```
   CCCCCCCCCCCC  LLL                 IIIIIIIII     UUU         UUU  TTTTTTTTTTTTTTT  LLL
   CCCCCCCCCCCC  LLL                 IIIIIIIII     UUU         UUU  TTTTTTTTTTTTTTT  LLL
   CCCCCCCCCCCC  LLL                 IIIIIIIII     UUU         UUU  TTTTTTTTTTTTTTT  LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
CCC              LLL                    III        UUU         UUU        TTT        LLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL     IIIIIIIII     UUUUUUUUUUUUUUU        TTT        LLLLLLLLLLLLLLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL     IIIIIIIII     UUUUUUUUUUUUUUU        TTT        LLLLLLLLLLLLLLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL     IIIIIIIII     UUUUUUUUUUUUUUU        TTT        LLLLLLLLLLLLLLL
```

```
  SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT  PPPPPPPP      000000    MM      MM  EEEEEEEEEE    SSSSSSSS    SSSSSSSS
  SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT  PPPPPPPP      000000    MM      MM  EEEEEEEEEE    SSSSSSSS    SSSSSSSS
SS          EE              TT      PP      PP  00      00  MMMM  MMMM  EE          SS          SS
SS          EE              TT      PP      PP  00      00  MMMM  MMMM  EE          SS          SS
SS          EE              TT      PP      PP  00    0000  MM  MM  MM  EE          SS          SS
SS          EE              TT      PP      PP  00    0000  MM  MM  MM  EE          SS          SS
  SSSSSS    EEEEEEEE        TT      PPPPPPPP    00  00  00  MM      MM  EEEEEEEE      SSSSSS      SSSSSS
  SSSSSS    EEEEEEEE        TT      PPPPPPPP    00  00  00  MM      MM  EEEEEEEE      SSSSSS      SSSSSS
        SS  EE              TT      PP          0000    00  MM      MM  EE                  SS          SS
        SS  EE              TT      PP          0000    00  MM      MM  EE                  SS          SS
        SS  EE              TT      PP          00      00  MM      MM  EE                  SS          SS      ....
        SS  EE              TT      PP          00      00  MM      MM  EE                  SS          SS      ....
SSSSSSSS    EEEEEEEEEE      TT      PP            000000    MM      MM  EEEEEEEEEE  SSSSSSSS    SSSSSSSS      ....
SSSSSSSS    EEEEEEEEEE      TT      PP            000000    MM      MM  EEEEEEEEEE  SSSSSSSS    SSSSSSSS      ....

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
```

```
;     1          0001  0 MODULE setp0$mess (IDENT = 'V04-000',
;     2          0002  0                     ADDRESSING_MODE(EXTERNAL=GENERAL)) =
;     3          0003  1 BEGIN
;     4          0004  1
;     5          0005  1 !
;     6          0006  1 !*****************************************************************************
;     7          0007  1 !*                                                                           *
;     8          0008  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;     9          0009  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    10          0010  1 !*  ALL RIGHTS RESERVED.                                                      *
;    11          0011  1 !*                                                                           *
;    12          0012  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND C( IED   *
;    13          0013  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUC': LICENSE  AND WIT  THE   *
;    14          0014  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY   THER   *
;    15          0015  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE    ANY   *
;    16          0016  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS   .REBY   *
;    17          0017  1 !*  TRANSFERRED.                                                              *
;    18          0018  1 !*                                                                           *
;    19          0019  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT  OTICE   *
;    20          0020  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EO' PMENT   *
;    21          0021  1 !*  CORPORATION.                                                              *
;    22          0022  1 !*                                                                           *
;    23          0023  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIAB'' .Y OF ITS   *
;    24          0024  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    25          0025  1 !*                                                                           *
;    26          0026  1 !*                                                                           *
;    27          0027  1 !*****************************************************************************
;    28          0028  1
;    29          0029  1 !++
;    30          0030  1 ! FACILITY:  SET Command (SETPO.EXE)
;    31          0031  1 !
;    32          0032  1 ! ABSTRACT:
;    33          0033  1 !
;    34          0034  1 !       This module processes the SET MESSAGE command
;    35          0035  1 !
;    36          0036  1 ! ENVIRONMENT:
;    37          0037  1 !
;    38          0038  1 !       VAX/VMS operating system. unprivileged user mode,
;    39          0039  1 !
;    40          0040  1 ! AUTHOR:  Tim Halvorsen, Dec 1979
;    41          0041  1 !
;    42          0042  1 ! Modified by:
;    43          0043  1 !
;    44          0044  1 !       V03-006 AEW0001         Anne E. Warner          20-Jul-1984
;    45          0045  1 !               Enable the message cli$_confqual to be returned when
;    46          0046  1 !               a file-spec is specified with the /DELETE qualifier.
;    47          0047  1 !
;    48          0048  1 !       V03-005 BLS0295         Benn Schreiber           7-APR-1984
;    49          0049  1 !               Disable SYSPRV.
;    50          0050  1 !
;    51          0051  1 !       V03-004 BLS0291         Benn Schreiber          24-MAR-1984
;    52          0052  1 !               Make routine name SET$MESSAGE.
;    53          0053  1 !
;    54          0054  1 !       V03-003 GAS0112         Gerry Smith             29-Mar-1983
;    55          0055  1 !               Remove references to old CLI interface.
;    56          0056  1 !
;    57          0057  1 !       V03-002 GAS0091         Gerry Smith             19-Oct-1982
```

```
;    58          0058  1 !                 Change input request for new CLD syntax.
;    59          0059  1 !
;    60          0060  1 !       V03-001 DWT0036         David Thiel     1-Apr-1982
;    61          0061  1 !                 Don't allow an indirect message section to be installed.
