```
EEEEEEEEEEEEEEEE   DDDDDDDDDDD      TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEE   DDDDDDDDDDD      TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEEE   DDDDDDDDDDD      TTTTTTTTTTTTTTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEEEEEEEEEEE       DDD       DDD          TTT
EEEEEEEEEEEE       DDD       DDD          TTT
EEEEEEEEEEEE       DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEE                DDD       DDD          TTT
EEEEEEEEEEEEEEEE   DDDDDDDDDDD            TTT
EEEEEEEEEEEEEEEE   DDDDDDDDDDD            TTT
EEEEEEEEEEEEEEEE   DDDDDDDDDDD            TTT
```

```
FFFFFFFFF   PPPPPPPP    UU      UU   TTTTTTTTTT    CCCCCCC   HH      HH      AAAAAA    RRRRRRR
FFFFFFFFF   PPPPPPPP    UU      UU   TTTTTTTTTT    CCCCCCC   HH      HH      AAAAAA    RRRRRRR
FF          PP     PP   UU      UU       TT       CC         HH      HH    AA    AA   RR      RR
FF          PP     PP   UU      UU       TT       CC         HH      HH    AA    AA   RR      RR
FF          PP     PP   UU      UU       TT       CC         HH      HH    AA    AA   RR      RR
FFFFFFFF    PPPPPPPP    UU      UU       TT       CC         HHHHHHHHHH    AA    AA   RRRRRRR
FFFFFFFF    PPPPPPPP    UU      UU       TT       CC         HHHHHHHHHH    AA    AA   RRRRRRR
FF          PP          UU      UU       TT       CC         HH      HH    AAAAAAAAAA RR  RR
FF          PP          UU      UU       TT       CC         HH      HH    AAAAAAAAAA RR  RR
FF          PP          UU      UU       TT       CC         HH      HH    AA    AA   RR    RR
FF          PP          UU      UU       TT                  HH      HH    AA    AA   RR    RR
FF          PP          UUUUUUUUUUU      TT       CCCCCCC    HH      HH    AA    AA   RR      RR
FF          PP          UUUUUUUUUUU      TT       CCCCCCC    HH      HH    AA    AA   RR      RR


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
LLLLLLLLLL   IIIIII      SSSSSSSS
LLLLLLLLLL   IIIIII      SSSSSSSS
```

```
   1    0001  0 %TITLE 'EDT$FPUTCHAR - store a character in the format buffer'
   2    0002  0 MODULE EDT$FPUTCHAR (                            ! Store a character in the format buffer
   3    0003  0             IDENT = 'V04-000'                    ! File: FPUTCHAR.BLI Edit: JBS1004
   4    0004  0             ) =
   5    0005  1 BEGIN
   6    0006  1
   7    0007  1 !****************************************************************************
   8    0008  1 !*                                                                          *
   9    0009  1 !*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                *
  10    0010  1 !*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                 *
  11    0011  1 !*   ALL RIGHTS RESERVED.                                                   *
  12    0012  1 !*                                                                          *
  13    0013  1 !*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
  14    0014  1 !*   ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
  15    0015  1 !*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
  16    0016  1 !*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
  17    0017  1 !*   OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY  *
  18    0018  1 !*   TRANSFERRED.                                                           *
  19    0019  1 !*                                                                          *
  20    0020  1 !*   THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
  21    0021  1 !*   AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
  22    0022  1 !*   CORPORATION.                                                           *
  23    0023  1 !*                                                                          *
  24    0024  1 !*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
  25    0025  1 !*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                *
  26    0026  1 !*                                                                          *
  27    0027  1 !*                                                                          *
  28    0028  1 !****************************************************************************
  29    0029  1
  30    0030  1
  31    0031  1 !++
  32    0032  1 ! FACILITY:      EDT -- The DEC Standard Editor
  33    0033  1 !
  34    0034  1 ! ABSTRACT:
  35    0035  1 !
  36    0036  1 !       Store a character in the format buffer.  Write out the buffer
  37    0037  1 !       if it overflows.
  38    0038  1 !
  39    0039  1 ! ENVIRONMENT:  Runs at any access mode - AST reentrant
  40    0040  1 !
  41    0041  1 ! AUTHOR: Bob Kushlis, CREATION DATE: March 18, 1979
  42    0042  1 !
  43    0043  1 ! MODIFIED BY:
  44    0044  1 !
  45    0045  1 ! 1-001 - Original.  DJS 19-FEB-1981.  This module was created by
  46    0046  1 !         extracting routine EDT$$PUT_CH  from module FORMAT.
  47    0047  1 ! 1-002 - Regularize headers.  JBS 05-Mar-1981
  48    0048  1 ! 1-003 - Use EDT$$K_FMT_BUFLEN.  JBS 29-Sep-1982
  49    0049  1 ! 1-004 - Fix a local symbol name.  JBS 04-Jan-1983
  50    0050  1 !--
  51    0051  1
```

```
   53      0052  1 %SBTTL 'Declarations'
   54      0053  1 !
   55      0054  1 ! TABLE OF CONTENTS:
   56      0055  1 !
   57      0056  1
   58      0057  1 REQUIRE 'EDTSRC:TRAROUNAM';
   59      0496  1
   60      0497  1 FORWARD ROUTINE
   61      0498  1     EDT$$STORE_FMTCH : NOVALUE;
   62      0499  1
   63      0500  1 !
   64      0501  1 ! INCLUDE FILES:
   65      0502  1 ! !
   66      0503  1
   67      0504  1 REQUIRE 'EDTSRC:EDTREQ';
   68      0639  1
   69      0640  1 !
   70      0641  1 ! MACROS:
   71      0642  1 !
   72      0643  1 !      NONE
   73      0644  1 !
   74      0645  1 ! EQUATED SYMBOLS:
   75      0646  1 !
   76      0647  1 !      NONE
   77      0648  1 !
   78      0649  1 ! OWN STORAGE:
   79      0650  1 !
   80      0651  1 !      NONE
   81      0652  1 !
   82      0653  1 ! EXTERNAL REFERENCES:
   83      0654  1 !
   84      0655  1 !      In the routine
```

