```
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
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
```

```
EXI

Mod
---
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
ED
SYS
LBI
LII
```

N 12

```
**FILE**ID**CHMCHKCC
```

```
  CCCCCCCC  HH      HH  MM      MM    CCCCCCCC  HH      HH  KK      KK    CCCCCCCC    CCCCCCCC
  CCCCCCCC  HH      HH  MM      MM    CCCCCCCC  HH      HH  KK      KK    CCCCCCCC    CCCCCCCC
CC          HH      HH  MMMM  MMMM  CC          HH      HH  KK      KK  CC          CC
CC          HH      HH  MMMM  MMMM  CC          HH      HH  KK      KK  CC          CC
CC          HH      HH  MM  MM  MM  CC          HH      HH  KK    KK    CC          CC
CC          HH      HH  MM  MM  MM  CC          HH      HH  KK    KK    CC          CC
CC          HHHHHHHHHH  MM      MM  CC          HHHHHHHHHH  KKKKKK      CC          CC
CC          HHHHHHHHHH  MM      MM  CC          HHHHHHHHHH  KKKKKK      CC          CC
CC          HH      HH  MM      MM  CC          HH      HH  KK    KK    CC          CC
CC          HH      HH  MM      MM  CC          HH      HH  KK    KK    CC          CC
CC          HH      HH  MM      MM  CC          HH      HH  KK      KK  CC          CC                ....
CC          HH      HH  MM      MM  CC          HH      HH  KK      KK  CC          CC                ....
  CCCCCCCC  HH      HH  MM      MM    CCCCCCCC  HH      HH  KK      KK    CCCCCCCC    CCCCCCCC      ....
  CCCCCCCC  HH      HH  MM      MM    CCCCCCCC  HH      HH  KK      KK    CCCCCCCC    CCCCCCCC      ....

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
```

```
0001  0 %TITLE 'EDT$CHMCHKCC - check for control C'
0002  0 MODULE EDT$CHMCHKCC (                                    ! Check for control C
0003  0                 IDENT = 'V04-000'                                ! File: CHMCHKCC.BLI Edit: REM1016
0004  0                 ) =
0005  1 BEGIN
0006  1 !
0007  1 !****************************************************************************
0008  1 !*                                                                          *
0009  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0010  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0011  1 !*  ALL RIGHTS RESERVED.                                                    *
0012  1 !*                                                                          *
0013  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0014  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0015  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0016  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0017  1 !*  OT4ER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0018  1 !*  TFANSFERRED.                                                            *
0019  1 !*                                                                          *
0020  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0021  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0022  1 !*  CORPORATION.                                                            *
0023  1 !*                                                                          *
0024  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0025  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0026  1 !*                                                                          *
0027  1 !*                                                                          *
0028  1 !****************************************************************************
0029  1 !
0030  1
0031  1 !++
0032  1 ! FACILITY:     EDT -- The DEC Standard Editor
0033  1 !
0034  1 ! ABSTRACT:
0035  1 !
0036  1 !       This module checks to see if a CTRL/C has been typed.
0037  1 !
0038  1 ! ENVIRONMENT:  Runs at any access mode - AST reentrant
0039  1 !
0040  1 ! AUTHOR: Bob Kushlis, CREATION DATE: Unknown
0041  1 !
0042  1 ! MODIFIED BY:
0043  1 !
0044  1 ! 1-001 - Original.  DJS 04-Feb-1981.  This module was created by
0045  1 !       extracting routine EDT$$CHK_CC  from module CHANGE.BLI.
0046  1 ! 1-002 - Regularize headers and remove control C checking, since we intend to
0047  1 !         re-do it to support journaling.  JBS 27-Feb-1981
0048  1 ! 1-003 - Fix module name.  JBS 02-Mar-1981