;    62          0062  1 !
;    63          0063  1 !       V001    TMH0001         Tim Halvorsen   26-Dec-1981
;    64          0064  1 !                 Complete the code to delete a message section when
;    65          0065  1 !                 desired (/DELETE) by unmaping the section and deassigning
;    66          0066  1 !                 the channel to the file.  Two new cells were added to P1
;    67          0067  1 !                 space to hold the virtual address range and the channel #.
;    68          0068  1 !--
;    69          0069  1
;    70          0070  1 !
;    71          0071  1 ! Include files
;    72          0072  1 !
;    73          0073  1
;    74          0074  1 LIBRARY 'SYS$LIBRARY:LIB.L32';  ! VAX/VMS common definitions
;    75          0075  1
```

```
;    77          0076  1 !
;    78          0077  1 ! Table of contents
;    79          0078  1 !
;    80          0079  1
;    81          0080  1 FORWARD ROUTINE
;    82          0081  1     set$message,                    ! Process SET MESSAGE
;    83          0082  1     map_section,                    ! Map message section into P1 region
;    84          0083  1     delete_section;                 ! Delete message section from P1 region
;    85          0084  1
;    86          0085  1 !
;    87          0086  1 ! Define literals
;    88          0087  1 !
;    89          0088  1
;    90          0089  1 LITERAL
;    91          0090  1     true = 1,
;    92          0091  1     false = 0;
;    93          0092  1
;    94          0093  1 !
;    95          0094  1 ! Macros
;    96          0095  1 !
;    97          0096  1
;    98          0097  1 MACRO
;    99        M 0098  1     perform(command) =
;   100        M 0099  1         BEGIN
;   101        M 0100  1         LOCAL status;
;   102        M 0101  1         status = command;
;   103        M 0102  1         IF NOT .status
;   104        M 0103  1         THEN
;   105        M 0104  1             BEGIN
;   106        M 0105  1             setp0$l_status = .status;
;   107        M 0106  1             RETURN true;
;   108        M 0107  1             END;
;   109          0108  1         END%,
;   110          0109  1
;   111        M 0110  1     return_if_error(command) =
;   112        M 0111  1         BEGIN
;   113        M 0112  1         LOCAL status;
;   114        M 0113  1         status = command;
;   115        M 0114  1         IF NOT .status
;   116        M 0115  1         THEN
;   117        M 0116  1             RETURN .status;
;   118          0117  1         END%;
;   119          0118  1
;   120          0119  1
;   121          0120  1 !
;   122          0121  1 ! External definitions
;   123          0122  1 !
;   124          0123  1
;   125          0124  1 EXTERNAL
;   126          0125  1     setp0$l_status,                 ! Status returned from option
;   127          0126  1     ctl$gb_msgmask:     BYTE,       ! Default message flags
;   128          0127  1     ctl$gl_ppmsg:       VECTOR [2], ! Process permanent message section
;   129          0128  1     ctl$gw_ppmsgchn:    WORD,       ! Channel to message section
;   130          0129  1     ctl$gl_ctlbasva;                ! Base of fixed P1 region
;   131          0130  1
;   132          0131  1 EXTERNAL LITERAL
;   133          0132  1     cli$_negated,                   ! Qualifier explicitly negated
```

```
;  134          0133  1      cli$_confqual;                  ! Conflicting qualifiers
;  135          0134  1
;  136          0135  1 EXTERNAL ROUTINE
;  137          0136  1      cli$get_value,                  ! Get a value from CLI
;  138          0137  1      cli$present;                    ! See if qualifier present
```

```
;  140          0138  1
;  141          0139  1 !
;  142          0140  1 ! Define flags to indicate which qualifiers are present
;  143          0141  1 !
;  144          0142  1
;  145          0143  1 LITERAL
;  146        P 0144  1     SEQULST(qual_,,0,1,
;  147        P 0145  1                     (delete,),
;  148          0146  1                     (file,));
;  149          0147  1
;  150          0148  1 !
;  151          0149  1 ! Work areas
;  152          0150  1 !
;  153          0151  1
;  154          0152  1 OWN
;  155          0153  1     cli_flags:  BITVECTOR[32],          ! CLI qualifier flags
;  156          0154  1     set_mask:   BYTE INITIAL(0),        ! Mask of bits to explicitly set
;  157          0155  1     clear_mask: BYTE INITIAL(0);        ! Mask of bits to explicitly clear
```

```
;  159          0156  1 GLOBAL ROUTINE set$message =          ! SET MESSAGE command
;  160          0157  1
;  161          0158  1 !---
;  162          0159  1 !
;  163          0160  1 !       This routine processes the SET MESSAGE command
;  164          0161  1 !
;  165          0162  1 ! Inputs:
;  166          0163  1 !       None.
;  167          0164  1 !
;  168          0165  1 ! Outputs:
;  169          0166  1 !
