

[illegible]

[illegible]

```

LL          IIIIII          SSSSSSSSS
LL          IIIIII          SSSSSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SSSSSSS
LL          II             SSSSSSS
LL          II             SS
LL          II             SS
LL          II             SS
LL          II             SS
LLLLLLLLLLLL IIIIII          SSSSSSSSS
LLLLLLLLLLLL IIIIII          SSSSSSSSS

```

```
0001 0 MODULE MAKSTR (
0002 0
0003 0     LANGUAGE (BLISS32),
0004 0     IDENT = 'V04-000'
0005 0 ) =
0006 1 BEGIN
0007 1
0008 1 *****
0009 1 *
0010 1 *  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0011 1 *  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0012 1 *  ALL RIGHTS RESERVED.
0013 1 *
0014 1 *  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0015 1 *  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0016 1 *  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0017 1 *  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0018 1 *  OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0019 1 *  TRANSFERRED.
0020 1 *
0021 1 *  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0022 1 *  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0023 1 *  CORPORATION.
0024 1 *
0025 1 *  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0026 1 *  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0027 1 *
0028 1 *****
0029 1 *****
0030 1
0031 1 ++
0032 1
0033 1 FACILITY: F11ACP Structure Level 1
0034 1
0035 1 ABSTRACT:
0036 1
0037 1     This routine converts a RAD-50 file name block into the
0038 1     equivalent ASCII name string.
0039 1
0040 1 ENVIRONMENT:
0041 1
0042 1     STARLET operating system, including privileged system services
0043 1     and internal exec routines.
0044 1
0045 1 --
0046 1
0047 1
0048 1 AUTHOR: Andrew C. Goldstein, CREATION DATE: 3-Jan-1977 11:11
0049 1
0050 1 MODIFIED BY:
0051 1
0052 1     V03-002 CDS0001      Christian D. Saether      4-May-1983
0053 1     Make decriptor pic.
0054 1
0055 1     V03-001 ACG0302      Andrew C. Goldstein,      3-Dec-1982 11:30
0056 1     Add support for $ and _ chars in file names
0057 1
```

MAKSTR
V04-000

C 10
16-Sep-1984 00:44:54
14-Sep-1984 12:30:35

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[F11X.SRC]MAKSTR.B32;1 Page 2 (1)

:	58	0058	1	:	V02-001	ACG0186	Andrew C. Goldstein,	4-Feb-1981	21:19
:	59	0059	1	:		Fix garbage in high word of returned length			
:	60	0060	1	:					
:	61	0061	1	:	V02-000	ACG0167	Andrew C. Goldstein,	16-Apr-1980	19:27
:	62	0062	1	:		Previous revision history moved to F11B.REV			
:	63	0063	1	:					
:	64	0064	1	:					
:	65	0065	1	:					
:	66	0066	1	:	LIBRARY	'SYSS\$LIBRARY:LIB.L32';			
:	67	0067	1	:	REQUIRE	'SRC\$:FCPDEF.B32';			

