

```
BBBBBBBBBBBBBB      AAAAAAAAAA      SSSSSSSSSSSS      RRRRRRRRRRRR      TTTTTTTTTTTTTT      LLL
BBBBBBBBBBBBBB      AAAAAAAAAA      SSSSSSSSSSSS      RRRRRRRRRRRR      TTTTTTTTTTTTTT      LLL
BBBBBBBBBBBBBB      AAAAAAAAAA      SSSSSSSSSSSS      RRRRRRRRRRRR      TTTTTTTTTTTTTT      LLL
BBB      BBB      AAA      AAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAA      AAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAA      AAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAA      AAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAA      AAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAA      AAA      SSS      RRR      RRR      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSS      RRRRRRRRRRRR      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSS      RRRRRRRRRRRR      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSS      RRRRRRRRRRRR      TTT      LLL
BBB      BBB      AAAAAAAAAAAAAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAAAAAAAAAAAAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAAAAAAAAAAAAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAA      AAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAA      AAA      SSS      RRR      RRR      TTT      LLL
BBB      BBB      AAA      AAA      SSS      RRR      RRR      TTT      LLL
BBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSSSS      RRR      RRR      TTT      LLLLLLLLLLLLLLLL
BBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSSSS      RRR      RRR      TTT      LLLLLLLLLLLLLLLL
BBBBBBBBBBBBBB      AAA      AAA      SSSSSSSSSSSS      RRR      RRR      TTT      LLLLLLLLLLLLLLLL
```

```

BBBBBBBBB      AAAAAA      SSSSSSSS      SSSSSSSS      EEEEEEEEEEE      GGGGGGGG
BBBBBBBBB      AAAAAA      SSSSSSSS      SSSSSSSS      EEEEEEEEEEE      GGGGGGGG
BB      BB      AA      AA      SS      SS      FF      GG
BB      BB      AA      AA      SS      SS      FF      GG
BB      BB      AA      AA      SS      SS      FF      GG
BB      BB      AA      AA      SS      SS      FF      GG
BBBBBBBBB      AA      AA      SSSSSS      SSSSSS      EEEEEEE      GG
BBBBBBBBB      AA      AA      SSSSSS      SSSSSS      EEEEEEE      GG
BB      BB      AAAAAAAAAA      SS      SS      FF      GG      GGGGGG
BB      BB      AAAAAAAAAA      SS      SS      FF      GG      GGGGGG
BB      BB      AA      AA      SS      SS      FF      GG      GG
BB      BB      AA      AA      SS      SS      FF      GG      GG
BBBBBBBBB      AA      AA      SSSSSSSS      SSSSSSSS      EEEEEEEEEEE      GGGGGG
BBBBBBBBB      AA      AA      SSSSSSSS      SSSSSSSS      EEEEEEEEEEE      GGGGGG

```

```

LL      IIIIII      SSSSSSSS
LL      IIIIII      SSSSSSSS
LL      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
LLLLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLLLL      IIIIII      SSSSSSSS

