C 10

```
EEEEEEEEEEEEEEE  DDDDDDDDDDDD     TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD     TTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD     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
EEEEEEEEEEEEEEE  DDDDDDDDDDDD           TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD           TTT
EEEEEEEEEEEEEEE  DDDDDDDDDDDD           TTT
```

```
EXE

Mod
---
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
SYS
LBR
LIB
```

K 7

```
**FILE**ID**TICHAR
```

```
TTTTTTTTTT    IIIIII      CCCCCCCC  HH      HH    AAAAAA    RRRRRRRR
TTTTTTTTTT    IIIIII      CCCCCCCC  HH      HH    AAAAAA    RRRRRRRR
    TT          II      CC          HH      HH  AA      AA  RR      RR
    TT          II      CC          HH      HH  AA      AA  RR      RR
    TT          II      CC          HH      HH  AA      AA  RR      RR
    TT          II      CC          HH      HH  AA      AA  RR      RR
    TT          II      CC          HHHHHHHHHH  AA      AA  RRRRRRRR
    TT          II      CC          HHHHHHHHHH  AA      AA  RRRRRRRR
    TT          II      CC          HH      HH  AAAAAAAAAA  RR  RR
    TT          II      CC          HH      HH  AAAAAAAAAA  RR  RR
    TT          II      CC          HH      HH  AA      AA  RR    RR        ....
    TT          II      CC          HH      HH  AA      AA  RR    RR        ....
    TT        IIIIII      CCCCCCCC  HH      HH  AA      AA  RR      RR      ....
    TT        IIIIII      CCCCCCCC  HH      HH  AA      AA  RR      RR      ....

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
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

ED
VO
;

```
0 %TITLE 'EDT$TICHAR - input a character'
0 MODULE EDT$TICHAR (                                   ! Input a character
0               IDENT = 'V04-000'                       ! File: TICHAR.BLI Edit: REM1018
0               ) =
1 BEGIN
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1
1 !++
1 ! FACILITY:     EDT -- The DEC Standard Editor
1 !
1 ! ABSTRACT:
1 !
1 !       Input a character.
1 !
1 ! ENVIRONMENT:  Runs at any access mode - AST reentrant
1 !
1 ! AUTHOR: Bob Kushlis, CREATION DATE: June 9, 1979
1 !
1 ! MODIFIED BY:
1 !
1 ! 1-001 - Original.  DJS 18-FEB-1981.  This module was created by
1 !       extracting routine FDT$$TI_INPCH  from module TINPUT.
1 ! 1-002 - Regularize headers.  JBS 10-Mar-1981
1 ! 1-003 - Revise journaling.  JBS 22-Jun-1981
1 ! 1-004 - Fix reading journal file.  JBS 08-Jul-1981
1 ! 1-005 - Don't flush the journal record in the middle of an escape sequence.
1 !         JBS 08-Jul-1981
1 ! 1-006 - Return 0 if the journal file ends.  JBS 01-Oct-1981
1 ! 1-007 - Use the global prompt string, if specified.  JBS 21-Oct-1981
1 ! 1-008 - Remove length of prompt string.  JBS 23-Oct-1981
1 ! 1-009 - Revise autorepeat logic, putting it in a subroutine.  JBS 30-Jan-1982
1 ! 1-010 - Flush the terminal buffer after turning off Autorepeat.
1 !         JBS 31-Jan-1982
1 ! 1-011 - Add EDT$$G_JOU_VALID.  JBS 09-Apr-1982
```

