```
BBBBBBBBBBBBB       AAAAAAAAA           SSSSSSSSSSSS    RRRRRRRRRRR      TTTTTTTTTTTTTTT  LLL
BBBBBBBBBBBBB       AAAAAAAAA           SSSSSSSSSSSS    RRRRRRRRRRR      TTTTTTTTTTTTTTTT LLL
BBBBBBBBBBBBB       AAAAAAAAA           SSSSSSSSSSSS    RRRRRRRRRRR      TTTTTTTTTTTTTTTT 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
BBBBBBBBBBBBB       AAA           AAA      SSSSSSSS      RRRRRRRRRRRR          TTT      LLL
BBBBBBBBBBBBB       AAA           AAA      SSSSSSSS      RRRRRRRRRRRR          TTT      LLL
BBBBBBBBBBBBB       AAA           AAA      SSSSSSSS      RRRRRRRRRRRR          TTT      LLL
BBB          BBB AAAAAAAAAAAAAAAA              SSS RRR RRR              TTT      LLL
BBB          BBB AAAAAAAAAAAAAAAA              SSS RRR RRR              TTT      LLL
BBB          BBB AAAAAAAAAAAAAAAA              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
BBBBBBBBBBBBB       AAA           AAA SSSSSSSSSSSS    RRR         RRR     TTT      LLLLLLLLLLLLLLLL
BBBBBBBBBBBBB       AAA           AAA SSSSSSSSSSSS    RRR         RRR     TTT      LLLLLLLLLLLLLLLL
BBBBBBBBBBBBB       AAA           AAA SSSSSSSSSSSS    RRR         RRR     TTT      LLLLLLLLLLLLLLLL
```

```
BBBBBBBB      AAAAAA      SSSSSSSS    CCCCCCCC    000000    NN      NN    CCCCCCCC      AAAAAA    TTTTTTTTTT
BBBBBBBB      AAAAAA      SSSSSSSS    CCCCCCCC    000000    NN      NN    CCCCCCCC      AAAAAA    TTTTTTTTTT
BB      BB  AA      AA  SS              CC      00      00  NN      NN  CC          AA      AA        TT
BB      BB  AA      AA  SS              CC      00      00  NNNN    NN  CC          AA      AA        TT
BB      BB  AA      AA  SS              CC      00      00  NNNN    NN  CC          AA      AA        TT
BBBBBBBB    AA      AA    SSSSSS        CC      00      00  NN  NN  NN  CC          AA      AA        TT
BBBBBBBB    AA      AA    SSSSSS        CC      00      00  NN  NN  NN  CC          AA      AA        TT
BB      BB  AAAAAAAAAA          SS      CC      00      00  NN    NNNN  CC          AAAAAAAAAA        TT
BB      BB  AAAAAAAAAA          SS      CC      00      00  NN    NNNN  CC          AAAAAAAAAA        TT
BB      BB  AA      AA          SS      CC      00      00  NN      NN  CC          AA      AA        TT
BB      BB  AA      AA          SS      CC      00      00  NN      NN  CC          AA      AA        TT      ....
BBBBBBBB    AA      AA  SSSSSSSS    CCCCCCCC    000000    NN      NN    CCCCCCCC  AA      AA          TT      ....
BBBBBBBB    AA      AA  SSSSSSSS    CCCCCCCC    000000    NN      NN    CCCCCCCC  AA      AA          TT      ....
```

```
LL              IIIIII      SSSSSSSS
LL              IIIIII      SSSSSSSS
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
LLLLLLLLL      IIIIII      SSSSSSSS
LLLLLLLLL      IIIIII      SSSSSSSS
```

```
    1     0001  0  MODULE BAS$CONCAT (                              ! Concatenate several strings
    2     0002  0                    IDENT = '1-001'                ! File: BASCONCAT.B32
    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: VAX-11 BASIC
   32     0032  1  !
