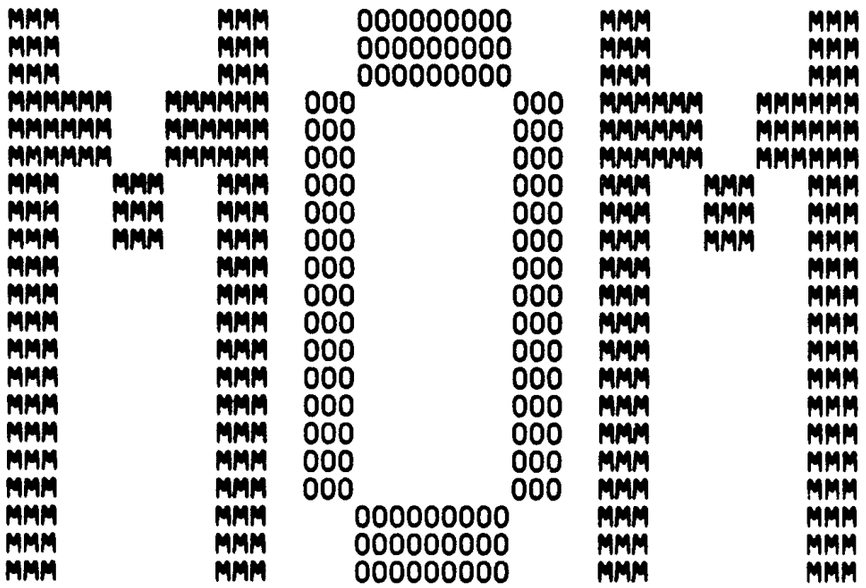

B C D E F G H I J K L M N B C D E F G H I J K L M N B C D E F G H I J K L M N B C D E F G H I

```
MM      MM    000000    MM      MM  MM      MM    AAAAAA      IIIIII    NN      NN
MM      MM    000000    MM      MM  MM      MM    AAAAAA      IIIIII    NN      NN
MMMM  MMMM  00      00  MMMM  MMMM  MMMM  MMMM  AA      AA      II      NN      NN
MMMM  MMMM  00      00  MMMM  MMMM  MMMM  MMMM  AA      AA      II      NN      NN
MM  MM  MM  00      00  MM  MM  MM  MM  MM  MM  AA      AA      II      NNNN    NN
MM  MM  MM  00      00  MM  MM  MM  MM  MM  MM  AA      AA      II      NNNN    NN
MM      MM  00      00  MM      MM  MM      MM  AA      AA      II      NN  NN  NN
MM      MM  00      00  MM      MM  MM      MM  AA      AA      II      NN  NN  NN
MM      MM  00      00  MM      MM  MM      MM  AAAAAAAAAA      II      NN    NNNN
MM      MM  00      00  MM      MM  MM      MM  AAAAAAAAAA      II      NN    NNNN
MM      MM  00      00  MM      MM  MM      MM  AA      AA      II      NN      NN      ....
MM      MM  00      00  MM      MM  MM      MM  AA      AA      II      NN      NN      ....
MM      MM    000000    MM      MM  MM      MM  AA      AA    IIIIII    NN      NN      ....
MM      MM    000000    MM      MM  MM      MM  AA      AA    IIIIII    NN      NN      ....

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 %TITLE 'Network Management Maintenance Operations main module'
;    2          0002  0 MODULE MOMMAIN (MAIN = MOM$MAIN,
;    3          0003  0                 ADDRESSING_MODE (NONEXTERNAL=GENERAL),
;    4          0004  0                 ADDRESSING_MODE (EXTERNAL=GENERAL),
;    5          0005  0                 IDENT = 'V04-000') =
;    6          0006  0
;    7          0007  1 BEGIN
;    8          0008  1
;    9          0009  1 !
;   10          0010  1 !*****************************************************************************
;   11          0011  1 !*                                                                           *
;   12          0012  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;   13          0013  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;   14          0014  1 !*  ALL RIGHTS RESERVED.                                                      *
;   15          0015  1 !*                                                                           *
;   16          0016  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;   17          0017  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;   18          0018  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;   19          0019  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;   20          0020  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;   21          0021  1 !*  TRANSFERRED.                                                              *
;   22          0022  1 !*                                                                           *
;   23          0023  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;   24          0024  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;   25          0025  1 !*  CORPORATION.                                                              *
;   26          0026  1 !*                                                                           *
;   27          0027  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;   28          0028  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;   29          0029  1 !*                                                                           *
;   30          0030  1 !*                                                                           *
;   31          0031  1 !*****************************************************************************
;   32          0032  1 !
