

PPPPPPPPPPPP	RRRRRRRRRRR	TTTTTTTTTTTT	SSSSSSSSSSSS	MMM	MMM	BBBBBBBBBBBB			
PPPPPPPPPPPP	RRRRRRRRRRR	TTTTTTTTTTTT	SSSSSSSSSSSS	MMM	MMM	BBBBBBBBBBBB			
PPPPPPPPPPPP	RRRRRRRRRRR	TTTTTTTTTTTT	SSSSSSSSSSSS	MMM	MMM	BBBBBBBBBBBB			
PPP	PPP	RRR	RRR	TTT	SSS	MMM MMM	BBBB	BBB	
PPP	PPP	RRR	RRR	TTT	SSS	MMMMMM	MMMMMM	BBBB	BBB
PPP	PPP	RRR	RRR	TTT	SSS	MMMMMM	MMMMMM	BBBB	BBB
PPP	PPP	RRR	RRR	TTT	SSS	MMMMMM	MMMMMM	BBBB	BBB
PPP	PPP	RRR	RRR	TTT	SSS	MMMMMM	MMMMMM	BBBB	BBB
PPP	PPP	RRR	RRR	TTT	SSS	MMMMMM	MMMMMM	BBBB	BBB
PPPPPPPPPPPP	RRRRRRRRRRR	TTTTTTTTTTTT	SSSSSSSSSSSS	MMM	MMM	BBBBBBBBBBBB	BBB	BBB	
PPPPPPPPPPPP	RRRRRRRRRRR	TTTTTTTTTTTT	SSSSSSSSSSSS	MMM	MMM	BBBBBBBBBBBB	BBB	BBB	
PPPPPPPPPPPP	RRRRRRRRRRR	TTTTTTTTTTTT	SSSSSSSSSSSS	MMM	MMM	BBBBBBBBBBBB	BBB	BBB	
PPP	RRR	RRR	RRR	TTT	SSS	MMMM	MMM	BBBB	BBB
PPP	RRR	RRR	RRR	TTT	SSS	MMMM	MMM	BBBB	BBB
PPP	RRR	RRR	RRR	TTT	SSS	MMMM	MMM	BBBB	BBB
PPP	RRR	RRR	RRR	TTT	SSS	MMMM	MMM	BBBB	BBB
PPP	RRR	RRR	RRR	TTT	SSS	MMMM	MMM	BBBB	BBB
PPP	RRR	RRR	RRR	TTT	SSS	MMMM	MMM	BBBB	BBB
PPP	RRR	RRR	RRR	TTT	SSS	MMMM	MMM	BBBB	BBB
PPP	RRR	RRR	RRR	TTT	SSS	MMMM	MMM	BBBB	BBB
PPP	RRR	RRR	RRR	TTT	SSS	MMMM	MMM	BBBB	BBB
PPPPPPPPPPPP	RRRRRRRRRRR	TTTTTTTTTTTT	SSSSSSSSSSSS	MMM	MMM	BBBBBBBBBBBB	BBB	BBB	
PPPPPPPPPPPP	RRRRRRRRRRR	TTTTTTTTTTTT	SSSSSSSSSSSS	MMM	MMM	BBBBBBBBBBBB	BBB	BBB	
PPPPPPPPPPPP	RRRRRRRRRRR	TTTTTTTTTTTT	SSSSSSSSSSSS	MMM	MMM	BBBBBBBBBBBB	BBB	BBB	



```

000000  UU  UU  TTTTTTTTTT  PPPPPPPP  UU  UU  TTTTTTTTTT
000000  UU  UU  TTTTTTTTTT  PPPPPPPP  UU  UU  TTTTTTTTTT
00  00  UU  UU  TT  PP  PP  UU  UU  TT
00  00  UU  UU  TT  PP  PP  UU  UU  TT
00  00  UU  UU  TT  PP  PP  UU  UU  TT
00  00  UU  UU  TT  PP  PP  UU  UU  TT
00  00  UU  UU  TT  PPPPPPPP  UU  UU  TT
00  00  UU  UU  TT  PPPPPPPP  UU  UU  TT
00  00  UU  UU  TT  PP  UU  UU  TT
00  00  UU  UU  TT  PP  UU  UU  TT
00  00  UU  UU  TT  PP  UU  UU  TT
00  00  UU  UU  TT  PP  UU  UU  TT
00  00  UU  UU  TT  PP  UU  UU  TT
00  00  UU  UU  TT  PP  UU  UU  TT
000000  UUUUUUUUUU  TT  PP  UUUUUUUUUU  TT  ....
000000  UUUUUUUUUU  TT  PP  UUUUUUUUUU  TT  ....

```

```

LL  IIIIII  SSSSSSSS
LL  IIIIII  SSSSSSSS
.L  II  SS
LL  II  SS
LL  II  SS
LL  II  SS
LL  II  SSSSSS
LL  II  SSSSSS
LL  II  SS
LL  II  SS
LL  II  SS
LL  II  SS
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS

```

```

1 0001 0 MODULE OUTPUT ( %TITLE 'Print Symbiont -- output routines'
2 0002 0 IDENT = 'V04-000',
3 0003 0 ADDRESSING_MODE (EXTERNAL = GENERAL)
4 0004 0 ) =
5 0005 1 BEGIN
6 0006 1
7 0007 1
8 0008 1 *****
9 0009 1 *
10 0010 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY *
11 0011 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS. *
12 0012 1 * ALL RIGHTS RESERVED. *
13 0013 1 *
14 0014 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
15 0015 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE *
16 0016 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER *
17 0017 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
18 0018 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY *
19 0019 1 * TRANSFERRED. *
20 0020 1 *
21 0021 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE *
22 0022 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT *
23 0023 1 * CORPORATION. *
24 0024 1 *
25 0025 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS *
26 0026 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL. *
27 0027 1 *
28 0028 1 *
29 0029 1 *****
30 0030 1
31 0031 1
32 0032 1 **
33 0033 1 FACILITY:
34 0034 1 Print Symbiont.
35 0035 1
36 0036 1 ABSTRACT:
37 0037 1 Output routines.
38 0038 1
39 0039 1 ENVIRONMENT:
40 0040 1 VAX/VMS user mode.
41 0041 1 --
42 0042 1
43 0043 1 AUTHOR: G. Robert, CREATION DATE: 31-Aug-1982
44 0044 1
45 0045 1 MODIFIED BY:
46 0046 1
47 0047 1 3B-006 RRB3006 Rowland R. Bradley 24-Jul-1984
48 0048 1 Add routine headers.
49 0049 1
50 0050 1 3B-005 GRR3005 Gregory R. Robert 29-Apr-1984
51 0051 1 FT2 bugfixes. Accept explicit OPEN/CLOSE calls instead
52 0052 1 of relying upon START_TASK.
53 0053 1
54 0054 1 3B-004 RRB3004 Rowland R. Bradley 27-Apr-1984
55 0055 1 Add support for storage, initialization, and ultimate
56 0056 1 display of the entire device name.
57 0057 1

