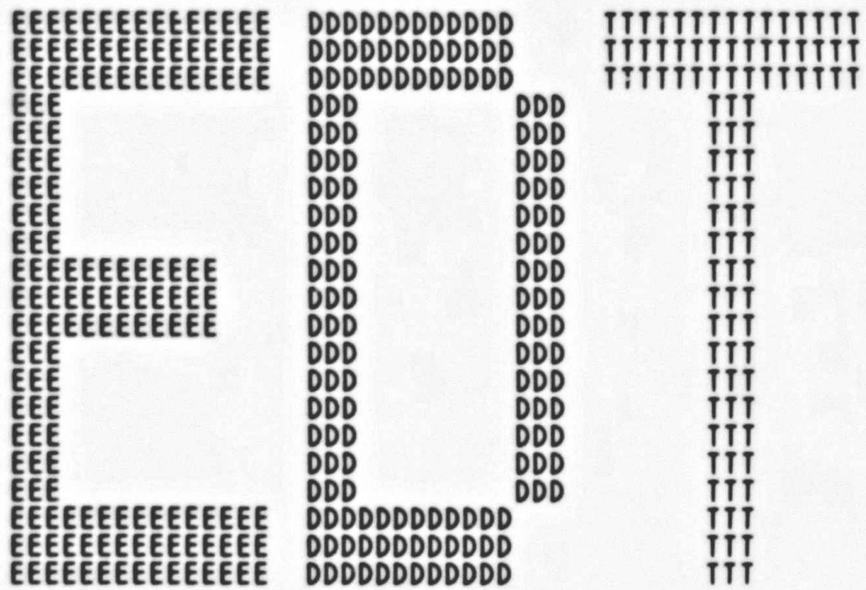
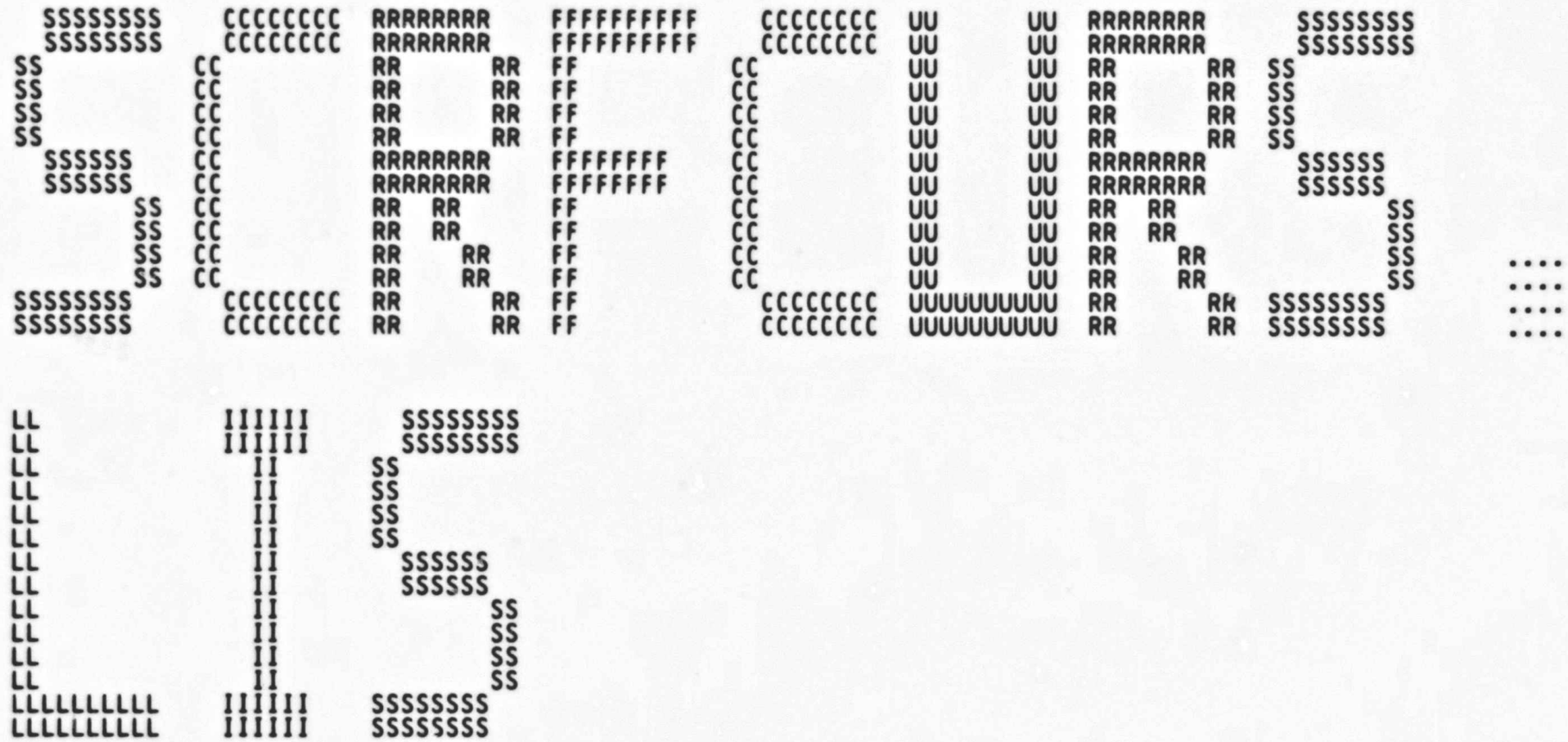