;   33          0033  1
;   34          0034  1 !++
;   35          0035  1 ! FACILITY:  DECnet-VAX Network Management Maintenance Operations Module (MOM)
;   36          0036  1 !
;   37          0037  1 ! ABSTRACT:
;   38          0038  1 !       This is the main module for MOM.
;   39          0039  1 !
;   40          0040  1 ! ENVIRONMENT:  VAX/VMS Operating System
;   41          0041  1 !
;   42          0042  1 ! AUTHOR:  Kathy Perko
;   43          0043  1 !
;   44          0044  1 ! CREATION DATE:  13-Jan-1983
;   45          0045  1 !
;   46          0046  1 ! MODIFIED BY:
;   47          0047  1 !       V03-002 MKP0002         Kathy Perko             11-April-1984
;   48          0048  1 !               Add NCP network management version number checking.
;   49          0049  1 !
;   50          0050  1 !       V03-001 MKP0001         Kathy Perko             21-July-1983
;   51          0051  1 !               Improve status returned by various routines.  Add SERVICE
;   52          0052  1 !               NODE VERSION parameter.  Clear CIBs at initialization time.
;   53          0053  1 !
;   54          0054  1 !--
;   55          0055  1
```

```
;    57          0056  1 %SBTTL 'Declarations'
;    58          0057  1
;    59          0058  1 !
;    60          0059  1 ! TABLE OF CONTENTS:
;    61          0060  1 !
;    62          0061  1
;    63          0062  1 FORWARD ROUTINE
;    64          0063  1     mom$main            : NOVALUE,
;    65          0064  1     mom_getmode         : NOVALUE,
;    66          0065  1     mom_initialize      : NOVALUE,
;    67          0066  1     mom_init_logging    : NOVALUE,
;    68          0067  1     mom_get_nice_msg,
;    69          0068  1     mom$send;
;    70          0069  1
;    71          0070  1 !
;    72          0071  1 ! INCLUDE FILES:
;    73          0072  1 !
;    74          0073  1
;    75          0074  1 LIBRARY 'LIB$:MOMLIB';                  ! Facility-wide definitions
;    76          0075  1 LIBRARY 'SHRLIB$:NMALIBRY';             ! NICE definitions
;    77          0076  1 LIBRARY 'SHRLIB$:NET';                  ! NETACP QIO interface
;    78          0077  1 LIBRARY 'SYS$LIBRARY:STARLET';          ! VMS common definitions
;    79          0078  1
;    80          0079  1
;    81          0080  1 !
;    82          0081  1 ! OWN STORAGE:
;    83          0082  1 !
;    84          0083  1 OWN
;    85          0084  1     mom$w_mbx_chan: WORD;
;    86          0085  1
;    87          0086  1 !
;    88          0087  1 ! Externals
;    89          0088  1 !
;    90          0089  1 $MOM_EXTERNALS;
;    91          0090  1
;    92          0091  1 EXTERNAL
;    93          0092  1     mom$ab_nml_mailbox_buf er: BBLOCK,
;    94          0093  1     npa$gl_logmask;
;    95          0094  1
;    96          0095  1 EXTERNAL LITERAL
;    97          0096  1     mom$k_nml_mbx_buf_len;
;    98          0097  1
;    99          0098  1 EXTERNAL ROUTINE
;   100          0099  1     mom$operservice,
;   101          0100  1     mom$autoservice,
;   102          0101  1     mom$trnlognum,
;   103          0102  1     mom$debug_msg;
;   104          0103  1
```

```
;   106          0104  1 %SBTTL 'MOM$MAIN  Main routine'
;   107          0105  1 GLOBAL ROUTINE MOM$MAIN : NOVALUE =
;   108          0106  1
;   109          0107  1 !++
;   110          0108  1 ! FUNCTIONAL DESCRIPTION:
;   111          0109  1 !
;   112          0110  1 !       This is the main routine for the DECnet-VAX Network Management
;   113          0111  1 !       Maintenance Operations Module (MOM).  Maintenance operations
;   114          0112  1 !       include down line load, up line dump, trigger, and loop node,
;   115          0113  1 !       circuit and line.
;   116          0114  1 !
;   117          0115  1 ! STARTUP CRITERIA:
;   118          0116  1 !       MOM is started up in one of two ways:
;   119          0117  1 !           The Network Management Listener (NML) gets a NICE message
;   120          0118  1 !           from NCP requesting a maintenance operation.  This is an
;   121          0119  1 !           operator requested maintenance operation.