```

; R

58
59
60
61
62
63
64
65
66
67
68
69
70

0058 1
0059 1
0060 1
0061 1
0062 1
0063 1
0064 1
0065 1
0066 1
0067 1
0068 1
0069 1
0070 1 **

3B-003	GRR3003	Gregory R. Robert	23-Aug-1983
	Bugfixes, page_setup_modules, form_setup_modules,		
	sheet_feed, symbiont-initiated pause_task and stop_stream,		
	hanguap code, read and write item services		
3B-002	GRR3002	Gregory R. Robert	03-Aug-1983
	Rewrite for new design.		
3B-001	GRR3001	Gregory R. Robert	23-Jul-1983
	Created new module.		

```
: 72      0071 1 LIBRARY 'SYSS$LIBRARY:LIB';  
: 73      0072 1 REQUIRE 'LIB$:SMBDEF';  
: 74      0564 1 REQUIRE 'SRC$:SMBREQ';  
: 75      1021 1  
: 76      1022 1 EXTERNAL ROUTINE  
: 77      1023 1         PSM$REPORT,  
: 78      1024 1         SMB$SEND TO JOBCTL,  
: 79      1025 1         PSM$STORE_ERRORS  
: 80      1026 1         :  
: 81      1027 1  
: 82      1028 1 EXTERNAL  
: 83      1029 1         PSM$HANGUP_DISPATCH_TABLE  
: 84      1030 1         :  
: 85      1031 1  
: 86      1032 1  
: 87      1033 1 FORWARD ROUTINE  
: 88      1034 1         PSM$OUTPUT      :  
: 89      1035 1         START_PRINTER  :  
: 90      1036 1         STOP_PRINTER   :  
: 91      1037 1         WRITE_AST      : NOVALUE  
: 92      1038 1         :  
: 93      1039 1
```

: R

```

95 1040 1 %SBTTL 'PSMS$OUTPUT'
96 1041 1 Functional Description:
97 1042 1 ?desc
98 1043 1
99 1044 1 Formal Parameters:
100 1045 1 ?desc
101 1046 1 Implicit Inputs:
102 1047 1 none
103 1048 1
104 1049 1 Implicit Outputs:
105 1050 1 none
106 1051 1
107 1052 1 Returned Value:
108 1053 1 none
109 1054 1
110 1055 1 Side Effects:
111 1056 1 none
112 1057 1 --
113 1058 1 GLOBAL ROUTINE PSMS$OUTPUT (
114 1059 1     SMB_CONTEXT      : REF $LONGWORD,
115 1060 1     USER_CONTEXT    : REF VECTOR,
116 1061 1     FUNCTION        : REF $LONGWORD,
117 1062 1     FUNC_DESC       : REF VECTOR,
118 1063 1     FUNC_ARG        : REF $LONGWORD
119 1064 1 ) =
120 1065 2 BEGIN
121 1066 2
122 1067 2 SELECTONEU .FUNCTION[] OF
123 1068 2 SET
124 1069 2
125 1070 2 [PSMSK_WRITE, PSMSK_WRITE_NOFORMAT]:
126 1071 3 BEGIN
127 1072 3 LOCAL IO_FUNC : INITIAL (IOS_WRITELBLK);
128 1073 3 LOCAL SIZE, ADDRESS;
129 1074 3
130 1075 3 BIND IOB = .SMB_CONTEXT[] : $BLOCK;
131 1076 3 BIND SCB = .IOB[IOB_A_CONTEXT] : $BLOCK;
132 1077 3
133 1078 3 ! If device was hungup then mark it unavailable and request stop stream
134 1079 3 !
135 1080 3 IF TESTBITSC (SCB[PSMSV_HANGUP])
136 1081 3 THEN
137 1082 4 BEGIN
138 1083 4     SCB[PSMSL_DEVICE_STATUS] = .SCB[PSMSL_DEVICE_STATUS] OR
139 1084 4     (SMBMSGM_UNAVAILABLE OR SMBMSGM_STOP_STREAM);
140 1085 4     RETURN SSS_HANGUP;
141 1086 3     END;
142 1087 3
143 1088 3 IF .FUNCTION[] EQL PSMSK_WRITE_NOFORMAT
144 1089 3 THEN
145 1090 3     IF .SCB[PSMSL_DEVICE_CLASS] EQL DC$TERM
146 1091 3     THEN
147 1092 3         IO_FUNC = IOS_WRITELBLK OR IOSM_NOFORMAT
148 1093 3     ELSE
149 1094 3         IO_FUNC = IOS_WRITEPBLK;
150 1095 3
151 1096 3 STR$ANALYZE_SDESC_R1 (.FUNC_DESC; SIZE, ADDRESS);

