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

Moc
---
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
ED1
SYS
LBF
LIE
```

```
**FILE**ID**PRGETCHR                                   M 14




PPPPPPPPP   RRRRRRRRR     GGGGGGGG  EEEEEEEEEE  TTTTTTTTTT    CCCCCCCC  HH      HH  RRRRRRRRR
PPPPPPPPP   RRRRRRRRR     GGGGGGGG  EEEEEEEEEE  TTTTTTTTTT    CCCCCCCC  HH      HH  RRRRRRRRR
PP      PP  RR      RR  GG          EE              TT      CC          HH      HH  RR      RR
PP      PP  RR      RR  GG          EE              TT      CC          HH      HH  RR      RR
PP      PP  RR      RR  GG          EE              TT      CC          HH      HH  RR      RR
PP      PP  RR      RR  GG          EE              TT      CC          HH      HH  RR      RR
PPPPPPPPP   RRRRRRRRR   GG          EEEEEEEE        TT      CC          HHHHHHHHHH  RRRRRRRRR
PPPPPPPPP   RRRRRRRRR   GG          EEEEEEEE        TT      CC          HHHHHHHHHH  RRRRRRRRR
PP          RR  RR      GG  GGGGGG  EE              TT      CC          HH      HH  RR  RR
PP          RR  RR      GG  GGGGGG  EE              TT      CC          HH      HH  RR  RR
PP          RR    RR    GG      GG  EE              TT      CC          HH      HH  RR    RR         ....
PP          RR    RR    GG      GG  EE              TT      CC          HH      HH  RR    RR         ....
PP          RR      RR    GGGGGG    EEEEEEEEEE      TT        CCCCCCCC  HH      HH  RR      RR       ....
PP          RR      RR    GGGGGG    EEEEEEEEEE      TT        CCCCCCCC  HH      HH  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
```

EDT
V04

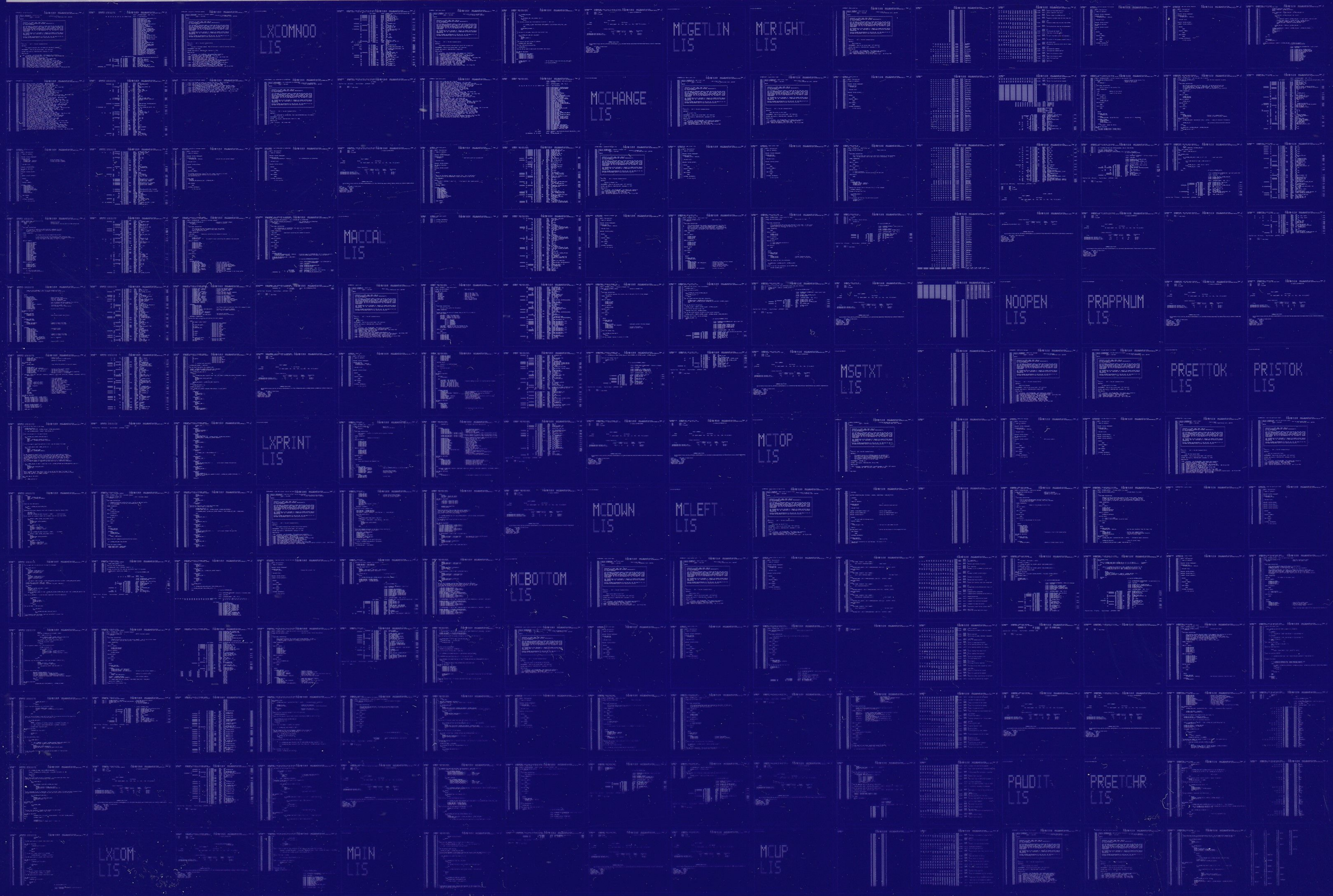
```
%TITLE 'EDT$PRGETCHR - get next parse character'
MODULE EDT$PRGETCHR (                           ! Get next parse character
                IDENT = 'V04-000'                       ! File: PRGETCHR.BLI Edit: JBS1005
                ) =
BEGIN
!
!****************************************************************************
!*                                                                          *
!*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
!*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
!*  ALL RIGHTS RESERVED.                                                    *
!*                                                                          *
!*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
!*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
!*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
!*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
!*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
!*  TRANSFERRED.                                                            *
!*                                                                          *
!*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
!*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
!*  CORPORATION.                                                            *
!*                                                                          *
!*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
!*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
!*                                                                          *
!*                                                                          *
!****************************************************************************
!