;  170          0167  1 !       status code
;  171          0168  1 !---
;  172          0169  1
;  173          0170  2 BEGIN
;  174          0171  2
;  175          0172  2 ROUTINE
;  176          0173  2     set_flags =
;  177          0174  2         (BUILTIN ap; ctl$gb_msgmask = .ap <0,4>; true); ! Set message flags

                                                        .TITLE  SETPO$MESS
                                                        .IDENT  \V04-000\

                                                        .PSECT  $OWN$,NOEXE,2

                                                00000 CLI_FLAGS:
                                                        .BLKB   4
                                              00  00004 SET_MASK:
                                                        .BYTE   0                       ;
                                              00  00005 CLEAR_MASK:
                                                        .BYTE   0                       ;

                                                        .EXTRN  SETPO$L_STATUS, CTL$GB_MSGMASK
                                                        .EXTRN  CTL$GL_PPMSG, CTL$GW_PPMSGCHN
                                                        .EXTRN  CTL$GL_CTLBASVA
                                                        .EXTRN  CLI$_NEGATED, CLI$_CONFQUAL
                                                        .EXTRN  CLI$GET_VALUE, CLI$PRESENT

                                                        .PSECT  $CODE$,NOWRT,2

                                                    0000 00000 SET_FLAGS:
                                                        .WORD   Save nothing            ; 0173
          50            5C                      04      00  EF 00002          EXTZV   #0, #4, AP, R0          ; 0174
                              00000000G         00      50  90 00007          MOVB    R0, CTL$GB_MSGMASK      ;
                                                50      01  D0 0000E          MOVL    #1, R0                  ;
                                                            04 00011          RET                             ;

; Routine Size:  18 bytes,    Routine Base:  $CODE$ + 0000

;  178          0175  2
;  179          0176  2 LOCAL
;  180          0177  2     status,                             ! status code
;  181          0178  2     priv_mask : $BBLOCK[8],             ! Mask to clear SYSPRV
;  182          0179  2     file_desc : $BBLOCK[dsc$c_s_bln],   ! File descriptor
;  183          0180  2     new_mask:   BYTE;                   ! New message bit mask
```

```
;  184          0181  2
;  185          0182  2 !
;  186          0183  2 ! Drop SYSPRV
;  187          0184  2 !
;  188          0185  2 ch$fill(0,8,priv_mask);
;  189          0186  2 priv_mask[prv$v_sysprv] = 1;
;  190          0187  2 $SETPRV(enbflg=0,prvadr=priv_mask);
;  191          0188  2 !
;  192          0189  2 !       Get message file name
;  193          0190  2 !
;  194          0191  2 $init_dyndesc(file_desc);
;  195          0192  2 cli$get_value(%ASCID 'FILE', file_desc);
;  196          0193  2 cli_flags[qual_delete] = cli$present(%ASCID 'DELETE');
;  197          0194  2
;  198          0195  2 !
;  199          0196  2 !       Get qualifiers on command line
;  200          0197  2 !
;  201          0198  2 status = cli$present(%ASCID 'TEXT');                    ! /[NO]TEXT
;  202          0199  2 IF .status
;  203          0200  2 THEN set_mask = .set_mask OR 1
;  204          0201  2 ELSE IF .status EQL cli$_negated
;  205          0202  2 THEN clear_mask = .clear_mask OR 1;
;  206          0203  2
;  207          0204  2 status = cli$present(%ASCID 'IDENT');                   ! /[NO]IDENT
;  208          0205  2 IF .status
;  209          0206  2 THEN set_mask = .set_mask OR 2
;  210          0207  2 ELSE IF .status EQL cli$_negated
;  211          0208  2 THEN clear_mask = .clear_mask OR 2;
;  212          0209  2
;  213          0210  2 status = cli$present(%ASCID 'SEVERITY');                ! /[NO]SEVERITY
;  214          0211  2 IF .status
;  215          0212  2 THEN set_mask = .set_mask OR 4
;  216          0213  2 ELSE IF .status EQL cli$_negated
;  217          0214  2 THEN clear_mask = .clear_mask OR 4;
;  218          0215  2
;  219          0216  2 status = cli$present(%ASCID 'FACILITY');                ! /[NO]FACILITY
;  220          0217  2 IF .status
;  221          0218  2 THEN set_mask = .set_mask OR 8
;  222          0219  2 ELSE IF .status EQL cli$_negated
;  223          0220  2 THEN clear_mask = .clear_mask OR 8;
;  224          0221  2
;  225          0222  2 !
;  226          0223  2 !       Set the new message display flag settings first
;  227          0224  2 !
;  228          0225  2
;  229          0226  2 IF .set_mask NEQ 0 OR .clear_mask NEQ 0 ! If we gotta change something,
;  230          0227  2 THEN
;  231          0228  3     BEGIN
;  232          0229  3     new_mask = (.ctl$gb_msgmask OR .set_mask) AND NOT .clear_mask;
;  233          0230  3
;  234        P 0231  3     perform($CMKRNL(ROUTIN = set_flags,
;  235          0232  3                     ARGLST = .new_mask));   ! Set new message flags
;  236          0233  2     END;
;  237          0234  2
;  238          0235  2 !
;  239          0236  2 !       If /DELETE was specified, then delete the current section
;  240          0237  2 !
```

```
;  241          0238  2 IF .cli_flags [qual_delete]             ! If /DELETE specified,
;  242          0239  2 THEN
;  243          0240  3     BEGIN
;  244          0241  3     IF .file_desc[dsc$w_length] NEQ 0   ! If file spec specified,
;  245          0242  3     THEN
;  246          0243  4         BEGIN
;  247          0244  4         setp0$l_status = cli$_confqual;
;  248          0245  4         RETURN                          ! return conflicting qualifiers
;  249          0246  3         END;
;  250          0247  3
;  251        P 0248  3     perform($CMKRNL(ROUTIN=delete_section, ! then delete the section
;  252          0249  3                     ARGLST=file_desc));
;  253          0250  2     END;
;  254          0251  2
;  255          0252  2 !
;  256          0253  2 !       If message file name specified, then map the file into
;  257          0254  2 !       the P1 region and set the process permanent message pointer.
;  258          0255  2 !