MAP
V04

: R
:
:
:
:

```

69 1058 1 GLOBAL ROUTINE MAKE_STRING (NAMEBLOCK, STRING) =
70 1059 1
71 1060 1 ++
72 1061 1
73 1062 1 FUNCTIONAL DESCRIPTION:
74 1063 1
75 1064 1     This routine converts a RAD-50 file name block into the
76 1065 1     equivalent ASCII name string.
77 1066 1
78 1067 1 CALLING SEQUENCE:
79 1068 1     MAKE_STRING (ARG1, ARG2)
80 1069 1
81 1070 1 INPUT PARAMETERS:
82 1071 1     ARG1: address of file name block
83 1072 1
84 1073 1 IMPLICIT INPUTS:
85 1074 1     NONE
86 1075 1
87 1076 1 OUTPUT PARAMETERS:
88 1077 1     ARG2: address of buffer for string
89 1078 1
90 1079 1 IMPLICIT OUTPUTS:
91 1080 1     NONE
92 1081 1
93 1082 1 ROUTINE VALUE:
94 1083 1     length of string generated
95 1084 1
96 1085 1 SIDE EFFECTS:
97 1086 1     NONE
98 1087 1
99 1088 1 --
100 1089 1
101 1090 2 BEGIN
102 1091 2
103 1092 2 MAP
104 1093 2     NAMEBLOCK      : REF BBLOCK, ! name block argument
105 1094 2     STRING       : REF VECTOR [,BYTE]; ! string buffer arg
106 1095 2
107 1096 2 LOCAL
108 1097 2     BLOCKP          : REF VECTOR [,WORD], ! pointer into name block
109 1098 2     STRINGP         : REF VECTOR [,BYTE], ! pointer into string
110 1099 2     CHARS           : VECTOR [3, BYTE], ! holding place for characters
111 1100 2     FORMAT          : VECTOR [2], ! FAO control string descriptor
112 1101 2     STRINGD         : VECTOR [2]; ! string descriptor for FAO
113 1102 2
114 1103 2 BIND
115 1104 2     DELIMITER        = UPLIT BYTE (';.')
116 1105 2     : VECTOR [,BYTE]; ! type and version delimiters
117 1106 2
118 1107 2
119 1108 2 ! Set up the pointers. Then start up the outer loop, which iterates
120 1109 2 ! over name and type fields.
121 1110 2
122 1111 2
123 1112 2 BLOCKP = NAMEBLOCK[NMBSW_NAME];
124 1113 2 STRINGP = .STRING;
125 1114 2
```

```
126 1115 2  DECR K FROM 2 TO 1 DO
127 1116     BEGIN
128 1117
129 1118 ! The next loop iterates over the RAD-50 words in the name block.
130 1119 ! There are 3 words for name, 1 for type. Expand each word into
131 1120 ! the 3 RAD-50 characters.
132 1121
133 1122
134 1123 3  DECR I FROM (IF .K THEN 1 ELSE 3) TO 1 DO
135 1124     BEGIN
136 1125 4  CHARS[0] = .BLOCKP[0] / (40*40);
137 1126 4  CHARS[1] = .BLOCKP[0]/40 MOD 40;
138 1127 4  CHARS[2] = .BLOCKP[0] MOD 40;
139 1128
140 1129 4  ! Now convert each character into the correct ASCII code and store it
141 1130 4  ! in the string buffer if it is not null.
142 1131 4  !
143 1132 4
144 1133 4  INCR J FROM 0 TO 2 DO
145 1134 4  IF .CHARS[J] NEQ 0
146 1135 4  THEN
147 1136 5  BEGIN
148 1137 5  STRINGP[0] =
149 1138 6  (
150 1139 6  IF .CHARS[J] LSS 27
151 1140 6  THEN .CHARS[J] - 1 + 'A'
152 1141 6  ELSE IF .CHARS[J] EQL 27
153 1142 6  THEN '$'
154 1143 6  ELSE IF .CHARS[J] EQL 28
155 1144 6  THEN '.'
156 1145 6  ELSE IF .CHARS[J] EQL 29
157 1146 6  THEN '!'
158 1147 6  ELSE .CHARS[J] - 30 + '0'
159 1148 5  );
160 1149 5  STRINGP = .STRINGP + 1;
161 1150 4  END;
162 1151 4  BLOCKP = .BLOCKP + 2;          ! move to next word
163 1152 3  END;                      ! end of word loop
164 1153
165 1154 ! At the end of each field, insert the appropriate field delimiter.
166 1155 !
167 1156
168 1157 STRINGP[0] = .DELIMITER[K];
169 1158 STRINGP = .STRINGP + 1;
170 1159 END;                      ! end of outer loop
171 1160
172 1161 ! Now build a descriptor for the remainder of the string buffer and
173 1162 ! call FA0 to convert the version number.
174 1163 !
175 1164
176 1165 STRINGD[0] = 6;
177 1166 STRINGD[1] = .STRINGP;
178 1167 PIC DESC ('!SW', FORMAT);
179 1168 $FA0 (FORMAT, STRINGD[0], STRINGD[0], .BLOCKP[0]);
180 1169
181 1170 RETURN .STRINGP + .STRINGD[0] - .STRING;    ! final byte count
182 1171 2
```