;   122          0120  1 !
;   123          0121  1 !       A MOP message is received from another node which requests a
;   124          0122  1 !           maintenance operation.  NETACP will start up MOM to perform
;   125          0123  1 !           the operation.  This is an automatic maintenance operation.
;   126          0124  1 !
;   127          0125  1 !
;   128          0126  1 ! ROUTINE VALUE:
;   129          0127  1 ! COMPLETION CODES:
;   130          0128  1 !
;   131          0129  1 !       Always returns success.
;   132          0130  1 !
;   133          0131  1 !--
;   134          0132  1
;   135          0133  2 BEGIN
;   136          0134  2
;   137          0135  2 BUILTIN
;   138          0136  2         fp;
;   139          0137  2
;   140          0138  2 LOCAL
;   141          0139  2     status;
;   142          0140  2 !
;   143          0141  2 ! Determine if MOM was started up by a MOP message from another node
;   144          0142  2 ! (autoservice) or by NML to process a NICE service request (operservice).
;   145          0143  2 !
;   146          0144  2 mom_initialize ();
;   147          0145  2 mom_getmode ();
;   148          0146  2 IF NOT .mom$gl_service_flags [mom$v_autoservice] THEN
;   149          0147  2     !
;   150          0148  2     ! MOM was started by NML to process a NICE service message.
;   151          0149  2     !
;   152          0150  3     BEGIN
;   153          0151  3     status = mom_get_nice_msg ();
;   154          0152  3     IF .status THEN
;   155          0153  3         mom$operservice ();
;   156          0154  3     END
;   157          0155  2 ELSE
;   158          0156  2     !
;   159          0157  2     ! MOM was started by NETACP because a MOP message was received from
;   160          0158  2     ! another node.  The MOP message can be a request for a down line
;   161          0159  2     ! load, an upline dump, a trigger, or a loop circuit or node.
;   162          0160  2     !
```

```
;  163          0161  3     BEGIN
;  164          0162  3     status = ss$_normal;
;  165          0163  3     mom$autoservice ();
;  166          0164  2     END;
;  167          0165  2 $EXIT (CODE = .status);
;  168          0166  2
;  169          0167  1 END;                        ! End of MOM


;                                                               .TITLE  MOMMAIN Network Management Maintenance Operatio
                                                                        ns main
                                                                .IDENT  \V04-000\

                                                                .PSECT  $OWN$,NOEXE,2

                                                        00000 MOM$W_MBX_CHAN:
                                                                .BLKB   2

                                                                .EXTRN  MOM$GL_LOGMASK, MOM$GL_SVD_INDEX
                                                                .EXTRN  MOM$AB_SERVICE_DATA
                                                                .EXTRN  MOM$GB_FUNCTION
                                                                .EXTRN  MOM$GB_OPTION_BYTE
                                                                .EXTRN  MOM$GB_ENTITY_CODE
                                                                .EXTRN  MOM$AB_ENTITY_BUF
                                                                .EXTRN  MOM$GQ_ENTITY_BUF_DSC
                                                                .EXTRN  MOM$GL_SERVICE_FLAGS
                                                                .EXTRN  MOM$AB_NPARSE_BLK
                                                                .EXTRN  MOM$AB_NICE_RCV_BUF
                                                                .EXTRN  MOM$AB_NICE_XMIT_BUF
                                                                .EXTRN  MOM$GQ_NICE_RCV_BUF_DSC
                                                                .EXTRN  MOM$GL_NICE_RCV_MSG_LEN
                                                                .EXTRN  MOM$GQ_NICE_XMIT_BUF_DSC
                                                                .EXTRN  MOM$AB_MSGBLOCK
                                                                .EXTRN  MOM$AB_ACPQIO_BUFFER
                                                                .EXTRN  MOM$GQ_ACPQIO_BUF_DSC
                                                                .EXTRN  MOM$AB_CIB, MOM$AB_LOOP_CIB
                                                                .EXTRN  MOM$AB_TRIGGER_CIB
                                                                .EXTRN  MOM$AB_MOP_XMIT_BUF
                                                                .EXTRN  MOM$GQ_MOP_XMIT_BUF_DSC
                                                                .EXTRN  MOM$AB_MOP_RCV_BUF
                                                                .EXTRN  MOM$GQ_MOP_RCV_BUF_DSC
                                                                .EXTRN  MOM$AB_MOP_MSG, MOM$GQ_MOP_MSG_DSC
                                                                .EXTRN  MOM$GW_EVT_CODE
                                                                .EXTRN  MOM$GB_EVT_POPR