```

```

: 152 1097
: 153 P 1098          RETURN IF_ERROR_ ($QIO (
: 154 P P 1099          FUNC          = .IO_FUNC,
: 155 P P 1100          CHAN           = .SCB[PSM$L_CHANNEL],
: 156 P P 1101          IOSB           = IOB[IOB_Q_IOSB],
: 157 P P 1102          ASTADR          = WRITE_AST,
: 158 P P 1103          ASTPRM          = IOB,
: 159 P 1104          P1              = .ADDRESS,
: 160 1105          P2              = .SIZE)),
: 161 1106          END;
: 162 1107
: 163 [PSM$K START_TASK]:
: 164 1109          BEGIN
: 165 1110          BIND SCB = .SMB_CONTEXT[] : $BBLOCK;
: 166 1111          LOCAL BUFFER : $BBLOCK [8];
: 167 1112          LOCAL DEVBUFSIZ : VOLATILE;
: 168 1113          LOCAL ITEM_LIST : $ITMLST_DECL (ITEMS=2);
: 169 1114
: 170 P 1115          $ITMLST_INIT (ITMLST=ITEM_LIST,
: 171 P 1116          (BUFADR=BUFFER +4, ITMCD=DVIS_DEVDEPEND OR DVIS$C_SECONDARY),
: 172 1117          (BUFADR=DEVBUFSIZ, ITMCD=DVIS_DEVBUFSIZ OR DVIS$C_SECONDARY));
: 173 1118
: 174 P 1119          RETURN IF_ERROR_ ($GETDVIW
: 175 1120          (CHAN=.SCB[PSM$L_CHANNEL], ITMLST=ITEM_LIST));
: 176 1121
: 177 1122          ! Set the page length so the driver can compute number of line
: 178 1123          ! feeds for devices lacking mechanical form feed -- preserve the
: 179 1124          ! class type and width information.
: 180 1125
: 181 1126          BUFFER [0,0,08,0] = .SCB[PSM$L_DEVICE_CLASS];
: 182 1127          BUFFER [1,0,08,0] = .SCB[PSM$L_DEVICE_TYPE];
: 183 1128          BUFFER [2,0,16,0] = .DEVBUFSIZ;
: 184 1129          BUFFER [7,0,08,0] = .SCB[PSM$L_FORM_LENGTH];
: 185 1130
: 186 P 1131          RETURN IF_ERROR_ ($QIOW
: 187 1132          (FONC=IOS_SETMODE, CHAN=.SCB[PSM$L_CHANNEL], P1=BUFFER, P2=8));
: 188 1133
: 189 1134          RETURN SS$_NORMAL;
: 190 1135          END;
: 191 1136
: 192 [PSM$K CLOSE]:
: 193 1137          BEGIN
: 194 1138          BIND SCB = .SMB_CONTEXT[] : $BBLOCK;
: 195 1139          RETURN STOP_PRINTER (SCB[PSM$L_CHANNEL], SCB[PSM$Q_DEVICE_NAME]);
: 196 1140          END;
: 197 1141
: 198 [PSM$K OPEN]:
: 199 1142          BEGIN
: 200 1143          BIND SCB = .SMB_CONTEXT[] : $BBLOCK;
: 201 1144          LOCAL DEVDEPEND : $BBLOCK[8];
: 202 1145
: 203 P 1146          RETURN IF_ERROR_ (START_PRINTER (
: 204 P 1147          .FUNC_DESC,           ! - specified device name
: 205 P 1148          .FUNC_ARG,           ! - resultant device status
: 206 P 1149          SCB[PSM$L_CHANNEL],    ! - channel assigned
: 207 P 1150          SCB[PSM$L_DEVICE_CLASS], ! - device class
: 208 P 1151          SCB[PSM$L_DEVICE_TYPE], ! - device type

```

```

: 209 P 1154 3 .FUNC DESC, ! - resultant device name
: 210 P 1155 3 DEVDEPEND ! - device dependent bits
: 211 1156 3 )));
: 212 1157 3
: 213 1158 3 IF .SCB[PSMS$L_DEVICE_CLASS] EQL DC$_TERM
: 214 1159 3 THEN
: 215 1160 4 BEGIN
: 216 1161 4 IF NOT .DEVDEPEND[TT$V_MECHTAB]
: 217 1162 4 THEN
: 218 1163 4 SCB[PSMS$V_EXPAND_TABS] = 1;
: 219 1164 4
: 220 P 1165 4 RETURN IF_ERROR ($QIO (
: 221 P 1166 4 FUNC = IO$ SETMODE OR IOSM_CTRLFAST,
: 222 P 1167 4 CHAN = .SCB[PSMS$L_CHANNEL],
: 223 P 1168 4 P1 = PSMS$HANGUP_DISPATCH_TABLE +
: 224 P 1169 4 (.SCB[PSMS$L_STREAM_INDEX] *
: 225 P 1170 4 PSMS$HANGUP_DISPATCH_ENTRY),
: 226 1171 4 P2 = 0));
: 227 1172 4 END
: 228 1173 3 ELSE
: 229 1174 3 IF .SCB[PSMS$L_DEVICE_TYPE] EQLU DT$_LP11
: 230 1175 4 AND (NOT .DEVDEPEND[CP$V_TAB])
: 231 1176 3 THEN
: 232 1177 3 SCB[PSMS$V_EXPAND_TABS] = 1;
: 233 1178 3
: 234 1179 3 RETURN SS$_NORMAL;
: 235 1180 2 END;
: 236 1181 2 [PSMSK_CANCEL]:
: 237 1182 2 BEGIN
: 238 1183 3 BIND SCB = .SMB_CONTEXT[] : $BBLOCK;
: 239 1184 3 RETURN_IF_ERROR_ ($CANCEL (CHAN=.SCB[PSMS$L_CHANNEL]));
: 240 1185 3 END;
: 241 1186 2
: 242 1187 2 [OTHERWISE]:
: 243 1188 2 RETURN PSMS_FUNNOTSUP;
: 244 1189 2
: 245 1190 2 TES;
: 246 1191 2
: 247 1192 2 PSMS_PENDING
: 248 1193 2
: 249 1194 2
: 250 1195 1 END;

```

```

.TITLE OUTPUT Print Symbiont -- output routines
.IDENT \V04-000\

.EXTRN BASSEDT, LBR$CLOSE
.EXTRN LBR$GET RECORD, LBR$INI CONTROL
.EXTRN LBR$LOOKUP KEY, LBR$OPEN
.EXTRN LBR$RET RMSSTV, LBR$SET_LOCATE
.EXTRN LIB$TRIM FILESPEC
.EXTRN LIB$GET_VM, LIB$FREE_VM
.EXTRN STR$ANALYZE_SDESC
.EXTRN STR$ANALYZE_SDESC R1
.EXTRN STR$APPEND, STR$CONCAT
.EXTRN STR$COPY_DX, STR$COPY_R