0049  1 ! 1-004 - Revise journaling to support control C.  JBS 22-Jun-1981
0050  1 ! 1-005 - Add logic for recovering during control C.  JBS 18-Dec-1981
0051  1 ! 1-006 - Debug control C journaling.  JBS 24-Dec-1981
0052  1 ! 1-007 - Change names of control C data.  JBS 29-Dec-1981
0053  1 ! 1-008 - Use two words for control C counters.  JBS 30-Dec-1981
0054  1 ! 1-009 - Print the 'working' message from this routine.  JBS 13-Jan-1982
0055  1 ! 1-010 - Use symbols instead of magic numbers for control C journaling.  JBS 24-May-1982
0056  1 ! 1-011 - Change the format of the working message.  SMB 28-Jun-1982
0057  1 ! 1-012 - Figure out whether the journal buffer should be flushed. STS 28-Sep-1982
```

```
;    58          0058  1 ! 1-013 - Don't destroy the cursor position, it is now being maintained accurately.  JBS 07-Oct-1982
;    59          0059  1 ! 1-014 - Clear EDT$$G_SECOND after printing or erasing the working message, so
;    60          0060  1 !          we do not spend all our time printing the working message on slow terminals.  JBS 07-Oct-1982
;    61          0061  1 ! 1-015 - Move setting of EDT$$G_PUT_JOU from within the high order increment of
;    62          0062  1 !         EDT$$G_CC_CNT1_ to just after the low order increment.  It wasn't
;    63          0063  1 !         getting set and therefore, ^C handling durring /RECOVERs didn't work
;    64          0064  1 !         at all.  REM 7-Oct-1983
;    65          0065  1 ! 1-016 - Moved the above EDT$$G_PUT_JOU back!  Now, we set it if
;    66          0066  1 !         EDT$$G_TIN_OBUFPOS GEQ JOU_MAX.  REM 10-Oct-1983
;    67          0067  1 !--
;    68          0068  1
;    70          0069   1 %SBTTL 'Declarations'
;    71          0070   1 !
;    72          0071   1 ! TABLE OF CONTENTS:
;    73          0072   1 !
;    74          0073   1
;    75          0074   1 REQUIRE 'EDTSRC:TRAROUNAM';
;    76          0513   1
;    77          0514   1 FORWARD ROUTINE
;    78          0515   1     EDT$$CHK_CC;                                ! Check to see if a CTRL/C has been typed
;    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 !
;    87          0658   1 ! MACROS:
;    88          0659   1 !
;    89          0660   1 !       NONE
;    90          0661   1 !
;    91          0662   1 ! EQUATED SYMBOLS:
;    92          0663   1 !
;    93          0664   1
;    94          0665   1 LITERAL
;    95          0666   1     JOU_MAX = 20,
;    96          0667   1     WORK_COL = 26;
;    97          0668   1
;    98          0669   1 !
;    99          0670   1 ! OWN STORAGE:
;   100          0671   1 !
;   101          0672   1 !       NONE
;   102          0673   1 !
;   103          0674   1 ! EXTERNAL REFERENCES:
;   104          0675   1 !
;   105          0676   1 !       In the routine
```

```
1 %SBTTL 'EDT$$CHK_CC  - check for control C'
1
1 GLOBAL ROUTINE EDT$$CHK_CC                              ! Check for control C
1     =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine checks to see if a CTRL/C has been typed.
1 !       It is called frequently enough that the response to typing a control C
1 !       is reasonably fast.  It keeps track of how often it has been called
1 !       since the last journal record was written so it can do the same during
1 !       recovery.
1 !
1 !       This routine is also responsible for printing the 'working' message.
1 !       The flag EDT$$G_SECOND is set once a second by an AST.  If it is set
1 !       this routine clears it and either prints or erases the 'working' message.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       EDT$$A_FMT_WRRUT
1 !       EDT$$G_MESSAGE_LINE
1 !       EDT$$G_RCOV_MOD
1 !       EDT$$G_CC
1 !       EDT$$G_CC_CNT1_LO
1 !       EDT$$G_CC_CNT1_HI
1 !       EDT$$G_CC_CNT2_LO
1 !       EDT$$G_CC_CNT2_HI
1 !       EDT$$G_CC_FLAG
1 !       EDT$$G_SECOND
1 !       EDT$$G_WORKCOUNT
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       EDT$$G_CC_CNT1_LO
1 !       EDT$$G_CC_CNT1_HI
1 !       EDT$$G_CC_CNT2_LO
1 !       EDT$$G_CC_CNT2_HI
1 !       EDT$$G_CC_FLAG
1 !       EDT$$G_SECOND
1 !       EDT$$G_WORKCOUNT
1 !
1 ! ROUTINE VALUE:
1 !
1 !       0               no control C typed
1 !       1               control C was typed
1 !
1 ! SIDE EFFECTS:
1 !
