BASRTL

BASTRM

LIS

```
0000    1           .TITLE  BAS$TRM
0000    2           .IDENT  /1-008/              ; File: BASTRM.MAR      Edit: SBL1008
0000    3
0000    4   ;
0000    5   ;*********************************************************************
0000    6   ;*                                                                   *
0000    7   ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                          *
0000    8   ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.           *
0000    9   ;*  ALL RIGHTS RESERVED.                                             *
0000   10   ;*                                                                   *
0000   11   ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED *
0000   12   ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE *
0000   13   ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER *
0000   14   ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY *
0000   15   ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY *
0000   16   ;*  TRANSFERRED.                                                      *
0000   17   ;*                                                                   *
0000   18   ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE *
0000   19   ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT *
0000   20   ;*  CORPORATION.                                                      *
0000   21   ;*                                                                   *
0000   22   ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS *
0000   23   ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.          *
0000   24   ;*                                                                   *
0000   25   ;*                                                                   *
0000   26   ;*********************************************************************
0000   27   ;
0000   28
0000   29   ;++
0000   30   ; FACILITY: BASIC code support
0000   31   ;
0000   32   ; ABSTRACT:
0000   33   ;
0000   34   ;       This module implements the BASIC-PLUS-2 EDIT$ function.
0000   35   ;
0000   36   ; ENVIRONMENT: User Mode, AST Reentrant
0000   37   ;
0000   38   ;--
0000   39   ; AUTHOR: R. WILL, CREATION DATE: 2-May-79
0000   40   ;
0000   41   ; MODIFIED BY:
0000   42   ;
0000   43   ; R. Will, : VERSION 1
0000   44   ; 1-001 - Original
0000   45   ; 1-002 - Fix order of source and dest strs in IRM. RW 16-May-79
0000   46   ; 1-003 - Enable IV in entry masks.  RW 23-May-79
0000   47   ; 1-004 - Change calls to STR$COPY.   JBS 16-JUL-1979
0000   48   ; 1-005 - Make PIC.  RW  13-Sept-79
0000   49   ; 1-006 - String cleanup, call STR$TRM.  RW  31-Oct-79
0000   50   ; 1-007 - Make TRM a separate module.  RW  13-Nov-79
0000   51   ; 1-008 - Use general mode addressing.  SBL 30-Nov--1981
```

