

OS - DUMMY DRI SERVICE

Page:- 03 Col:- 00

Step	Instruction	Address	Comment	Octal	Step
00	*ENTRY		Output *BASE*	←BA→	00
01	DATIA/START			015400	01
02	JSBR	0020	Wait for DONE flag		02
03	DATIA/STOP			016200	03
04	AND		[INCA]AISE]		04
05	JUMP	I 0000	Return		05
06	→ NOOP		*STATUS		06
07	STA	0013	Status		07
10	CLA				10
11	STA	0014	Feedback Counter		11
12	JUMP	0066			12
13			Line Feed Count / Status	-	13
14			→ Buffer Word 1	-	14
15			Count	-	15
16	*ENTRY		SERVICE INTERRUPT	←BA→	16
17	JUMP	I 0020	Resume Channel Program		17
20	*ENTRY		WAIT 0/0021	←BA→	20
21	JUMP	I 0016	Dismiss Interrupt		21
22	INT OFF		*START CHANNEL PROGRAM	002005	22
23	DATIA/STOP			016200	23
24	A=0		[AASB]		24
25	JUMP	0006	Status - abort		25
26	→ LDB	I 0014	= word 1		26
27	BPOS				27
30	LDA	Z 0213	"NUL VT"		30
31	→ BND				31
32	LDA	Z 0214	"NUL FF"		32
33	→ A=0				33
34	JUMP	0044			34
35	→ STB	0013	Line Feed Count		35
36	JUMP	0041			36
37	LDA	Z 0212	"NUL LF"		37
40	JSBR	0000	Output character (Line Feed)		40
41	DESZ	0013	Line feed count		41
42	JUMP	0037	Auto reset line feed		42
43	→ SHIP				43
44	JSBR	0000	Output character (Vertical Tab & Form Feed)		44
45	INSZ	0015	Count (characters)		45
46	LDA	0014			46
47	ADA	0014	x2		47
50	ADA	Z 0303	CFB		50
51	STA	0014	Status x2		51
52	JUMP	0061			52
53	LDB	0014	Status x2	* Next character	53
54	INSZ	0014			54
55	JSBR	I 1417	Read Block into Buffer		55
56	CMHA	Z 0260	"NUL B"		56
57	ADA	Z 0227	"NUL 0"		57
60	→ JSBR	0000	Output character		60
61	DESZ	0015	Count		61
62	JUMP	0053	Auto reset character		62
63	→ LDA	Z 0215	"NUL CR"		63
64	JSBR	0000	Output character (Carriage Return)		64
65	JUMP	0010	Complete		65
66	JSBR	0020	Wait		66
67	STOP			012000	67
70	JUMP	0066			70
71					71
72					72
73					73
74					74
75					75
76					76
77					77

OS - DUMMY CDC 400 SERVICE

Page:- 03 Col:- 01

Step	Instruction	Address	Comment	Octal	Step
00	*ENTRY		Output *BASE*	← BA →	00
01	DAT12B/STOP			012200	01
02	B=0				02
03	JUMP	0012	Status - abort		03
04	→ DAT02H		(Data Channel Mode)	014500	04
05	DAT01A/START			015400	05
06	JSBR	0024	Wait for DONE flag		06
07	DAT12B/STOP			012200	07
10	BND				10
11	JUMP	I 0000	Return		11
12	→ NOOP		↓ STATUS		12
13	STB	0017	Status		13
14	CIA				14
15	STA	0020	Indicate Completion		15
16	JSBR	0024	WAIT		16
17			Status	-	17
20			→ Buffer Word 1	-	20
21			Count	-	21
22	*ENTRY		SERVICE INTERRUPT	← BA →	22
23	JUMP	I 0024	Resume Channel Program		23
24	*ENTRY		WAIT 0/0021	← BA →	24
25	JUMP	I 0022	Dismiss Interrupt		25
26	LDB	0020	→ Buffer Word 1 *START CHANNEL PROGRAM		26
27	ADB	0060	CF67		27
30	LDA	I2 B	Insert CR at end of line		30
31	ANDH	Z 1753	Top byte		31
32	JORH	Z 0215	"MIX CR"		32
33	STA	I2 B			33
34	LDB	I 0020	= Word 1		34
35	LDA	0061	→ 000415 (Line Feed Buffer)		35
36	BPOS				36
37	INCA..		→ 021015. (Wait and take buffer)		37
40	→ BND				40
41	ADA	Z 0202	→ 020015 (Form Feed Buffer)		41
42	→ STB	0021	Count (Line feeds)		42
43	INT OFF			000005	43
44	CHPA	0061	→ 000415		44
45	JUMP	0056			45
46	→ JSBR	0000	Output buffer: (Vertical tab or Form feed)		46
47	LDA	0020	→ Word 1		47
50	INCH		→ data		50
51	JSBR	0000	Output buffer: (data line)		51
52	JUMP	0014	Complete		52
53	LDA	0061	→ 000415 → Next line feed		53
54	JSBR	0000	Output buffer: (line feed)		54
55	DESZ	0021	Count		55
56	JUMP	0053	Or to next line feed		56
57	→ JUMP	0047	Output data line		57
60			CF 67	000103	60
61			→ 000415 (Line Feed Buffer)	3/1575	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 BCL Printer Service