```



```

.EXTRN STR$FREE1_DX, STR$FREE1_DX_R4
.EXTRN STR$GET1_DX, STR$LEFT
.EXTRN STR$PREFIX, STR$RIGHT
.EXTRN PSMS$ HANGUP DISPATCH ENTRY
.EXTRN PSMS$ BUFFEROVF, PSMS$ EOF
.EXTRN PSMS$ _ESCAPE, PSMS$ _FLOSH
.EXTRN PSMS$ _FUNNOTSUP, PSMS$ _INVITMCOB
.EXTRN PSMS$ _INVVMSOSC, PSMS$ _MODNOTFND
.EXTRN PSMS$ _NEWPAGE, PSMS$ _NOFILEID
.EXTRN PSMS$ _OSCTOOLON, PSMS$ _PENDING
.EXTRN PSMS$ _SUSPEND, PSMS$ _TOOMANYLEV
.EXTRN SMBS$ _INVSTMNBR, SMBS$ _INVSTRLEV
.EXTRN SMBS$ _NOMOREITEMS
.EXTRN PSMS$REPORT, SMBS$SEND_TO_JOBCTL
.EXTRN PSMS$STORE ERRORS
.EXTRN PSMS$HANGUP DISPATCH TABLE
.EXTRN SYSS$QIO, SYSS$GETDVIQ
.EXTRN SYSS$QIOW, SYSS$CANCEL

```

.PSECT CODE, NOWRT, 2

```

.ENTRY PSMS$OUTPUT, Save R2,R3,R4,R5,R6,R7,R8,R9,- R10,R11 : 1058
MOVAB SYSS$QIO, R6
SUBL2 #40, SP
MOVL @FUNCTION, R3 : 1067
CMPL R3, #9 : 1070
BLSSU 4$
CMPL R3, #10
BGTRU 4$
MOVL #32, IO_FUNC : 1071
MOVL @SMB_CONTEXT, R4 : 1075
MOVL 20(R4), R2 : 1076
BBCC #2, 12(R2), 1$ : 1080
BISB2 #160, 84(R2) : 1084
MOVZWL #716, R0 : 1085
RET
CMPL R3, #10 : 1088
BNEQ 3$
CMPL 416(R2), #66 : 1090
BNEQ 2$
MOVZWL #288, IO_FUNC : 1092
BRB 3$
MOVL #11, IO_FUNC : 1094
MOVL FUNC_DESC, R0 : 1096
JSB STR$ANALYZE_SDESC_R1
CLRQ -(SP) : 1105
CLRQ -(SP)
PUSHL SIZE
PUSHL ADDRESS
PUSHL R4
PUSHAB WRITE_AST
PUSHAB 12(R4)
PUSHL IO_FUNC
PUSHL 396(R2)
CLRL -(SP)
CALLS #12, SYSS$QIO

```

```

OFFC 00000
56 00000000G 00 9E 00002
5E 28 C2 00009
53 0C BC D0 0000C
09 53 D1 00010
63 1F 00013
0A 53 D1 00015
5E 1A 00018
55 20 D0 0001A
54 04 BC D0 0001D
52 14 A4 D0 00021
OB 0C A2 02 E5 00025
54 54 A2 A0 8F 88 0002A
50 02CC 8F 3C 0002F
04 00034
0A 53 D1 00035 1$:
15 12 00038
00000042 8F 01A0 C2 D1 0003A
07 12 00043
55 0120 8F 3C 00045
03 11 0004A
55 0B D0 0004C 2$:
50 10 AC D0 0004F 3$:
00000000G 00 16 00053
7E 7C 00059
7E 7C 0005B
50 DD 0005D
51 DD 0005F
54 DD 00061
0000V CF 9F 00063
0C A4 9F 00067
55 DD 0006A
018C C2 DD 0006C
7E D4 00070
66 0C FB 00072

```

			0122	31	00075		BRW	13\$			
	10		53	D1	00078	4\$:	CMPL	R3, #16		1108	
			72	12	0007B		BNEQ	6\$			
	52	04	BC	D0	0007D		MOVL	@SMB_CONTEXT, R2		1110	
	50		6E	9E	00081		MOVAB	ITEM_LIST, \$\$ITMBLKPTR		1117	
	80	000B0004	8F	D0	00084		MOVL	#720900, (\$\$ITMBLKPTR)+			
	80		AE	9E	00088		MOVAB	BUFFER+4, (\$\$ITMBLKPTR)+			
			80	D4	0008F		CLRL	(\$\$ITMBLKPTR)+			
	80	00090004	8F	D0	00091		MOVL	#589828, (\$\$ITMBLKPTR)+			
	80		AE	9E	00098		MOVAB	DEVBUFSIZ, (\$\$ITMBLKPTR)+			
			80	7C	0C09C		CLRQ	(\$\$ITMBLKPTR)+			
			7E	7C	0009E		CLRQ	-(SP)		1120	
			7E	7C	000A0		CLRQ	-(SP)			
			AE	9F	000A2		PUSHAB	ITEM_LIST			
			7E	D4	000A5		CLRL	-(SP)			
		018C	C2	DD	000A7		PUSHL	396(R2)			
			7E	D4	000AB		CLRL	-(SP)			
00000000G	00		08	FB	000AD		CALLS	#8, SYSS\$GETDVIW			
	79		50	E9	000B4		BLBC	STATUS, 8\$			
	20	AE	01A0	C2	90	000B7	MOVB	416(R2), BUFFER		1126	
	21	AE	01A4	C2	90	000BD	MOVB	420(R2), BUFFER+1		1127	
	22	AE	1C	AE	80	000C3	MOVW	DEVBUFSIZ, BUFFER+2		1128	
	27	AE	78	A2	90	000C8	MOVB	120(R2), BUFFER+7		1129	
			7E	7C	000CD		CLRQ	-(SP)		1132	
			7E	7C	000CF		CLRQ	-(SP)			
			08	DD	000D1		PUSHL	#8			
			AE	9F	000D3		PUSHAB	BUFFER			
			7E	7C	000D6		CLRQ	-(SP)			
	7E		23	7D	000D8		MOVQ	#35, -(SP)			
		018C	C2	DD	000DB		PUSHL	396(R2)			
			7E	D4	000DF		CLRL	-(SP)			
00000000G	00		0C	FB	000E1		CALLS	#12, SYSS\$QIOW			
	03		50	E9	000E8		BLBC	STATUS, 5\$			
			0094	31	000EB		BRW	11\$			
			04	000EE	5\$:	RET				1134	
	02		53	D1	000EF	6\$:	CMPL	R3, #2		1137	
			18	12	000F2		BNEQ	7\$			
7E	04	BC	0000004C	8F	C1	000F4	ADDL3	#76, @SMB_CONTEXT, -(SP)		1140	
7E	04	BC	0000018C	8F	C1	000FD	ADDL3	#396, @SMB_CONTEXT, -(SP)			
	0000V	CF		02	FB	00106	CALLS	#2, STOP_PRINTER			
				04	0010B		RET				
	04		53	D1	0010C	7\$:	CMPL	R3, #4		1143	
			75	12	0010F		BNEQ	12\$			
	52	04	BC	D0	00111		MOVL	@SMB_CONTEXT, R2		1145	
		20	AE	9F	00115		PUSHAB	DEVDEPEND		1156	
		10	AC	DD	00118		PUSHL	FUNC_DESC			
		01A4	C2	9F	0011B		PUSHAB	420(R2)			
		01A0	C2	9F	0011F		PUSHAB	416(R2)			
		018C	C2	9F	00123		PUSHAB	396(R2)			
	7E	10	AC	7D	00127		MOVQ	FUNC_DESC, -(SP)			
0000V	CF		07	FB	0012B		CALLS	#7, START_PRINTER			
	7A		50	E9	00130	8\$:	BLBC	STATUS, 18\$			
00000042	8F	01A0	C2	D1	00133		CMPL	416(R2), #66		1158	
			34	12	0013C		BNEQ	10\$			
	04	21	AE	E8	0013E		BLBS	DEVDEPEND+1, 9\$		1161	
	0C	A2	02	88	00142		BISB2	#2, 12(R2)		1163	
			7E	7C	00146	9\$:	CLRQ	-(SP)		1171	