```
:    58          0058  1 ! 1-012 - Allow for 8-bit keyboards.  JBS 17-Aug-1982
:    59          0059  1 ! 1-013 - Add SS3 for 8-bit keyboards.  JBS 20-Aug-1982
:    60          0060  1 ! 1-014 - Revise read-ahead logic.  JBS 02-Sep-1982
:    61          0061  1 ! 1-015 - Don't write out to the journal file here. 07-Oct-1982
:    62          0062  1 ! 1-016 - Add VT220 support conditional.  JBS 11-Feb-1983
:    63          0063  1 ! 1-017 - Added logic to maintain EDT$$G_TIN_OBUFPOS durring /RECOVERY mode.
:    64          0064  1 !         REM 10-Oct-1983
:    65          0065  1 ! 1-018 - Added more logic to reset the ^C counter & logic if about to read
:    66          0066  1 !         ANOTHER journal record.  REM 18-Oct-1983
:    67          0067  1 !--
:    68          0068  1
:    70        0069  1 %SBTTL 'Declarations'
:    71        0070  1 !
:    72        0071  1 ! TABLE OF CONTENTS:
:    73        0072  1 !
:    74        0073  1
:    75        0074  1 REQUIRE 'EDTSRC:TRAROUNAM';
:    76        0513  1
:    77        0514  1 FORWARD ROUTINE
:    78        0515  1     EDT$$TI_INPCH;
:    79        0516  1
:    80        0517  1 !
:    81        0518  1 ! INCLUDE FILES:
:    82        0519  1 !
:    83        0520  1
:    84        0521  1 REQUIRE 'EDTSRC:EDTREQ';
:    85        0656  1
:    86        0657  1 LIBRARY 'EDTSRC:SUPPORTS';
:    87        0658  1
:    88        0659  1 !
:    89        0660  1 ! MACROS:
:    90        0661  1 !
:    91        0662  1 !       NONE
:    92        0663  1 !
:    93        0664  1 ! EQUATED SYMBOLS:
:    94        0665  1 !
:    95        0666  1 !       NONE
:    96        0667  1 !
:    97        0668  1 ! OWN STORAGE:
:    98        0669  1 !
:    99        0670  1 !       NONE
:   100        0671  1 !
:   101        0672  ' ! EXTERNAL REFERENCES:
:   102        0673  1 !
:   103        0674  1 !       In the routine
```

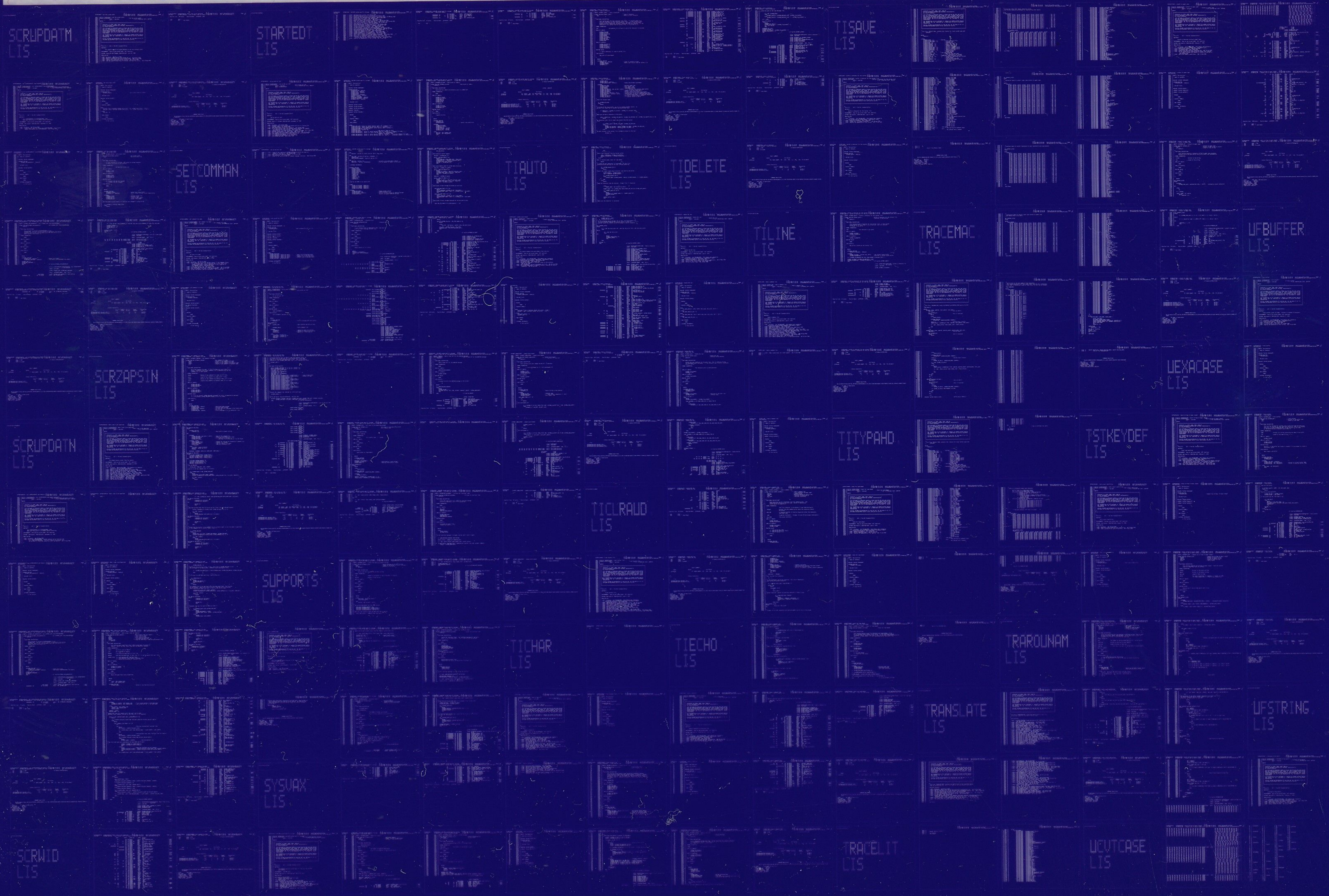
```
1 %SBTTL 'EDT$$TI_INPCH  - input a character'
1
1 GLOBAL ROUTINE EDT$$TI_INPCH (                       ! Input a character
1     BUF                                              ! Where to store character
1     ) =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine gets the next character from input.  If recovery mode is on
1 !       then the character is read from the journal file.   If not, then we check
1 !       to see if the character was read previously, and take it from the type-
1 !       ahead character; otherwise, we read a chraracter from the terminal.  If the
1 !       character is not coming from the journal file, it is written to it.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !  BUF                  the address of a fullword in which the character is returned.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       EDT$$G_RCOV_MOD
1 !       EDT$$G_K_AUTRPT
1 !       EDT$$G_TIN_IRECEND
1 !       EDT$$G_RDAHED
1 !       EDT$$T_RDAHED
1 !       EDT$$T_TIN_INREC
1 !       EDT$$A_TIN_IBUFPTR
1 !       EDT$$T_PMT_KPD
1 !       EDT$$T_PMT_NOKPD
1 !       EDT$$G_KPAD
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       EDT$$G_TIN_IRECLEN
1 !       EDT$$G_RDAHED
1 !       EDT$$A_TIN_IBUFPTR
1 !       EDT$$G_JOU_VALID
1 !       EDT$$G_TIN_OBUFPOS
1 !
1 ! ROUTINE VALUE:
1 !
1 !       1 = got a character, 0 = end of journal file
1 !
1 ! SIDE EFFECTS:
1 !
1 !       May turn off auto-repeat.
1 !
1 !--
1
2     BEGIN
2
2     EXTERNAL ROUTINE
2         edt$$clr_cc : NOVALUE,                       ! Clear ^C counters & logic
2         EDT$$RD_JOUTXT,                              ! Read a text record from the journal file
2         EDT$$OUT_FMTBUF,
2         EDT$$FMT_LIT,