BASSTRM
1-008

F 11

DECLARATIONS

15-SEP-1984 23:58:04 VAX/VMS Macro V04-00   Page 2
6-SEP-1984 10:38:28 [BASRTL.SRC]BASTRM.MAR;1   (2)

```
                      0000    53              .SBTTL  DECLARATIONS
                      0000    54 ;
                      0000    55 ; INCLUDE FILES:
                      0000    56 ;
                      0000    57
                      0000    58              $DSCDEF                          ; Define descriptor offsets
                      0000    59
                      0000    60 ;
                      0000    61 ; EXTERNAL DECLARATIONS:
                      0000    62 ;
                      0000    63              .DSABL  GBL                      ; Prevent undeclared
                      0000    64                                              ; symbols from being
                      0000    65                                              ; automatically global.
                      0000    66
                      0000    67              .EXTRN  STR$TRIM                 ; routine to trim blanks and tabs
                      0000    68
                      0000    69 ;
                      0000    70 ; MACROS:
                      0000    71 ;
                      0000    72
                      0000    73 ;+
                      0000    74 ; Macro to generate tables to translate ASCII characters.  This macro
                      0000    75 ; will generate a table of bytes starting at the designated character
                      0000    76 ; for the series of characters needed.
                      0000    77 ;-
                      0000    78
                      0000    79              .MACRO  GEN_TRANS_TABLE, NUMBER_OF_CHARS, STARTING_CHAR
                      0000    80              I = STARTING_CHAR
                      0000    81              .REPT   NUMBER_OF_CHARS
                      0000    82              .BYTE   I
                      0000    83              I = I + 1
                      0000    84              .ENDR
                      0000    85              .ENDM
                      0000    86
                      0000    87 ;
                      0000    88 ; EQUATED SYMBOLS:
                      0000    89 ;
                      0000    90
            00000020  0000    91              blank            = ^A/ /
            00000001  0000    92              space_or_tab     = ^X0001
            00000002  0000    93              junk_byte        = ^X0002
            00000004  0000    94              quote            = ^X0004
            00000003  0000    95              space_tab_junk   = ^X0003
                      0000    96
            00000000  0000    97              trim_parity_bit = 0
            00000001  0000    98              disc_sp_tab_bit = 1
            00000002  0000    99              disc_junk_bit   = 2
            00000003  0000   100              disc_lead_bit   = 3
            00000004  0000   101              compact_bit     = 4
            00000005  0000   102              upper_case_bit  = 5
            00000006  0000   103              parens_bit      = 6
            00000007  0000   104              disc_trail_bit  = 7
            00000008  0000   105              ignor_quote_bit = 8
                      0000   106
                      0000   107 ;
                      0000   108 ; OWN STORAGE:
                      0000   109 ;
```

```
          0000    110
          0000    111 ;
          0000    112 ; PSECT DECLARATIONS:
          0000    113 ;
      00000000    114         .PSECT _BAS$CODE PIC, USR, CON, REL, LCL, SHR, -
          0000    115                 EXE, RD, NOWRT, LONG
          0000    116
```

```
                              0000  118
                              0000  119              .SBTTL  BASSTRM              - Delete trailing blanks and spaces
                              0000  120 ;++
                              0000  121 ; FUNCTIONAL DESCRIPTION:
                              0000  122 ;
                              0000  123 ;         This routine deletes trailing tabs and spaces from the source string
                              0000  124 ;         and then uses STR$COPY to write the string in the destination.  It
                              0000  125 ;         finds the last non-blank character by starting at the end of the
                              0000  126 ;         string and using each character to index into a table that has a
                              0000  127 ;         flag for spaces and tabs.
                              0000  128 ;
                              0000  129 ; CALLING SEQUENCE:
                              0000  130 ;
                              0000  131 ;         CALL BASSTRM (dest_string.wx.dx, src_string.rx.dx)
                              0000  132 ;
                              0000  133 ; INPUT PARAMETERS:
                              0000  134 ;
                 00000008     0000  135 ;         src_string = 8
                              0000  136 ;
                              0000  137 ; IMPLICIT INPUTS:
                              0000  138 ;
                              0000  139 ;         NONE
                              0000  140 ;
                              0000  141 ; OUTPUT PARAMETERS:
                              0000  142 ;
                 00000004     0000  143 ;         dest_string = 4
                              0000  144 ;
                              0000  145 ; IMPLICIT OUTPUTS:
                              0000  146 ;
                              0000  147 ;         NONE
                              0000  148 ;
                              0000  149 ; FUNCTION VALUE:
                              0000  150 ; COMPLETION CODES:
                              0000  151 ;
                              0000  152 ;         NONE
                              0000  153 ;
                              0000  154 ; SIDE EFFECTS:
                              0000  155 ;
                              0000  156 ;         STR$COPY may allocate space to the destination string.  Any of
                              0000  157 ;         STR$COPY's errors may be signalled.
                              0000  158 ;
                              0000  159 ;--
                              0000  160
                      0000    0000  161              .ENTRY BASSTRM, ^M<>
                              0002  162
                              0002  163 ;+
                              0002  164 ; Register usage
                              0002  165 ;-
                              0002  166
00000000'GF  6C   FA  0002  167              CALLG   (AP), G^STRSTRIM
             04       0009  168              RET                                ; End of BASSTRM
                              000A  169
                              000A  170
```

```
                   000A    172  ;+
                   000A    173  ; The following tables are used for the SPANC, SCANC, and MOVTC instructions
                   000A    174  ; used in the BAS$EDIT routine
                   000A    175  ;-
                   000A    176
                   000A    177
                   000A    178  ;+
                   000A    179  ; Table used to 'remove parity bit' from all characters in the character
                   000A    180  ; string.  Used to ensure that the high bit is clear on all characters.
                   000A    181  ;-
                   000A    182
                   000A    183  PARITY_TAB:
                   000A    184       GEN_TRANS_TABLE 128, 0              ; fill ASCII character set
                   008A    185       GEN_TRANS_TABLE 128, 0              ; truncate hi bit
                   010A    186
                   010A    187  ;+
                   010A    188  ; Table used to translate lower case to uppercase.  Translate rest of
                   010A    189  ; ASCII character set to self.
                   010A    190  ;-
                   010A    191
                   010A    192  UPCASE_TAB:
                   010A    193       GEN_TRANS_TABLE 97, 0               ; fill table to start of lowercase
                   016B    194       GEN_TRANS_TABLE 26, 65             ; translate lower to upper
                   0185    195       GEN_TRANS_TABLE 133, 123           ; finish table, translate to self
                   020A    196
                   020A    197  SQBRAK_TO_PAREN:
                   020A    198       GEN_TRANS_TABLE 91, 0              ; translate to self
            28     0265    199       .BYTE           40                ; translate [ to (
            5C     0266    200       .BYTE           92                ; translate to self
            29     0267    201       .BYTE           41                ; translate ] to )
                   0268    202       GEN_TRANS_TABLE 162, 94            ; translate remainder to self
                   030A    203
                   030A    204  UPCASE_PAREN:
                   030A    205       GEN_TRANS_TABLE 91, 0              ; translate to self
            28     0365    206       .BYTE           40                ; translate [ to (
            5C     0366    207       .BYTE           92                ; translate to self
            29     0367    208       .BYTE           41                ; translate ] to )
                   0368    209       GEN_TRANS_TABLE 3, 94              ; translate to self
                   036B    210       GEN_TRANS_TABLE 26, 65            ; translate lower to upper
                   0385    211       GEN_TRANS_TABLE 133, 123          ; finish table, translate to self
                   040A    212
                   040A    213  SCAN_SPAN_TABLE:
            02     040A    214       .BYTE           junk_byte         ; NUL
00'00'00'00'00'00'00'00'  040B    215       .BYTE           0[8]              ; skip over
            01     0413    216       .BYTE           space_or_tab      ; tab
            02     0414    217       .BYTE           junk_byte         ; LF
            00     0415    218       .BYTE           0                 ; skip over
       02'02'     0416    219       .BYTE           junk_byte[2]      ; FF and CR
00'00'00'00'00'00'00'00'00'00'00'  0418    220   .BYTE   0[13]             ; skip over
            00'    0424
            02     0425    221       .BYTE           junk_byte         ; ESC
   00'00'00'00'    0426    222       .BYTE           0[4]              ; skip over
            01     042A    223       .BYTE           space_or_tab      ; space
            00     042B    224       .BYTE           0                 ; skip over
            04     042C    225       .BYTE           quote             ; double quote
   00'00'00'00'    042D    226       .BYTE           0[4]              ; skip over
            04     0431    227       .BYTE           quote             ; single quote
```

```
00'00'00'00'00'00'00'00'00'00'00' 0432      228           .BYTE          0[87]          ; skip over
00'00'00'00'00'00'00'00'00'00'00' 043E
00'00'00'00'00'00'00'00'00'00'00' 044A
00'00'00'00'00'00'00'00'00'00'00' 0456
00'00'00'00'00'00'00'00'00'00'00' 0462
00'00'00'00'00'00'00'00'00'00'00' 046E
00'00'00'00'00'00'00'00'00'00'00' 047A
                     00'00'00' 0486
                           02 0489      229           .BYTE          junk byte      ; DEL
00'00'00'00'00'00'00'00'00'00'00' 048A      230           .BYTE          0[128]         ; skip over
00'00'00'00'00'00'00'00'00'00'00' 0496
00'00'00'00'00'00'00'00'00'00'00' 04A2
00'00'00'00'00'00'00'00'00'00'00' 04AE
00'00'00'00'00'00'00'00'00'00'00' 04BA
00'00'00'00'00'00'00'00'00'00'00' 04C6
00'00'00'00'00'00'00'00'00'00'00' 04D2
00'00'00'00'00'00'00'00'00'00'00' 04DE
00'00'00'00'00'00'00'00'00'00'00' 04EA
00'00'00'00'00'00'00'00'00'00'00' 04F6
                     00'00'00'00'00'00'00'00' 0502
                              050A      231           .END
```

```
BAS$TRM                          00000000 RG    02
JUNK_BYTE                       = 00000002
PARITY_TAB                       0000000A R     02
QUOTE                          = 00000004
SCAN_SPAN_TABLE                  0000040A R     02
SPACE_OR_TAB                   = 00000001
SQBRAK_TO_PAREN                  0000020A R     02
STR$TRIM                         ********  X    00
UPCASE_PAREN                     0000030A R     02
UPCASE_TAB                       0000010A R     02
```

```
                              +-----------------+
                              ! Psect synopsis !
                              +-----------------+
```