EEEEEEEEEEEEEEE DDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTTTTTTTTTTTTTTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD TTT
EEEEEEEEEEEE DDD DDD 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 DDDDDDDDDDD TTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTT
EEEEEEEEEEEEEEE DDDDDDDDDDDD TTT

FILEID**SCRFCURS



```
;     1          0001  0 %TITLE 'EDT$SCRFCURS - absolute cursor position'
;     2          0002  0 MODULE EDT$SCRFCURS (                          ! Absolute cursor position
;     3          0003  0                     IDENT = 'V04-000'                  ! File: SCRFCURS.BLI Edit: JBS1007
;     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 !       This module forces an absolute cursor position sequence to a specified
;    37          0037  1 !       line and position to be placed in the formatted output buffer.
;    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: September 8, 1979
;    42          0042  1 !
;    43          0043  1 ! MODIFIED BY:
;    44          0044  1 !
;    45          0045  1 ! 1-001 - Original.  DJS 12-Feb-1981.  This module was created by
;    46          0046  1 !       extracting the routine EDT$$SC_POSABS  from module SCREEN.
;    47          0047  1 ! 1-002 - Regularize headers.  JBS 13-Mar-1981
;    48          0048  1 ! 1-004 - Output a CR before positioning to the last column, to defeat
;    49          0049  1 !         autowrap.  JBS 16-Apr-1982
;    50          0050  1 ! 1-005 - Remove edit 1-004, since it doesn't help on a VT100, and use
;    51          0051  1 !         a shorter sequence to position to home.  JBS 07-Oct-1982
;    52          0052  1 ! 1-006 - Don't allow negative cursor locations.  JBS 10-Oct-1982
;    53          0053  1 ! 1-007 - Preserve EDT$$G_FMT_LNPOS, it may be modified by EDT$$FMT_DCML.  JBS 28-Oct-1982
;    54          0054  1 !--
;    55          0055  1
```

```
;    57          0056  1 %SBTTL 'Declarations'
;    58          0057  1 !
;    59          0058  1 ! TABLE OF CONTENTS:
;    60          0059  1 !
;    61          0060  1
;    62          0061  1 REQUIRE 'EDTSRC:TRAROUNAM';
;    63          0500  1
;    64          0501  1 FORWARD ROUTINE
;    65          0502  1     EDT$$SC_POSABS : NOVALUE;
;    66          0503  1
;    67          0504  1 !
;    68          0505  1 ! INCLUDE FILES:
;    69          0506  1 !
;    70          0507  1
;    71          0508  1 REQUIRE 'EDTSRC:EDTREQ';
;    72          0643  1
;    73          0644  1 !
;    74          0645  1 ! MACROS:
;    75          0646  1 !
;    76          0647  1 !       NONE
;    77          0648  1 !
;    78          0649  1 ! EQUATED SYMBOLS:
;    79          0650  1 !
;    80          0651  1 !       NONE
;    81          0652  1 !
;    82          0653  1 ! OWN STORAGE:
;    83          0654  1 !
;    84          0655  1 !       NONE
;    85          0656  1 !
;    86          0657  1 ! EXTERNAL REFERENCES:
;    87          0658  1 !
;    88          0659  1 !       In the routine
```

```
;    90          0660  1 %SBTTL 'EDT$$SC_POSABS  - absolute cursor position'
;    91          0661  1
;    92          0662  1 GLOBAL ROUTINE EDT$$SC_POSABS (                    ! Absolute cursor position
;    93          0663  1     LINE,                                          ! Desired line
;    94          0664  1     POS                                            ! Desired column
;    95          0665  1     ) : NOVALUE =
;    96          0666  1
;    97          0667  1 !++
;    98          0668  1 ! FUNCTIONAL DESCRIPTION:
;    99          0669  1 !
;   100          0670  1 !       This routine forces a cursor position sequence to the specified line
;   101          0671  1 !       and position to be placed in the formatted output buffer.
;   102          0672  1 !
;   103          0673  1 !       The new line and position are saved in EDT$$G_PRV_LN  and
;   104          0674  1 !       EDT$$G_PRV_COL for use by future positions.
;   105          0675  1 !
;   106          0676  1 ! FORMAL PARAMETERS:
;   107          0677  1 !
;   108          0678  1 !  LINE                 Desired line number
;   109          0679  1 !
;   110          0680  1 !  POS                  Desired column number
;   111          0681  1 !
;   112          0682  1 ! IMPLICIT INPUTS:
;   113          0683  1 !