M 4

EDT$FPUTCHAR   EDT$FPUTCHAR - store a character in the format   16-Sep-1984 00:28:45   VAX-11 BLiss-32 V4.0-742   Page 3   EDT$
V04-000   EDT$$STORE_FMTCH - store a character in the fo 14-Sep-1984 12:23:12   DISK$VMSMASTER:[EDT.SRC]FPUTCHAR.BLI;1   (3)   V04-

```
  86      0656  1    %SBTTL 'EDT$$STORE_FMTCH  - store a character in the format buffer'
  87      0657  1
  88      0658  1    GLOBAL ROUTINE EDT$$STORE_FMTCH (                      ! Store a character in the format buffer
  89      0659  1        C                                                 ! The character to store
  90      0660  1        ) : NOVALUE =
  91      0661  1
  92      0662  1    !++
  93      0663  1    ! FUNCTIONAL DESCRIPTION:
  94      0664  1    !
  95      0665  1    !        Place a character in the formatted output buffer.  If the pointer is at the
  96      0666  1    !        end of the buffer then write out the line and start over again.
  97      0667  1    !
  98      0668  1    ! FORMAL PARAMETERS:
  99      0669  1    !
 100      0670  1    !   C                    The character to put out.
 101      0671  1    !
 102      0672  1    ! IMPLICIT INPUTS:
 103      0673  1    !
 104      0674  1    !        EDT$$T_FMT_BUF
 105      0675  1    !        EDT$$A_FMT_CUR
 106      0676  1    !        EDT$$G_FMT_LNPOS
 107      0677  1    !
 108      0678  1    ! IMPLICIT OUTPUTS:
 109      0679  1    !
 110      0680  1    !        NONE
 111      0681  1    !
 112      0682  1    ! ROUTINE VALUE:
 113      0683  1    !
 114      0684  1    !        NONE
 115      0685  1    !
 116      0686  1    ! SIDE EFFECTS:
 117      0687  1    !
 118      0688  1    !        NONE
 119      0689  1    !
 120      0690  1    !--
 121      0691  1
 122      0692  2        BEGIN
 123      0693  2
 124      0694  2        EXTERNAL ROUTINE
 125      0695  2            EDT$$OUT_FMTBUF;
 126      0696  2
 127      0697  2        EXTERNAL
 128      0698  2            EDT$$T_FMT_BUF : BLOCK [CH$ALLOCATION (EDT$$K_FMT_BUFLEN)],      ! The formatted output buffer
 129      0699  2            EDT$$A_FMT_CUR,                                  ! Pointer to next character in above
 130      0700  2            EDT$$G_FMT_LNPOS;                                ! The current column number
 131      0701  2
 132      0702  2        LOCAL
 133      0703  2            SAV_LNPOS;
 134      0704  2
 135      0705  2        BIND
 136      0706  2            END_BUF = CH$PTR (EDT$$T_FMT_BUF, EDT$$K_FMT_BUFLEN);   ! Pointer to end of format buffer
 137      0707  2    !+
 138      0708  2    ! Check for end of buffer.
 139      0709  2    !-
 140      0710  2
 141      0711  2
 142      0712  3        IF CH$PTR_EQL (END_BUF, .EDT$$A_FMT_CUR)
```