;  162      0732  2          EDT$$TI_GETCH,
;  163      0733  2          EDT$$TI_BUFCH : NOVALUE,
;  164      0734  2          EDT$$TI_ENBLAUTREP : NOVALUE;          ! Enable and disable autorepeat
;  165      0735  2
;  166      0736  2      EXTERNAL
;  167      0737  2          EDT$$G_RCOV_MOD,                       ! Recovery mode flag
;  168      0738  2          EDT$$G_K_AUTRPT,                       ! Auto repeat flag
;  169      0739  2          EDT$$G_RDAHED,                         ! Type ahead counter
;  170      0740  2          EDT$$T_RDAHED,                         ! Type-ahead string
;  171      0741  2          EDT$$A_TIN_IBUFPTR : REF VECTOR [, BYTE],      ! Pointer into journal input buffer
;  172      0742  2          EDT$$G_TIN_IRECLEN,                    ! Length of journal input record
;  173      0743  2          EDT$$T_TIN_INREC,                      ! Journal input record buffer
;  174      0744  2          EDT$$G_TIN_IRECEND,                    ! Pointer to end of journal input buffer
;  175      0745  2          EDT$$T_PMT_KPD : VECTOR [, BYTE],      ! Counted ASCII string for keypad mode prompt
;  176      0746  2          EDT$$T_PMT_NOKPD : VECTOR [, BYTE],    ! Counted ASCII string for nokeypad mode prompt
;  177      0747  2          EDT$$G_KPAD,                           ! 1 = keypad change mode, 0 = nokeypad change mode
;  178      0748  2          EDT$$G_TIN_OBUFPOS,                    ! Position in journal output buffer
;  179      0749  2          EDT$$G_JOU_VALID;                      ! 1 = journal record is valid
;  180      0750  2
;  181      0751  2  !+
;  182      0752  2  ! Check for recovery mode.
;  183      0753  2  !-
;  184      0754  2
;  185      0755  2      IF .EDT$$G_RCOV_MOD
;  186      0756  2      THEN
;  187      0757  3          BEGIN
;  188      0758  3
;  189      0759  3  !+
;  190      0760  3  ! If at end of the input buffer, we will be reading another record.  So
;  191      0761  3  ! we must first reset the CC counters and associated logic.
;  192      0762  3  !-
;  193      0763  3
;  194      0764  3          IF CH$PTR_EQL (.edt$$a_tin_ibufptr, .edt$$g_tin_irecend) THEN
;  195      0765  3              edt$$clr_cc ();
;  196      0766  3
;  197      0767  3  !+
;  198      0768  3  ! Make sure we have a character in the buffer.
;  199      0769  3  !-
;  200      0770  3
;  201      0771  3          WHILE (CH$PTR_EQL (.EDT$$A_TIN_IBUFPTR, .EDT$$G_TIN_IRECEND) OR (.EDT$$A_TIN_IBUFPTR EQL 0)) DO
;  202      0772  4              BEGIN
;  203      0773  4  !+
;  204      0774  4  ! No characters left, try to read a new record from the journal
;  205      0775  4  ! file.
;  206      0776  4  !+
;  207      0777  4
;  208      0778  4              IF EDT$$RD_JOUTXT (EDT$$T_TIN_INREC, EDT$$G_TIN_IRECLEN)
;  209      0779  4              THEN
;  210      0780  5                  BEGIN
;  211      0781  5                  EDT$$G_TIN_IRECEND = CH$PTR (EDT$$T_TIN_INREC, .EDT$$G_TIN_IRECLEN);
;  212      0782  5                  EDT$$A_TIN_IBUFPTR = CH$PTR (EDT$$T_TIN_INREC, 0);
;  213      0783  5                  END
;  214      0784  4              ELSE
;  215      0785  4  !+
;  216      0786  4  ! No more records in the journal file, arrange to return to the root.
;  217      0787  4  !-
;  218      0788  4                  RETURN (0);
;  219          0789  4
;  220          0790  3             END;
;  221          0791  3
;  222          0792  3 !+
;  223          0793  3 ! Return next character from the journal buffer.
;  224          0794  3 !-
;  225          0795  3         .BUF = CH$RCHAR_A (EDT$$A_TIN_IBUFPTR);
;  226          0796  3         EDT$$G_TIN_OBUFPOS = .EDT$$G_TIN_OBUFPOS + 1