50	022C	C2	00000000G	7E	7C	00148	CLRQ	-(SP)		
				7E	D4	0014A	CLRL	-(SP)		
				8F	75	0014C	MULL3	#PSM\$S HANGUP DISPATCH ENTRY, 556(R2), R0		
			00000000G0040	9F	9F	00156	PUSHAB	PSM\$HANGUP_DISPATCH_TABLE[R0]		
				7E	7C	0015D	CLRQ	-(SP)		
				7E	D4	0015F	CLRL	-(SP)		
		7E	A3	8F	9A	00161	MOVZBL	#163, -(SP)		
			018C	C2	DD	00165	PUSHL	396(R2)		
				7E	D4	00169	CLRL	-(SP)		
		66		CC	FB	0016B	CALLS	#12, SYSS\$QIO		
		11		50	E8	0016E	BLBS	STATUS, 11\$		
					04	00171	RET			
		01	01A4	C2	D1	00172	10\$:	C MPL	420(R2), #1	1174
				09	12	00177		BNEQ	11\$	
04	20	AE		05	E0	00179		BBS	#5, DEVDEPEND, 11\$	1175
	0C	A2		02	88	0017E		BISB2	#2, 12(R2)	1177
		50		01	D0	00182	11\$:	MOVL	#1, R0	1179
					04	00185		RET		
		01		53	D1	00186	12\$:	C MPL	R3, #1	1182
				13	12	00189		BNEQ	14\$	
		50	04	BC	D0	0018B		MOVL	@SMB CONTEXT, R0	1184
			018C	C0	DD	0018F		PUSHL	396(R0)	1185
		00000000G	00	01	FB	00193		CALLS	#1, SYSS\$CANCEL	
			09	50	E8	0019A	13\$:	BLBS	STATUS, 15\$	
					04	0019D		RET		
		50	00000000G	8F	D0	0019E	14\$:	MOVL	#PSM\$_FUNNOTSUP, R0	1189
					04	001A5		RET		
		50	00000000G	8F	D0	001A6	15\$:	MOVL	#PSM\$_PENDING, R0	1195
					04	001AD	16\$:	RET		

; Routine Size: 430 bytes, Routine Base: CODE + 0000

```

: 252 1196 1 %SBTTL 'START PRINTER'
: 253 1197 1 Functional Description:
: 254 1198 1 ?desc
: 255 1199 1
: 256 1200 1 Formal Parameters:
: 257 1201 1 ?desc
: 258 1202 1 Implicit Inputs:
: 259 1203 1 none
: 260 1204 1
: 261 1205 1 Implicit Outputs:
: 262 1206 1 none
: 263 1207 1
: 264 1208 1 Returned Value:
: 265 1209 1 none
: 266 1210 1
: 267 1211 1 Side Effects:
: 268 1212 1 none
: 269 1213 1 --
: 270 1214 1 ROUTINE START_PRINTER ( %SBTTL 'START_PRINTER'
: 271 1215 1 DEVICE_NAME,
: 272 1216 1 DEVICE_STATUS : REF $LONGWORD,
: 273 1217 1 CHANNEL : REF $WORD,
: 274 1218 1 DEVICE_CLASS : REF $LONGWORD,
: 275 1219 1 DEVICE_TYPE : REF $LONGWORD,
: 276 1220 1 OUT_DEVICE_NAME : ,
: 277 1221 1 OUT_DEVDEPEND : REF VECTOR
: 278 1222 1 ) =
: 279 1223 2 BEGIN
: 280 P 1224 2 PARAMETER_INDEX (DEVICE_NAME, DEVICE STATUS, CHANNEL,
: 281 1225 2 DEVICE_CLASS, DEVICE_TYPE, OUT_DEVICE_NAME, OUT_DEVDEPEND);
: 282 1226 2
: 283 1227 2 LOCAL
: 284 1228 2 CHAN : INITIAL (0),
: 285 1229 2 ITEM_LIST : $ITMLST DECL (ITEMS=4),
: 286 1230 2 DEVDEPEND : $BBLOCK[4] VOLATILE,
: 287 1231 2 DEVSTATUS : INITIAL (0),
: 288 1232 2 DEVCLASS : VOLATILE,
: 289 1233 2 DEVTYPE : VOLATILE,
: 290 1234 2 DEVNAMSIZ : INITIAL (0),
: 291 1235 2 DEVNAMBUF : VECTOR [64, BYTE] VOLATILE,
: 292 1236 2 STATUS_1
: 293 1237 2 ;
: 294 1238 2
: 295 1239 2 MAP
: 296 1240 2 DEVSTATUS : $BBLOCK;
: 297 1241 2
: 298 P 1242 2 $ITMLST INIT (ITMLST=ITEM_LIST,
: 299 P 1243 2 (BUFADR=DEVDEPEND, ITMCOD=DVIS_DEVDEPEND OR DVISC_SECONDARY),
: 300 P 1244 2 (BUFADR=DEVCLASS, ITMCOD=DVIS_DEVCLASS OR DVISC_SECONDARY),
: 301 P 1245 2 (BUFADR=DEVTYPE, ITMCOD=DVIS_DEVTYPE OR DVISC_SECONDARY),
: 302 P 1246 2 (BUFADR=DEVNAMBUF, ITMCOD=DVIS_FULLDEVNAM OR DVISC_SECONDARY,
: 303 P 1247 2 BUFSIZ=%ALLOCATION(DEVNAMBUF), RETLEN=DEVNAMSIZ)
: 304 1248 2 );
: 305 1249 2
: 306 1250 2 IF PARAMETER_PRESENT_ (CHANNEL)
: 307 1251 2 THEN
: 308 1252 2 CHANNEL[] = 0;

