
06	CLA					06
07	JSBR	I2	1725	STA Min.		07
10	P1= 3401 -					10
11	LDA	Z	0361	177777		11
12	JSBR	I2	1725	STA Hour		12
13	P1=3402 -					13
14	JSBR	2	1770	Get Octal "PROCESS?"		14
15	P1= 0352 -					15
16	AN∅					16
17	JUMP		0332			17
20	JSBR	I2	1725	STA Min		20
21	P1= 3401 -					21
22	JSBR	I2	1725	STA Hours.		22
23	P1= 3402 -					23
24	JSBR		1770	Get Octal "LAST"		24
25	P1= 0363 -					25
26	AN∅					26
27	JUMP		0332			27
30	JSBR	I2	1725	STA		30
31	P1=3402 -					31
32	JSBR	I2	1651	Spool & Part.		32
33	JUMP	Z	1402	to PROGRAM!		33
34						34
35						35
36						36
37						37
40				CR		40
41				P R		41
42				O G		42
43				R A		43
44				M SP		44
45				M A		45
46				P NUL		46
47				0/1402 P0		47
50				SPLIT "PROCESS?"	300400	50
51				(PMAP, ZERO) 0352 -		51
52				CR P		52
53				R O		53
54				C E		54
55				S S		55
56				? NUL		56
57				CR F		57
60				I R		60
61				S T		61
62				SP NUL		62
63				SP SP		63
64				L A		64
65				S T		65
66				SP NUL		66
67						67
70						70
71						71
72	LDA		0076	(patch from 0126)		72
73	STA	I	0077			73
74	JUMP		0127			74
75	JORA	Z	0323	Ad 9 (Interrogation arrived) patch from 0416		75
76	STA		0461			76
77	JUMP		0417			77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS-Pyle 1.

Page:- Col:- 04

Step	Instruction	Address		Comment	Octal	Step
00	JSBR	I2	1652	PUT "PROGRAM LIBRARY MAINTENANCE"		00
01	P_1= 0700-					01
02	JSBR	I2	1634	Specify Escape Point		02
03	P_1=0406-					03
04	LDA	Z	0201	Bit 1		04
05	JSBR	I2	1636	INHIBITED?		05
06	JSBR	I2	1640	SPLIT "AMEND DIRECTORY?"		06
07	P_1=1610-					07
10	JUMP		0543	No.		10
11	JSBR	I2	1640	Yes. SPLIT " "INPUT"? "		11
12	P_1=1602-					12
13	JUMP		0476	No. - Overstart.		13
14	LDA	Z	0203	Yes.		14
15	STA		0424			15
16	JUMP		0375	Patch.		16
17	LDA		0475			17
20	STA		0462			20
21	JSBR	I2	1640	GET "NAME"		21
22	P_1= 1604-					22
23	JSBR	I2	1670	FETCH block directory element		23
24	P_1= /					24
25	P_2= 1674-			→ Name		25
26	P_3= 1674-			→ Extract Area		26
27	JUMP		0500	Not found.		27
30	JUMP		0470	Match!! → Amendment.		30
31	JSBR	I2	1612	Octal → ASCII		31
32	P_1=0747-					32
33	JSBR		1770	GET "MODULE n?"		33
34	P_1= 0743-					34
35	LDB	Z	0045	No. of characters input.		35
36	BNØ					36
37	JUMP		0451	bypass - no amendment.		37
40	ANØ					40
41	JUMP		0510	Deletion request.		41
42	APOS					42
43	JUMP	I2	1641	Error - outside limit.		43
44	CMPA	Z	0344	004000		44
45	NOOP					45
46	SKNGT					46
47	JUMP	I2	1641	Error - outside limits		47
50	STA		1676	New Module No.		50
51	LDA		1677	= Entry Point		51
52	JSBR	I2	1605	Address → ASCII		52
53	P_1= 0756½ -					53
54	JSBR		1770	GET "ENTRY x ?"		54
55	P_1=0753-					55
56	A=Ø					56
57	STA		1677	New entry point.		57
60	JSBR	I2	1670	FETCH block directory element		60
61	P_1= /					61
62	P_2= 1/			→ Name		62
63	P_3= 0			No extraction		63
64	JSBR	I2	1777	HALT - name disappeared, or Directory Full.		64
65	JSBR	I2	1672	OVERWRITE'		65
66	P_1= 1674-					66
67	JUMP		0406			67
70	JSBR	I2	1652	Put "AMENDMENT" Match from 0430		70
71	P_1= 1040-					71
72	LDA		1676	Module No.		72
73	JUMP		0431			73
74				→ Nul Name 1670-		74
75				→ Name 1674-		75
76	LDA	Z	0204			76
77	JUMP		0415			77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS- Pgm 1.

Page:- Col:- 05

Step	Instruction	Address		Comment	Octal	Step
00	NOOP	I2	1652	"NEW NAME" + Add new name.		00
01	NOOP					01
02	CLA					02
03	STA		1676			03
04	STA		1677			04
05	LDA		0474	→ Null name.		05
06	STA		0462			06
07	JUMP		0472	Merge with "1 word" rtn.		07
10	LDA		0461	+ Delete directory entry.		10
11	STA		0513			11
12	JSBR	I2	1670	FETCH BLOCK element		12
13	P_1 = /			Filed.		13
14	P_2 = 1674 -			→ Name		14
15	P_3 = 0			No extraction		15
16	JSBR	I2	1777	HALT - name disappeared.		16
17	JSBR	I2	1672	OVERWRITE (clear entry)		17
20	P_1 = 1670 -					20
21	JSBR	I2	1741	Move Code (name)		21
22	P_1 = 1674 -					22
23	P_2 = 0773 -					23
24	P_3 = Adeap					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	Onto next question.		40
41						41
42						42
43	JSBR		1770	Yes. Get "MODULE"		43
44	P_1 = 0273½ -					44
45	ANO					45
46	JUMP	I2	1641	End		46
47	CMPA	Z	0344	004000		47
50	NOOP					50
51	SKNGT					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 -			[illegible]		57
60	LDA	I2	0151	<1st word of record.		60
61	ANO					61
62	JUMP		0566	Not loaded		62
63	JSBR	I2	1652	PUT "AMENDMENT"		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 = 1021½ -					75
76	STA		1665			76
77	JUMP		0606			77

Programmer:-

BUSINESS COMPUTERS LTD.