;  227          0797  3         END
;  228          0798  2     ELSE
;  229          0799  2 !+
;  230          0800  2 ! Not in recovery mode, check for type ahead.
;  231          0801  2 !-
;  232          0802  3         BEGIN
;  233          0803  3 !+
;  234          0804  3 ! If there are any characters in type-ahead, use the first.
;  235          0805  3 ! Otherwise block waiting for the user to type something.
;  236          0806  3 !-
;  237          0807  3
;  238          0808  4         IF (.EDT$$G_RDAHED GTR 0)
;  239          0809  3         THEN
;  240          0810  4             BEGIN
;  241          0811  4 !+
;  242          0812  4 ! Return the oldest typed ahead character, and shuffle the buffer.
;  243          0813  4 !-
;  244          0814  4             EDT$$G_RDAHED = .EDT$$G_RDAHED - 1;
;  245          0815  4             .BUF = CH$RCHAR (EDT$$T_RDAHED);
;  246          0816  4
;  247          0817  5             IF (.EDT$$G_RDAHED GTR 0)           !
;  248          0818  4             THEN
;  249          0819  4                 EDT$$CPY_MEM (.EDT$$G_RDAHED, CH$PLUS (EDT$$T_RDAHED, 1), EDT$$T_RDAHED);
;  250          0820  4
;  251          0821  4             END
;  252          0822  3         ELSE
;  253          0823  4             BEGIN
;  254          0824  4 !+
;  255          0825  4 ! Read the character from the terminal.  Prompt first if requested.
;  256          0826  4 !-
;  257          0827  4
;  258          0828  4             LOCAL
;  259          0829  4                 PROMPT_ADDR : REF VECTOR [8, BYTE];
;  260          0830  4
;  261          0831  4             PROMPT_ADDR = (IF .EDT$$G_KPAD THEN EDT$$T_PMT_KPD ELSE EDT$$T_PMT_NOKPD);
;  262          0832  4
;  263          0833  5             IF (.PROMPT_ADDR [0] GTR 0)
;  264          0834  4             THEN
;  265          0835  5                 BEGIN
;  266          0836  5                 EDT$$FMT_LIT (PROMPT_ADDR [1], .PROMPT_ADDR [0]);
;  267          0837  5                 EDT$$OUT_FMTBUF ();
;  268          0838  4                 END;
;  269          0839  4
;  270          0840  4             EDT$$TI_GETCH (.BUF);
;  271          0841  3             END;
;  272          0842  3
;  273          0843  3 !+
;  274          0844  3 ! Make sure the character is journaled.
;  275          0845  3 !-
;  276        0846  3             EDT$$TI_BUFCH (..BUF);
;  277        0847  3             EDT$$G_JOU_VALID = 1;
;  278        0848  3 !+
;  279        0849  3 ! If the character was an escape, CSI or SS3, turn auto-repeat off.
;  280        0850  3 ! This also disables dumping the journal buffer, so that a
;  281        0851  3 ! journal record can contain an entire escape or control sequence.
;  282        0852  3 !-
;  283        0853  3
;  284      L 0854  3 %IF SUPPORT_VT220
;  285        0855  3 %THEN
;  286        0856  3
;  287        0857  4             IF ((..BUF EQL ASC_K_ESC) OR (..BUF EQL ASC_K_CSI) OR (..BUF EQL ASC_K_SS3))
;  288        0858  3             THEN
;  289      U 0859  3 %ELSE
;  290      U 0860  3
;  291      U 0861  3                 IF (..BUF EQL ASC_K_ESC)
;  292      U 0862  3                 THEN
;  293        0863  3 %FI
;  294        0864  3
;  295        0865  4                     BEGIN
;  296        0866  4                     EDT$$TI_ENBLAUTREP (0);
;  297        0867  4                     EDT$$OUT_FMTBUF ();
;  298        0868  3                     END;
;  299        0869  3
;  300        0870  2             END;
;  301        0871  2
;  302        0872  2         RETURN (1);
;  303        0873  1         END;                                        ! of routine EDT$$TI_INPCH