!++
! FACILITY:     EDT -- The DEC Standard Editor
!
! ABSTRACT:
!
!       Get the next character to be parsed.
!
! ENVIRONMENT:  Runs at any access mode - AST reentrant
!
! AUTHOR: Bob Kushlis, CREATION DATE: December 12, 1978
!
! MODIFIED BY:
!
! 1-001 - Original.  DJS 25-Feb-1981.  This module was created by
!       extracting routine GET_CHAR from module PARSER.
! 1-002 - Regularize headers.  JBS 12-Mar-1981
! 1-003 - Change alphabetic processing.  JBS 19-Jul-1982
! 1-004 - Don't strip diacritical marks.  JBS 13-Dec-1982
! 1-005 - Add VT220 support conditional.  JBS 11-Feb-1983
!--

```

EDT
V04

```
;    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$$PA_GETCH : NOVALUE;
;    62         0499   1
;    63         0500   1 !
;    64         0501   1 ! INCLUDE FILES:
;    65         0502   1 !
;    66         0503   1
;    67         0504   1 REQUIRE 'EDTSRC:EDTREQ';
;    68         0639   1
;    69         0640   1 LIBRARY 'EDTSRC:SUPPORTS';
;    70         0641   1
;    71         0642   1 !
;    72         0643   1 ! MACROS:
;    73         0644   1 !
;    74         0645   1 !       NONE
;    75         0646   1 !
;    76         0647   1 ! EQUATED SYMBOLS:
;    77         0648   1 !
;    78         0649   1 !       NONE
;    79         0650   1 !
;    80         0651   1 ! OWN STORAGE:
;    81         0652   1 !
;    82         0653   1 !       NONE
;    83         0654   1 !
;    84         0655   1 ! EXTERNAL REFERENCES:
;    85         0656   1 !
;    86         0657   1 !       In the routine
;    88         0658  1 %SBTTL 'EDT$$PA_GETCH  - get the next char to parse'
;    89         0659  1
;    90         0660  1 GLOBAL ROUTINE EDT$$PA_GETCH                          ! Get the next character to parse
;    91         0661  1     : NOVALUE =
;    92         0662  1
;    93         0663  1 !++
;    94         0664  1 ! FUNCTIONAL DESCRIPTION:
;    95         0665  1 !
;    96         0666  1 !       Get the next character from the input line.  If we are pointing at the
;    97         0667  1 !       end of the record, return a '!', otherwise, get the character pointed
;    98         0668  1 !       to by EDT$$A_CMD_BUF  and bump EDT$$A_CMD_BUF .  The character
;    99         0669  1 !       is placed in the global EDT$$C_PA_CH.  Lower case alphabetics are converted
;   100         0670  1 !       to upper case.
;   101         0671  1 !
;   102         0672  1 ! FORMAL PARAMETERS:
;   103         0673  1 !
;   104         0674  1 !       NONE
;   105         0675  1 !
;   106         0676  1 ! IMPLICIT INPUTS:
;   107         0677  1 !
;   108         0678  1 !       EDT$$A_CMD_BUF
;   109         0679  1 !       EDT$$A_CMD_END
;   110         0680  1 !
;   111         0681  1 ! IMPLICIT OUTPUTS:
;   112         0682  1 !
;   113         0683  1 !       EDT$$C_PA_CH
;   114         0684  1 !
;   115         0685  1 ! ROUTINE VALUE:
;   116         0686  1 !
;   117         0687  1 !       NONE
;   118         0688  1 !
;   119         0689  1 ! SIDE EFFECTS:
;   120         0690  1 !
;   121         0691  1 !       NONE
;   122         0692  1 !
;   123         0693  1 !--
;   124         0694  1
;   125         0695  2     BEGIN
;   126         0696  2
;   127         0697  2     EXTERNAL
;   128         0698  2         EDT$$A_CMD_BUF,                               ! Pointer into command buffer.
;   129         0699  2         EDT$$A_CMD_END,                               ! Pointer to end of current command.
;   130         0700  2
;   131       L 0701  2 %IF SUPPORT_VT220
;   132         0702  2 %THEN
;   133         0703  2         EDT$$B_CHAR_INFO : BLOCKVECTOR [256, 1, BYTE],  ! table of character information
;   134         0704  2 %FI
;   135         0705  2
;   136         0706  2         EDT$$C_PA_CH;                                 ! the currently being processed character
;   137         0707  2
;   138         0708  2     LOCAL
;   139         0709  2         CHAR;
;   140         0710  2
;   141         .711  3     IF CH$PTR_GTR (.EDT$$A_CMD_BUF, .EDT$$A_CMD_END)
;   142         0712  2     THEN
;   143         0713  2         EDT$$C_PA_CH = %C'!'
;   144         0714  2     ELSE
;  145          0715  3          BEGIN
;  146          0716  3          CHAR = CH$RCHAR_A (EDT$$A_CMD_BUF);
;  147          0717  3
;  148        L 0718  4 %IF ( NOT SUPPORT_VT220)
;  149        U 0719  3 %THEN
;  150        U 0720  3          CHAR = .CHAR AND %X'7F';
;  151          0721  3 %FI
;  152          0722  3
;  153        L 0723  3 %IF SUPPORT_VT220
;  154          0724  3 %THEN
;  155          0725  3
;  156          0726  3          IF .EDT$$B_CHAR_INFO [.CHAR, 0, 0, 1, 0]          ! Lower case letter
;  157          0727  3          THEN
;  158        U 0728  3 %ELSE
;  159        U 0729  3
;  160        U 0730  3              IF ((.CHAR GEQ %C'a') AND (.CHAR LEQ %C'z'))
;  161        U 0731  3              THEN
;  162          0732  3 %FI
;  163          0733  3
;  164          0734  3                  EDT$$C_PA_CH = .CHAR - %C'a' + %C'A'
;  165          0735  3              ELSE
;  166          0736  3                  EDT$$C_PA_CH = .CHAR;
;  167          0737  3
;  168          0738  2          END;
;  169          0739  2
;  170          0740  1      END;                                               ! of routine EDT$$PA_GETCH