                                                                .EXTRN  MOM$GB_EVT_PRSN
                                                                .EXTRN  MOM$GB_EVT_PSER
                                                                .EXTRN  SVD$GK_PCNO_ADD
                                                                .EXTRN  SVD$GK_PCNO_SDV
                                                                .EXTRN  SVD$GK_PCNO_CPU
                                                                .EXTRN  SVD$GK_PCNO_STY
                                                                .EXTRN  SVD$GK_PCNO_DAD
                                                                .EXTRN  SVD$GK_PCNO_DCT
                                                                .EXTRN  SVD$GK_PCNO_IHO
                                                                .EXTRN  SVD$GK_PCNO_NNA
                                                                .EXTRN  SVD$GK_PCNO_SLI
                                                                .EXTRN  SVD$GK_PCNO_SPA
                                                                .EXTRN  SVD$GK_PCNO_HWA
```

```
                                                .EXTRN  SVD$GK_PCNO_SNV
                                                .EXTRN  SVD$GK_PCNO_LOA
                                                .EXTRN  SVD$GK_PCNO_SLO
                                                .EXTRN  SVD$GK_PCNO_TLO
                                                .EXTRN  SVD$GK_PCNO_DFL
                                                .EXTRN  SVD$GK_PCNO_SID
                                                .EXTRN  SVD$GK_PCNO_DUM
                                                .EXTRN  SVD$GK_PCNO_SDU
                                                .EXTRN  SVD$GK_PCNO_$HNA
                                                .EXTRN  SVD$GK_PCNO_$HHW
                                                .EXTRN  SVD$GK_PCNO_$FTY
                                                .EXTRN  SVD$GK_PCNO_PHA
                                                .EXTRN  SVD$GK_PCNO_$DA
                                                .EXTRN  SVD$GK_PCNO_LPC
                                                .EXTRN  SVD$GK_PCNO_LPL
                                                .EXTRN  SVD$GK_PCNO_LPD
                                                .EXTRN  SVD$GK_PCNO_LPH
                                                .EXTRN  SVD$GK_PCNO_LPA
                                                .EXTRN  SVD$GK_PCNO_LPN
                                                .EXTRN  SVD$GK_PCNO_$LNA
                                                .EXTRN  SVD$GK_PCNO_$LNH
                                                .EXTRN  SVD$GK_PCNO_LAN
                                                .EXTRN  SVD$GK_PCNO_$LNN
                                                .EXTRN  SVD$GK_PCNO_$LAH
                                                .EXTRN  SVD$GK_PCLI_STI
                                                .EXTRN  SVD$C_ENTRY_COUNT
                                                .EXTRN  MOM$AB_NML_MAILBOX_BUFFER
                                                .EXTRN  NPA$GL_LOGMASK, MOM$K_NML_MBX_BUF_LEN
                                                .EXTRN  MOM$OPERSERVICE
                                                .EXTRN  MOM$AUTOSERVICE
                                                .EXTRN  MOM$TRNLOGNUM, MOM$DEBUG_MSG
                                                .EXTRN  SYS$EXIT

                                                .PSECT  $CODE$,NOWRT,2

                               0004 00000       .ENTRY  MOM$MAIN, Save R2                  ; 0105
00000000V  00             00  FB 00002          CALLS   #0, MOM_INITIALIZE                 ; 0144
00000000V  00             00  FB 00009          CALLS   #0, MOM_GETMODE                    ; 0145
        16 00000000G      00  E8 00010          BLBS    MOM$GL_SERVICE_FLAGS, 1$           ; 0146
00000000V  00             00  FB 00017          CALLS   #0, MOM_GET_NICE_MSG               ; 0151
           52             50  D0 0001E          MOVL    R0, STATUS                         ;
           13             52  E9 00021          BLBC    STATUS, 2$                         ; 0152
00000000G  00             00  FB 00024          CALLS   #0, MOM$OPERSERVICE                ; 0153
                          0A  11 0002B          BRB     2$                                 ; 0146
           52             01  D0 0002D  1$:     MOVL    #1, STATUS                         ; 0162
00000000G  00             00  FB 00030          CALLS   #0, MOM$AUTOSERVICE                ; 0163
                          52  DD 00037  2$:     PUSHL   STATUS                             ; 0165
00000000G  00             01  FB 00039          CALLS   #1, SYS$EXIT
                              04 00040          RET                                        ; 0167

; Routine Size:  65 bytes,    Routine Base:  $CODE$ + 0000
```

```
;   171          0168  1 %SBTTL 'MOM_GETMODE  Determine the mode in MOM is running'
;   172          0169  1 ROUTINE MOM_GETMODE: NOVALUE =
;   173          0170  1
;   174          0171  1 !++
;   175          0172  1 ! FUNCTIONAL DESCRIPTION:
;   176          0173  1 !