MOLECULAR 18

Sheet:-

Date:-

OS- Pgml

Page:- Col:- 06

Step	Instruction	Address		Comment	Octal	Step
00					154166	00
01				(G+D) FIRST BSC NO.	1661. —	01
02					113176 —	02
03				MIN	113175 —	03
04				MAX	113171 —	04
05						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 File		11
12	LDA	I	0642	No. of sectors in use		12
13	STA		1666	Start sector 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	SKNGT					17
20	JUMP		0632	File Full		20
21	JUMP		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	OLFRWRITE		26
27	P_1 = 1664 –					27
30	NOOP					30
31	JUMP		0406	Onto next operation		31
32	JSBR	I2	1652	PUT "NO SPACE"		32
33	P_1 = 1047 –					33
34	JUMP		0406			34
35	STA	I	0642	No. of sectors now in use patch from 0621		35
36	INSZ		1666	Start sector No.		36
37	JSBR	I2	Ø667	Store control Record		37
40	JUMP		0622			40
41						41
42				→ No. of Sectors In Use Count		42
43				MASK	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 –					50
51	LDB	Z	0045	No. of chars input		51
52	BNØ					52
53	JUMP		Ø657	bypass – no deletion or amendment		53
54	ANØ					54
55	JUMP		0530	Delete Module		55
56	STA		1666	New Start Sector No.		56
57	LDA		1665	Core Address		57
60	JSBR	I2	1605	Address → ASCII		60
61	P_1 = 1656 –					61
62	JSBR		1770	Get "LOADS AT..?"		62
63	P_1 = 1651 –					63
64	A=Ø					64
65	STA		1665	New Start Address		65
66	JUMP		0622	Update		66
67						67
70						70
71						71
72						72
73						73
74						74
75						75
76						76
77						77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS-Pgm1.

Page:- Col:- 07

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

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS Pgm 1.

Page:- Col:- 10

Step	Instruction		Address	Comment	Octal	Step
00				E D		00
01				NUL CR		01
02				[illegible]		02
03				W [illegible]		03
04				[illegible] /		04
05				[illegible] R		05
06				[illegible]		06
07				[illegible] .		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 '		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				BEL SP		40
41				SP A		41
42				M E		42
43				N D		43
44				M E		44
45				N T		45
46				NUL		46
47				SP SP		47
50				BEL 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:-

BUSINESS COMPUTERS LTD.

MOLECULAR 18

Sheet:-

Date:-

OS–Pgm l

Page:- Col:- 11

Step	Instruction		Address	Comment	Octal	Step
00	JSBR	I2	1652	PUT ~~File~~ ↓ ZERO		00
01	P_1=1055–					01
02	LDA	Z	0201	Bit 1		02
03	JSBR	I2	1636	Initialised ?		03
04	JSBR	I2	1634	Specify Escape Point		04
05	P_1=0/1400					05
06	JSBR	I2	1770	Get "DISC NO"		06
07	P_1=1065½–					07
10	IORA		1172	200 400 "Write/sector" Jump	1142	10
11	STA		1175			11
12	JSBR		1770	Get "START SECTOR (OCT)"		12
13	P_1=1626–					13
14	STA		1177			14
15	JSBR	I2	1640	Get "SECTORS (DEC)"		15
16	P_1=1621–					16
17	JSBR	I2	1640	SPLIT "PROCESS?"		17
20	P_1=0350–					20
21	JUMP		1106	No		21
22	JSBR	I2	1710	Yes CLEAR CORE (Master Buffer)		22
23	P_1=3700–	I2	1710	Master Buffer		23
24	P_2=128 words					24
25	P– LDA	Z	0072	→ Master Buffer		25
26	STA		1176			26
27	JSBR	I2	1707	Duplicate (Load @ data)		27
30	P_1=1174 –					30
31	P_2=0/0142					31
32	P_3=4 words					32
33	JSBR	I2	1623	LOADQ (Write/sector)		33
34	INSZ	Z	0145	Sector No.		34
35	DESZ	Z	1173	Sector Count		35
36	JUMP		1133	Next Sector		36
37	JSBR	I2	1652	PUT "BEL"		37
40	P_1=0/0207½					40
41	JUMP	Z	1106			41
42	ANDA	Z	1752			42
43	A=D					43
44	Jump	I	3			44
45	JSBR		1776			45
46	IOCS –					46
47	Jump		1155			47
50	JSBR	I2	1634	Specify Escape Point (from 1202)		50
51	P_1= 0/1400					51
52	LDA	Z	0201	Bit 1		52
53	P JSBR	I2	1636	Initialised ?		53
54	JUMP		1204			54
55	IORA		1172			55
56	Jump		1111			56
57	COM. A		1263			57
60	Jump		747			60
61	Jump		1223			61
62	Jump		117	SP O		62
63				V E		63
64				R W		64
65				R I		65
66				T E		66
67				? NUL		67
70				SPLIT "OVERWRITE!" {	30 0000	70
71				{ 1162 –		71
72				MASK	200400	72
73				Sector Count	/	73
74				{ Options	000000	74
75				{ R/W, 1 sector, Disc No.	–	75
76				{ → Buffer	–	76
77				{ Sector Number	–	77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS-Page 1 READ