                                                                        .TITLE  EDT$TICHAR EDT$TICHAR - input a character
                                                                        .IDENT  \V04-000\

                                                                        .EXTRN  EDT$$CLR_CC, EDT$$RD_JOUTXT
                                                                        .EXTRN  EDT$$OUT_FMTBUF
                                                                        .EXTRN  EDT$$FMT_LIT, EDT$$TI_GETCH
                                                                        .EXTRN  EDT$$TI_BUFCH, EDT$$TI_ENBLAUTREP
                                                                        .EXTRN  EDT$$G_RCOV_MOD
                                                                        .EXTRN  EDT$$G_K_AUTRPT
                                                                        .EXTRN  EDT$$G_RDAHED, EDT$$T_RDAHED
                                                                        .EXTRN  EDT$$A_TIN_IBUFPTR
                                                                        .EXTRN  EDT$$G_TIN_IRECLEN
                                                                        .EXTRN  EDT$$T_TIN_INREC
                                                                        .EXTRN  EDT$$G_TIN_IRECEND
                                                                        .EXTRN  EDT$$T_PMT_KPD, EDT$$T_PMT_NOKPD
                                                                        .EXTRN  EDT$$G_KPAD, EDT$$G_TIN_OBUFPOS
                                                                        .EXTRN  EDT$$G_JOU_VALID