;  259          0256  2
;  260          0257  2 IF .file_desc[dsc$w_length] NEQ 0       ! If parameter specified,
;  261          0258  2     AND NOT .cli_flags [qual_delete]    ! and not deleting the section
;  262          0259  2 THEN
;  263          0260  3     BEGIN
;  264          0261  3     !
;  265          0262  3     !   Only allow 1 section to be mapped at a time for now
;  266          0263  3     !   This can be extended at a later date.
;  267          0264  3     !
;  268          0265  3     IF .ctl$gl_ppmsg NEQ 0              ! If section already mapped,
;  269          0266  3     THEN
;  270          0267  3         perform($CMKRNL(ROUTIN=delete_section,ARGLST=file_desc));! unmap it
;  271          0268  3     !
;  272          0269  3     !   Map the specified file as the current section
;  273          0270  3     !
;  274        P 0271  3     perform($CMKRNL(ROUTIN=map_section, ! map the section
;  275          0272  3                     ARGLST=file_desc));
;  276          0273  2     END;
;  277          0274  2
;  278          0275  2 RETURN true;
;  279          0276  2
;  280          0277  1 END;
; INFO#212                L1:0244
; Null expression appears in value-required context


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                        45  4C  49  46  00000 P.AAB:  .ASCII  \FILE\                                  ;
                                          010E0004  00004 P.AAA:  .LONG   17694724                                  ;
                                          00000000' 00008         .ADDRESS P.AAB                                    ;
                            00  00  45  54  45  4C  45  44  0000C P.AAD:  .ASCII  \DELETE\<0><0>                       ;
                                          010E0006  00014 P.AAC:  .LONG   17694726                                  ;
                                          00000000' 00018         .ADDRESS P.AAD                                    ;
                                        54  58  45  54  0001C P.AAF:  .ASCII  \TEXT\                                  ;
                                          010E0004  00020 P.AAE:  .LONG   17694724                                  ;
                                          00000000' 00024         .ADDRESS P.AAF                                    ;
                            00  00  00  54  4E  45  44  49  00028 P.AAH:  .ASCII  \IDENT\<0><0><0>                     ;
```

```
                                    010E0005  00030 P.AAG:  .LONG   17694725                                                  ;
                                    00000000' 00034         .ADDRESS P.AAH                                                 ;
                59  54  49  52  45  56  45  53  00038 P.AAJ:  .ASCII  \SEVERITY\                                            ;
                                    010E0008  00040 P.AAI:  .LONG   17694728                                               ;
                                    00000000' 00044         .ADDRESS P.AAJ                                                 ;
                59  54  49  4C  49  43  41  46  00048 P.AAL:  .ASCII  \FACILITY\                                            ;
                                    010E0008  00050 P.AAK:  .LONG   17694728                                               ;
                                    0000C000' 00054         .ADDRESS P.AAL                                                 ;

                                                            .EXTRN  SYS$SETPRV, SYS$CMKRNL

                                                            .PSECT  $CODE$,NOWRT,2

                                        0FFC 0C000          .ENTRY  SET$MESSAGE, Save R2,R3,R4,R5,R6,R7,R8,R9,-            ; 0156
                                                                    R10,R11
                        5B 00000000G    00   9E 00002       MOVAB   SETPO$L_STATUS, R11                                    ;
                        5A      0000'   CF   9E 00009       MOVAB   P.AAA, R10                                             ;
                        59 00000000G    00   9E 0000E       MOVAB   SYS$CMKRNL, R9                                         ;
                        58 00000000G    8F   D0 00015       MOVL    #CLI$_NEGATED, R8                                      ;
                        57 00000000G    00   9E 0001C       MOVAB   CLI$PRESENT, R7                                        ;
                        56      0000'   CF   9E 00023       MOVAB   CLEAR_MASK, R6                                         ;
                        5E              10   C2 00028       SUBL2   #16, SP                                                ;
        08      00      6E              00   2C 0002B       MOVC5   #0, (SP), #0, #8, PRIV_MASK                            ; 0185
                                08      AE   00030                                                                         ;
                            0B  AE          10   88 00032   BISB2   #16, PRIV_MASK+3                                       ; 0186
                                            7E   7C 00036   CLRQ    -(SP)                                                  ; 0187
                                10      AE   9F   00038     PUSHAB  PRIV_MASK                                              ;
                                            7E   D4 0003B   CLRL    -(SP)                                                  ;
                    00000000G   00          04   FB 0003D   CALLS   #4, SYS$SETPRV                                       ;
                                6E 020E0000     8F   D0 00044   MOVL    #34471936, FILE_DESC                           ; 0191
                                        04      AE   D4 0004B   CLRL    FILE_DESC+4                                    ;
                                    4400        8F   BB 0004E   PUSHR   #^M<R10,SP>                                    ; 0192
                    00000000G   00          02   FB 00052   CALLS   #2, CLI$GET_VALUE                                   ;
                                        10      AA   9F 00059   PUSHAB  P.AAC                                          ; 0193
                                67              01   FB 0005C   CALLS   #1, CLI$PRESENT                                 ;
    FB  A6          01          00          50   F0 0005F   INSV    R0, #0, #1, CLI_FLAGS                              ;
                                        1C      AA   9F 00065   PUSHAB  P.AAE                                          ; 0198
                                67              01   FB 00068   CALLS   #1, CLI$PRESENT                                 ;
                                06              50   E9 0006B   BLBC    STATUS, 1$                                     ; 0199
                            FF  A6              01   88 0006E   BISB2   #1, SET_MASK                                    ; 0200