```

```

1 0001 0 MODULE BAS$SEG (
2 0002 0 IDENT = '1-005'
3 0003 0 ) =
4 0004 1 BEGIN
5 0005 1
6 0006 1 *****
7 0007 1 *
8 0008 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
9 0009 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
10 0010 1 * ALL RIGHTS RESERVED.
11 0011 1 *
12 0012 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
13 0013 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
14 0014 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
15 0015 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
16 0016 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
17 0017 1 * TRANSFERRED.
18 0018 1 *
19 0019 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
20 0020 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
21 0021 1 * CORPORATION.
22 0022 1 *
23 0023 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
24 0024 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
25 0025 1 *
26 0026 1 *
27 0027 1 *****
28 0028 1
29 0029 1
30 0030 1 ++
31 0031 1 FACILITY: String support library
32 0032 1
33 0033 1 ABSTRACT:
34 0034 1
35 0035 1 This module extracts a substring according to the BASIC-PLUS-2
36 0036 1 syntax. It finds the substring of a main string starting at the
37 0037 1 character position specified by the second input parameter and
38 0038 1 continues for the number of characters specified by the third
39 0039 1 input parameter. This substring is copied to the destination string.
40 0040 1
41 0041 1 ENVIRONMENT: User mode, AST level or not or mixed
42 0042 1
43 0043 1 AUTHOR: R. Will, CREATION DATE: 21-Feb-79
44 0044 1
45 0045 1 MODIFIED BY:
46 0046 1
47 0047 1 R. Will, 28-Feb-79: VERSION 01
48 0048 1 01 - original
49 0049 1 02 - change linkage and call to COPY routine. 15-Mar-79 RW
50 0050 1 1-003 - Make the string linkages start with STR$. JBS 04-JUN-1979
51 0051 1 1-004 - Change call to STR$COPY. JBS 16-JUL-1979
52 0052 1 1-005 - CALL STR$POS EXTR, DELETE THIS MODULE WHEN COMPILER CALLS STR$.
53 0053 1 RW 1-NOV-79
54 0054 1 --
55 0055 1
56 0056 1 !<BLF/PAGE>
    
```

```

: 58      0057 1 |
: 59      0058 1 | SWITCHES:
: 60      0059 1 |
: 61      0060 1 |
: 62      0061 1 | SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
: 63      0062 1 |
: 64      0063 1 |
: 65      0064 1 | LINKAGES:
: 66      0065 1 |
: 67      0066 1 |
: 68      0067 1 |
: 69      0068 1 | TABLE OF CONTENTS:
: 70      0069 1 |
: 71      0070 1 |
: 72      0071 1 | FORWARD ROUTINE
: 73      0072 1 |     BAS$SEG : NOVALUE;                ! Find the SEG of a string
: 74      0073 1 |
: 75      0074 1 |
: 76      0075 1 | INCLUDE FILES:
: 77      0076 1 |
: 78      0077 1 |
: 79      0078 1 | REQUIRE 'RTLIN:RTLPSECT';           ! Declare PSECTs code
: 80      0173 1 |
: 81      0174 1 |
: 82      0175 1 | MACROS:
: 83      0176 1 |
: 84      0177 1 |
: 85      0178 1 | EQUATED SYMBOLS:
: 86      0179 1 |
: 87      0180 1 |
: 88      0181 1 | PSECT DECLARATIONS
: 89      0182 1 |
: 90      0183 1 | DECLARE_PSECTS (BAS);
: 91      0184 1 |
: 92      0185 1 | OWN STORAGE:
: 93      0186 1 |
: 94      0187 1 |
: 95      0188 1 | EXTERNAL REFERENCES:
: 96      0189 1 |
: 97      0190 1 |
: 98      0191 1 | EXTERNAL ROUTINE
: 99      0192 1 |     STR$POS_EXTR;                    ! Routine to do the copy
: 100     0193 1 |