;   177          0174  1 !       This routine determines the mode in which MOM is running by
;   178          0175  1 !       determining how the image was activated.  MOM can be started
;   179          0176  1 !       up in two modes:
;   180          0177  1 !               Autoservice - a target node has sent a MOP message requesting
;   181          0178  1 !                       a maintenance function (load, dump, loop, or trigger).
;   182          0179  1 !               Operservice - The maintenance function is being requested
;   183          0180  1 !                       via a NICE message from NCP.
;   184          0181  1 !       MOM gets started up in one of three ways:
;   185          0182  1 !               By NMLSHR running as a sharable image with NCP and communicating
;   186          0183  1 !                       with NCP via buffers.
;   187          0184  1 !               By NMLSHR running as a sharable image with NML and communicating
;   188          0185  1 !                       with NCP via a logical link.
;   189          0186  1 !               By NETACP when it receives an autoservice request on a circuit.
;   190          0187  1 !
;   191          0188  1 ! IMPLICIT OUTPUTS:
;   192          0189  1 !
;   193          0190  1 !       If MOM was started up by an autoservice function,
;   194          0191  1 !       mom$gl_service_flags [mom$v_autoservice] is set.
;   195          0192  1 !       Otherwise it is clear.
;   196          0193  1 !
;   197          0194  1 !--
;   198          0195  1
;   199          0196  2 BEGIN
;   200          0197  2
;   201          0198  2 LOCAL
;   202          0199  2     bufdesc: VECTOR [2],                ! Descriptor of above buffer
;   203          0200  2     status;                             ! Temporary status
;   204          0201  2
;   205          0202  2 !
;   206          0203  2 ! Translate SYS$NET to determine what mode MOM was started up in.
;   207          0204  2 !
;   208          0205  2 bufdesc [0] = mom$k_nice_buf_len;       ! Setup result buffer descriptor
;   209          0206  2 bufdesc [1] = mom$ab_nice_rcv_buf;
;   210          0207  2
;   211        P 0208  2 status = $TRNLOG (LOGNAM = %ASCID 'SYS$NET',
;   212        P 0209  2                   RSLLEN = bufdesc [0],
;   213          0210  2                   RSLBUF = bufdesc);
;   214          0211  2 !
;   215          0212  2 ! If SYS$NET translates, MOM was activated by an NML talking to NCP via
;   216          0213  2 ! a logical link (operservice), or by NETACP (autoservice).  It
;   217          0214  2 ! was NML if the translation contains the "::" indicating a node name.
;   218          0215  2 ! If not, it is autoservice.
;   219          0216  2 !
;   220          0217  2 IF .status EQLU ss$_normal AND
;   221          0218  2    CH$FAIL (CH$FIND_SUB (.bufdesc [0],
;   222          0219  2                          mom$ab_nice_rcv_buf,
;   223          0220  2                          2,
;   224          0221  2                          UPLIT ('::'))) THEN
;   225          0222  3     BEGIN
;   226          0223  3     !
;   227          0224  3     ! MOM was activated by NETACP for an autoservice operation.
```

```
;   228          0225  3     ! Sys$net translates to the name of the circuit over which the
;   229          0226  3     ! autoservice MOP request was received.  Save the circuit ID both
;   230          0227  3     ! as an entity ID and in the Service Data Table.
;   231          0228  3     !
;   232          0229  3     mom$gl_service_flags [mom$v_autoservice] = true;
;   233          0230  3     mom$gq_entity_buf_dsc [0] = .bufdesc [0];
;   234          0231  3     mom$gq_entity_buf_dsc [1] = mom$ab_nice_rcv_buf;
;   235          0232  3     mom$ab_service_data [svd$gk_pcno_sli, svd$b_string_len] = .bufdesc [0];
;   236          0233  3     CH$MOVE (.bufdesc [0],
;   237          0234  3              mom$ab_nice_rcv_buf,
;   238          0235  3              mom$ab_service_data [svd$gk_pcno_sli, svd$t_string]);
;   239          0236  3     mom$ab_service_data [svd$gk_pcno_sli, svd$v_msg_param] = true;
;   240          0237  3     END
;   241          0238  2 ELSE
;   242          0239  2     !