                                58              08   11 00072   BRB     2$                                             ;
                                58              50   D1 00074 1$:  CMPL    STATUS, R8                                     ; 0201
                                                03   12 00077   BNEQ    2$                                             ;
                                66              01   88 00079   BISB2   #1, CLEAR_MASK                                  ; 0202
                                        2C      AA   9F 0007C 2$:  PUSHAB  P.AAG                                          ; 0204
                                67              01   FB 0007F   CALLS   #1, CLI$PRESENT                                 ;
                                06              50   E9 00082   BLBC    STATUS, 3$                                     ; 0205
                            FF  A6              02   88 00085   BISB2   #2, SET_MASK                                    ; 0206
                                58              08   11 00089   BRB     4$                                             ;
                                58              50   D1 0008B 3$:  CMPL    STATUS, R8                                     ; 0207
                                                03   12 0008E   BNEQ    4$                                             ;
                                66              02   88 00090   BISB2   #2, CLEAR_MASK                                  ; 0208
                                        3C      AA   9F 00093 4$:  PUSHAB  P.AAI                                          ; 0210
                                67              01   FB 00096   CALLS   #1, CLI$PRESENT                                 ;
                                06              50   E9 00099   BLBC    STATUS, 5$                                     ; 0211
                            FF  A6              04   88 0009C   BISB2   #4, SET_MASK                                    ; 0212
                                                08   11 000A0   BRB     6$                                             ;
```

```
                58              50  D1 000A2 5$:    CMPL    STATUS, R8                              ; 0213
                                03  12 000A5       BNEQ    6$
                66              04  88 000A7       BISB2   #4, CLEAR_MASK                         ; 0214
                      4C        AA  9F 000AA 6$:    PUSHAB  P.AAK                                 ; 0216
                67              01  FB 000AD       CALLS   #1, CLI$PRESENT
                06              50  E9 000B0       BLBC    STATUS, 7$                            ; 0217
             FF A6              08  88 000B3       BISB2   #8, SET_MASK                          ; 0218
                                08  11 000B7       BRB     8$
                58              50  D1 000B9 7$:    CMPL    STATUS, R8                            ; 0219
                                03  12 000BC       BNEQ    8$
                66              08  88 000BE       BISB2   #8, CLEAR_MASK                        ; 0220
                50        FF    A6  9A 000C1 8$:    MOVZBL  SET_MASK, R0                          ; 0226
                                04  12 000C5       BNEQ    9$
                                66  95 000C7       TSTB    CLEAR_MASK
                                1B  13 000C9       BEQL    10$
                52 00000000G    00  9A 000CB 9$:    MOVZBL  CTL$GB_MSGMASK, R2                    ; 0229
                50              52  C8 000D2       BISL2   R2, R0
             51 50              66  8B 000D5       BICB3   CLEAR_MASK, R0, NEW_MASK
                7E              51  9A 000D9       MOVZBL  NEW_MASK, -(SP)                       ; 0232
                      FF0E      CF  9F 000DC       PUSHAB  SET_FLAGS
                69              02  FB 000E0       CALLS   #2, SYS$CMKRNL
                45              50  E9 000E3       BLBC    STATUS, 14$
                19        FB    A6  E9 000E6 10$:   BLBC    CLI_FLAGS, 12$                        ; 0238
                                6E  B5 000EA       TSTW    FILE_DESC                             ; 0241
                                09  13 000EC       BEQL    11$
                6B 00000000G    8F  D0 000EE       MOVL    #CLI$_CONFQUAL, SETPO$L_STATUS        ; 0244
                                3B  11 000F5       BRB     16$                                   ; 0243
                                5E  DD 000F7 11$:   PUSHL   SP                                    ; 0249
                      0000V     CF  9F 000F9       PUSHAB  DELETE_SECTION
                69              02  FB 000FD       CALLS   #2, SYS$CMKRNL
                28              50  E9 00100       BLBC    STATUS, 14$
                                6E  B5 00103 12$:   TSTW    FILE_DESC                             ; 0257
                                27  13 00105       BEQL    15$
                23        FB    A6  E8 00107       BLBS    CLI_FLAGS, 15$                        ; 0258
                   00000000G    00  D5 0010B       TSTL    CTL$GL_PPMSG                          ; 0265
                                0C  13 00111       BEQL    13$
                                5E  DD 00113       PUSHL   SP                                    ; 0267
                      0000V     CF  9F 00115       PUSHAB  DELETE_SECTION
                69              02  FB 00119       CALLS   #2, SYS$CMKRNL
                0C              50  E9 0011C       BLBC    STATUS, 14$
                                5E  DD 0011F 13$:   PUSHL   SP                                    ; 0272
                      0000V     CF  9F 00121       PUSHAB  MAP_SECTION
                69              02  FB 00125       CALLS   #2, SYS$CMKRNL
                03              50  E8 00128       BLBS    STATUS, 15$
                6B              50  D0 0012B 14$:   MOVL    STATUS, SETPO$L_STATUS
                50              01  D0 0012E 15$:   MOVL    #1, R0                                ; 0275
                                    04 00131       RET
                                50  D4 00132 16$:   CLRL    R0                                    ; 0277
                                    04 00134       RET

; Routine Size:  309 bytes,    Routine Base:  $CODE$ + 0012
```

```
;  282          0278  1 ROUTINE map_section =
;  283          0279  1
;  284          0280  1 !---
;  285          0281  1 !
;  286          0282  1 !       This routine maps the message file into the P1 region
;  287          0283  1 !       as a process permanent message section.  The file name
;  288          0284  1 !       is described by the parameter descriptor block.
;  289          0285  1 !
;  290          0286  1 ! Inputs:
;  291          0287  1 !
;  292          0288  1 !       ap = Address of parameter descriptor block
;  293          0289  1 !