                                                                            .TITLE  EDT$PRGETCHR EDT$PRGETCHR - get next parse char
;                                                                                                    acter
                                                                            .IDENT  \V04-000\

                                                                            .EXTRN  EDT$$A_CMD_BUF, EDT$$A_CMD_END
                                                                            .EXTRN  EDT$$B_CHAR_INFO
                                                                            .EXTRN  EDT$$C_PA_CH

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

                                                         000C 00000         .ENTRY  EDT$$PA_GETCH, Save R2,R3                        ; 0660
                                        53 00000000G  00   9E 00002         MOVAB   EDT$$A_CMD_BUF, R3                               ;
                                        52 00000000G  00   9E 00009         MOVAB   EDT$$C_PA_CH, R2                                 ;
                                 00000000G  00         63  D1 00010         CMPL    EDT$$A_CMD_BUF, EDT$$A_CMD_END                   ; 0711
                                                       04  1B 00017         BLEQU   1$                                               ;
                                            62         21  D0 00019         MOVL    #33, EDT$$C_PA_CH                                ; 0713
                                                           04 0001C         RET                                                      ;
                                            51         63  D0 0001D 1$:     MOVL    EDT$$A_CMD_BUF, R1                               ; 0716
                                            50         61  9A 00020         MOVZBL  (R1), CHAR                                       ;
                                                       63  D6 00023         INCL    EDT$$A_CMD_BUF                                   ;
                              05 00000000G0040         00  E1 00025         BBC     #0, EDT$$B_CHAR_INFO[CHAR], 2$                   ; 0726
                                            62     E0  A0  9E 0002E         MOVAB   -32(R0), EDT$$C_PA_CH                            ; 0734
                                                           04 00032         RET                                                      ;
                                            62         50  D0 00033 2$:     MOVL    CHAR, EDT$$C_PA_CH                               ; 0736
                                                           04 00036         RET                                                      ; 0740

; Routine Size:  55 bytes,    Routine Base:  _EDT$CODE + 0000
;  171          0741  1
;  172          0742  1 !<BLF/PAGE>
;  174           0743  1 END                                                      ! of module EDT$PRGETCHR
;  175           0744  1
;  176           0745  0 ELUDOM





;                                        PSECT SUMMARY
;
;       Name                             Bytes                      Attributes
;
;  _EDT$CODE                                  55  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         1          0           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$:PRGETCHR/OBJ=OBJ$:PRGETCHR MSRC$:PRGETCHR.BLI/UPDATE=(ENH$:PRGE
;       TCHR)

; Size:           55 code + 0 data bytes
; Run Time:          00:10.1
; Elapsed Time:      00:13.5
; Lines/CPU Min:      4425
; Lexemes/CPU-Min: 12124
; Memory Used:  58 pages
; Compilation Complete
```

0137 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY


LXCOMNOO
LIS

MCGETLIN
LIS

MCRIGHT
LIS

MCCHANGE
LIS

MACCAL
LIS

NOOPEN
LIS

PRAPPNUM
LIS

MSGTXT
LIS

PRGETTOK
LIS

PRISTOK
LIS

LXPRINT
LIS

MCTOP
LIS

MCDOWN
LIS

MCLEFT
LIS

MCBOTTOM
LIS

PAUDIT
LIS

PRGETCHR
LIS

LXCOM
LIS

MAIN
LIS

MCUP
LIS