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

                                                     0FFC 00000         .ENTRY  EDT$$TI_INPCH, Save R2,R3,R4,R5,R6,R7,R8,-   ; 0677
                                                                                R9,R10,R11                                   ;
                                  5B 00000000G  00   9E 00002           MOVAB   EDT$$G_TIN_IRECLEN, R11                      ;
                                  5A 00000000G  00   9E 00009           MOVAB   EDT$$T_RDAHED, R10                           ;
                                  59 00000000G  00   9E 00010           MOVAB   EDT$$G_RDAHED, R9                            ;
                                  58 00000000G  00   9E 00017           MOVAB   EDT$$T_TIN_INREC, R8                         ;
                                  57 00000000G  00   9E 0001E           MOVAB   EDT$$G_TIN_IRECEND, R7                       ;
```

```
               56 00000000G  00  9E 00025          MOVAB   EDT$$A_TIN_IBUFPTR, R6                        ;
               47 00000000G  00  E9 0002C          BLBC    EDT$$G_RCOV_MOD, 5$                           ; 0755
               67            66  D1 00033          CMPL    EDT$$A_TIN_IBUFPTR, EDT$$G_TIN_IRECEND        ; 0764
                             07  12 00036          BNEQ    1$                                            ;
   00000000G   00            00  FB 00038          CALLS   #0, EDT$$CLR_CC                               ; 0765
               50            66  D0 0003F 1$:      MOVL    EDT$$A_TIN_IBUFPTR, R0                        ; 0771
               67            50  D1 00042          CMPL    R0, EDT$$G_TIN_IRECEND                        ;
                             04  13 00045          BEQL    2$                                            ;
                             50  D5 00047          TSTL    R0                                            ;
                             1D  12 00049          BNEQ    4$                                            ;
                      0900   8F  BB 0004B 2$:      PUSHR   #^M<R8,R11>                                   ; 0778
   00000000G   00            02  FB 0004F          CALLS   #2, EDT$$RD_JOUTXT                            ;
               03            50  E8 00056          BLBS    R0, 3$                                        ;
                           00AC  31 00059          BRW     13$                                           ;
               50            68  9E 0005C 3$:      MOVAB   EDT$$T_TIN_INREC, R0                          ; 0781
67             50            6B  C1 0005F          ADDL3   EDT$$G_TIN_IRECLEN, R0, EDT$$G_TIN_IRECEND    ;
               66            68  9E 00063          MOVAB   EDT$$T_TIN_INREC, EDT$$A_TIN_IBUFPTR          ; 0782
                             D7  11 00066          BRB     1$                                            ; 0778
               50            66  D0 00068 4$:      MOVL    EDT$$A_TIN_IBUFPTR, R0                        ; 0795
         04    BC            60  9A 0006B          MOVZBL  (R0), @BUF                                    ;
                             66  D6 0006F          INCL    EDT$$A_TIN_IBUFPTR                            ;
                  00000000G  00  D6 00071          INCL    EDT$$G_TIN_OBUFPOS                            ; 0796
                           008A  31 00077          BRW     12$                                           ;
                             69  D5 0007A 5$:      TSTL    EDT$$G_RDAHED                                 ; 0808
                             12  15 0007C          BLEQ    6$                                            ;
                             69  D7 0007E          DECL    EDT$$G_RDAHED                                 ; 0814
         04    BC            6A  9A 00080          MOVZBL  EDT$$T_RDAHED, @BUF                           ; 0815
               50            69  D0 00084          MOVL    EDT$$G_RDAHED, R0                             ; 0817
                             40  15 00087          BLEQ    10$                                           ;
6A       01    AA            50  28 00089          MOVC3   R0, EDT$$T_RDAHED+1, EDT$$T_RDAHED            ; 0819
                             39  11 0008E          BRB     10$                                           ; 0808
               09 00000000G  00  E9 00090 6$:      BLBC    EDT$$G_KPAD, 7$                               ; 0831
               50 00000000G  00  9E 00097          MOVAB   EDT$$T_PMT_KPD, PROMPT_ADDR                   ;
                             07  11 0009E          BRB     8$                                            ;
               50 00000000G  00  9E 000A0 7$:      MOVAB   EDT$$T_PMT_NOKPD, PROMPT_ADDR                 ;
                             60  95 000A7 8$:      TSTB    (PROMPT_ADDR)                                 ; 0833
                             14  13 000A9          BEQL    9$                                            ;
               7E            60  9A 000AB          MOVZBL  (PROMPT_ADDR), -(SP)                          ; 0836
                        01   A0  9F 000AE          PUSHAB  1(PROMPT_ADDR)                                ;
   00000000G   00            02  FB 000B1          CALLS   #2, EDT$$FMT_LIT                              ;
   00000000G   00            00  FB 000B8          CALLS   #0, EDT$$OUT_FMTBUF                           ; 0837
                        04   AC  DD 000BF 9$:      PUSHL   BUF                                           ; 0840
   00000000G   00            01  FB 000C2          CALLS   #1, EDT$$TI_GETCH                             ;
                        04   BC  DD 000C9 10$:     PUSHL   @BUF                                          ; 0846
   00000000G   00            01  FB 000CC          CALLS   #1, EDT$$TI_BUFCH                             ;
   00000000G   00            01  D0 000D3          MOVL    #1, EDT$$G_JOU_VALID                          ; 0847
               1B       04   BC  D1 000DA          CMPL    @BUF, #27                                     ; 0857
                             14  13 000DE          BEQL    11$                                           ;
   0000009B    8F       04   BC  D1 000E0          CMPL    @BUF, #155                                    ;
                             0A  13 000E8          BEQL    11$                                           ;
   0000008F    8F       04   BC  D1 000EA          CMPL    @BUF, #143                                    ;
                             10  12 000F2          BNEQ    12$                                           ;
                             7E  D4 000F4 11$:     CLRL    -(SP)                                         ; 0866
   00000000G   00            01  FB 000F6          CALLS   #1, EDT$$TI_ENBLAUTREP                        ;
   00000000G   00            00  FB 000FD          CALLS   #0, EDT$$OUT_FMTBUF                           ; 0867
               50            01  D0 00104 12$:     MOVL    #1, R0                                        ; 0872
                                 04 00107          RET                                                   ;
```