;  294          0290  1 ! Outputs:
;  295          0291  1 !
;  296          0292  1 !       The file is mapped; else status returned in setp0$l_status.
;  297          0293  1 !---
;  298          0294  1
;  299          0295  2 BEGIN
;  300          0296  2
;  301          0297  2 BUILTIN
;  302          0298  2     AP,                                 ! Register AP
;  303          0299  2     PROBER;                             ! Built-in function PROBER
;  304          0300  2
;  305          0301  2 MAP
;  306          0302  2     ap:         REF BLOCK [,BYTE];      ! Address of parm desc. block
;  307          0303  2
;  308          0304  2 LOCAL
;  309          0305  2     fab:        BLOCK [fab$c_bln,BYTE], ! FAB to access file with
;  310          0306  2     status,                             ! temporary status
;  311          0307  2     range:      VECTOR [2],             ! CRMPSC address range
;  312          0308  2     section:    REF BLOCK[,BYTE];       ! Use to access actual section
;  313          0309  2
;  314          0310  2 !
;  315          0311  2 !       Open the message file and obtain a channel to it
;  316          0312  2 !
;  317          0313  2
;  318        P 0314  2 $FAB_INIT(FAB = fab,                      ! Initialize FAB
;  319        P 0315  2           FNA = .ap [dsc$a_pointer],      ! Address of file name string
;  320        P 0316  2           FNS = .ap [dsc$w_length],       ! Length of file name string
;  321        P 0317  2           DNA = UPLIT('.EXE'),            ! Default name string
;  322        P 0318  2           DNS = 4,
;  323          0319  2           FOP = UFO);                     ! User file open
;  324          0320  2
;  325          0321  2 return_if_error($OPEN(FAB = fab));       ! Open the file
;  326          0322  2
;  327          0323  2 !
;  328          0324  2 !       Map the section into P1 space
;  329          0325  2 !
;  330          0326  2
;  331          0327  2 range [0] = %X'40000000';               ! Signal to EXPREG in P1 region
;  332          0328  2 range [1] = .range [0];
;  333          0329  2
;  334        P 0330  2 return_if_error($CRMPSC(INADR = range,  ! Map the section into P1 space
;  335        P 0331  2                         RETADR = range,  ! Get resulting address range
;  336        P 0332  2                         FLAGS = sec$m_expreg,  ! Expand region to map section
;  337        P 0333  2                         CHAN = .fab [fab$l_stv],! Channel number of open file
;  338        P 0334  2                         ACMODE = psl$c_exec,   ! Owned by exec mode, read only
```

```
;  339     0335  2                 VBN = 2));                  ! Start after image header
;  340     0336  2                                             ! NOTE: Assume 1 block image header
;  341     0337  2
;  342     0338  2
;  343     0339  2 !
;  344     0340  2 !       Change the page protection to user read only.  This
;  345     0341  2 !       is so that the probing in $GETMSG works for all callers.
;  346     0342  2 !
;  347     0343  2
;  348     0344  2 status = $SETPRT(INADR=range,PROT=prt$c_ur);    ! Change to user read only
;  349     0345  2
;  350     0346  2 IF NOT .status                                  ! If error changing protection
;  351     0347  2 THEN
;  352     0348  3     BEGIN
;  353     0349  3     $DELTVA(INADR=range);                       ! Unmap section
;  354     0350  3     $DASSGN(CHAN=.fab [fab$l_stv]);             ! and deassign channel
;  355     0351  3     RETURN .status;                             ! return with status
;  356     0352  2     END;
;  357     0353  2
;  358     0354  2 !
;  359     0355  2 !       Verify validity of message section
;  360     0356  2 !
;  361     0357  2
;  362     0358  2 section = .range [0];                           ! Address the contents of the section
;  363     0359  2
;  364     0360  3 IF (.section [plv$l_type] NEQ plv$c_typ_msg)    ! Verify validity of section
;  365     0361  3     OR (.section [plv$l_version] NEQ 0)
;  366     0362  3     OR (.section [plv$l_msgdsp] NEQ 6)
;  367     0363  3     OR (.section [plv$l_exec] NEQ %X'65160101')
;  368     0364  3     OR (
;  369     0365  4         BEGIN
;  370     0366  4
;  371     0367  4         BIND
;  372     0368  4             rel = section [plv$l_usrundwn],     ! self-relative pointer
;  373     0369  4             msc = rel + .rel : BLOCK [, BYTE];  ! message section
;  374     0370  4
;  375     0371  4         IF PROBER (%REF (psl$c_user), %REF (1), msc [msc$b_type])
;  376     0372  4         THEN
;  377     0373  4             .msc [msc$b_type] NEQ msc$c_msg     ! only message text type allowed
;  378     0374  4         ELSE
;  379     0375  4             true
;  380     0376  4         END
;  381     0377  3         )
;  382     0378  2 THEN                                            ! Message file illegal, unmap it
;  383     0379  3     BEGIN
;  384     0380  3     $DELTVA(INADR=range);                       ! Unmap section
;  385     0381  3     $DASSGN(CHAN=.fab [fab$l_stv]);             ! and deassign channel
;  386     0382  3     RETURN ss$_ivsecidctl;                      ! Signal invalid message file
;  387     0383  2     END;
;  388     0384  2
;  389     0385  2 ctl$gl_ppmsg [0] = .range [0];                  ! Set address of section vector
;  390     0386  2 ctl$gl_ppmsg [1] = .range [1];
;  391     0387  2 ctl$gw_ppmsgchn = .fab [fab$l_stv];             ! Save channel to message section
;  392     0388  2
;  393     0389  2 ctl$gl_ctlbasva = .range [0];                   ! Set new base of fixed P1 region
;  394     0390  2
;  395     0391  2 RETURN true;                                    ! Return successful
```

```
;  396          0392  2
;  397          0393  1 END;


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

                                        45  58  45  2E  00058 P.AAM:   .ASCII  \.EXE\                                  ;

                                                        .EXTRN  SYS$OPEN, SYS$CRMPSC
                                                        .EXTRN  SYS$SETPRT, SYS$DELTVA
                                                        .EXTRN  SYS$DASSGN

                                                        .PSECT  $CODE$,NOWRT,2

                                                00FC 00000 MAP_SECTION:
                                                        .WORD   Save R2,R3,R4,R5,R6,R7                          ; 0278
                                    57 00000000G    00  9E 00002        MOVAB   SYS$DASSGN, R7                  ;
                                    56 00000000G    00  9E 00009        MOVAB   SYS$DELTVA, R6                  ;
                                    5E        A8    AE  9E 00010        MOVAB   -88(SP), SP                     ;
                0050    8F              00          6E    00  2C 00014  MOVC5   #0, (SP), #0, #80, $RMS_PTR      ; 0319