;   114          0684  1 !       EDT$$G_TI_TYP
;   115          0685  1 !
;   116          0686  1 ! IMPLICIT OUTPUTS:
;   117          0687  1 !
;   118          0688  1 !       EDT$$G_PRV_LN
;   119          0689  1 !       EDT$$G_PRV_COL
;   120          0690  1 !
;   121          0691  1 ! ROUTINE VALUE:
;   122          0692  1 !
;   123          0693  1 !       NONE
;   124          0694  1 !
;   125          0695  1 ! SIDE EFFECTS:
;   126          0696  1 !
;   127          0697  1 !       NONE
;   128          0698  1 !
;   129          0699  1 !--
;   130          0700  1
;   131          0701  2     BEGIN
;   132          0702  2
;   133          0703  2     EXTERNAL ROUTINE
;   134          0704  2         EDT$$STORE_FMTCH : NOVALUE,                ! Store a character in the format buffer
;   135          0705  2         EDT$$FMT_DCML : NOVALUE;                   ! Expand a number in decimal into the format buffer
;   136          0706  2
;   137          0707  2     EXTERNAL
;   138          0708  2         EDT$$G_PRV_LN,                             ! Previous line number.
;   139          0709  2         EDT$$G_PRV_COL,                            ! Previous column number.
;   140          0710  2         EDT$$G_TI_TYP,                             ! Terminal type.
;   141          0711  2         EDT$$G_FMT_LNPOS;                          ! Column number for formatting purposes
;   142          0712  2
;   143          0713  2     LOCAL
;   144          0714  2         SAVE_FMTLNPOS;
;   145          0715  2
;   146          0716  2 !+
```

```
;  147          0717  2 ! The line and position parameters must not be negative.
;  148          0718  2 !-
;  149          0719  2     ASSERT (.LINE GEQ 0);
;  150          0720  2     ASSERT (.POS GEQ 0);
;  151          0721  2 !+
;  152          0722  2 ! Preserve EDT$$G_FMT_LNPOS, since the call to EDT$$FMT_DCML will modify it,
;  153          0723  2 ! but those characters do not really increment the column.
;  154          0724  2 !-
;  155          0725  2     SAVE_FMTLNPOS = .EDT$$G_FMT_LNPOS;
;  156          0726  2 !+
;  157          0727  2 ! Always start with an escape.
;  158          0728  2 !-
;  159          0729  2     EDT$$STORE_FMTCH (ASC_K_ESC);
;  160          0730  2 !+
;  161          0731  2 ! If this is a VT100, the escape is always followed by a [, to make a CSI.
;  162          0732  2 !-
;  163          0733  2
;  164          0734  2     IF (.EDT$$G_TI_TYP EQL TERM_VT100) THEN EDT$$STORE_FMTCH (%C'[');
;  165          0735  2
;  166          0736  2 !+
;  167          0737  2 ! If we are positioning to home, use the shorter sequence.
;  168          0738  2 !-
;  169          0739  2
;  170          0740  3     IF ((.LINE EQL 0) AND (.POS EQL 0))
;  171          0741  2     THEN
;  172          0742  3         BEGIN
;  173          0743  3         EDT$$STORE_FMTCH (%C'H');
;  174          0744  3         EDT$$G_PRV_LN = .LINE;
;  175          0745  3         EDT$$G_PRV_COL = .POS;
;  176          0746  3         EDT$$G_FMT_LNPOS = .SAVE_FMTLNPOS;
;  177          0747  3         RETURN;
;  178          0748  2         END;
;  179          0749  2
;  180          0750  2 !+
;  181          0751  2 ! We are not positioning to home, use the longer sequence.
;  182          0752  2 !-
;  183          0753  2
;  184          0754  3     IF (.EDT$$G_TI_TYP EQL TERM_VT52)
;  185          0755  2     THEN
;  186          0756  3         BEGIN
;  187          0757  3         EDT$$STORE_FMTCH (%C'Y');
;  188          0758  3         EDT$$STORE_FMTCH (.LINE + 32);
;  189          0759  3         EDT$$STORE_FMTCH (.POS + 32);
;  190          0760  3         EDT$$G_PRV_LN = .LINE;
;  191          0761  3         EDT$$G_PRV_COL = .POS;
;  192          0762  3         EDT$$G_FMT_LNPOS = .SAVE_FMTLNPOS;
;  193          0763  3         RETURN;
;  194          0764  2         END;
;  195          0765  2
;  196          0766  2     ASSERT (.EDT$$G_TI_TYP EQL TERM_VT100);
;  197          0767  2
;  198          0768  2     IF (.LINE NEQ 0) THEN EDT$$FMT_DCML (.LINE + 1);
;  199          0769  2
;  200          0770  3     IF (.POS NEQ 0)
;  201          0771  2     THEN
;  202          0772  3         BEGIN
;  203          0773  3         EDT$$STORE_FMTCH (%C';');
```

```
;  204          0774  3          EDT$$FMT_DCML (.POS + 1);