1 !       May print or erase the 'working' message.
1 !
1 !--
1
```

```
;  164          0734  2       BEGIN
;  165          0735  2
;  166          0736  2       EXTERNAL
;  167          0737  2           EDT$$G_PUT_JOU,                     ! flag to put journal buffer
;  168          0738  2           EDT$$G_TIN_OBUFPOS,                 ! Position in journal output buffer
;  169          0739  2           EDT$$A_FMT_WRRUT,                   ! Address of output formatter
;  170          0740  2           EDT$$G_MESSAGE_LINE,                ! Working on this line + 1
;  171          0741  2           EDT$$G_CC : VOLATILE,               ! 1 = a control C has been typed
;  172          0742  2           EDT$$G_CC_CNT1_LO,                  ! Counts number of "no control C's" returned.
;  173          0743  2           EDT$$G_CC_CNT1_HI,                  ! High half of the above
;  174          0744  2           EDT$$G_CC_CNT2_LO,                  ! Loaded with value from control C record
;  175          0745  2           EDT$$G_CC_CNT2_HI,                  ! High half of the above
;  176          0746  2           EDT$$G_CC_FLAG,                     ! 1 = a control C record was read or written
;  177          0747  2           EDT$$G_RCOV_MOD,                    ! 1 = /RECOVER
;  178          0748  2           EDT$$G_SECOND : VOLATILE,           ! Set once a second
;  179          0749  2           EDT$$G_WORKCOUNT;                   ! Counter to support the 'working' message
;  180          0750  2
;  181          0751  2       EXTERNAL ROUTINE
;  182          0752  2           EDT$$JOU_PUTREC,
;  183          0753  2           EDT$$TI_WRSTR,                      ! String format for working
;  184          0754  2           EDT$$OUT_FMTBUF,                    ! Output format buffer to terminal
;  185          0755  2           EDT$$SC_ERATOEOL,                   ! Erase to end of current line
;  186          0756  2           EDT$$SC_POSCSIF,                    ! Absolute cursor positioning
;  187          0757  2           EDT$$MSG_TOSTR,                     ! Write out the working message
;  188          0758  2           EDT$$TI_FLUSHJOUFI : NOVALUE;       ! Write a record on the journal file
;  189          0759  2
;  190          0760  2       LOCAL
;  191          0761  2           FORMAT_ROUTINE;                     ! Save the format routine entered with
;  192          0762  2
;  193          0763  2       MESSAGES ((WORKING));
;  194          0764  2   !+
;  195          0765  2   ! If a second has passed since we were last here, print or erase the
;  196          0766  2   ! 'working' message.
;  197          0767  2   !-
;  198          0768  2
;  199          0769  3       IF (.EDT$$G_SECOND NEQ 0)
;  200          0770  2       THEN
;  201          0771  3           BEGIN
;  202          0772  3           EDT$$G_WORKCOUNT = .EDT$$G_WORKCOUNT + 1;
;  203          0773  3           FORMAT_ROUTINE = .EDT$$A_FMT_WRRUT;
;  204          0774  3           EDT$$A_FMT_WRRUT = EDT$$TI_WRSTR;
;  205          0775  3           EDT$$SC_POSCSIF (.EDT$$G_MESSAGE_LINE + 1, WORK_COL);
;  206          0776  3
;  207          0777  3           IF .EDT$$G_WORKCOUNT THEN EDT$$MSG_TOSTR (EDT$_WORKING) ELSE EDT$$SC_ERATOEOL ();
;  208          0778  3
;  209          0779  3           EDT$$OUT_FMTBUF ();
;  210          0780  3           EDT$$A_FMT_WRRUT = .FORMAT_ROUTINE;
;  211          0781  3           EDT$$G_SECOND = 0;
;  212          0782  2           END;
;  213          0783  2
;  214          0784  2   !+
;  215          0785  2   ! If we are in recovery mode, use the counters to
;  216          0786  2   ! simulate typing a control C at the right point.
;  217          0787  2   !-
;  218          0788  2
;  219          0789  2       IF .EDT$$G_RCOV_MOD
;  220          0790  2       THEN
```

```
3         BEGIN
3 !+
3 ! If EDT$$G_CC_FLAG is zero, there has been no control C record.
3 !-
3         IF .EDT$$G_CC_FLAG THEN
4             BEGIN
4 !+
4 ! If the counters match, it is time to return a control C.
4 !-
5             IF ((.EDT$$G_CC_CNT1_LO EQL .EDT$$G_CC_CNT2_LO) AND (.EDT$$G_CC_CNT1_HI EQL .EDT$$G_CC_CNT2_HI))
4             THEN
5                 BEGIN
5                 EDT$$G_PUT_JOU = 1;