   33     0033  1  ! ABSTRACT:
   34     0034  1  !
   35     0035  1  !       This module takes up to 254 input strings and concatenates
   36     0036  1  !       them into a result string.  The strings can be of any class
   37     0037  1  !       and data type, provided that the length field of the descriptor
   38     0038  1  !       indicates the length of the string in bytes.
   39     0039  1  !
   40     0040  1  ! ENVIRONMENT: VAX-11 User mode
   41     0041  1  !
   42     0042  1  ! AUTHOR: John Sauter, CREATION DATE: 18-OCT-1979
   43     0043  1  !
   44     0044  1  ! MODIFIED BY:
   45     0045  1  !
   46     0046  1  ! 1-001 - Original.
   47     0047  1  !--
   48     0048  1
   49     0049  1  !<BLF/PAGE>
```

```
51      0050    1   ! SWITCHES:
52      0051    1   !
53      0052    1
54      0053    1
55      0054    1   SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
56      0055    1
57      0056    1   !
58      0057    1   ! LINKAGES:
59      0058    1   !
60      0059    1   !     NONE
61      0060    1
62      0061    1   ! TABLE OF CONTENTS:
63      0062    1   !
64      0063    1
65      0064    1   FORWARD ROUTINE
66      0065    1       BAS$CONCAT : NOVALUE;                        ! Concatenate two or more strings
67      0066    1
68      0067    1   !
69      0068    1   ! INCLUDE FILES:
70      0069    1   !
71      0070    1
72      0071    1   REQUIRE 'RTLIN:RTLPSECT';                        ! Declare PSECTS code
73      0166    1
74      0167    1   LIBRARY 'RTLSTARLE';                             ! System symbols
75      0168    1
76      0169    1   ! MACROS:
77      0170    1   !
78      0171    1   !
79      0172    1   !     NONE
80      0173    1   !
81      0174    1   ! EQUATED SYMBOLS:
82      0175    1   !
83      0176    1   !     NONE
84      0177    1   !
85      0178    1   ! PSECT DECLARATIONS
86      0179    1
87      0180    1   DECLARE_PSECTS (BAS);
88      0181    1   !
89      0182    1   ! OWN STORAGE:
90      0183    1   !
91      0184    1   !     NONE
92      0185    1   !
93      0186    1   ! EXTERNAL REFERENCES:
94      0187    1   !
95      0188    1
96      0189    1   EXTERNAL ROUTINE
97      0190    1       STR$CONCAT,                                  ! Concatenate strings
98      0191    1       BAS$$STOP : NOVALUE;                         ! Signal fatal BASIC error
99      0192    1
100     0193    1   !+
101     0194    1   ! The following are the error messages used in this module:
102     0195    1   !-
103     0196    1
104     0197    1   EXTERNAL LITERAL
105     0198    1       BAS$K_STRTOOLON : UNSIGNED (8);              ! String too long
106     0199    1
```

```
108    0200   1   GLOBAL ROUTINE BAS$CONCAT (                          ! Concatenate strings
109    0201   1           DEST_DESC                                   ! pointer to destination descriptor
110    0202   1       ) : NOVALUE =
111    0203   1
112    0204   1   !++
113    0205   1   !   FUNCTIONAL DESCRIPTION
114    0206   1   !
115    0207   1   !       This routine takes up to 254 source strings of any DTYPE and
116    0208   1   !       CLASS, concatenates them, and assigns that value to the
117    0209   1   !       destination string.
118    0210   1   !
119    0211   1   !   FORMAL PARAMETERS:
120    0212   1   !
121    0213   1   !       DEST_DESC.wt.dx            Pointer to destination descriptor
122    0214   1   !       [INPUT].rt.dx              Pointer to input string.  There can
123    0215   1   !                                  be up to 254 of these.
124    0216   1   !
125    0217   1   !   IMPLICIT INPUTS:
126    0218   1   !
127    0219   1   !       NONE
128    0220   1   !