Page:- 03 Col:- 02

Step	Instruction	Address	Comment	Octal	Step
00	*ENTRY		Output to BASE	← BA →	00
01	DATIRA/START			015400	01
02	JSBR	0022	Wait for DONE flag		02
03	DATIRA/STOP			016200	03
04	AND				04
05	JUMP	I 0000	Return		05
06	NOOP			↓ STATUS	06
07	STA	0015	Status		07
10	CYA				10
11	STA	0016	Indicate Completion		11
12	JSBR	0022	WAIT		12
13			Program = 66 lines	000102	13
14			line feeds to next page	000102	14
15			↑ Count of Status	-	15
16			→ Buffer Word 1	-	16
17			Count	-	17
20	*ENTRY		SERVICE INTERRUPT	← BA →	20
21	JUMP	I 0022	Resume Channel Program		21
22	*ENTRY		WAIT	0/0021 ← BA →	22
23	JUMP	I 0020	Divide Interrupt		23
24	INT OFF		↓ START CHANNEL PROGRAM	000005	24
25	DATIRA/STOP			016200	25
26	A=0				26
27	JUMP	0006	Status - abort		27
30	LDB	I 0016	= word 1		30
31	BPOS				31
32	LDB	Z 0202	(Perhaps Vertical Tab by 2 line feeds)		32
33	BNA				33
34	LDB	0014	(Perhaps Form Feed by no. of line feeds to next page)		34
35	STB	0015	line feed count		35
36	SKIP				36
37	JSBR	0064	line feed		37
40	DESZ	0015	line feed count		40
41	JUMP	0037	up next line feed		41
42	IN SZ	0017	Count (Character)		42
43	LDA	0016			43
44	ADA	0016			44
45	ADA	Z 0203	CF3		45
46	STA	0016	Source x2		46
47	JUMP	0056			47
50	LDB	0016	Source x2	*NEXT CHARACTER	50
51	IN SZ	0016			51
52	JSBR	IL 1417	Load Alternate Byte		52
53	CHPA	Z 0260	"NIL 0"		53
54	ADA	Z 0237	"NIL 0"		54
55	JSBR	0000	Output Character		55
56	DESZ	0017	Count		56
57	JUMP	0050	Next next character		57
60	JSBR	0064	line feed		60
61	LDA	Z 0215	"NIL CR"		61
62	JSBR	0000	Output Character (Correct Return)		62
63	JUMP	0010	Complete		63
64	*ENTAX		LINE FEED	← BA →	64
65	LDA	Z 0212	"NIL LF"		65
66	JSBR	0000	Output Character (line feed)		66
67	DESZ	0014	line feeds to next page		67
70	JUMP	I 0064	Return	← BA →	70
71	LDA	0013	Page Stop		71
72	STA	0014	Reset		72
73	JUMP	I 0064	Return		73
74					74
75					75
76					76
77					77

OS - On-line security (SW intercept)
called by TASK 0

Page:- 3 Col:-03

Step	Instruction	Address	Comment	Octal	Step
00	*ENTRY		SWITCH REGISTER Intercept	← BA →	00
01	ESWRA				01
02	CMPA	0373	Action Mask (bits 12, 11, 8, 7, 4, 3)		02
03	SKIP				03
04	JUMP	I 0300	Return		04
05	LDA	IZ 0022	= 1st word of OLS control block.		05
06	APOS				06
07	JUMP	I 0300	Return (ignore leaving REMD)		07
10	AND				10
11	JUMP	I 0300	Return (ignore)		11
12	STA	0473	= No. of OLS Master Discs in system. (Digits 12, 11, 8, 7, 4, 3)		12
13	HAAT				13
14	ESWRA				14
15	ANEG				15
16	JUMP	I 0300	Return - false alarm.		16
17	STA	0372	Save Security Reg.		17
20	LDA	Z 0032	→ Control block		20
21	INCA		*NEXT PAIR		21
22	STA	Z 0176	→ 1st word		22
23	INCA				23
24	STA	Z 0177	→ 2nd word		24
25	INCA				25
26	STA	0474	→ 3rd word		26
27	JSBR	0234	Process disc pair		27
30	LDA	0474			30
31	DESI	0473	Counter		31
32	JUMP	0321	Auto next pair		32
33	JUMP	0374	End.		33
34	*ENTRY		PROCESS DISC PAIR	← BA →	34
35	LDA	IZ 0176	1st word		35
36	JORH	IZ 0177	2nd word		36
37	LDB	0372	Security Reg.		37
40	LRB/BMSB/USE				40
41	JUMP	0351	Ignore - both switches are set.		41
42	APOS				42
43	JUMP	0351	Ignore - 1 cartridge is already mounted.		43
44	BLSB				44
45	JUMP	0354	Disconnect MASTER		45
46	LRB/BLSB		+ load		46
47	JUMP	0360	Disconnect SLAVE		47
50	SKIP				50
51	LSB				51
52	STB	0372	Security Reg.		52
53	JUMP	I 0334	Return		53
54	LDA	IZ 0177	2nd word *MARK MASTER	176	54
55	CISA/COMISA		*MARK Out-of-Commission		55
56	STA	IZ 0177		176	56
57	JUMP	0351		176 0474	57
60	STB	0372	Security Reg. *MARK SLAVE		60
61	LDA	IZ 0176	1st word		61
62	CISA/COMISA		*MARK Out-of-Commission		62
63	STA	IZ 0176		176 0177	63
64	LDA	I 0474	= 3rd word		64
65	AND		Enable?		65
66	JUMP	I 0334	No-return.		66
67	JUMP	0572	Yes - reattach Master Vesta		67
70					70
71					71
72			Switch Register		72
73			Action Mask	006314	73
74	CHA		(from 0333)		74
75	STA	Z 1717	Indicate No Discs Cdb.		75
76	JSBR	IL 1674	Re-attach to unattached Disc Queue.		76
77	JUMP	I 0300	Return done.		77