Page:- 0 Col:- 12 -

Step	Instruction	Address		Comment	Octal	Step
00	JSBR	I2	1652	NUT Title *READ		00
01	P1=1254-					01
02	JUMP		1150	Fetch		02
03					1427 —	03
04	JSBR		1770	Get Octal "DISC No"		04
05	P1=1065½-					05
06	ANDA	Z	1752	Bottom Byte		06
07	ANØ					07
10	JUMP		1300			10
11	IORA	Z	0323	000400 Read 1 sector		11
12	STA		1251			12
13	LDA		1251	*NEXT SECTOR		13
14	CLSA			Ensure "Read"		14
15	STA		1251			15
16	JSBR		1770	Get Octal "SECTOR (OCT)"		16
17	P1=1262½-					17
20	APOS					20
21	JUMP		1233	1231		21
22	STA		1253			22
23	JSBR	I2	1615	Transfer (Read into Master Buffer)		23
24	P1=1250 —					24
25	LDA	Z	0144	→ Buffer		25
26	LDB	Z	0316	Record Length 128 words		26
27	JSBR		1325			27
30	JUMP		1213	Onto next		30
31	CLSA					31
32	A=B					32
33	CLSA			STA 1253		33
34	A=Ø			JSBR I2 1640		34
35	STA		1253	Clear Sector No. 1170 —		35
36	JSBR	I2	1640	SPLIT "OVERWRITE!" Jump 1213		36
37	P1= 1170 —			LDA 1251		37
40	JUMP		1213	No. CLSA, COMPSA		40
41	LDA		1251	Yes. STA 1251		41
42	CLSA/COMPSA			Indicate "Write" Jump 1223		42
43	STA		1251	LDA 1374		43
44	JUMP		1223	Here. STA 1574		44
45				JSBR I2 1761		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 A		71
72				T SP		72
73						73
74						74
75						75
76						76
77				NUL		77

Programmer:-

Sheet:-

Date:-

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	P1= 1500-					01
02	ANDA	Z	4752	000377		02
03	STA		0140	STA 1246		03
04	IORA		1374	045400 Jump 1243		04
05	STA		1574	JSBR IL 1640		05
06	JSBR	IL	1640	SPLIT "TEST?" 1571 -		06
07	P1= 1571 -			Jump 1600		07
10	JUMP		1314	No. LDA 1574		10
11	LDA		1574	Yes XORA 1563		11
12	XORA	Z	0355	Bit15 } Remove "No Test" option STA 1574		12
13	STA		1574	Jump 1600		13
14	JSBR	IL	1640	GET "KEY"		14
15	P1=1574-					15
16	A=Ø					16
17	JUMP		0100			17
20	LDA	Z	0151	→ Record in Buffer		20
21	LDB	Z	0152	= Record Length		21
22	JSBR		1325	Display		22
23	JUMP		1314	Gets next Key.		23
24						24
25	*ENTRY		[illegible]	DISPLAY Record.	←BA→	25
26	STB		1375	Word Count		26
27	JSBR	IZ	1605	Address → ASCII		27
30	P1=1520½-					30
31	JSBR	IL	1652	PUT "BUFFER."		31
32	P1=1514-					32
33	LDA	Z	0215	"NUL CR"		33
34	STA		0400			34
35	STA		0441			35
36	LDA		0001	→ Extract Item		36
37	STA		1377	→ Current Word.		37
40	JSBL	IL	1731	Space fill *NEXT LINE		40
41	P1=0401-					41
42	P1=32words					42
43	LDA	Z	0210	OF8		43
44	STA		1376	Line counter		44
45	LDA		1341	→ Buffer		45
46	STA		1351			46
47	LDA	I	1377	= Next Word *NEXT Word.		47
50	JSBR	IL	1612	Octal → ASCII		50
51	P1= //					51
52	LDA		1351			52
53	ADA	Z	0204			53
54	STA		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	P1=0400½ -					61
62	JUMP	I	1325	Return.		62
63	DESZ		1376	Line Count		63
64	JUMP		1347	Gets next word		64
65	JSBR	IL	1652	PUT line.		65
66	P1= 0400½-					66
67	JUMP		1340	Gets next line.		67
70						70
71						71
72						72
73						73
74				MASK	045400	74
75				Word Count	-	75
76				Line Count	-	76
77				→ Current Word.	-	77

Programmer:-

Sheet:-

Date:-

OS-1

Page:- Col:- 14-

Step	Instruction	Address		Comment	Octal	Step
00	JSBR	I2	1652	PUT "RESTART TASK" * R		00
01	P.=1525½-					01
02	JSBR	I2	1652	PUT "AT OWN RISK"		02
03	P.=3106½-			[illegible]		03
04	NOOP			[illegible]		04
05	JSBR	I2	1634	Specify Escape Point		05
06	P.=011402					06
07	JSBR		1770	Get Octal "TASK NO"		07
10	P.=1534½-					10
11	CMPA	Z	0051	Max Task No.		11
12	NOOP					12
13	SKNGT					13
14	JUMP	I2	1641	Exit Test limits		14
15	APOS					15
16	JUMP	I2	1641	"		16
17	ANØ					17
20	JUMP	I2	1641	"		20
21	STA		1477	TASK NO.		21
22	JSBR		1770	Get Octal "ADDRESS"		22
23	P.=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	ANØ					30
31	LDA	I2	B	Use default restart address		31
32	SFB	Z	0206	→ Escape Point address		32
33	ANØ					33
34	LDA	I2	B			34
35	NOOP					35
36	SFB	Z	0256	→ Base Address.		36
37	APOS			Replace Offset address		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	377777		44
45	JUMP		1560	Task is halted.		45
46	STA	Z	0176	= Restart Address *Search Device Q.		46
47	LDA		1477	Task No.		47
50	LDB	Z	0052	Device Control Table origin		50
51	ADB	Z	0267	→Unallocated Device Q		51
52	STB	Z	0175	Save→previous *Next Vector		52
53	LDB	I2	B			53
54	B=Ø			End of chain?		54
55	JUMP		1461	No.		55
56	JSBR	I2	1652	Put "NOT HALTED!"		56
57	P.=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 vector in Q.		65
66	STA	Z	0177	Task No.		66
67	DECB			→Vector		67
70	LDA	IZ	B	→Next Vector Remove Vecto from Unalloc. Q.		70
71	STA	IZ	0175			71
72	LDA	Z	0054	→1st free Q element.		72
73	STA	IZ	B	Add to free chain.		73
74	STB	Z	0054			74
75	JSBR	I2	1616	STACK restart address		75
76	JUMP	Z	1402	→ PROGRAM?		76
77				TASK NO.		77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

05-Apr 1.