```

```

102 0194 1 GLOBAL ROUTINE BAS$SEG (DEST_DESC,      ! Pointer to destination descriptor
103 0195 1     SRC_DESC,                             ! Pointer to source descriptor
104 0196 1     START_POS,                         ! First character to be included
105 0197 1     END_POS) : NOVALUE =              ! Last character position to include
106 0198 1
107 0199 1
108 0200 1 ++
109 0201 1     FUNCTIONAL DESCRIPTION:
110 0202 1         This routine extracts the characters starting at the
111 0203 1         character position in the source string specified by the 3rd input
112 0204 1         parameter and continuing to the character position in the source
113 0205 1         string specified by the 4th input parameter, and copies that substring
114 0206 1         to the destination string (by JSB to STR$$COPY_R_R8) according
115 0207 1         to the syntax of the class of the destination string.
116 0208 1         If the starting position is < 1, 1 is used. If the ending position is
117 0209 1         > length of the source string, the length of the source string is
118 0210 1         used. If the starting position > the ending position, a null string
119 0211 1         is returned.
120 0212 1
121 0213 1     FORMAL PARAMETERS:
122 0214 1
123 0215 1         DEST_DESC.wt.dx      pointer to destination string descriptor
124 0216 1         SRC_DESC.rt.dx       pointer to source string descriptor
125 0217 1         START_POS.rw.v    character position in src to start substring
126 0218 1         END_POS.rw.v      character position in src to end substring
127 0219 1
128 0220 1     IMPLICIT INPUTS:
129 0221 1
130 0222 1         NONE
131 0223 1
132 0224 1     IMPLICIT OUTPUTS:
133 0225 1
134 0226 1         NONE
135 0227 1
136 0228 1     ROUTINE VALUE:
137 0229 1     COMPLETION CODES:
138 0230 1
139 0231 1         NONE
140 0232 1
141 0233 1     SIDE EFFECTS:
142 0234 1
143 0235 1         NONE
144 0236 1
145 0237 1     --
146 0238 1
147 0239 2     BEGIN
148 0240 2
149 0241 2     MAP
150 0242 2         DEST_DESC : REF BLOCK [8,BYTE];
151 0243 2
152 0244 2     MAP
153 0245 2         SRC_DESC : REF BLOCK [8,BYTE];
154 0246 2
155 0247 2     STR$POS EXTR (DEST_DESC [0,0,0,0], SRC_DESC [0,0,0,0],
156 0248 2         START_POS, END_POS);
157 0249 2     RETURN;
158 0250 1     END;                                     !End of BAS$SEG

```

```

.TITLE BASSSEG
.IDENT \1-005\
.EXTRN STR$POS_EXTR
.PSECT _BASSCODE, NOWRT, SHR, PIC, 2
.ENTRY BASSSEG, Save nothing
PUSHAB END_POS
PUSHAB START_POS
MOVQ DEST_DESC, -(SP)
CALLS #4, STR$POS_EXTR
RET

```

```

: 0194
: 0247
:
: 0250

```

```

00000000G 7E 00
10 AC 9F 00002
0C AC 9F 00005
04 AC 7D 00008
04 FB 0000C
04 00013

```

: Routine Size: 20 bytes, Routine Base: _BASSCODE + 0000

```

: 159 0251 1
: 160 0252 1 END
: 161 0253 1
: 162 0254 0 ELUDOM

```

!End of module

PSECT SUMMARY

Name	Bytes	Attributes
_BASSCODE	20	NOVEC, NOWRT, RD, EXE, SHR, LCL, REL, CON, PIC, ALIGN(2)

COMMAND QUALIFIERS

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

```

: Size: 20 code + 0 data bytes
: Run Time: 00:01.8
: Elapsed Time: 00:04.5
: Lines/CPU Min: 8327
: Lexemes/CPU-Min: 21770
: Memory Used: 19 pages
: Compilation Complete

```

0031 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION CONFIDENTIAL AND PROPRIETARY

This image displays a grid of 160 small terminal window screenshots, arranged in 10 rows and 16 columns. Each window shows a different interface or data output. Several windows have prominent titles:

- Row 2, Column 5: BASRTDIM LIS
- Row 2, Column 6: BASSARITH LIS
- Row 3, Column 11: BASSCALE LIS
- Row 3, Column 12: BASSIGNAL LIS
- Row 4, Column 6: BASRUNIMI LIS
- Row 4, Column 11: BASSCRATIC LIS
- Row 5, Column 1: BASRSTSFI LIS
- Row 6, Column 14: BASSLEEP LIS
- Row 8, Column 15: BASSTOP LIS
- Row 9, Column 11: BASSEG LIS

The remaining windows contain various data visualizations, including bar charts, histograms, and text-based reports. The overall appearance is that of a multi-processor system or a large-scale data processing application running on a VAX/VMS architecture.