OS - On-line - Security

Page:- 03 Col:- 04

Step	Instruction	Address	Comment	Octal	Step
00	*ENTRY		SKIP IF O.S. NOT BUSY	←BA→	00
01	JSBR	0410	Skip if O.S. Master Disc		01
02	JUMP	0406	Not O.S. therefore not busy.		02
03	INSZ	Z 0177	→ 3rd word		03
04	LDA	IZ 0177	= 3rd word (→ Vector if disc is busy)		04
05	ANφ		Busy?		05
06	INSZ	0400	No.		06
07	JUMP	I 0400	Yes. Return.		07
10	*ENTRY		SKIP IF O.S. MASTER DISC	←BA→	10
11	LDA	IZ 0032	= No. of O.S. Master Discs		11
12	ANφ				12
13	JUMP	I 0410	Return - No O.S. discs		13
14	APOS				14
15	JUMP	0441	Program "HEND" is in action		15
16	STA	Z 0177	Counter (No. of Discs)		16
17	LDA	Z 0032			17
20	INCA		→ 1st field		20
21	CMPB	IZ A	←NEXT DISC		21
22	JUMP	0427	this is an O.S. Master Disc		22
23	ADA	Z 0203			23
24	DESZ	Z 0177	Counter		24
25	JUMP	0421	into next disc		25
26	JUMP	I 0410	Return - not in list (at Slave out of use)		26
27	INCA				27
30	STA	Z 0177	→ InUse / Slave Disc No.		30
31	LDA	IZ A	= Full / Slave Disc No.		31
32	APOS		Master Full?		32
33	JUMP	0436	No.		33
34	INSZ	0410	Yes. (sleep)		34
35	JUMP	I 0410	Return.		35
36	CLSA				36
37	LDB	Z A	Replace Master Disc by Slave Disc.		37
40	JUMP	I 0410	Return (no sleep)		40
41	CLSA		←HEND in action.		41
42	CMPA	Z 0040	Current Task = Program "HEND"?		42
43	JUMP	I 0410	Yes - goto 41 (no sleep)		43
44	LDA	Z 0040	Current Task		44
45	APOS		Task Scheduler?		45
46	JUMP	I 0410	Yes - return (no sleep)		46
47	JSBR	I 0477	12/0540 Define Callbac Holder		47
50	JSBR	IZ 1777	HALT - attempt to Read/Write during "HEND"		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			Slave Disc No. (Count Discs)	-	73
74			→ 3rd word of Control Block	-	74
75			→ Current Control Block word. 12/1776 I		75
76			12/1730		76
77			12/0540		77

OS - on-line security

Page:- 03 Col:- 05

Step	Instruction	Address	Comment	Octal	Step
00	* ENTRY		Completion Test	← BA →	00
01	STA	I 0476	1211730 → Vectat		01
02	ADA	Z 0203	CF3		02
03	LDA	I2 A	=R/W, No. Sector, Dir. No.		03
04	ANEG		Recast?		04
05	JUMP	I 0500	Yes - return.		05
06	ANDA	Z 1752	(Leave Dir. No.)		06
07	LDB	Z A	= Dir. No.		07
10	JSR	0410	Stop if OLS Master Dir.		10
11	JUMP	I 0500	Not OLS - return.		11
12	STA	0473	Save Slave Dir. No.		12
13	LDA	Z 0177			13
14	INCH				14
15	STA	0474	→ 3rd word of Control Block		15
16	LDB	I 0475	1211776I = Unit Control Word (→ Vectat)		16
17	CLSB				17
20	STB	I 0474	Indicate OLS Dir. is Busy		20
21	STB	0527	→ Vectat		21
22	LDA	I2 B	→ Next Vectat in Q? Advances Unit Controller		22
23	STA	I 0475	1211776I J. sets vectat @ vectat		23
24	A=0				24
25	JSR	I2 1674	Re-schedule Dir. Q for this Unit.		25
26	JSR	I2 1707	Duplicate (Vectat from Q) & guarantee Slave Vectat		26
27	K=				27
30	R=0/0140				30
31	R=16units				31
32	LDA	Z 0143	R/W, No. Sector (R/W)		32
33	ANDA	Z 1753	Top B/Ae { Invert Slave Dir. No.		33
34	JORA	0473	Slave Dir. No.		34
35	STA	Z 0143			35
36	LDA	0474	→ 3rd word of Control Block		36
37	STA	Z 0150	into Slave Vectat		37
40	CLA/COMP				40
41	STA	Z 1717	Indicate No. Devices Busy		41
42	JSR	I2 1673	WRITE to SLAVE		42
43	LDA	I2 0150	→ Master Vectat		43
44	AN0		Has Slave been taken out of commission?		44
45	JUMP	I2 1624	Yes - ignore. (to Task Scheduler)		45
46	STA	0554			46
47	CLA				47
50	STA	I2 0150	Indicate OLS is Not Busy		50
51	STA	Z 1717	No. Devices Busy		51
52	JSR	I2 1674	Reschedule Unallocated Dir. Q		52
53	JSR	I2 1707	Duplicate (Retrieves Master Vectat)		53
54	R=				54
55	R=0/0140				55
56	R=16units				56
57	LDA	Z 0054	→ 1st/2nd relevant		57
60	STA	I 0554	} return to prod chain		60
61	LDA	0554			61
62	STA	Z 0054			62
63	JUMP	I2 0157	Return from LOADQ of WRITE.		63
64					64
65					65
66					66
67					67
70					70
71					71
72	LDB	Z 0107	→ 1st Unalloc. Vectat (from 0367)		72
73	STB	I2 A	} Master Vectat to Head of Unallocated Q.		73
74	STA	Z 0107			74
75	CLA				75
76	STA	I 0474	Indicate not in Use.		76
77	JUMP	I 0334	Return		77