N 4

EDT$FPUTCHAR   EDT$FPUTCHAR - store a character in the format 16-Sep-1984 00:28:45   VAX-11 Bliss-32 V4.0-742   Page   4
V04-000   EDT$$STORE_FMTCH - store a character in the fo 14-Sep-1984 12:23:12   DISK$VMSMASTER:[EDT.SRC]FPUTCHAR.BLI;1   (3)

```
;   143    0713  2        THEN
;   144    0714  3            BEGIN
;   145    0715  3            SAV_LNPOS = .EDT$$G_FMT_LNPOS;
;   146    0716  3            EDT$$OUT_FMTBUF ();
;   147    0717  3            EDT$$G_FMT_LNPOS = .SAV_LNPOS;
;   148    0718  2            END;
;   149    0719
;   150    0720  2        CH$WCHAR_A (.C, EDT$$A_FMT_CUR);
;   151    0721  1        END;                                           ! of routine EDT$$STORE_FMTCH
;
;


                                                      .TITLE   EDT$FPUTCHAR EDT$FPUTCHAR - store a character i
                                                                           n the format
                                                      .IDENT   \V04-000\

                                                      .EXTRN   EDT$$OUT_FMTBUF
                                                      .EXTRN   EDT$$T_FMT_BUF, EDT$$A_FMT_CUR
                                                      .EXTRN   EDT$$G_FMT_LNPOS

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

                                  001C 00000          .ENTRY   EDT$$STORE_FMTCH, Save R2,R3,R4            ; 0658
            54 00000000G  00  9E 00002                MOVAB    EDT$$G_FMT_LNPOS, R4
            53 00000000G  00  9E 00009                MOVAB    EDT$$A_FMT_CUR, R3
            50 00000000G  00  9E 00010                MOVAB    END_BUF, R0                               ; 0712
            50            63  D1 00017                CMPL     EDT$$A_FMT_CUR, R0
                          0D  12 0001A                BNEQ     1$
            52            64  D0 0001C                MOVL     EDT$$G_FMT_LNPOS, SAV_LNPOS               ; 0715
   00000000G 00           00  FB 0001F                CALLS    #0, EDT$$OUT_FMTBUF                       ; 0716
            64            52  D0 00026                MOVL     SAV_LNPOS, EDT$$G_FMT_LNPOS               ; 0717
            50            63  D0 00029 1$:             MOVL     EDT$$A_FMT_CUR, R0                        ; 0720
            60       04   AC  90 0002C                MOVB     C, (R0)
                          63  D6 00030                INCL     EDT$$A_FMT_CUR
                          04 00032                    RET                                               ; 0721
```

. Routine Size: 51 bytes,   Routine Base: _EDT$CODE + 0000

```
;   152    0722  1
;   153    0723  1  !<BLF/PAGE>
```

B 5

EDT$FPUTCHAR   EDT$FPUTCHAR - store a character in the format 16-Sep-1984 00:28:45   VAX-11 Bliss-32 V4.0-742   Page 5   EDT$
V04-000   EDT$$STORE_FMTCH - store a character in the fo 14-Sep-1984 12:23:12   DISK$VMSMASTER:[EDT.SRC]FPUTCHAR.BLI;1   (4)   V04-

```
;  155        0724 1 END                                              ! of module EDT$FPUTCHAR
;  156        0725 1
;  157        0726 0 ELUDOM
```

;
;
;   PSECT SUMMARY
;
;   Name   Bytes   Attributes
;
;   _EDT$CODE   51 NOVEC,NOWRT, RD , EXE, SHR, LCL, REL, CON, PIC,ALIGN(2)   ; R
;
;   ; ‾
;   ; 1
;   Library Statistics
;
;   -------- Symbols --------   Pages   Processing
;   File   Total   Loaded   Percent   Mapped   Time
;
;   _255DUA28:[EDT.SRC]EDT.L32;1   377   2   0   40   00:00.2
;   _255DUA28:[EDT.SRC]PSECTS.L32;1   2   1   50   7   00:00.1
;
;
;
;
;   COMMAND QUALIFIERS
;
;   BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS$:FPUTCHAR/OBJ=OBJ$:FPUTCHAR MSRC$:FPUTCHAR.BLI/UPDATE=(ENH$:FPUT
;   CHAR)

; Size: 51 code + 0 data bytes
; Run Time: 00:09.6
; Elapsed Time: 00:11.3
; Lines/CPU Min: 4518
; Lexemes/CPU-Min: 12404
; Memory Used: 57 pages
; Compilation Complete

FPUTMES
LIS

FPUT
LIS

FTEXT
LIS

INIT
LIS

FIXNOTRUN
LIS

HANDLER
LIS

FPUTCHAR
LIS

GETXLATE
LIS

IOMOD
LIS

FLITERAL
LIS

FSTRING
LIS

INPUT
LIS

FIXNOTRUN
LIS

HELP
LIS