5                 RETURN (1);
4                 END;
4             END
3         END
2     ELSE
3         BEGIN
3 !+
3 ! We are not recovering.
3 !-
3         IF .EDT$$G_CC
3         THEN
4             BEGIN
4 !+
4 ! A control C was typed.  Write out a control C record in case
4 ! we must recover to this point, unless we have already written it.
4 !-
4             EDT$$G_PUT_JOU = 1;                 ! indicate we must do a put
4
5             IF (.EDT$$G_CC_FLAG EQL 0)
4             THEN
5                 BEGIN
5                 EDT$$TI_FLUSHJOUFI (%C'C');
5                 EDT$$G_CC_FLAG = 1;
4                 END;
4
4             RETURN (1);
3             END;
2         END;
2
2 !+
2 ! Keep track of the number of times we are called but no control C has
2 ! been typed; that is, count the number of times we return 0.  This
2 ! number will be written to the journal file later, when and if we see a
2 ! control C, so that we can read it into the second counter on recovery,
2 ! and thus return 1 at the same point in EDT's execution.
2 !-
2     EDT$$G_CC_CNT1_LO = .EDT$$G_CC_CNT1_LO + 1;
2
3     IF (.EDT$$G_CC_CNT1_LO EQL CC_CTR_MAX)
2     THEN
3         BEGIN
3         EDT$$G_CC_CNT1_LO = 0;
3         EDT$$G_CC_CNT1_HI = .EDT$$G_CC_CNT1_HI + 1;
3         EDT$$G_PUT_JOU = 1;
```

```
;  278          0848  3            ASSERT (.EDT$$G_CC_CNT1_HI LEQ CC_CTR_MAX);
;  279          0849  2            END;
;  280          0850  2
;  281          0851  2 !+
;  282          0852  2 ! check to see if the journal buffer should be written out and
;  283          0853  2 ! the control-c counter restarted
;  284          0854  2 !-
;  285          0855  2
;  286          0856  2         IF (.EDT$$G_TIN_OBUFPOS GEQ JOU_MAX) THEN EDT$$G_PUT_JOU = 1;
;  287          0857  2
;  288          0858  2         RETURN (0);
;  289          0859  2
;  290          0860  1         END;                            !  of EDT$$CHK_CC


                                                                      .TITLE  EDT$CHMCHKCC EDT$CHMCHKCC - check for control C
                                                                      .IDENT  \V04-000\

                                                                      .EXTRN  EDT$$G_PUT_JOU, EDT$$G_TIN_OBUFPOS
                                                                      .EXTRN  EDT$$A_FMT_WRRUT
                                                                      .EXTRN  EDT$$G_MESSAGE_LINE
                                                                      .EXTRN  EDT$$G_CC, EDT$$G_CC_CNT1_LO
                                                                      .EXTRN  EDT$$G_CC_CNT1_HI
                                                                      .EXTRN  EDT$$G_CC_CNT2_LO
                                                                      .EXTRN  EDT$$G_CC_CNT2_HI
                                                                      .EXTRN  EDT$$G_CC_FLAG, EDT$$G_RCOV_MOD
                                                                      .EXTRN  EDT$$G_SECOND, EDT$$G_WORKCOUNT
                                                                      .EXTRN  EDT$$JOU_PUTREC
                                                                      .EXTRN  EDT$$TI_WRSTR, EDT$$OUT_FMTBUF
                                                                      .EXTRN  EDT$$SC_ERATOEOL
                                                                      .EXTRN  EDT$$SC_POSCSIF
                                                                      .EXTRN  EDT$$MSG_TOSTR, EDT$$TI_FLUSHJOUFI
                                                                      .EXTRN  EDT$_WORKING, EDT$$INTER_ERR

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

                                                  03FC 00000          .ENTRY  EDT$$CHK_CC, Save R2,R3,R4,R5,R6,R7,R8,R9   ; 0679
                         59 00000000G  00   9E 00002          MOVAB   EDT$$G_SECOND, R9                         ;
                         58 00000000G  00   9E 00009          MOVAB   EDT$$G_WORKCOUNT, R8                      ;
                         57 00000000G  00   9E 00010          MOVAB   EDT$$G_CC_CNT1_HI, R7                     ;
                         56 00000000G  00   9E 00017          MOVAB   EDT$$G_CC_FLAG, R6                        ;