;   243          0240  2     ! NML activated MOM to process a NICE message requesting a maintenance
;   244          0241  2     ! operation.
;   245          0242  2     !
;   246          0243  2     mom$gl_service_flags [mom$v_autoservice] = false;
;   247          0244  2
;   248          0245  1 END;                              ! End of MOM_GETMODE


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

                    00  54  53  4E  24  53  59  53  00000 P.AAB:  .ASCII  \SYS$NET\<0>                                    ;
                                      010E0007  00008 P.AAA:  .LONG   17694727                                          ;
                                      00000000' 0000C         .ADDRESS P.AAB                                            ;
                                00  00  3A  3A  00010 P.AAC:  .ASCII  \::\<0><0>                                         ;

                                                        .EXTRN  SYS$TRNLOG

                                                        .PSECT  $CODE$,NOWRT,2

                                            00FC 00000 MOM_GETMODE:
                                                        .WORD   Save R2,R3,R4,R5,R6,R7                                  ; 0169
                                    57 00000000G    00  9E 00002        MOVAB   MOM$GL_SERVICE_FLAGS, R7                ;
                                    56 00000000G    00  9E 00009        MOVAB   MOM$AB_NICE_RCV_BUF, R6                 ;
                                    5E              04  C2 00010        SUBL2   #4, SP                                  ;
                                    7E          C5  8F  9A 00013        MOVZBL  #197, BUFDESC                           ; 0205
                                04  AE              66  9E 00017        MOVAB   MOM$AB_NICE_RCV_BUF, BUFDESC+4          ; 0206
                                                    7E  7C 0001B        CLRQ    -(SP)                                   ; 0210
                                                    7E  D4 0001D        CLRL    -(SP)                                   ;
                                                0C  AE  9F 0001F        PUSHAB  BUFDESC                                 ;
                                                10  AE  9F 00022        PUSHAB  BUFDESC                                 ;
                                    00000000'   00  9F 00025        PUSHAB  P.AAA                                   ;
                            00000000G   00          06  FB 0002B        CALLS   #6, SYS$TRNLOG                          ;
                                        01          50  D1 00032        CMPL    STATUS, #1                              ; 0217
                                                    3B  12 00035        BNEQ    2$                                      ;
66                      6E  00000000'   00          02  39 00037        MATCHC  #2, P.AAC, BUFDESC, MOM$AB_NICE_RCV_BUF ; 0221
                                                    03  13 00040        BEQL    1$                                      ;
                                        53          02  D0 00042        MOVL    #2, R3                                  ;
                                        53          02  C2 00045 1$:     SUBL2   #2, R3                                  ;
                                                    28  12 00048        BNEQ    2$                                      ;
                                        67          01  88 0004A        BISB2   #1, MOM$GL_SERVICE_FLAGS                ; 0229
                            00000000G   00          6E  D0 0004D        MOVL    BUFDESC, MOM$GQ_ENTITY_BUF_DSC          ; 0230
```

```
                    00000000G  00          66  9E 00054          MOVAB   MOM$AB_NICE_RCV_BUF, -                    ; 0231
                                                                  MOM$GQ_ENTITY_BUF_DSC+4
                    00000000*  00          6E  90 0005B          MOVB    BUFDESC, <<MOM$AB_SERVICE_DATA+-           ; 0232
                                                                  <SVD$GK_PCNO_SLI*137>>+8>
          00000000*  00                    66  6E 28 00062       MOVC3   BUFDESC, MOM$AB_NICE_RCV_BUF, <-           ; 0235
                                                                  <MOM$AB_SERVICE_DATA+<SVD$GK_PCNO_SLI*137>>-
                                                                  +9>
                    00000000*  00          01  88 0006A          BISB2   #1, <<MOM$AB_SERVICE_DATA+<SVD$GK_PCNO_SLI*-  ; 0236
                                                                  137>>+7>
                                              04 00071          RET                                                ; 0217
                               67           01  8A 00072 2$:     BICB2   #1, MOM$GL_SERVICE_FLAGS                  ; 0243
                                              04 00075          RET                                                ; 0245

; Routine Size:  118 bytes,    Routine Base:  $CODE$ + 0041
```

```
;  250          0246  1 %SBTTL 'MOM_INITIALIZE   Initialization for MOM'
;  251          0247  1 ROUTINE MOM_INITIALIZE: NOVALUE =
;  252          0248  1
;  253          0249  1 !++
;  254          0250  1 !