Page:- Col:- 15-

Step	Instruction		Address	Comment	Octal	Step
00				CR F		00
01				I 2		01
02				E SP		02
03				I D		03
04				SP NUL		04
05				SP SP		05
06				T E		06
07				S T		07
10				? NUL		10
11				CR K		11
12				E Y		12
13				SP NUL		13
14				SP SP		14
15				B U		15
16				F F		16
17				E R		17
20				SP		20
21						21
22						22
23						23
24						24
25				NUL CR		25
26				R E		26
27				S T		27
30				A R		30
31				T SP		31
32				T H		32
33				S K		33
34				NUL CR		34
35				T A		35
36				S K		36
37				SP N		37
40				O SP		40
41				NUL SP		41
42				SP A		42
43				D D		43
44				R E		44
45				S S		45
46				SP NUL		46
47				SP SP		47
50				BEL N		50
51				O T		51
52				SP H		52
53				A L		53
54				T E		54
55				D !		55
56				NUL		56
57				0/1402		57
60	STA	IZ	0177			60
61	JUMP	Z	1402	to "PROG ADDR"		61
62					20377	62
63						63
64				(G+B) LAST REC NO.	140106	64
65					220 -	65
66					117171 -	66
67					116110 -	67
70					110171 -	70
71				SPLIT "TEST!"	300000	71
72				1505-		72
73				Po 1300-		73
74					/	74
75				GET EFFECT'N Key 1511-		75
76				-Key 3640-	12 -	76
77				2000-		77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS-Ipul.

Page:- Col:- 16

Step	Instruction	Address	Comment	Octal	Step
00	LDA	1374		300000	00
01	JUMP	1645	1013 -		01
02			Split "INPUT"?	300000	02
03			0727½ -		03
04				220004	04
05			Get "NAME" 0737 -		05
06			1674 -		06
07				001400	07
10			Split "Need Directory?"	300400	10
11			0716½ -		11
12				040100	12
13			1010 -		13
14			GET "SECTORS" (Prim) 1677 -		14
15			Min CF1 0/0201		15
16			Max CF13 0/0215		16
17			Split "AUTO-RESERVE!"	300000	17
20			1030½ -		20
21				040100	21
22			GET "SECTORS" (Zero) 1010 -		22
23			1173 -		23
24			Min BF1 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			SP NUL		40
41			CR S		41
42			E C		42
43			T O		43
44			R SP		44
45	ADDA	1542		-	45
46	STA	1161		-	46
47	JUMP	1314		-	47
50			? NUL		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			" NUL		63
64			Nsects, "OR"	-	64
65			Core Address	-	65
66			StartSector 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			Mod.ob No.	-	76
77			No. of Sectors. Extn H.	-	77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS-Page 1

Page:- Col:- 17-

Step	Instruction	Address		Comment	Octal	Step
00	*ENTRY				←BA→	00
01	LDA	Z	0066	→Input Buffer		01
02	ADA	Z	0066	x2		02
03	STA	Z	0177	Source x 2		03
04	CLA					04
05	STA	Z	0176	Octal Word		05
06	STA	Z	0175	"/" indicator		06
07	LDB	Z	0177	Source x 2 *NEXT CHAR.		07
10	INSZ	Z	0177			10
11	JUMP		1715			11
12						12
13						13
14						14
15	JSBR	IZ	1415	Load Absolute Byte.		15
16	A=Ø					16
17	JUMP		1722			17
20	LDA	Z	0176	= Octal Word		20
21	JUMP	I	1700	Return.		21
22	CMPA	Z	0257	"N42 /"		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	"N42 7"		27
30	NOOP					30
31	SKNGT					31
32	JUMP	IZ	1641	Not octal - error		32
33	SFA	Z	0260			33
34	DESZ	Z	0175	"/" input previously ?		34
35	JUMP		1743	No.		35
36	ANØ/CLC					36
37	JUMP		1745	Ø input		37
40	CMPA	Z	0201	CF1		40
41	JUMP		1745	1 input		41
42	JUMP	IZ	1641	Not Ø or 1 - error		42
43	CLC/LSB					43
44	LSB					44
45	BBS/SKNC					45
46	JUMP	IZ	1641	Overflow! Error.		46
47	LSB					47
50	ADB	Z	A			50
51	SKNC					51
52	CLSB/COMPSB					52
53	STB	Z	0176	Octal Word		53
54	JUMP		1707	Onto next character		54
55	LDA	Z	0201	to "/" input		55
56	STA	Z	0175	Set indicator		56
57	JUMP		1707	Onto next char.		57
60	CMPA	Z	0255	"N42 -"		60
61	CLSB/COMPSB/STCR			"[illegible] -		61
62	JUMP	IZ	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	IZ	1640	GET		74
75	K=1766-					75
76	JSBR		1700	Octal → Binary		76
77	JUMP	I	1770	Return.		77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS-Pgm 1