PSECT name	Allocation		PSECT No.	Attributes										
. ABS .	00000000	(0.)	00 (0.)	NOPIC	USR	CON	ABS	LCL	NOSHR	NOEXE	NORD	NOWRT	NOVEC	BYTE
ABS	00000000	(0.)	01 (1.)	NOPIC	USR	CON	ABS	LCL	NOSHR	EXE	RD	WRT	NOVEC	BYTE
_BAS$CODE	0000050A	(1290.)	02 (2.)	PIC	USR	CON	REL	LCL	SHR	EXE	RD	NOWRT	NOVEC	LONG

```
                              +----------------------------+
                              ! Performance indicators !
                              +----------------------------+
```

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.07	00:00:00.54
Command processing	104	00:00:00.48	00:00:02.44
Pass 1	226	00:00:07.60	00:00:15.55
Symbol table sort	0	00:00:00.19	00:00:00.40
Pass 2	57	00:00:01.72	00:00:03.86
Symbol table output	2	00:00:00.02	00:00:00.02
Psect synopsis output	2	00:00:00.03	00:00:00.03
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	422	00:00:10.11	00:00:22.84

The working set limit was 1350 pages.
53938 bytes (106 pages) of virtual memory were used to buffer the intermediate code.
There were 10 pages of symbol table space allocated to hold 155 non-local and 0 local symbols.
231 source lines were read in Pass 1, producing 15 object records in Pass 2.
9 pages of virtual memory were used to define 8 macros.

```
                              +----------------------------+
                              ! Macro library statistics !
                              +----------------------------+
```

Macro library name	Macros defined
_255DUA28:[SYSLIB]STARLET.MLB;2	4

190 GETS were required to define 4 macros.

There were no errors, warnings or information messages.

MACRO/ENABLE=SUPPRESSION/DISABLE=(GLOBAL,TRACEBACK)/LIS=LIS$:BASTRM/OBJ=OBJ$:BASTRM MSRC$:BASTRM/UPDATE=(ENH$:BASTRM)

BASSTRING
LIS

BASTERMIO
LIS

BASTRM
LIS

BASUDFWF
LIS

BASTAB
LIS

BASUDFRM
LIS

BASSYS
LIS

BASSTR
LIS

BASUDFRL
LIS