OS-PRINTER COMPLETION

Page:- 03 Col:- 06

Step	Instruction	Address	Comment	Octal	Step	
00	LDB	Z 0074	→ I/O Control Area * Printer Completion		00	
01	ADB	Z 0212			01	
02	LDA	Z 1720	Stacks		02	
03	← A=0				03	
04	JUMP	0625	Stacks, Printers		04	
05	→ STA	I2 B	Clear Count Stacks (3732-)		05	
06	ADB	Z 0206	CF6		06	
07	← DESZ	I2 B	Order Buffers In Use Count		07	
10	← SKIP				10	
11	→ JUMP	I2 1624	WAIT for further work		11	
12	ADB	Z 0210			12	
13	STB	Z 0176	→ Order Buffer A		13	
14	INCB				14	
15	STB	Z 0177	→ Order Buffer B		15	
16	LDA	I2 0176			16	
17	LDB	I2 0177			17	
20	STA	I2 0177			20	
21	STB	I2 0176			21	
22	JSBR	0645	START PRINTER (Buffer B) * Retry		22	
23	INT ON			000004	23	
24	JUMP	I2 1624	WAIT		24	
25	← CMPA	I2 B	Count Stacks * STATUS FAIL		25	
26	← JUMP	0640	No change - don't flush		26	
27	→ STA	I2 B			27	
30	JSBR	I2 1612	Stacks → Octal		30	
31	P ₁ = 13/1771½				31	
32	JSBR	I2 1771	Load Byte		32	
33	P ₁ = 3733½ -		Printer Identifier		33	
34	P ₁ = JSBR	I2 1775		Store Byte		34
35	P ₁ = 13/1764					35
36	P ₁ = JSBR	I2 1653	FLASH - "PRINTER STATUS"		36	
37	P ₁ = 13/1756½				37	
40	P ₁ = JUMP	0774			40	
41					41	
42			Stacks	000000	42	
43			→ Buffer Word 1		43	
44			Count		44	
45	* ENTRY		START PRINTER	→ BA →	45	
46	STA	0643	→ Buffer		46	
47	LDA	I2 0074	?		47	
50	← JUMP	0667	Set "Full" indicator @ 3720-		50	
51	STA	I2 0074			51	
52	LDB	Z 0024	→ Device Table Origin		52	
53	ADB	Z 1717	Device Code		53	
54	LDA	I2 B	= Device ID		54	
55	CHSA				55	
56	SFA	Z 0103	CF3		56	
57	STA	0664			57	
60	ADA	Z 0207	CF7		60	
61	STA	Z 0177			61	
62	JSBR	I2 1707	DUPLICATE		62	
63	P ₁ = 3/0642				63	
64	P ₂ = ✓				64	
65	P ₃ = 4 words				65	
66	JUMP	I2 0177	to Channel Program.		66	
67	CHSA	1717		(New 0650)	67	
70	STB	Z 1717	Device Code		70	
71	COMPSA				71	
72	LDB	I 0643	= Word 0 (Device Count)		72	
73	STB	0644			73	
74	INGZ	0643	→ Word 1		74	
75	JUMP	0651			75	
76					76	
77					77	