```
                                                   50  D4 00108 13$:     CLRL     R0                                       ; 0873
                                                       04 0010A          RET                                               ;

; Routine Size:  267 bytes,    Routine Base:  _EDT$CODE + 0000


;  304            0874  1
;  305            0875  1 !<BLF/PAGE>
;   307          0876   1 END                                                   ! of module EDT$TICHAR
;   308          0877   1
;   309          0878   0 ELUDOM




;                                            PSECT SUMMARY
;
;        Name                           Bytes                         Attributes
;
;   _EDT$CODE                                267  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)




;                                    Library Statistics
;
;                                                    -------- Symbols --------       Pages        Processing
;        File                                         Total    Loaded   Percent       Mapped       Time
;
;   _$255$DUA28:[EDT.SRC]EDT.L32;1                      377         4         1          40         00:00.2
;   _$255$DUA28:[EDT.SRC]PSECTS.L32;1                     2         1        50           7         00:00.1
;   _$255$DUA28:[EDT.SRC]SUPPORTS.L32;1                   2         1        50           5         00:00.1





;                                              COMMAND QUALIFIERS

;        BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS$:TICHAR/OBJ=OBJ$:TICHAR MSRC$:TICHAR.BLI/UPDATE=(ENH$:TICHAR)

; Size:          267 code + 0 data bytes
; Run Time:         00:16.1
; Elapsed Time:     00:21.2
; Lines/CPU Min:      3280
; Lexemes/CPU-Min:   9332
; Memory Used:  103 pages
; Compilation Complete
```

0140 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



SCRUPDATM
LIS

STARTEDT
LIS

TISAVE
LIS

SETCOMMAN
LIS

TIAUTO
LIS

TIDELETE
LIS

TILINE
LIS

TRACEMAC
LIS

UFBUFFER
LIS

SCRZAPSIN
LIS

UEXACASE
LIS

SCRUPDATN
LIS

TITYPAHD
LIS

TSTKEYDEF
LIS

TICLRAUD
LIS

SUPPORTS
LIS

TICHAR
LIS

TIECHO
LIS

TRAROUNAM
LIS

TRANSLATE
LIS

UFSTRING
LIS

SYSVAX
LIS

SCRWID
LIS

TRACELIT
LIS

UCVTCASE
LIS