;  255          0251  1 !       This routine performs various initialization for MOM at the
;  256          0252  1 !       beginning of each NICE command or autoservice function.
;  257          0253  1 !       The most notable of these is to set the default values
;  258          0254  1 !       for the service data in the Service Data table.
;  259          0255  1 !
;  260          0256  1 ! Inputs:
;  261          0257  1 !       none
;  262          0258  1 !
;  263          0259  1 !--
;  264          0260  1
;  265          0261  2 BEGIN
;  266          0262  2
;  267          0263  2 MAP
;  268          0264  2     mom$gl_service_flags;
;  269          0265  2
;  270          0266  2 mom$gl_service_flags = 0;
;  271          0267  2 !
;  272          0268  2 ! Initialize the Service Data Table to indicate that none of the parameter
;  273          0269  2 ! values currently in the table came from a NICE or MOP message.  The
;  274          0270  2 ! svd$v_msg_param field is used to make sure the NICE and MOP message
;  275          0271  2 ! parameter values are used over the parameter values from the volatile
;  276          0272  2 ! database.
;  277          0273  2 !
;  278          0274  2 INCR i FROM 0 TO svd$c_entry_count DO
;  279          0275  3     BEGIN
;  280          0276  3     mom$ab_service_data [.i, svd$b_flags] = 0;
;  281          0277  3     mom$ab_service_data [.i, svd$b_string_len] = 0;
;  282          0278  3     mom$ab_service_data [.i, svd$l_param] = 0;
;  283          0279  2     END;
;  284          0280  2 !
;  285          0281  2 ! Default the line service timer to infinity.
;  286          0282  2 !
;  287          0283  2 mom$ab_service_data [svd$gk_pcli_sti, svd$l_param] = -1;
;  288          0284  2 !
;  289          0285  2 ! Just in case it's a loop command, default the loop message length,
;  290          0286  2 ! loop message data type, and number of messages looped.
;  291          0287  2 !
;  292          0288  2 mom$ab_service_data [svd$gk_pcno_lpl, svd$l_param] = nma$c_loop_dsiz;
;  293          0289  2 mom$ab_service_data [svd$gk_pcno_lpd, svd$l_param] = nma$c_loop_mix;
;  294          0290  2 mom$ab_service_data [svd$gk_pcno_lpc, svd$l_param] = nma$c_loop_dcnt;
;  295          0291  2 mom$ab_service_data [svd$gk_pcno_lph, svd$l_param] = mom$k_no_loop_help;
;  296          0292  2 !
;  297          0293  2 ! Just in case it's a load, default the first file loaded to the
;  298          0294  2 ! secondary loader and default the system file to be loaded to
;  299          0295  2 ! the operating system (as opposed to diagnostics) and Phase IV.
;  300          0296  2 !
;  301          0297  2 mom$ab_service_data [svd$gk_pcno_sty, svd$l_param] = nma$c_soft_secl;
;  302          0298  2 mom$ab_service_data [svd$gk_pcno_$fty, svd$l_param] = mop$c_sid_osy;
;  303          0299  2 mom$ab_service_data [svd$gk_pcno_snv, svd$l_param] = nma$c_nodsnv_ph4;
;  304          0300  2 !
;  305          0301  2 ! Clear the (Channel Information Blocks) CIBs - these blocks contain
;  306          0302  2 ! information about the I/O channel over which the service function
```

```
;  307          0303  2 ! is being performed.
;  308          0304  2 !
;  309          0305  2 CH$FILL (0, cib$c_ciblen, mom$ab_cib);
;  310          0306  2 CH$FILL (0, cib$c_ciblen, mom$ab_loop_cib);
;  311          0307  2 mom_init_logging ();
;  312          0308  2 !
;  313          0309  2 ! Clear out the skip MOP message buffer.  This buffer is used to save
;  314          0310  2 ! certain MOP messages that must be ignored if received more than once.
;  315          0311  2 !