OS -

Page:- 03 Col:- 07

Step	Instruction	Address	Comment	Octal	Step	
00	* ENTRY		GET PASSWORD	← BA →	00	
01	LDA	0700			01	
02	JSBR	I2 1725	STA (Save return addresses)		02	
03	P _i = 3730-	0700			03	
04	NOOP				04	
05	JSBR	I2 1640	GET "PASSWORD" (12 characters max)		05	
06	P _i = 310751				06	
07	JSBR	I2 1721	LDA		07	
10	P _i = 3730-		= 8 } next 8 return addresses		10	
11	STA	0700			11	
12	NOOP				12	
13	LDB	I 0700	= P _i → Password		13	
14	JSBR	I2 1414	Resolve Absolute Byte		14	
15	NOOP				15	
16	STB	Z 0176	Password x2		16	
17	LDB	Z 0066	→ Input Buffer		17	
20	HDB	Z 0066			20	
21	STB	Z 0177	In x2		21	
22	LDB	Z 0177		* Next Input char.	22	
23	INSZ	Z 0177			23	
24	JUMP	0730			24	
25					25	
26					26	
27					27	
30	JSBR	I2 1415	Load Absolute Byte		30	
31	STA	Z 0175	Save		31	
32	LDB	Z 0176		* Next Password char.	32	
33	INSZ	Z 0176			33	
34	JUMP	0740			34	
35					35	
36					36	
37					37	
40	JSBR	I2 1415	Load Absolute Byte		40	
41	CMPH	Z 0175	Same Character - Input?		41	
42	(SKIP)		Yes		42	
43	→ JUMP	I2 1641	No - Error		43	
44	(A = 0)		end of Password?		44	
45	→ JUMP	0722	No - cuts next character		45	
46	→ INSZ	0700	Yes - Matched		46	
47	JUMP	I 0700	Return.		47	
50				011402 P ₀	50	
51			GET "PASSWORD"	R10414	51	
52					310753	52
53			CR P			53
54			# S			54
55			S W			55
56			O R			56
57			D SP			57
60			NUL		60	
61	* ENTRY			← BA →	61	
62	ANDA	Z 0474	Program Permit Marks		62	
63	AND				63	
64	(JUMP	I 0761	Return - program may run		64	
65	→ JSBR	I2 1652	GET "AUTHORISED!"		65	
66	P _i = 1311744!				66	
67	JUMP	Z 1402	to "BLOCKED"		67	
70					70	
71					71	
72	* ENTRY		COMPUTE CHECK DIGIT	← BA →	72	
73	JUMP	1000			73	
74	JSBR	I2 1625	SUSPEND		74	
75	JSBR	I2 1721	LDA → octal buffer B	10627	75	
76	P _i = 3751-				76	
77	JUMP	0622	10627		77	

OS

Page:- 03 Col:- 10

Step	Instruction	Address	Comment	Octal	Step
00	CASH				00
01	JDBK	Z 1745	Divide by 10 Unrounded		01
02	STB	Z 0176	Unit Digit		02
03	JSBR	Z 1745	Divide by 10 Unrounded		03
04	STB	Z 0175	10's Digit		04
05	JSBR	Z 1745	Divide by 10 Unrounded		05
06	LSA				06
07	STA	Z 0177			07
10	LSA				10
11	ADA	Z 0177	= 1000's digit x 6		11
12	ADB	Z 0175	= 100's digit + 10's digit		12
13	LSB				13
14	LSB				14
15	ADA	Z B			15
16	ADA	Z 0175	= 1000's x 6 + 100's x 4 + 10's x 5		16
17	ADA	Z 0176			17
20	ADA	Z 0176			20
21	ADA	Z 0176			21
22	STA	Z 0177	= 1000's x 6 + 100's x 4 + 10's x 5 + 1's x 3		22
23	JSBR	Z 1732	DIVIDE by 26 with Remainder		23
24	R=10/0177		Remainder		24
25	R=10/0232		CF26		25
26	R=10/0176		Product		26
27	LDA	Z 0177	= Remainder (0-25)		27
30	ADA	Z 0303	000101 (ASCII A-2)		30
31	JUMP	Z 0772	Notes		31
32	*ENTRY		MOVE to PAD		32
33	LDB	I 1032	= R1 -> Source		33
34	JSBR	Z 1414	Number Absolute Byte		34
35	NOOP				35
36	STB	Z 0177	Source x 2		36
37	INSZ	1032			37
40	LDB	I 1032	= R2 -> Target		40
41	JSBR	Z 1414	Number Absolute Byte		41
42	NOOP				42
43	STB	Z 0176	Target x 2		43
44	INSZ	1032			44
45	LDA	I 1032	= R3 = No of Characters in Target		45
46	INCH				46
47	STA	Z 0175	Counter		47
50	INSZ	1032			50
51	LDB	Z 0177	Source x 2		51
52	BNP		← NEXT SOURCE CHAR.		52
53	JUMP	1065	Pad.		53
54	INSZ	Z 0177			54
55	JUMP	1061			55
56					56
57					57
60					60
61	JSBR	Z 1415	Word Absolute Byte		61
62	A=0				62
63	JUMP	1066			63
64	STA	Z 0177	Indicate End of Source Passed		64
65	LDA	Z 0240	"NUM SP" Padding		65
66	DESZ	Z 0175	Counter		66
67	JUMP	1072			67
70	LDA	Z 0177	= 0 if Source < Target in length		70
71	JUMP	I 1032	Return		71
72	LDB	Z 0176	Target x 2		72
73	INSZ	Z 0176	*STORE CHAR. in TARGET		73
74	JSBR	Z 1416	Stop Absolute Byte		74
75	JUMP	1051	End of program		75
76					76
77	STA	Z 0177			77