                                                        08  AE    0001B
                                08      AE      5003    8F  B0 0001D    MOVW    #20483, $RMS_PTR                ;
                                0C      AE  00020000    8F  D0 00023    MOVL    #131072, $RMS_PTR+4             ;
                                1E      AE              02  90 0002B    MOVB    #2, $RMS_PTR+22                 ;
                                27      AE              02  90 0002F    MOVB    #2, $RMS_PTR+31                 ;
                                34      AE          04  AC  D0 00033    MOVL    4(AP), $RMS_PTR+44              ;
                                38      AE        0000' CF  9E 00038    MOVAB   P.AAM, $RMS_PTR+48              ;
                                3C      AE              6C  90 0003E    MOVB    (AP), $RMS_PTR+52               ;
                                3D      AE              04  90 00042    MOVB    #4, $RMS_PTR+53                 ;
                                            08  AE  9F  00046    PUSHAB  FAB                                     ; 0321
                        00000000G   00              01  FB  00049    CALLS   #1, SYS$OPEN                    ;
                                    2D              50  E9  00050    BLBC    STATUS, 1$                      ;
                                    6E  40000000    8F  D0  00053    MOVL    #1073741824, RANGE              ; 0327
                                04  AE              6E  D0  0005A    MOVL    RANGE, RANGE+4                  ; 0330
                                                    7E  7C  0005E    CLRQ    -(SP)                           ; 0333
                                                    02  DD  00060    PUSHL   #2                              ;
                                                    7E  D4  00062    CLRL    -(SP)                           ;
                                            24  AE  DD  00064        PUSHL   FAB+12                          ;
                                                    7E  7C  00067    CLRQ    -(SP)                           ;
                                                    7E  D4  00069    CLRL    -(SP)                           ;
                                        00020000    8F  DD  0006B    PUSHL   #131072                         ;
                                                    01  DD  00071    PUSHL   #1                              ;
                                            28  AE  9F  00073        PUSHAB  RANGE                           ;
                                            2C  AE  9F  00076        PUSHAB  RANGE                           ;
                        00000000G   00              0C  FB  00079    CALLS   #12, SYS$CRMPSC                 ;
                                    01              50  E8  00080 1$:  BLBS    STATUS, 2$                      ;
                                                        04  00083    RET                                     ;
                                    7E              0F  7D  00084 2$:  MOVQ    #15, -(SP)                      ; 0344
                                                    7E  7C  00087    CLRQ    -(SP)                           ;
                                            10  AE  9F  00089        PUSHAB  RANGE                           ;
                        00000000G   00              05  FB  0008C    CALLS   #5, SYS$SETPRT                  ;
                                    52              50  D0  00093    MOVL    R0, STATUS                      ;
                                    12              52  E8  00096    BLBS    STATUS, 3$                      ; 0346
                                                    7E  7C  00099    CLRQ    -(SP)                           ; 0349
                                            08  AE  9F  0009B        PUSHAB  RANGE                           ;
                                    66              03  FB  0009E    CALLS   #3, SYS$DELTVA                  ;
```

```
                        14   AE  DD 000A1          PUSHL   FAB+12                                  ; 0350
                    67       01  FB 000A4          CALLS   #1, SYS$DASSGN
                    50       52  D0 000A7          MOVL    STATUS, R0                              ; 0351
                             04  000AA             RET
                    50       6E  D0 000AB 3$:      MOVL    RANGE, SECTION                          ; 0358
                    02       60  D1 000AE          CMPL    (SECTION), #2                           ; 0360
                             25  12 000B1          BNEQ    4$
                        04   A0  D5 000B3          TSTL    4(SECTION)                              ; 0361
                             20  12 000B6          BNEQ    4$
                    06       08  A0  D1 000B8      CMPL    8(SECTION), #6                          ; 0362
                             1A  12 000BC          BNEQ    4$
                65160101     8F  0C  A0  D1 000BE  CMPL    12(SECTION), #1695940865                ; 0363
                             10  12 000C6          BNEQ    4$
                    50       10  C0 000C8          ADDL2   #16, R0                                 ; 0368
                    50       60  C0 000CB          ADDL2   (R0), R0                                ; 0369
            60      01       03  0C 000CE          PROBER  #3, #1, (R0)                            ; 0371
                             04  13 000D2          BEQL    4$
                             60  95 000D4          TSTB    (R0)                                    ; 0373
                             14  13 000D6          BEQL    5$
                             7E  7C 000D8 4$:      CLRQ    -(SP)                                   ; 0380
                        08   AE  9F 000DA          PUSHAB  RANGE
                    66       03  FB 000DD          CALLS   #3, SYS$DELTVA
                        14   AE  DD 000E0          PUSHL   FAB+12                                  ; 0381
                    67       01  FB 000E3          CALLS   #1, SYS$DASSGN
                    50   02E4    8F  3C 000E6      MOVZWL  #740, R0                                ; 0382
                             04  000EB             RET
          00000000G  00          6E  7D 000EC 5$:  MOVQ    RANGE, CTL$GL_PPMSG                     ; 0385
          00000000G  00      14  AE  B0 000F3      MOVW    FAB+12, CTL$GQ_PPMSGCHN                 ; 0387
          00000000G  00          6E  D0 000FB      MOVL    RANGE, CTL$GL_CTLBASVA                  ; 0389
                    50       01  D0 00102          MOVL    #1, R0                                  ; 0391
                             04  00105            RET                                             ; 0393

; Routine Size:  262 bytes,    Routine Base:  $CODE$ + 0147
```

```
;  399          0394  1 ROUTINE delete_section =
;  400          0395  1
;  401          0396  1 !---
;  402          0397  1 !