;  205          0775  2          END;
;  206          0776  2
;  207          0777  2      EDT$$STORE_FMTCH (%C'H');
;  208          0778  2      EDT$$G_PRV_LN = .LINE;
;  209          0779  2      EDT$$G_PRV_COL = .POS;
;  210          0780  2      EDT$$G_FMT_LNPOS = .SAVE_FMTLNPOS;
;  211          0781  1      END;                                        ! of routine EDT$$SC_POSABS


;                                                                       .TITLE  EDT$SCRFCURS EDT$SCRFCURS - absolute cursor pos
                                                                                ition
                                                                        .IDENT  \V04-000\

                                                                        .EXTRN  EDT$$STORE_FMTCH
                                                                        .EXTRN  EDT$$FMT_DCML, EDT$$G_PRV_LN
                                                                        .EXTRN  EDT$$G_PRV_COL, EDT$$G_TI_TYP
                                                                        .EXTRN  EDT$$G_FMT_LNPOS
                                                                        .EXTRN  EDT$$INTER_ERR

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

                                        03FC 00000                      .ENTRY  EDT$$SC_POSABS, Save R2,R3,R4,R5,R6,R7,R8,- ; 0662
                                                                                R9                                          ;
                                59 00000000G   00  9E 00002             MOVAB   EDT$$FMT_DCML, R9                           ;
                                58 00000000G   00  9E 00009             MOVAB   EDT$$G_FMT_LNPOS, R8                        ;
                                57 00000000G   00  9E 00010             MOVAB   EDT$$INTER_ERR, R7                          ;
                                56 00000000G   00  9E 00017             MOVAB   EDT$$G_TI_TYP, R6                           ;
                                55 00000000G   00  9E 0001E             MOVAB   EDT$$STORE_FMTCH, R5                        ;
                                53       04    AC  D0 00025             MOVL    LINE, R3                                    ; 0719
                                               03  18 00029             BGEQ    1$                                          ;
                                67             00  FB 0002B             CALLS   #0, EDT$$INTER_ERR                          ;
                                52       08    AC  D0 0002E 1$:         MOVL    POS, R2                                     ; 0720
                                               03  18 00032             BGEQ    2$                                          ;
                                67             00  FB 00034             CALLS   #0, EDT$$INTER_ERR                          ;
                                54             68  D0 00037 2$:         MOVL    EDT$$G_FMT_LNPOS, SAVE_FMTLNPOS             ; 0725
                                               1B  DD 0003A             PUSHL   #27                                         ; 0729
                                65             01  FB 0003C             CALLS   #1, EDT$$STORE_FMTCH                        ;
                                02             66  D1 0003F             CMPL    EDT$$G_TI_TYP, #2                           ; 0734
                                               07  12 00042             BNEQ    3$                                          ;
                                7E       5B    8F  9A 00044             MOVZBL  #91, -(SP)                                  ;
                                65             01  FB 00048             CALLS   #1, EDT$$STORE_FMTCH                        ;
                                               53  D5 0004B 3$:         TSTL    R3                                          ; 0740
                                               04  12 0004D             BNEQ    4$                                          ;
                                               52  D5 0004F             TSTL    R2                                          ;
                                               38  13 00051             BEQL    8$                                          ;