OS - Support Structures

Page:- 3 Col:- 11

Step	Instruction	Address	Comment	Octal	Step
00					00
01					01
02					02
03					03
04					04
05					05
06					06
07	* ENTRY		HALT	← BA →	07
10	JSBR	I2 1605	Address → ASCII (Address)		10
11	R= 711666				11
12	LDA	I 1107			12
13	JSBR	I2 1605	Address → ASCII (Back address)		13
14	R= 711660				14
15	LDA	Z 0040			15
16	JSBR	I2 1612	Octal → ASCII (Task Number)		16
17	R= 711645				17
20	JSBR	I2 1741	How to read (Program Name)		20
21	R= 5.111716				21
22	R= 711650				22
23	R= 4clm				23
24	JSBR	I2 1653	FLASH "TASK HALTED"		24
25	R= 711641				25
26	LDA	Z 0074	→ I/O Control Area		26
27	ADA	Z 0202	CF2		27
30	CHB/COMPB				30
31	STB	I2 A	Indicate no restart address		31
32	JSBR	I2 1625	SUSPEND		32
33	LDA	Z 0074	→ I/O Control Area		33
34	ADA	Z 0202	CF2		34
35	LDA	I2 A			35
36	COMPA	Z 0376	377777		36
37	JUMP	I 1132	Jump		37
40	JSBR	Z 1630	Resolve object		40
41	JUMP	I2 A	RESTART TASK.		41
42					42
43	STA	Z 0177	Save packet date (Patch from 1202)		43
44	A=0				44
45	JUMP	I 1203	Continue		45
46	STA	Z 0160			46
47	JUMP	I 1236	Space field forget		47
50					50
51	* ENTRY		Specify Default Restart Address	← BA →	51
52	LDA	I 1151	=P ₁ (Restart address)		52
53	JSBR	I2 1725	STA into TCA		53
54	R= 3772				54
55	INSZ	I 1151			55
56	JUMP	I 1151	Return.		56
57	* ENTRY		STORE BYTE	← BA →	57
60	* ANDA	Z 1752	Bottom Byte	← BA →	60
61	LDB	I 1157	=P ₁		61
62	INSZ	I 1157			62
63	JSBR	I2 1414	Resolve Absolute Byte		63
64	JSBR	I2 1416	Store Absolute Byte		64
65	JUMP	I 1157	Return.		65
66	* ENTRY		LOAD BYTE	← BA →	66
67	LDB	I 1166	=P ₁		67
70	INSZ	I 1166			70
71	JSBR	I2 1414	Resolve Absolute Byte		71
72	JSBR	I2 1415	Load Absolute Byte		72
73	JUMP	I 1166	Return.		73
74					74
75					75
76					76
77					77

OS

Page:- 03 Col:- 12

Step	Instruction	Address	Comment	Octal	Step
00	JUMP	I 1157	Return.		00
01	*ENTRY		Count PACKED DATE → ASCII	← BA →	01
02	JUMP	Z 1143	Patch.		02
03	ANDA	Z 0237	Mask 000057		03
04	STA	Z 0176	= day.		04
05	JSBR	I2 1765	Day → ASCII		05
06					06
07					07
10					10
11	LDA	Z 0177	= Pack Date		11
12	RSA				12
13	RSA				13
14	RSA				14
15	RSA				15
16	ANDA	Z 0236	Mask 000036 (gives Monthx2)		16
17	ADA	1247	+ Month Table origin		17
20	LDB	I2 A	} set up "SP 24 24 m"		20
21	STB	Z 0161			21
22	INCH				22
23	LDB	I2 A			23
24	STB	Z 0162			24
25	LDA	Z 0177	= Pack Date		25
26	RSA				26
27	ANDA	Z 1753	Top Byte		27
30	SWAPH				30
31	STA	Z 0176	= Year		31
32	JSBR	I2 1765	Year → ASCII		32
33					33
34					34
35					35
36	LDA	I 1201	→ Target		36
37	NOOP				37
40	STA	1243	P2		40
41	JSBR	I2 1741	Move B PAD (ASCII date → target)		41
42					42
43					43
44					44
45	JMSZ	I 1201			45
46	JUMP	I 1201	Return.		46
47			→ MONTH TABLE ORIGIN	3/1246	47
50				SP J	50
51				A N	51
52				SP F	52
53				E B	53
54				SP M	54
55				A R	55
56				SP A	56
57				P R	57
60				SP M	60
61				A Y	61
62				SP J	62
63				U N	63
64				SP J	64
65				U L	65
66				SP H	66
67				U G	67
70				SP S	70
71				E P	71
72				SP O	72
73				C T	73
74				SP N	74
75				O V	75
76				SP D	76
77				E C	77

OS

Page:- 03 Col:- 13

Step	Instruction	Address	Comment	Octal	Step
00	*ENTRY		CONVERT TO BINARY & TEST LIMITS	← BA →	00
01	JSR	I2 1731	LDA A reg. of last OUB call		01
02	P=3773-				02
03	ANZ		End of Input string reached?		03
04	JUMP	I 1317	Yes - take input field as zero.		04
05	INCB				05
06	LDA	I2 3	3774- → Next Position in Input buffer?		06
07	STA	I 1311	P ₁		07
10	JSR	I2 1762	Convert to Binary		10
11	P ₁ =				11
12	P ₂ =0/0177		Watermark		12
13	P ₃ =0,1,0				13
14	APOS				14
15	JUMP	I2 1641	Error		15
16	LDA	Z 0177			16
17	CMPA	I 1300	Minimum * test limits		17
20	JUMP	I 1323			20
21	SHGT				21
22	JUMP	I2 1641	Error < Minimum		22
23	INVSZ	I 1300			23
24	CMPA	I 1300	Maximum		24
25	JUMP	I 1330			25
26	SHGT				26
27	JUMP	I2 1641	Error > Maximum		27
30	INVSZ	I 1300			30
31	JUMP	I 1300	Return (A contains binary number)		31
32	*ENTRY		PACK DATE	← BA →	32
33	JSR	I2 1606	Convert Day to Binary		33
34	P=1		Min		34
35	B=31		Max		35
36	STA	Z 0176			36
37	JSR	I2 1606	Convert Month to Binary		37
40	P=1		Min		40
41	B=12		Max		41
42	SWAPA				42
43	RSA		} Move field to bits 9-6.		43
44	RSA				44
45	RSA				45
46	JORA	Z 0176	Merge with Day		46
47	STA	Z 0176			47
50	JSR	I2 1606	Convert Year to Binary		50
51	P=10				51
52	B=99				52
53	SWAPA		} Move field to bits 16-10		53
54	LSA				54
55	JORA	Z 0176	Merge with Month and Day		55
56	JUMP	I 1332	Return.		56
57	*ENTRY		OCTAL → ASCII CONVERSION	← BA →	57
60	STA	Z 0177	Source Octal Digit		60
61	LDB	I 1357	→ Target String		61
62	JSR	I2 1414	Negative Overflow Byte		62
63	WOOP				63
64	RDB	Z 0205	CF5 → End of Target String		64
65	STB	Z 0176	Target x 2		65
66	INVSZ	I 1357			66
67	LDA	Z 0206	CF6		67
70	STA	Z 0175	Counter		70
71	LDA	Z 0177	*NEXT Source Digit		71
72	LDB	Z 0177			72
73	BPOS/CAC				73
74	COMPC				74
75	PRBC				75
76	RSB				76
77	RSB				77