                         55 00000000G  00   9E 0001E          MOVAB   EDT$$A_FMT_WRRUT, R5                      ;
                         54 00000000G  00   9E 00025          MOVAB   EDT$$G_PUT_JOU, R4                        ;
                         53 00000000G  00   9E 0002C          MOVAB   EDT$$G_CC_CNT1_LO, R3                     ;
                                       69   D5 00033          TSTL    EDT$$G_SECOND                             ; 0769
                                       42   13 00035          BEQL    3$                                        ;
                                       68   D6 00037          INCL    EDT$$G_WORKCOUNT                          ; 0772
                         52            65   D0 00039          MOVL    EDT$$A_FMT_WRRUT, FORMAT_ROUTINE          ; 0773
                         65 00000000G  00   9E 0003C          MOVAB   EDT$$TI_WRSTR, EDT$$A_FMT_WRRUT           ; 0774
                                       1A   DD 00043          PUSHL   #26                                       ; 0775
              7E 00000000G  00         01   C1 00045          ADDL3   #1, EDT$$G_MESSAGE_LINE, -(SP)            ;
                 00000000G  00         02   FB 0004D          CALLS   #2, EDT$$SC_POSCSIF                       ;
                        0F             68   E9 00054          BLBC    EDT$$G_WORKCOUNT, 1$                      ; 0777
                            00000000G  8F   DD 00057          PUSHL   #EDT$_WORKING                             ;
                 00000000G  00         01   FB 0005D          CALLS   #1, EDT$$MSG_TOSTR                        ;
                                       07   11 00064          BRB     2$                                        ;
                 00000000G  00         00   FB 00066 1$:      CALLS   #0, EDT$$SC_ERATOEOL                      ;
```

```
                          00000000G  00                  00   FB 0006D 2$:      CALLS   #0, EDT$$OUT_FMTBUF                                        ; 0779
                                     65                  52   D0 00074          MOVL    FORMAT_ROUTINE, EDT$$A_FMT_WRRUT                           ; 0780
                                                         69   D4 00077          CLRL    EDT$$G_SECOND                                              ; 0781
                                     1A 00000000G        00   E9 00079 3$:      BLBC    EDT$$G_RCOV_MOD, 4$                                        ; 0789
                                     37                  66   E9 00080          BLBC    EDT$$G_CC_FLAG, 6$                                         ; 0795
                          00000000G  00                  63   D1 00083          CMPL    EDT$$G_CC_CNT1_LO, EDT$$G_CC_CNT2_LO                       ; 0800
                                                         2E   12 0008A          BNEQ    6$                                                         ;
                          00000000G  00                  67   D1 0008C          CMPL    EDT$$G_CC_CNT1_HI, EDT$$G_CC_CNT2_HI                       ;
                                                         25   12 00093          BNEQ    6$                                                         ;
                                     64                  01   D0 00095          MOVL    #1, EDT$$G_PUT_JOU                                         ; 0803
                                                         1C   11 00098          BRB     5$                                                         ; 0804
                                     19 00000000G        00   E9 0009A 4$:      BLBC    EDT$$G_CC, 6$                                              ; 0813
                                     64                  01   D0 000A1          MOVL    #1, EDT$$G_PUT_JOU                                         ; 0820
                                                         66   D5 000A4          TSTL    EDT$$G_CC_FLAG                                             ; 0822
                                                         0E   12 000A6          BNEQ    5$                                                         ;
                                     7E          43      8F   9A 000A8          MOVZBL  #67, -(SP)                                                 ; 0825