! end of routine MAKE_STRING

				.TITLE	MAKSTR			
				.IDENT	\V04-000\			
				.PSECT	\$CODE\$,NOWRT,2			
2E	3B	20	00000	P.AAA:	.ASCII \ ;.\	:		
			00003		.BLKB 1			
00	57	53	21	00004	P.AAB: .ASCII \!SW\<0>	:		
				DELIMITER=	P.AAA			
				.EXTRN	SYSSFAO			
				.ENTRY	MAKE_STRING, Save R2,R3,R4,R5,R6	: 1058		
50	04	5E	14	C2	00002	SUBL2 #20, SP	: 1112	
		AC	06	C1	00005	ADDL3 #6, NAMEBLOCK, BLOCKP	: 1113	
		52	08	AC	D0	0000A	MOVL STRING, STRINGP	: 1115
		55	02	D0	0000E	MOVL #2, K	: 1123	
		05	55	E9	00011	1\$: BLBC K, 2\$		
		56	01	D0	00014	MOVL #1, R6		
			03	11	00017	BRB 3\$		
		56	03	D0	00019	2\$: MOVL #3, R6		
			56	D6	0001C	3\$: INCL I		
			71	11	0001E	BRB 12\$		
		51	60	3C	00020	4\$: MOVZWL (BLOCKP), R1	: 1125	
		51	8F	C6	00023	DIVL2 #1600, R1		
		6E	51	90	0002A	MOVB R1, CHARS		
		51	60	3C	0002D	MOVZWL (BLOCKP), R1	: 1126	
		51	28	C6	00030	DIVL2 #40, R1		
7E	00	51	01	7A	00033	EMUL #1, R1, #0, -(SP)		
	51		28	7B	00038	EDIV #40, (SP)+, R1, R1		
		01	51	90	0003D	MOVB R1, CHARS+1		
			60	3C	00041	MOVZWL (BLOCKP), R1	: 1127	
		51	01	7A	00044	EMUL #1, R1, #0, -(SP)		
7E	00	51	28	7B	00049	EDIV #40, (SP)+, R1, R1		
	51		51	90	0004E	MOVB R1, CHARS+2		
		02	51	D4	00052	CLRL J	: 1133	
			53	6E41	9A	00054	5\$: MOVZBL CHARSE[J], R3	: 1134
				30	13	00058	BEQL 11\$	
		1B	53	91	0005A	CMPB R3, #27	: 1139	
			09	1E	0005D	BGEQU 6\$		
		54	40	A3	9E	0005F	MOVAB 64(R3), R4	: 1140
		53	54	D0	00063	MOVL R4, R3		
			1F	11	00066	BRB 10\$		
			05	12	00068	6\$: BNEQ 7\$	: 1141	
		53	24	D0	0006A	MOVL #36, R3		
			18	11	0006D	BRB 10\$		
		1C	53	91	0006F	7\$: CMPB R3, #28	: 1143	
			05	12	00072	BNEQ 8\$		
		53	2E	D0	00074	MOVL #46, R3		
			0E	11	00077	BRB 10\$		
		1D	53	91	00079	8\$: CMPB R3, #29	: 1145	
			06	12	0007C	BNEQ 9\$		
		53	5F	8F	9A	0007E	MOVZBL #95, R3	
			03	11	00082	BRB 10\$		

```

      53      12 C0 00084 9$: ADDL2 #18, R3      : 1147
      82      53 90 00087 10$: MOVB R3, (STRINGP)+ : 1138
C6    51      02 F3 0008A 11$: AOBLEQ #2, J, 5$    : 1134
      50      02 C0 0008E      : 1151
      8C      56 F5 00091 12$: SOBGTR I, 4$        : 1123
      82      45 90 00094      : 1157
      02      55 F5 0009A      : 1115
      FF5F CF 03 11 0009D      : 1115
      FF6F 31 0009F 13$: BRB 1$
      04      06 D0 000A2 14$: BRW 1$
      08      52 D0 000A6      : 1165
      0C      03 D0 000AA      : 1166
      10      AE CF 9E 000AE      : 1167
      7E      60 3C 000B4      : 1168
      08      AE 9F 000B7      :
      0C      AE 9F 000BA      :
      18      AE 9F 000BD      :
00000000G 00      04 FB 000C0      :
      52      04 AE C0 000C7      : 1170
      52      08 AC C2 000CB      :
      50      52 D0 000CF      :
      04      04 000D2      : 1172
      RET
```

; Routine Size: 211 bytes, Routine Base: \$CODE\$ + 0008

```

: 184      1173 1
: 185      1174 1 END
: 186      1175 0 ELUDOM
```

PSECT SUMMARY

Name	Bytes	Attributes
\$CODE\$	219	NOVEC, NOWRT, RD, EXE, NOSHR, LCL, REL, CON, NOPIC, ALIGN(2)

Library Statistics

File	----- Total	Symbols Loaded	----- Percent	Pages Mapped	Processing Time
_\$255\$DUA28:[SYSLIB]LIB.L32;1	18619	19	0	1000	00:01.8

MAKSTR
V04-000

W 10
16-Sep-1984 00:44:54
14-Sep-1984 12:30:35

VAX-11 Bliss-32 V4.0-742
DISK\$VMSMASTER:[F11X.SRC]MAKSTR.B32;1 Page (2)

COMMAND QUALIFIERS

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

; Size: 211 code + 8 data bytes
; Run Time: 00:10.3
; Elapsed Time: 00:25.3
; Lines/CPU Min: 6871
; Lexemes/CPU-Min: 27339
; Memory Used: 139 pages
; Compilation Complete

0171 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