129    0221   1   !   IMPLICIT OUTPUTS:
130    0222   1   !
131    0223   1   !       NONE
132    0224   1   !
133    0225   1   !   COMPLETION CODES:
134    0226   1   !
135    0227   1   !       NONE
136    0228   1   !
137    0229   1   !   SIDE EFFECTS:
138    0230   1   !
139    0231   1   !       Signals the same as STR$CONCAT, and also signals String too long
140    0232   1   !       if the result descriptor is dynamic and STR$CONCAT had to truncate.
141    0233   1   !
142    0234   1   !--
143    0235   1
144    0236   2       BEGIN
145    0237   2
146    0238   2       MAP
147    0239   2           DEST_DESC : REF BLOCK [8, BYTE];
148    0240
149    0241   2       BUILTIN
150    0242   2           AP,
151    0243   2           CALLG;
152    0244   2
153    0245   2       LOCAL
154    0246   2           CONCAT_STATUS;
155    0247   2
156    0248   2   !+
157    0249   2   ! Call the string concatenate routine, giving it our parameters.
158    0250   2   !-
159    0251   2       CONCAT_STATUS = CALLG (.AP, STR$CONCAT);
160    0252   2
161    0253   3       IF ( NOT .CONCAT_STATUS)
162    0254   2       THEN
163    0255   3           BEGIN
164    0256   3
```

```
; 165         0257 3              IF (.DEST_DESC [DSC$B_CLASS] EQL DSC$K_CLASS_D) THEN BAS$$STOP (BAS$K_STRTOOLON);
; 166         0258 3
; 167         0259 2          END;
; 168         0260 2
; 169         0261 1      END;                                                      ! End of BAS$CONCAT


                                                 .TITLE  BAS$CONCAT
                                                 .IDENT  \1-001\

                                                 .EXTRN  STR$CONCAT, BAS$$STOP
                                                 .EXTRN  BAS$K_STRTOOLON

                                                 .PSECT  _BAS$CODE,NOWRT,  SHR,  PIC,2

                                 0000 00000      .ENTRY  BAS$CONCAT, Save nothing              ; 0200
                 00000000G  00         6C FA 00002      CALLG   (AP), STR$CONCAT              ; 0251
                           15         50 E8 00009      BLBS    CONCAT_STATUS, 1$             ; 0253
                           50      04 AC D0 0000C      MOVL    DEST_DESC, R0                 ; 0257
                           02      03 A0 91 00010      CMPB    3(R0), #2
                                    0B 12 00014      BNEQ    1$
                           7E      00G 8F 9A 00016      MOVZBL  #BAS$K_STRTOOLON, -(SP)
                 00000000G  00         01 FB 0001A      CALLS   #1, BAS$$STOP
                                       04 00021 1$:    RET                                    ; 0261

; Routine Size: 34 bytes,   Routine Base: _BAS$CODE + 0000


; 170         0262 1
; 171         0263 1 END                                                            !End of module BAS$CONCAT
; 172         0264 1
; 173         0265 0 ELUDOM
```

;   PSECT SUMMARY
;
;   Name   Bytes   Attributes
;
; _BAS$CODE   34 NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)

;   Library Statistics
;
;   -------- Symbols --------   Pages   Processing
;   File   Total   Loaded   Percent   Mapped   Time
;
; _255DUA28:[SYSLIB]STARLET.L32;1   9776   2   0   581   00:01.0

```
;                             COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACE/LIS=LIS$:BASCONCAT/OBJ=OBJ$:BASCONCAT MSRC$:BASCONCAT/UPDATE=(ENH$:BASCONCAT
;       )

; Size:          34 code + 0 data bytes
; Run Time:         00:03.2
; Elapsed Time:     00:08.4
; Lines/CPU Min:    4937
; Lexemes/CPU-Min: 12577
; Memory Used:  37 pages
; Compilation Complete
```