OS

Page:- 03 Col:- 14

Step	Instruction	Address	Comment	Octal	Step
00	STB	Z 0177			00
01	ANDA	Z 0207	000007 Bottom Digit	0207	01
02	JORA	Z 0260	000060 Count to ASCII		02
03	LDB	Z 0176	Format x2 *STORE CHAR in TRASET		03
04	DESZ	Z 0176			04
05	JSBR	IZ 1416	Store Absolute Byte		05
06	DESZ	Z 0175	Counter		06
07	JUMP	I 1371	Auto next digit		07
10	JUMP	I 1357	Return		10
11					11
12					12
13					13
14					14
15					15
16					16
17					17
20					20
21	JUMP	I 1357	Return		21
22	* ENTRY		SPECIFY PROGRAM ESCAPE POINT	← BA →	22
23	LDB	Z 0074	→ Info Control Area		23
24	ADB	Z 0244	CF36		24
25	LDA	IZ B	= Previous Escape Point		25
26	STA	Z 0177	Save		26
27	LDA	I 1422	= New Escape Point		27
30	JSBR	Z 1630	Resolves Offset		30
31	STA	IZ B	Info Control Area		31
32	LDA	Z 0177			32
33	INSZ	IZ 1422			33
34	JUMP	I 1422	Return		34
35	JSBR	Z 1630	Resolves Offset		35
36	JUMP	I 1435	→ Segment		36
37	* ENTRY		COMPLEMENT DOUBLE WORD	← BA →	37
40	LDA	I 1437	→ Binary No. (Word)		40
41	INSZ	IZ 1437			41
42	JSBR	Z 1630	Resolves Offset		42
43	LDB	IZ A	} 1st Word		43
44	COMPG				
45	STB	IZ A			45
46	INCA		→ 2nd word		46
47	LDB	IZ A			47
50	BNP				50
51	JUMP	I 1456			51
52	JSBR	Z 1570	COMPG/CASB		52
53	INCB				53
54	STB	IZ A			54
55	JUMP	I 1437	Return		55
56	DECA		→ 1st word		56
57	INSZ	IZ A			57
60	NOOP				60
61	JUMP	I 1437	Return		61
62	* ENTRY		RESTORE	← BA →	62
63	STA	I 1435	= length required (words)		63
64	ADB	Z 0240	0/00 to		64
65	DECB		* NEXT SEGMENT		65
66	STB	I 1436	→ Previous Segment		66
67	LDB	I 1436	→ Current Segment		67
70	BNP				70
71	JUMP	I 1565	→ Insufficient core available		71
72	INCB				72
73	LDA	IZ B	= length of segment		73
74	SFA	I 1435	length required		74
75	A-POS				75
76	JUMP	I 1465	this segment has finished try next		76
77	STA	IZ B	= length remaining		77

OS

Page:- 03 Col:- 15

Step	Instruction	Address	Comment	Octal	Step
00	CHPA	Z 0202	CFZ (Minimum Segment Size)		00
01	JUMP	1504			01
02	SKBT				02
03	JUMP	1506	Remove segment from free chain.		03
04	ADA	I 1436			04
05	JUMP	I 1462	Return.		05
06	-DECB				06
07	LDB	I2 B	→ next segment		07
10	LDA	I 1436			10
11	STB	I 1436			11
12	JUMP	I 1462	Return.		12
13	*ENTRY		BREAKPOINT	← BA →	13
14	STA	Z 0175			14
15	STB	Z 0176			15
16	SKINC/CA				16
17	INCA				17
20	SKINGT				20
21	ADA	Z 0204	CFZ		21
22	STA	Z 0177			22
23	JSBR	IL 1721	LDA Control Test No.		23
24	P=3731-				24
25	A=0		Under Breakpoint Control?		25
26	JUMP	1531	Yes-		26
27	LDA	1513			27
30	JUMP	Z 1366	HALT - End of Control		30
31	ADA	Z 0047	+Test Control Table		31
32	LDB	I2 A	→ TCA of Control Table		32
33	SFB	Z 0343	3720 (→) End of Control Portion		33
34	LDB	I2 B	→ Breakpoint 152.		34
35	JUMP	I2 B			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	LDA	1462	Return address (from 1471)		65
66	HALT		INSUFFICIENT FREECORE for RETURN		66
67	JUMP	Z 1365	HALT - User's Unfinished End		67
70	*ENTRY		Dr. Mem Program 172.	← BA →	70
71	COMVB				71
72	CRSB				72
73	JUMP	I 1570	Return.		73
74					74
75			Line Feed Buffer	000415	75
76			Vertical Tab Buffer	021015	76
77			Form Feed Buffer	020015	77