Page:- 1 | Col:- 00- 20-

Step	Instruction	Address		Comment	Octal	Step
00	JSBR		0100	Initialise *"OP" Entry		00
01	JUMP		1200			01
02	JSBR	I2	1635	GET PASSWORD *"HALT" Entry		02
03	Pi=011324					03
04	HALT					04
05	JUMP		0002			05
06	IORA		167	E		06
07	STA		11			07
10	JSBR	I2	1766			10
11		I1	[illegible]			11
12	STA		176	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	1640			22
23	6DD –					23
24	ANEB					24
25	JUMP		37			25
26	COMPA	2	363	"a"		26
27	JUMP		33	KD		27
30	JUMP	I2	1641			30
31	JSBR	I2	1646			31
32	1564 –					32
33	LDA		11			33
34	ANDA	2	1752			34
35	SETSA					35
36	STA		46			36
37	JSBR	I2	1610	FETCH + LOGIC		37
40	4E					40
41	113176 –					41
42	IB 3400 –					42
43	JSBR	I2	1716			43
44	P1 – 3400 –					44
45						45
46	JSBR	I2	1671			46
47	LDA		176			47
50	COMPA		171			50
51	JUMP		54			51
52	INSZ		176			52
53	JUMP		37			53
54	LDA		1752			54
55	JUMP		1735			55
56	JSBR	I2	1721	LDA FileTable Offset }		56
57	Pi=3775–					57
60	APOS			Obtain FileTable		60
61	INCA			Number		61
62	ARA					62
63	ANDA	Z	0202			63
64	ADA	Z	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:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS - Page 1

Page:- 01 Col:- 01- 21-

Step	Instruction	Address		Comment	Octal	Step
00	*ENTRY			INITIALIZE	← BA →	00
01	JSBR	I2	1627	Resolve Offset Addresses		01
02	P₁=S.1/0760					02
03	JSBR	I2	1627	" " "		03
04	P₁=S.0/0100					04
05	TLDA	Z	0201			05
06	P₁= 0 JSBR	I2	1636	Initialized?		06
07	JSBR	I2	1707	Duplicate		07
10	P₁= 3200-					10
11	P₂= 3200-					11
12	P₃= 320 words				000500	12
13	TLDA		0766			13
14	STA		1342			14
15	JSBR	I2	1643	Specify Default Restart Address		15
16	P₁= 3200-					16
17	LDA	Z	0047	→ Task Control Table		17
20	ADA	Z	0040	+ Current routine no.		20
21	LDA	I2	A	→ Control Block, this routine		21
22	LDA	I2	A	= Device Code, current routine		22
23	STA	Z	0177	Save		23
24	ANDA	Z	1752	Bottom Byte (leaves output device code)		24
25	IORA		0703			25
26	STA		1515			26
27	LDA	Z	0177			27
30	SWAPA					30
31	ANDA	Z	1752	Bottom Byte (leaves input device code)		31
32	STA	Z	0176	Save.		32
33	ADA		0704	}		33
34	STA		1505			34
35	ADA		0705			35
36	STA		1604	Set up service rtn.		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||000		44
45	STA		1677	Set up Completion Address		45
46	LDB	I	1677	= Old Service pointer } Swap service		46
47	LDA		0763	= New " " routines		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	SKNGT					55
56	JUMP	I	0166			56
57	CLA			+ ANK.		57
60	STA		1471	} eliminate decode.		60
61	STA		1472			61
62	JSBR	I2	1707	Duplicate revised terminator set		62
63	P₁= 3172-					63
64	P₂= 3344-					64
65	P₃= 6 words					65
66	JUMP		0056			66
67				UMSK	001400	67
70				Kq { UCu Reset	-	70
71				UCx Reset	-	71
72				Reject NUL CAN	000030	72
73				Initial Ser. NUL GS	000035	73
74				Backward Space NUL CR	000015	74
75				Forward Space NUL TAB	000011	75
76				Send − NUL −	000055	76
77				Send + NUL ETB	000027	77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS - Pgm 1

Page:- S.1 Col:- 02→06- As for S.1/12→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:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS - Pgm 1

Page:- S.I Col:- 07 27-

Step	Instruction	Address		Comment		Octal	Step
00				Escape to "PROGRAM?"	0/1400 P6	001400	00
01						200407	01
02				GET "PASSWORD?"	S.1/0751½	302751	02
03						015400	03
04						010200	04
05					MASKS f. o/code.	000600	05
06						002000	06
07						003100	07
10					P6 3004-	200177	10
11				GET "TEXT"		200577	11
12					2722½-		12
13						-	13
14							14
15							15
16							16
17							17
20							20
21							21
22					CR		22
23					T E		23
24					X T		24
25					SP NUL		25
26					BEL SP		26
27					N O		27
30					SP S		30
31					P A		31
32					C E		32
33					NUL CR		33
34					S E		34
35					W D		35
36					SP M		36
37					E S		37
40					S A		40
41					G E		41
42					NUL CR		42
43					T A		43
44					S K		44
45					SP NUL		45
46					?		46
47					NUL S		47
50					NUL P		50
51					NUL CR		51
52					Y A		52
53					S S		53
54					W O		54
55					R D		55
56					? NUL		56
57					! NUL		57
60				→ Overlay Start	0000-		60
61				→ "OP!" Code	3200-		61
62				→ DECODE	2600-		62
63				→ Realscreen Servicing Rtn.	3504-		63
64				→ Key Area for Directory	2072-		64
65				→ Data Area	2074-		65
66				→ Breakpoint Handling Entry point	0400-		66
67	O Start Addresses.					0	67
70						0	70
71						0	71
72						0	72
73						0	73
74						0	74
75						0	75
76				→ Octal "GET" Rtn.	1770-		76
77						0	77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS-1