                          00000000G  00                  01   FB 000AC          CALLS   #1, EDT$$TI_FLUSHJOUFI                                     ;
                                     66                  01   D0 000B3          MOVL    #1, EDT$$G_CC_FLAG                                         ; 0826
                                     50                  01   D0 000B6 5$:      MOVL    #1, R0                                                     ; 0829
                                                              04 000B9          RET                                                                ;
                                                         63   D6 000BA 6$:      INCL    EDT$$G_CC_CNT1_LO                                          ; 0840
                          00007530   8F                  63   D1 000BC          CMPL    EDT$$G_CC_CNT1_LO, #30000                                  ; 0842
                                                         17   12 000C3          BNEQ    7$                                                         ;
                                                         63   D4 000C5          CLRL    EDT$$G_CC_CNT1_LO                                          ; 0845
                                                         67   D6 000C7          INCL    EDT$$G_CC_CNT1_HI                                          ; 0846
                                     64                  01   D0 000C9          MOVL    #1, EDT$$G_PUT_JOU                                         ; 0847
                          00007530   8F                  67   D1 000CC          CMPL    EDT$$G_CC_CNT1_HI, #30000                                  ; 0848
                                                         07   15 000D3          BLEQ    7$                                                         ;
                          00000000G  00                  00   FB 000D5          CALLS   #0, EDT$$INTER_ERR                                         ;
                                     14 00000000G        00   D1 000DC 7$:      CMPL    EDT$$G_TIN_OBUFPOS, #20                                    ; 0856
                                                         03   19 000E3          BLSS    8$                                                         ;
                                     64                  01   D0 000E5          MOVL    #1, EDT$$G_PUT_JOU                                         ;
                                                         50   D4 000E8 8$:      CLRL    R0                                                         ; 0858
                                                              04 000EA          RET                                                                ; 0860

; Routine Size:  235 bytes,    Routine Base:  _EDT$CODE + 0000


;  291          0861  1
;  292          0862  1 !<BLF/PAGE>
;  294         0863   1 END
;  295         0864   1
;  296         0865   0 ELUDOM



;                                     PSECT SUMMARY
;
;       Name                          Bytes                     Attributes
;
;   _EDT$CODE                              235  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          3         0         40         00:00.2
;   _$255$DUA28:[EDT.SRC]PSECTS.L32;1              2          1        50          7         00:00.1




;                                     COMMAND QUALIFIERS
;
;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/NOTRACEBACK/LIS=LIS$:CHMCHKCC/OBJ=OBJ$:CHMCHKCC MSRC$:CHMCHKCC.BLI/UPDATE=(ENH$:CHMC
;       HKCC)

; Size:          235 code + 0 data bytes
; Run Time:         00:15.2
; Elapsed Time:     00:18.7
; Lines/CPU Min:     3405
; Lexemes/CPU-Min:  9220
; Memory Used:  92 pages
; Compilation Complete
```

0130 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

ERRMSG
REQ

TRACEOFF
REQ

SUPPORTS
REQ

TRAROUNAM
REQ

BADKEY
LIS

CHMCMBLIN
LIS

TRACEON
REQ

TRANSLATE
REQ

CALLWIO
LIS

CHMBEGSEN
LIS

CHMBELL
LIS

SYSSYM
REQ

TRANNAMES
REQ

CHMCHGC
LIS

CHMDELLIN
LIS

TRACELIT
REQ

EDTREQ
REQ

KEYPADDEF
REQ

CALLFIO
LIS

CHMBEGWRD
LIS

CHMCLRCC
LIS

CHMBEEP
LIS

CHMCHANGE
LIS

TRACEMAC
REQ

CHMDELCHR
LIS

PSECTS
REQ

VERSION
REQ

CHMCHKCC
LIS

CHMEINPUT
LIS