OS - NAME & ADDRESS PROCESSOR

Page:- 03 Col:- 16

Step	Instruction	Address	Comment	Octal	Step
00	*ENTRY			← BA →	00
01	LDA	1600			01
02	JSBR	I2 1725	STA Save return address in I/O Counter.		02
03	P ₁ = 3776-				03
04	CIA				04
05	INCB				05
06	STA	I2 B	Clear exit counter.		06
07	LDA	I 1600	=P ₁ #Trickily Works		07
10	JSBR	Z 1630	Resolve		10
11	STA	Z 0177			11
12	LDB	Z 0177	→ Next parameter block. * Next Parameter blocks.		12
13	BNZ				13
14	JUMP	1627	End of list		14
15	→ LDA	I2 B	→ next parameter block		15
16	JSBR	Z 1630	Resolve		16
17	STA	Z 0177			17
20	LDA	Z B	→ W ₁		20
21	INVCB		→ W ₂		21
22	ADA	Z 0204	→ W ₅		22
23	LDB	I2 B	=W ₂ (→ Source Address)		23
24	JSBR	I2 1414	Resolve Absolute Byte		24
25	STB	I2 A	=W ₂ block = Workarea = Source x 2.		25
26	JUMP	1612	Outs next parameter block.		26
27	JSBR	I2 1721	LDA * NEXT LINE.		27
30	P ₁ = 3776-				30
31	STA	1600	= return address		31
32	LDA	I 1600	=P ₁		32
33	STA	Z 0170	→ next parameter block		33
34	INSZ	1600			34
35	LDA	I 1600	=P ₂		35
36	JSBR	Z 1630	Resolve		36
37	STA	Z 0171	→ Exit Routine		37
40	INSZ	1600			40
41	LDA	I 1600	=P ₃ (No. of lines)		41
42	INSZ	1600			42
43	INVCB		→ Exit Counter (3777-		43
44	CMPI	I2 B			44
45	JUMP	I 1600	Return - all lines processed		45
46	→ INSZ	I2 B	Exit Counter.		46
47	LDA	Z 0170	→ Next parameter block & next address		47
50	AND				50
51	JUMP	1657	End of parameter blocks		51
52	→ JSBR	Z 1630	Resolve		52
53	LDB	I2 A			53
54	STB	Z 0170	→ next parameter block.		54
55	JSBR	1663	Set up address line.		55
56	JUMP	1647	Outs next address.		56
57	JSBR	I2 1721	LDA Exit Counter		57
60	P ₁ = 3777-				60
61	JSBR	I2 0171	Exit Routine		61
62	JUMP	1627	Outs next line.		62
63	*ENTRY		SET-UP ADDRESS LINE	← BA →	63
64	ADA	Z 0202	CF2		64
65	STA	Z 0175	Parameter pointer		65
66	LDB	I2 A	=W ₃		66
67	JSBR	I2 1414	Resolve Absolute Byte		67
70	NOOP				70
71	STB	Z 0176	Target x2		71
72	INSZ	Z 0175			72
73	LDA	I2 0175	=P ₄ (Target length, characters)		73
74	STA	Z 0177	Counter		74
75	INSZ	Z 0175	→ P ₅ (Source x2)		75
76	LDB	I2 0175	Source x2 * NEXT Source Char.		76
77	BNZ				77

Programmer:-

OS - NAME ADDRESS PROCESSOR

Page:- 03 Col:-17

Step	Instruction	Address	Comment	Octal	Step
00	JUMP	I 1663	End of line (No source string left)		00
01	INSZ	I2 0175	Source x2		01
02	JUMP	1706			02
03					03
04					04
05					05
06	JSBR	I2 1415	Load Absolute Byte		06
07	AEQ				07
10	JUMP	1713			10
11	STA	I2 0175	Source x2		11
12	JUMP	I 1663	End of line		12
13	CPMA	Z 0215	"MUL CR"		13
14	JUMP	I 1663	End of line		14
15	LDB	Z 0176	Target x2 *NEXT Target dist.		15
16	BNB				16
17	JUMP	1676	bypass (count exhausted)		17
20	INSZ	Z 0176	Target x2		20
21	JSBR	I2 1416	Store Absolute Byte		21
22	DESL	Z 0177	Counter		22
23	JUMP	1676	Auto next same dist.		23
24	CHR				24
25	STB	Z 0176	Target x2 (critical target is full)		25
26	JUMP	1676	auto next same dist.		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			CR S0		47
50			BEL T		50
51			A S		51
52			K SP		52
53					53
54			Task No.		54
55					55
56			SP A		56
57			W A		57
60			I T		60
61			S SP		61
62			D I		62
63			S C		63
64			SP		64
65			Proc No.		65
66					66
67			SP		67
70			A T		70
71			SP		71
72					72
73			Block Address		73
74					74
75					75
76			SI SP		76
77			WHL		77