Page:- 1 Col:- 10- 30-

Step	Instruction	Address		Comment	Octal	Step
00	JSBR	I2	1652	PUT "Send Message"		00
01	P_1= 2733½-					01
02	JSBR	I2	1627	Realise Offset Block.		02
03	P_1 = 2776 –					03
04	JSBR	I	0776	Get Offset "TASK"		04
05	P_1 = 2742½ –					05
06	APOS					06
07	JUMP	I2	1641	End -		07
10	CMPA	Z	0055	No. of I/O Stations.		10
11	NOOP					11
12	SKNGT					12
13	JUMP	I2	1641	End		13
14	STA		1074			14
15	JSBR	I2	1640	GET "TEXT"		15
16	P_1= 2714 —					16
17	LDA	Z	0066	→ Input Buffer		17
20	ADA	Z	0066	x2		20
21	ADA	Z	0045	No. of Chars input.		21
22	STA	Z	0177	~~Target~~ X2.		22
23	LDA	Z	0217	"NUL SI"		23
24	JSBR		1100			24
25	LDA	Z	0240	"NUL SP" } Store parameters.		25
26	JSBR		1100			26
27	CLA			"NUL NUL"		27
30	JSBR		1100			30
31	LDA	Z	0040	Task No.		31
32	JSBR	I2	1612	Get ad → DSCJI		32
33	P_1 = 3070 –					33
34	JSBR	IL	1707	Duplicate		34
35	P_1 = 3064 –	=		—		35
36	P_2 = 3570 –					36
37	P_3 = 8 cards					37
40	LDA		1074	Target Task No.		40
41	JSBR	I2	1654	Flash Single Task		41
42	P_1 =3570 –					42
43	JUMP		1004	auto vect		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	I2	1634	Specify Escape Point. +SEND -		75
76	P_1 = 0/1402					76
77	JUMP		1000			77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS - Pgm 1

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

Step	Instruction	Address		Comment	Octal	Step
00	XENTRY				2—BA→	00
01	LDB	Z	0177	Tarot×L		01
02	INSZ	Z	0177			02
03	JSBR	IZ	1416	Store Absolute Byte		03
04	JUMP	I	1100	Return		04
05						05
06				SP		06
07				SO A		07
10				T SP		10
11				O W		11
12				N SP		12
13				R I		13
14				S T		14
15				SI NUL		15
16				Max. Check Digit { 999999999	0356 32	16
17					144777	17
20	CHA/COMPSA			* Check Digit List		20
21	JSBR	IZ	1613	Specify Dvcid Q		21
22	JSBR	IZ	1652	PUT "Check Digit List"		22
23	P₁ = 3140-					23
24	JSBR	IZ	1640	GET "FIRST No."		24
25	P₁ = 3173-					25
26	JSBR	IZ	1640	GET "LAST No."		26
27	P₁ = 3165-					27
30	NOOP					30
31	NOOP					31
32	JSBR	IZ	1651	SPOOL & POST		32
33	JUMP	-	1124			33
34						34
35	JSBR	Z	1630	—		35
36	Jump	IZ	A			36
37					1363 -	37
40				CR C		40
41				H E		41
42				C K		42
43				SP D		43
44				I G		44
45				I T		45
46				SP L		46
47				I S		47
50				T NUL		50
51				CR F		51
52				I R		52
53				S T		53
54				SP N		54
55				O SP		55
56				NUL SP		56
57				SP L		57
60				A S		60
61				T SP		61
62				N O		62
63				SP NUL		63
64				3124- P₀		64
65					040600	65
66				3156½-		66
67				Get "LAST No" 3403-		67
70				Min 3401-		70
71				Max 3116-		71
72				011400 P₀		72
73					040600	73
74				3151-		74
75				Get "FIRST No" 3401-		75
76				Min 010200		76
77				Max 3116-		77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS - Pgm I

Page:- S.1 Col:- 12 32 -

Step	Instruction		Address	Comment	Octal	Step
00	CCLA			★ OP entry		00
01	JUMP		1365			01
02	NOOP					02
03	CMPB		1343	"NUL S" (from 1337)		03
04	JUMP		1260	Breakpoint Handler		04
05	CMPB	Z	0304	"NUL F"		05
06	JUMP		1264	File-Control-Block		06
07	CMPB		1662	"NUL R"		07
10	JUMP		1474	Read Program		10
11	CMPB		1661	"NUL W"		11
12	JUMP		1240	Write Program		12
13	JUMP		1272	~~Error~~ F.		13
14						14
15						15
16						16
17						17
20						20
21						21
22						22
23						23
24	*ENTRY			GET Octal Input		24
25	LDA	I	1224	=P1		25
26	STA		1457	→ ASCII prompt.		26
27	INSZ		1224			27
30	CLA					30
31	STA		1672	Clear Octal Word		31
32	STA		1674	Indicate Octal Mode		32
33	JSBR	I2	1640	GET Octal Input		33
34	P1=1/1456-					34
35	LDA		1672	Octal Word Input		35
36	LDB		1666	Terminating Character		36
37	JUMP	I	1224	Return.		37
40	CMPA		1664	= Overlay No. Read? *WRITE OVERLAY		40
41	SKIP					41
42	JUMP		1272	'Error'		42
43	JSBR	I2	1670	STOW Overlay		43
44	P1=Read, Write, Privilege, "02"				201402	44
45	P2=S.1/1664			→ Program No.		45
46	LDA	Z	0146	} Save New Hash total.		46
47	STA		1672			47
50	JSBR	I2	1670	FETCH & LOCK Overlay Index		50
51	P1=Lock, Privilege, "01"					51
52	P2=S.1/1664			→ Program No.		52
53	P3=0/0170			Workarea (4 words)		53
54	LDA		1672	} Insert new hash total.		54
55	STA	Z	0173			55
56	JSBR	I2	1671	REWRITE Overlay Index		56
57	JUMP		1301			57
60	JSBR	I2	1670	FETCH Breakpoint Handler		60
61	P1=00000Z					61
62	P2=0/0202			Octal 002		62
63	JUMP	I	1342	0400-		63
64	STA	Z	1266	★ FILE CONTROL Block		64
65	JSBR	I2	1700	Extract from FCB		65
66	P1=/					66
67	LDA	Z	1B			67
70	JUMP		1300			70
71						71
72	CLA/COMPA			*ERROR		72
73	STA		1675	Indicate 'Error' Mode		73
74	JUMP	I2	1641	'Error'		74
75	LDA		1673	*RETURN to PROGRAM		75
76	STA	I	1677	Restore Original Service routine.		76
77	JUMP	Z	1402	to "PROGRAM?"		77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS - Pgm 1