;  403          0398  1 !       This routine deletes an existing process permanent
;  404          0399  1 !       message file currently mapped in P1 space.
;  405          0400  1 !
;  406          0401  1 ! Inputs:
;  407          0402  1 !
;  408          0403  1 !       ap = Address of parameter descriptor block
;  409          0404  1 !
;  410          0405  1 ! Outputs:
;  411          0406  1 !
;  412          0407  1 !       The section is unmapped; or status returned in setp0$l_status.
;  413          0408  1 !---
;  414          0409  1
;  415          0410  2 BEGIN
;  416          0411  2
;  417          0412  2 IF .ctl$gl_ctlbasva EQL .ctl$gl_ppmsg [0]
;  418          0413  2 THEN
;  419          0414  2     ctl$gl_ctlbasva = .ctl$gl_ppmsg [1] + 1;
;  420          0415  2
;  421          0416  2 setp0$l_status = $DELTVA(INADR=ctl$gl_ppmsg);
;  422          0417  2
;  423          0418  2 $DASSGN(CHAN = .ctl$gw_ppmsgchn);        ! Deassign channel to file
;  424          0419  2 ctl$gw_ppmsgchn = 0;                     ! Indicate channel no longer valid
;  425          0420  2
;  426          0421  2 ctl$gl_ppmsg [0] = 0;                   ! Zero address of section
;  427          0422  2 ctl$gl_ppmsg [1] = 0;
;  428          0423  2
;  429          0424  2 RETURN true;
;  430          0425  2
;  431          0426  1 END;


                                 001C 00000 DELETE_SECTION:
                                            .WORD   Save R2,R3,R4                                   ; 0394
                    54 00000000G   00  9E 00002         MOVAB   CTL$GL_CTLBASVA, R4                 ;
                    53 00000000G   00  9E 00009         MOVAB   CTL$GW_PPMSGCHN, R3                 ;
                    52 00000000G   00  9E 00010         MOVAB   CTL$GL_PPMSG, R2                    ;
                    62             64  D1 00017         CMPL    CTL$GL_CTLBASVA, CTL$GL_PPMSG       ; 0412
                                   05  12 0001A         BNEQ    1$                                  ;
                64        04       A2  01  C1 0001C         ADDL3   #1, CTL$GL_PPMSG+4, CTL$GL_CTLBASVA ; 0414
                                   7E  7C 00021 1$:     CLRQ    -(SP)                               ; 0416
                                   52  DD 00023         PUSHL   R2                                  ;
            00000000G  00          03  FB 00025         CALLS   #3, SYS$DELTVA                      ;
            00000000G  00          50  D0 0002C         MOVL    R0, SETP0$L_STATUS                  ;
                       7E          63  3C 00033         MOVZWL  CTL$GW_PPMSGCHN, -(SP)              ; 0418
            00000000G  00          01  FB 00036         CALLS   #1, SYS$DASSGN                      ;
                                   63  B4 0003D         CLRW    CTL$GW_PPMSGCHN                     ; 0419
                                   62  7C 0003F         CLRQ    CTL$GL_PPMSG                        ; 0421
                    50             01  D0 00041         MOVL    #1, R0                              ; 0424
                                       04 00044         RET                                         ; 0426
```

; Routine Size: 69 bytes,   Routine Base: $CODE$ + 024D

```
;  433          0427  1 END
;  434          0428  0 ELUDOM


;                                   PSECT SUMMARY

;       Name                    Bytes                           Attributes

;  $OWN$                          6  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                       658  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                        92  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)


;                                 Library Statistics

;                                          -------- Symbols --------      Pages       Processing
;       File                               Total    Loaded   Percent      Mapped      Time

;  _$255$DUA28:[SYSLIB]LIB.L32;1           18619        76         0      1000          00:01.8


; Information:  1
; Warnings:     0
; Errors:       0


;                                        COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:SETPOMESS/OBJ=OBJ$:SETPOMESS MSRC$:SETPOMESS/UPDATE=(ENH$:SETPOMESS)

; Size:          658 code + 98 data bytes
; Run Time:          00:15.5
; Elapsed Time:      00:51.7
; Lines/CPU Min:      1658
; Lexemes/CPU-Min:   25612
; Memory Used:   160 pages
; Compilation Complete
```

SETFILE
LIS

SETP0MESS
LIS

SETP0DISP
LIS

SETMISC
LIS

SETPRO
LIS

SETMAIN
LIS