```

```

: 309 1253 2
: 310 1254 2 STATUS_1 = $ALLOC (DEVNAM=.DEVICE_NAME);
: 311 1255 2 IF .STATUS_1 EQL SSS_DEVALRALLOC
: 312 1256 2 THEN
: 313 1257 2 STATUS_1 = SSS_DEVALLOC;
: 314 1258 2 IF NOT .STATUS_1
: 315 1259 2 THEN
: 316 1260 2 RETURN .STATUS_1;
: 317 1261 2
: 318 1262 2 RETURN_IF_ERROR_ ($ASSIGN (DEVNAM=.DEVICE_NAME, CHAN=CHAN));
: 319 1263 2
: 320 1264 2 IF PARAMETER_PRESENT_ (CHANNEL)
: 321 1265 2 THEN
: 322 1266 2 CHANNEL[] = .CHAN;
: 323 1267 2
: 324 1268 2 RETURN_IF_ERROR_ ($GETDVIW (CHAN=.CHAN, ITMLST=ITEM_LIST));
: 325 1269 2
: 326 1270 2 IF PARAMETER_PRESENT_ (OUT_DEVDEPEND)
: 327 1271 2 THEN
: 328 1272 2 OUT_DEVDEPEND[0] = .DEVDEPEND;
: 329 1273 2
: 330 1274 2 SELECTONE .DEVCLASS OF
: 331 1275 2 SET
: 332 1276 2
: 333 1277 2 [DC$ LP]:
: 334 1278 2 IF .DEVDEPEND[LPSV_LOWER]
: 335 1279 2 THEN
: 336 1280 2 DEVSTATUS[SMBMSG$V_LOWERCASE] = 1;
: 337 1281 2
: 338 1282 2 [DC$ TERM]:
: 339 1283 2 BEGIN
: 340 1284 2 DEVSTATUS[SMBMSG$V_TERMINAL] = 1;
: 341 1285 2 IF .DEVDEPEND[TT$V_MODEM]
: 342 1286 2 THEN
: 343 1287 2 IF .DEVDEPEND[TT$V_REMOTE]
: 344 1288 2 THEN
: 345 1289 2 DEVSTATUS[SMBMSG$V_REMOTE] = 1
: 346 1290 2 ELSE
: 347 1291 2 RETURN SSS_HANGUP;
: 348 1292 2 IF .DEVDEPEND[TT$V_LOWER]
: 349 1293 2 THEN
: 350 1294 2 DEVSTATUS[SMBMSG$V_LOWERCASE] = 1;
: 351 1295 2 END;
: 352 1296 2
: 353 1297 2 [OTHERWISE]:
: 354 1298 2 BEGIN
: 355 1299 2 RETURN_IF_ERROR_ (STOP_PRINTER (CHANNEL[], .DEVICE_NAME));
: 356 1300 2 RETURN 0;          !*! Better message needed.
: 357 1301 2 END;
: 358 1302 2
: 359 1303 2 TES;
: 360 1304 2
: 361 1305 2 IF NOT PARAMETER_PRESENT_ (CHANNEL)
: 362 1306 2 THEN
: 363 1307 2 RETURN_IF_ERROR_ (STOP_PRINTER (CHANNEL[], .DEVICE_NAME));
: 364 1308 2
: 365 1309 2 IF PARAMETER_PRESENT_ (DEVICE_STATUS)

```

```

: 366 1310 2 THEN
: 367 1311 2     DEVICE_STATUS[] = .DEVSTATUS;
: 368 1312 2
: 369 1313 2 IF PARAMETER_PRESENT_ (DEVICE_CLASS)
: 370 1314 2 THEN
: 371 1315 2     DEVICE_CLASS[] = .DEVCLASS;
: 372 1316 2
: 373 1317 2 IF PARAMETER_PRESENT_ (DEVICE_TYPE)
: 374 1318 2 THEN
: 375 1319 2     DEVICE_TYPE[] = .DEVTYPE;
: 376 1320 2
: 377 1321 2 IF PARAMETER_PRESENT_ (OUT_DEVICE_NAME)
: 378 1322 2 THEN
: 379 1323 2     RETURN_IF_ERROR_ (STR$COPY_R (.OUT_DEVICE_NAME, DEVNAMSIZ, DEVNAMBUF));
: 380 1324 2
: 381 1325 2 SSS_NORMAL
: 382 1326 2
: 383 1327 1 END;

```

.EXTRN SYSSALLOC, SYSSASSIGN

0004 00000 START_PRINTER:

	5E	FF7C	CE	9E	00002	.WORD	Save R2	: 1214
			7E	D4	00007	MOVAB	-132(SP), SP	: 1223
			52	D4	00009	CLRL	CHAN	
		04	AE	D4	0000B	CLRL	DEVSTATUS	
	50	54	AE	9E	0000E	CLRL	DEVNAMSIZ	: 1248
	80	000B0004	8F	D0	00012	MOVAB	ITEM LIST, \$\$ITMBLKPTR	
	80	50	AE	9E	00019	MOVL	#720900, (\$\$ITMBLKPTR)+	
			80	D4	0001D	MOVAB	DEVDEPEND, (\$\$ITMBLKPTR)+	
	80	00050004	8F	D0	0001F	CLRL	(\$\$ITMBLKPTR)+	
	80	4C	AE	9E	00026	MOVL	#327684, (\$\$ITMBLKPTR)+	
			80	D4	0002A	MOVAB	DEVCLASS, (\$\$ITMBLKPTR)+	
	80	00070004	8F	D0	0002C	CLRL	(\$\$ITMBLKPTR)+	
	80	48	AE	9E	00033	MOVL	#458756, (\$\$ITMBLKPTR)+	
			80	D4	00037	MOVAB	DEVTYPE, (\$\$ITMBLKPTR)+	
	80	00E90040	8F	D0	00039	CLRL	(\$\$ITMBLKPTR)+	
	80	08	AE	9E	00040	MOVL	#15269952, (\$\$ITMBLKPTR)+	
	80	04	AE	9E	00044	MOVAB	DEVNAMBUF, (\$\$ITMBLKPTR)+	
			80	D4	00048	MOVAB	DEVNAMSIZ, (\$\$ITMBLKPTR)+	
	03		6C	91	0004A	CLRL	(\$\$ITMBLKPTR)+	: 1250
			08	1F	0004D	CMPB	(AP), #3	
		0C	AC	D5	0004F	BLSSU	1\$	
			03	13	00052	TSTL	12(AP)	
		0C	BC	B4	00054	BEQL	1\$: 1252
			7E	7C	00057	CLRW	@CHANNEL	: 1254
			7E	7C	00059	CLRQ	-(SP)	
		04	AC	DD	0005B	CLRQ	-(SP)	
	00000000G	00	05	FB	0005E	PUSHL	DEVICE_NAME	
	00000641	8F	50	D1	00065	CALLS	#5, SYSSALLOC	: 1255
			05	12	0006C	CPL	STATUS_1, #1601	
	50	0840	8F	3C	0006E	BNEQ	2\$: 1257
	35		50	E9	00073	MOVZWL	#2112, STATUS_1	: 1258
			7E	7C	00076	BLBC	STATUS_1, 4\$: 1262
						CLRQ	-(SP)	

			08	AE	9F	00078		PUSHAB	CHAN		
			04	AC	DD	0007B		PUSHL	DEVICE_NAME		
	00000000G	00		C4	FB	0007E		CALLS	#4, SYSS\$ASSIGN		
		79		50	E9	00085		BLBC	STATUS, 9\$		
		03		6C	91	00088		CMPB	(AP), #3		1264
				09	1F	0008B		BLSSU	3\$		
			0C	AC	D5	0008D		TSTL	12(AP)		
				04	13	00090		BEQL	3\$		
	0C	BC		6E	B0	00092		MOVW	CHAN, @CHANNEL		1266
				7E	7C	00096	3\$:	CLRQ	-(SP)		1268
				7E	7C	00098		CLRQ	-(SP)		
			64	AE	9F	0009A		PUSHAB	ITEM_LIST		
				7E	D4	0009D		CLRL	-(SP)		
			18	AE	DD	0009F		PUSHL	CHAN		
				7E	D4	000A2		CLRL	-(SP)		
	00000000G	00		08	FB	000A4		CALLS	#8, SYSS\$GETDVIW		
		6E		50	E9	000AB	4\$:	BLBC	STATUS, 12\$		
		07		6C	91	000AE		CMPB	(AP), #7		1270
				0A	1F	000B1		BLSSU	5\$		
			1C	AC	D5	000B3		TSTL	28(AP)		
				05	13	000B6		BEQL	5\$		
	1C	BC		50	AE	D0	000B8	MOVW	DEVDEPEND, @OUT_DEVDEPEND		1272
		50		4C	AE	D0	000BD	5\$:	DEVCLASS, R0		1274
	00000043	8F		50	D1	000C1		CMPL	R0, #67		1277
				22	13	000C8		BEQL	7\$		
	00000042	8F		50	D1	000CA		CMPL	R0, #66		1282
				23	12	000D1		BNEQ	8\$		
		52		40	8F	88	000D3	BISB2	#64, DEVSTATUS		1284
10	52	AE		05	E1	000D7		BBC	#5, DEVDEPEND+2, 7\$		1285
05	51	AE		05	E1	000DC		BBC	#5, DEVDEPEND+1, 6\$		1287
		52		04	88	000E1		BISB2	#4, DEVSTATUS		1289
				06	11	000E4		BRB	7\$		
		50		02CC	8F	3C	000E6	6\$:	MOVZWL	#716, R0	1291
					04	000EB		RET			
				50	AE	95	000EC	7\$:	TSTB	DEVDEPEND	1292
					16	18	000EF		BGEQ	10\$	
		52			01	88	000F1		BISB2	#1, DEVSTATUS	1294
					11	11	000F4		BRB	10\$	1274
				04	AC	DD	000F6	8\$:	PUSHL	DEVICE_NAME	1299
				0C	AC	DD	000F9		PUSHL	CHANNEC	
	0000V	CF		02	FB	000FC		CALLS	#2, STOP_PRINTER		
		67		50	E9	00101	9\$:	BLBC	STATUS, T8\$		
				50	D4	001C4		CLRL	R0		1300
					04	00106		RET			
		03		6C	91	00107	10\$:	CMPB	(AP), #3		1305
				05	1F	0010A		BLSSU	11\$		
			0C	AC	D5	0010C		TSTL	12(AP)		
				0E	12	0010F		BNEQ	13\$		
			04	AC	DD	00111	11\$:	PUSHL	DEVICE_NAME		1307
			0C	AC	DD	00114		PUSHL	CHANNEC		
	0000V	CF		02	FB	00117		CALLS	#2, STOP_PRINTER		
		4C		50	E9	0011C	12\$:	BLBC	STATUS, T8\$		
		02		6C	91	0011F	13\$:	CMPB	(AP), #2		1309
				09	1F	00122		BLSSU	14\$		
			08	AC	D5	00124		TSTL	8(AP)		
				04	13	00127		BEQL	14\$		
	08	BC		52	D0	00129		MOVL	DEVSTATUS, @DEVICE_STATUS		1311

	04		6C 91 0012D	14\$:	CMPB	(AP), #4	:	1313
			0A 1F 00130		BLSSU	15\$:	
		10	AC D5 00132		TSTL	16(AP)	:	
			05 13 00135		BEQL	15\$:	
	10	BC	4C AE D0 00137		MOVL	DEVCLASS, @DEVICE_CLASS	:	1315
		05	6C 91 0013C	15\$:	CMPB	(AP), #5	:	1317
			0A 1F 0013F		BLSSU	16\$:	
		14	AC D5 00141		TSTL	20(AP)	:	
			05 13 00144		BEQL	16\$:	
	14	BC	48 AE D0 00146		MOVL	DEVTYPE, @DEVICE_TYPE	:	1319
		06	6C 91 0014B	16\$:	CMPB	(AP), #6	:	1321
			18 1F 0014E		BLSSU	17\$:	
		18	AC D5 00150		TSTL	24(AP)	:	
			13 13 00153		BEQL	17\$:	
		08	AE 9F 00155		PUSHAB	DEVNAMBUF	:	1323
		08	AE 9F 00158		PUSHAB	DEVNAMSIZ	:	
		18	AC DD 0015B		PUSHL	OUT_DEVICE_NAME	:	
00000000G	00		03 FB 0015E		CALLS	#3, -STR\$COPY_R	:	
	03		50 E9 00165		BLBC	STATUS, 18\$:	
	50		01 D0 00168	17\$:	MOVL	#1, R0	:	1327
			04 0016B	18\$:	RET		:	