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

Step	Instruction	Address		Comment	Octal	Step
00	STA		1665	Set Location Pointer *Amend Interrogate.		00
01	JSBR		1460	Set up ASCII *Get Octal Input		01
02	JSBR		1224	GET Octal Input		02
03	P₁= 1/1642-					03
04	JUMP		1311			04
05	LDA		1341			05
06						06
07	JSBR		1000			07
10	JUMP		1275			10
11	CMPB	Z	0233	"NUL ESC"		11
12	JUMP		1305	"Escape to "PROGRAM?"		12
13	CMPB	Z	0303	"NUL A"		13
14	JUMP		1300	Reset Address Pointer		14
15	CMPB	Z	0351	"NUL GS" (Enter +)		15
16	JUMP		1357	Amend word.		16
17	CMPB	Z	0212	"NUL LF"		17
20	JUMP		1357	Amend word.		20
21	CMPB	Z	0231	"NUL EM" (Reset)		21
22	JUMP		1301	Re-output		22
23	CMPB		1347	"NUL NAK" (→)		23
24	JUMP		1403	Forward Space		24
25	CMPB		1346	"NUL BS" (←)		25
26	JUMP		1413	Back space		26
27	CMPB		1345	"NUL SUB" (↑)		27
30	JUMP		1423	Literal Mode		30
31	CMPB	Z	0240	"NUL SP"		31
32	JUMP		1434	Text Output		32
33	CMPB		1641	"NUL I"		33
34	JUMP		1442	Set Address Pointer Indirect flg		34
35	CMPB		1370	"NUL O"		35
36	JUMP		1365	Offset Address pointer		36
37	JUMP		1203			37
40						40
41				Breakpoint [illegible]	/	41
42				Breakpoint [illegible]	/	42
43				NUL S		43
44				Reject NUL ETX	000003	44
45				Literal Switch NUL SUB	000032	45
46	Terminator Cert			Backward Space NUL BS	000010	46
47	(overwritten ∦ NAK)			Forward Space NUL NAK	000025	47
50				Load − NUL FS	000034	50
51				Load + NUL GS	000035	51
52	CMPA	Z	0256	"NUL ." (from 1624)		52
53	SKIP					53
54	JUMP		1611	Not octal		54
55	ADB	Z	0356	Bit 16		55
56	JUMP		1603			56
57	STA	I	1665	*AMEND CORE		57
60	JSBR		1460	Setup ASCII		60
61	INSZ		1665			61
62	JSBR	IZ	1652	PUT new ASCII		62
63	P₁=S.1/1646½					63
64	JUMP		1301			64
65	CASA/COMPSA			*OFFSET POINTER		65
66	JSBR	Z	1630	Resolve Offset		66
67	JUMP		1300	Reset Address pointer		67
70				NUL O		70
71	STA		1664	Prog No *READ OVERLAY		71
72	JSBR	IZ	1670	FETCH Overlay		72
73	P₁=Text, "O2"				100002	73
74	P₂=S.1/1664			→ Program No:		74
75	JUMP	IZ	1641	Error - not found		75
76	LDA	Z	0144	= Start Core Address (Absolute)	213144	76
77	JUMP		1300	Amend/Interrogate		77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS – Pgm 'I'

Page:- S.1 Col:- 14 34–

Step	Instruction		Address	Comment	Octal	Step
00	ANØ	=	1665	(Jan 1477)	[illegible]	00
01	JUMP	I2	1644	Error	[illegible]	01
02	JUMP		1371	[illegible]	[illegible]	02
03	ANØ			*FORWARDS	007500	03
04	INSZ		1672	Counter=1	040672	04
05	SKIP				004020	05
06	JSBR		1447	Display	031447	06
07	INSZ		1665	MA } Display Loop.	041665	07
10	DESZ		1672	Counter	051672	10
11	JUMP		1406		021406	11
12	JUMP		1301		021301	12
13	ANØ			*BACKWARDS	007500	13
14	INSZ		1672	Counter=1	041672	14
15	SKIP				004020	15
16	JSBR		1447	Display	031447	16
17	DESZ		1665	MA } Display Loop	051665	17
20	DESZ		1672	Counter	051672	20
21	JUMP		1416		021416	21
22	JUMP		1301		021301	22
23	ANØ			*LITERAL	007500	23
24	LDA		1665	MA	211665	24
25	LRA				003240	25 ←
26	STA		1674	=In x2	251674	26
27	JSBR	I2	1640	GET Literal	037640	27
30	P.=S.1/1454				A	30
31	LDA		1674	In x2	211674	31
32	RSA			(ignore odd byte)	003100	32 ←
33	JUMP		1300		021300	33
34	ANØ			*TEXT	007500	34
35	LDA	=	1665	MA	211665	35
36	STA		1440		251440	36 ←
37	JSBR	I2	1652	Put Text	037652	37
40	P.=0				–	40
41	JUMP		1301	Re-output octal.	021301	41
42	ANØ			*INDIRECT	007500	42
43	LDA		1665	MA	211665	43
44	LDA	I2	A		216000	44
45	JUMP		1300	Reset MA.	021300	45
46				Save Value	/	46
47	* ENTRY			PUT MA + Contents	←BA→	47
50	JSBR		1460		031460	50
51	JSBR	I2	1652	PUT	037652	51
52	P.=S 1/1642				A	52
53	JUMP	I	1447		025447	53
54				GET Literal	200000	54
55				S.1/1633½	303633	55
56				GET Octal	200000	56
57					[illegible]	57
60	*ENTRY			Setup ASCII.o/	←BA→	60
61	LDA		1665	Location Pointer		61
62	JSBR	I2	1605	Address → ASCII		62
63	P.=S.1/1642½					63
64	LDA	I	1665	Location Contents	215665	64
65	STA		1446	(Save to avoid double "Parity" error)	251446	65
66	JSBR	I2	1612	Octal → ASCII	037612	66
67	P.=S.1/1647				–	67
70	LDA		1446		211446	70
71	JSBR	I2	1604	DECODE	037604	71
72	P.=S.1/1652½					72
73	JUMP	I	1460		025460	73
74	LDB		1341	*READ octal		74
75	B=Ø					75
76	JSBR		1015			76
77	JUMP		1400			77