                                01             66  D1 00053 4$:         CMPL    EDT$$G_TI_TYP, #1                           ; 0754
                                               12  12 00056             BNEQ    5$                                          ;
                                7E       59    8F  9A 00058             MOVZBL  #89, -(SP)                                  ; 0757
                                65             01  FB 0005C             CALLS   #1, EDT$$STORE_FMTCH                        ;
                                         20    A3  9F 0005F             PUSHAB  32(R3)                                      ; 0758
                                65             01  FB 00062             CALLS   #1, EDT$$STORE_FMTCH                        ;
                                         20    A2  9F 00065             PUSHAB  32(R2)                                      ; 0759
                                               25  11 00068             BRB     9$                                          ;
                                02             66  D1 0006A 5$:         CMPL    EDT$$G_TI_TYP, #2                           ; 0766
                                               03  13 0006D             BEQL    6$                                          ;
                                67             00  FB 0006F             CALLS   #0, EDT$$INTER_ERR                          ;
```

```
                                                53  D5 00072 6$:     TSTL    R3                                      ; 0768
                                                06  13 00074        BEQL    7$                                      ;
                                            01  A3  9F 00076        PUSHAB  1(R3)                                   ;
                                        69      01  FB 00079        CALLS   #1, EDT$$FMT_DCML                       ;
                                                52  D5 0007C 7$:     TSTL    R2                                      ; 0770
                                                0B  13 0007E        BEQL    8$                                      ;
                                                3B  DD 00080        PUSHL   #59                                     ; 0773
                                        65      01  FB 00082        CALLS   #1, EDT$$STORE_FMTCH                    ;
                                            01  A2  9F 00085        PUSHAB  1(R2)                                   ; 0774
                                        69      01  FB 00088        CALLS   #1, EDT$$FMT_DCML                       ;
                                        7E  48  8F  9A 0008B 8$:     MOVZBL  #72, -(SP)                              ; 0777
                                        65      01  FB 0008F 9$:     CALLS   #1, EDT$$STORE_FMTCH                    ;
                            00000000G   00          53  D0 00092        MOVL    R3, EDT$$G_PRV_LN                       ; 0778
                            00000000G   00          52  D0 00099        MOVL    R2, EDT$$G_PRV_COL                      ; 0779
                                        68          54  D0 000A0        MOVL    SAVE_FMTLNPOS, EDT$$G_FMT_LNPOS         ; 0780
                                                    04  000A3        RET                                             ; 0781

; Routine Size:  164 bytes,    Routine Base:  _EDT$CODE + 0000

;  212          0782  1
;  213          0783  1 !<BLF/PAGE>
```

```
;  215          0784  1 END                                   ! of module EDT$SCRFCRUS
;  216          0785  1
;  217          0786  0 ELUDOM
```

```
;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  _EDT$CODE                       164  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
```

```
;                                   COMMAND QUALIFIERS

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

; Size:          164 code + 0 data bytes
; Run Time:          00:13.5
; Elapsed Time:      00:19.4
; Lines/CPU Min:      3495
; Lexemes/CPU-Min: 10171
; Memory Used:  90 pages
; Compilation Complete
```

PRNUMRAN
LIS

REAJOUTEX
LIS

PRPARCOMN
LIS

SCRCHKREV
LIS

SCRDELETE
LIS

SCRESCR
LIS

SCRCURS
LIS

PSECTS
LIS

PRMACCAL
LIS

PRPUSH
LIS

SCRFIND
LIS

PRPARCOM
LIS

PRPARDRV
LIS

RANRPOS
LIS

SCRCOMCUR
LIS

RANNEXT
LIS

SCRBLOB
LIS

SCRELINE
LIS

SCRFCURS
LIS

PRSEMRTN
LIS

SAUDIT
LIS