; Routine Size: 364 bytes, Routine Base: CODE + 01AE


```

385 1328 1 %SBTTL 'STOP_PRINTER'
386 1329 1 Functional-Description:
387 1330 1 ?desc
388 1331 1
389 1332 1 Formal Parameters:
390 1333 1 ?desc
391 1334 1 Implicit Inputs:
392 1335 1 none
393 1336 1
394 1337 1 Implicit Outputs:
395 1338 1 none
396 1339 1
397 1340 1 Returned Value:
398 1341 1 none
399 1342 1
400 1343 1 Side Effects:
401 1344 1 none
402 1345 1 --
403 1346 1 ROUTINE STOP_PRINTER (
404 1347 1 CHANNEL : REF $WORD,
405 1348 1 DEVICE_NAME
406 1349 1 ) =
407 1350 2 BEGIN
408 1351 2
409 1352 2 LOCAL
410 1353 2 STATUS_1,
411 1354 2 STATUS_2
412 1355 2 ;
413 1356 2
414 1357 2 STATUS_1 = $DASSGN (CHAN=.CHANNEL[]);
415 1358 2
416 1359 2 CHANNEL[] = 0;
417 1360 2
418 1361 2 STATUS_2 = $DALLOC (DEVNAM=.DEVICE_NAME);
419 1362 2
420 1363 2 IF NOT .STATUS_1
421 1364 2 THEN
422 1365 2 RETURN .STATUS_1;
423 1366 2
424 1367 2 .STATUS_2
425 1368 2
426 1369 1 END;

```

.EXTRN SYSSDASSGN, SYSSDAL'LOC

```

0004 00000 STOP_PRINTER:
00000000G 7E 04 BC 3C 00002 .WORD Save R2 : 1346
00000000G 00 01 FB 00006 MOVZWL @CHANNEL, -(SP) : 1357
00000000G 52 50 D0 0000D CALLS #1, SYSSDASSGN
00000000G 04 BC B4 00010 MOVL R0, STATUS_1 : 1359
00000000G 7E D4 00013 CLRW @CHANNEL : 1361
00000000G 08 AC DD 00015 CLRL -(SP)
00000000G 00 02 FB 00018 PUSHL DEVICE_NAME
00000000G 03 52 E8 0001F CALLS #2, SYSSDALLOC : 1363
00000000G 03 52 E8 0001F BLBS STATUS_1, 1$

```



```

: 428 1370 1 %SBTTL 'WRITE AST'
: 429 1371 1 Functional Description:
: 430 1372 1 ?desc
: 431 1373 1
: 432 1374 1 Formal Parameters:
: 433 1375 1 ?desc
: 434 1376 1 Implicit Inputs:
: 435 1377 1 none
: 436 1378 1
: 437 1379 1 Implicit Outputs:
: 438 1380 1 none
: 439 1381 1
: 440 1382 1 Returned Value:
: 441 1383 1 none
: 442 1384 1
: 443 1385 1 Side Effects:
: 444 1386 1 none
: 445 1387 1 --
: 446 1388 1 ROUTINE WRITE_AST (
: 447 1389 1 IOB : REF $BBLOCK
: 448 1390 1 ) : NOVALUE =
: 449 1391 2 BEGIN
: 450 1392 2
: 451 1393 2 LOCAL
: 452 1394 2 FUNCTION_STATUS
: 453 1395 2 ;
: 454 1396 2
: 455 1397 2 FUNCTION_STATUS = .IOB[IOB_W_IO_STATUS];
: 456 1398 2
: 457 1399 2 PSM$REPORT (IOB, FUNCTION_STATUS);
: 458 1400 2
: 459 1401 1 END;

```

```

0000 0000 WRITE_AST:
          50      04  AC  D0 00002      .WORD      Save nothing      : 1388
          7E      0C  AO  3C 00006      MOVL      IOB, R0          : 1397
          0000000G 00      04  AC  9F 0000A      MOVZWL   12(R0), FUNCTION_STATUS
          02      FB 0000F      PUSHL   SP              : 1399
          04      04 00016      PUSHAB  IOB
          CALLS #2, PSM$REPORT
          RET
          : 1401

```

; Routine Size: 23 bytes, Routine Base: CODE + 0340

OUTPUT
V04-000

Print Symbiont -- output routines
WRITE_AST

I 12
16-Sep-1984 02:21:40
14-Sep-1984 12:55:12

VAX-11 Bliss-32 V4.0-742
[PRTSMB.SRC]OUTPUT.B32;1

Page 18
(7)

SEP
V04

: 461
: 462
1402 1 END
1403 0 ELUDOM

PSECT SUMMARY

Name Bytes Attributes
CODE 855 NOVEC,NOWRT, RD , EXE,NOSHR, LCL, REL, CON,NOPI,ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	76	0	1000	00:01.8

COMMAND QUALIFIERS

BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS\$:OUTPUT/OBJ=OBJ\$:OUTPUT MSRC\$:OUTPUT/UPDATE=(ENH\$:OUTPUT)

: Size: 855 code + 0 data bytes
: Run Time: 00:26.6
: Elapsed Time: 00:58.1
: Lines/CPU Min: 3170
: Lexemes/CPU-Min: 32415
: Memory Used: 284 pages
: Compilation Complete

0310 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

MESSAGE LIS OUTPUT LIS SEPARATE LIS
 INPUT LIS
 PRTSMB LIS
 MEMORY LIS
 PSMMSG LIS

The image displays a dense grid of approximately 15 columns and 10 rows of small, low-resolution terminal screen captures. Each individual screen shows a monospaced text-based interface, likely representing various system utilities or diagnostic tools from the VAX/VMS V4.0 era. The text is too small to read clearly, but the overall layout suggests a comprehensive set of system management or monitoring screens.