Programmer:-

BCL/MI8/PSI

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS - Pgm 1.

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

Step	Instruction	Address		Comment	Octal	Step
00				FASHBACK DEVICE CODE	–	00
01				In x2	–	01
02				Count	–	02
03				Status / Actual EOF	–	03
04	*ENTRY			INPUT SERVICE Rtn.	←— BA —→	04
05	DATIAB			(0102 dd)	[illegible]	05
06	DATIA/STOP			(0161dd)	[illegible]	06
07	BNØ				006500	07
10	JUMP		1515	O.K	021515	10
11	IOPAS			$STATUS (0130dd)	[illegible]	11
12	SWAPB/CASB/COM/SB				004016	12
13	STB		1666	Terminating character	261666	13
14	JUMP		1612	Complete	021612	14
15	DATOA/START			FASHBACK (0154dd)	[illegible]	15
16	JSBR	I2	1772	Convert to upper case.	[illegible]	16
17	CMPB		1675	Error Mode?	241675	17
20	JUMP		1527	No.	021527	20
21	CMPA		1344	Cancel EOF?	[illegible]	21
22	SKIP				004020	22
23	JUMP		1612	Re-output "ERROR" (not acknowledged)	021612	23
24	STB		1675	Clear "Error Mode" indicator	261675	24
25	STB		1672	Clear Octal Word	261672	25
26	JUMP		1620	Complete (re-output prompt)	021620	26
27	CMPB		1674	Octal Mode?	241674	27
30	JUMP		1550	Yes.	021550	30
31	CMPA	Z	0233	"NUL ESC" ←LITERAL MODE	[illegible]	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	STA		1672	Workspace	251672	37
40	LDA	Z	1752	Bottom 3 bits	213752	40
41	SKNC				007040	41
42	SWAPA				005010	42
43	ANDA	I2	1		066001	43
44	IORA		1672		071672	44
45	STA	I2	1		256001	45
46	INSZ		1674	In x2	041674	46
47	JUMP		1604	Start	021604	47
50	CMPA		1344	Cancel EOF? ←OCTAL MODE	[illegible]	50
51	JUMP		1525	Yes - re-output prompt	021525	51
52	CMPA	Z	0257	"NUL /"	232257	52
53	JUMP		1606		021606	53
54	LDB		1672	Octal word so far.	221672	54
55	SHGT				006410	55
56	JUMP		1622		021622	56
57	CMPA	Z	0267	"NUL 7"	232267	57
60	NOOP				000000	60
61	SHNGT				006010	61
62	JUMP		1611	Not octal	021611	62
63	SFA	Z	0260	"NUL Ø"	132260	63
64	DESZ		1667	"/" input previously?	051667	64
65	JUMP		1573	No.	021573	65
66	ANØ/CLC				007520	66
67	JUMP		1575	Ø input	021575	67
70	CMPA	Z	0201	CFI	232201	70
71	JUMP		1575	1 input	021575	71
72	JUMP		1611	Not Ø or 1	021611	72
73	CLC/LSB				002700	73
74	LSB				002300	74
75	BPOS/SKNC				006240	75
76	JUMP		1512	Overflow! (force Error)	021512	76
77	LSB				002300	77

Programmer:-

BUSINESS COMPUTERS LTD. **MOLECULAR 18**

Sheet:-

Date:-

OS – Page 1.

Page:- 11 Col:- 16– 36–

Step	Instruction	Address		Comment	Octal	Step
00	ADB	Z	A		122000	00
01	SKNC					01
02	ChSB/COMPSB					02
03	STB		1672	Octal Word.		03
04	START			(0110dd)		04
05	JUMP	IZ	0010	Dismiss (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 Accept) *COMPLETE		12
13	STA		1503	"Actual" EOF for Supervisor to detect.		13
14	ChA					14
15	STA		1501	Indicate Completion		15
16	JUMP	IZ	0010	Dismiss (Completed)		16
17						17
20	LDA	Z	0203	"NUL ETX" (EOF Reject)		20
21	JUMP		1613			21
22	CMPA	Z	1350	"NUL FS" Exits –?		22
23	ChSB/COMPSB/SKIP					23
24	JUMP		1352	Test for "." (2nd byte)		24
25	STB		1672	Octal Word.		25
26	LDA	Z	1351	"NUL GS" (Exit +)		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				decode		55
56						56
57						57
60				SI · SPE		60
61				NUL W		61
62				NUL R		62
63				"Privileged 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 / Outcome	–	72
73				⇒ Peripheral Device Ptr. B504	–	73
74				Octal Mode Indicator (≠0) & INX 2	–	74
75				Error Mode Indicator (If ≠0)	–	75
76						76
77				Service Pointer, Input Devices, current route	–	77

Programmer:-