;  316          0312  2 mom$gq_mop_msg_dsc [0] = 0;
;  317          0313  1 END;                                ! of MOM_INITIALIZE


                                                007C 00000 MOM_INITIALIZE:
                                                                .WORD   Save R2,R3,R4,R5,R6                                     ; 0247
                                 56 00000000G   00    9E 00002          MOVAB   MOM$AB_SERVICE_DATA+7, R6
                                    00000000G   00    D4 00009          CLRL    MOM$GL_SERVICE_FLAGS                          ; 0266
                                 50             01    CE 0000F          MNEGL   #1, I                                         ; 0274
                                                15    11 00012          BRB     2$
                              51 50 00000089    8F    C5 00014 1$:      MULL3   #137, I, R1                                   ; 0276
                                          6641  94    00 0001C          CLRB    MOM$AB_SERVICE_DATA+7[R1]
                                       01 A641  94    00 0001F          CLRB    MOM$AB_SERVICE_DATA+8[R1]                     ; 0277
                                       02 A641  9F    00 00023          PUSHAB  MOM$AB_SERVICE_DATA+9[R1]                     ; 0278
                                                9E    D4 00027          CLRL    @(SP)+
                              E3 50 00000000G   8F    F3 00029 2$:      AOBLEQ  #SVD$C_ENTRY_COUNT, I, 1$                     ; 0274
                 00000000*    00                01    CE 00031          MNEGL   #1, <<MOM$AB_SERVICE_DATA+<SVD$GK_PCLI_STI*-  ; 0283
                                                                                  137>>+9>
                 00000000*    00                28    D0 00038          MOVL    #40, <<MOM$AB_SERVICE_DATA+<SVD$GK_PCNO_LPL-  ; 0288
                                                                                  *137>>+9>
                 00000000*    00                02    D0 0003F          MOVL    #2, <<MOM$AB_SERVICE_DATA+<SVD$GK_PCNO_LPD*-  ; 0289
                                                                                  137>>+9>
                 00000000*    00                01    D0 00046          MOVL    #1, <<MOM$AB_SERVICE_DATA+<SVD$GK_PCNO_LPC*-  ; 0290
                                                                                  137>>+9>
                 00000000*    00                01    CE 0004D          MNEGL   #1, <<MOM$AB_SERVICE_DATA+<SVD$GK_PCNO_LPH*-  ; 0291
                                                                                  137>>+9>
                                    00000000*    00    D4 00054          CLRL    <<MOM$AB_SERVICE_DATA+<SVD$GK_PCNO_STY*137>-  ; 0297
                                                                                  >+9>
                 00000000*    00          FF    8F    9A 0005A          MOVZBL  #255, <<MOM$AB_SERVICE_DATA+-                 ; 0298
                                                                                  <SVD$GK_PCNO_$FTY*137>5+9>
                 00000000*    00                0f    D0 00062          MOVL    #1, <<MOM$AB_SERVICE_DATA+<SVD$GK_PCNO_SNV*-  ; 0299
                                                                                  137>>+9>
        004C     8F           00                6E          00    2C 00069  MOVC5   #0, (SP), #0, #76, MOM$AB_CIB               ; 0305
                                                      00000000G   00    00070
        004C     8F           00                6E          00    2C 00075  MOVC5   #0, (SP), #0, #76, MOM$AB_LOOP_CIB          ; 0306
                                                      00000000G   00    0007C
                 00000000V    00                      00    FB 00081  CALLS   #0, MOM_INIT_LOGGING                          ; 0307
                                                      00000000G   00    D4 00088  CLRL    MOM$GQ_MOP_MSG_DSC                    ; 0312
                                                                  04    0008E  RET                                            ; 0313

; Routine Size:  143 bytes,    Routine Base:  $CODE$ + 00B7
```

```
;  319          0314  1 %SBTTL 'MOM_INIT_LOGGING  Initialization debug logging'
;  320          0315  1
;  321          0316  1 ROUTINE MOM_INIT_LOGGING: NOVALUE =
;  322          0317  1
;  323          0318  1 !++
;  324          0319  1 !
;  325          0320  1 !       This routine initializes the internal logging flags for MOM debugging.
;  326          0321  1 !       The logical name NML$LOG is translated to get the flag settings.
;  327          0322  1 !
;  328          0323  1 ! Inputs:
;  329          0324  1 !
;  330          0325  1 !       The logical name NML$LOG
;  331          0326  1 !
;  332          0327  1 !--
;  333          0328  1
;  334          0329  2 BEGIN
;  335          0330  2
;  336          0331  2 !
;  337          0332  2 ! Set internal logging flags if NML$LOG is defined.
;  338          0333  2 !
;  339          0334  2 mom$trnlognum ($ASCID ('NML$LOG'), mom$gl_logmask);
;  340          0335  2
;  341          0336  2 !
;  342          0337  2 ! If the NPARSE logging flag is set then set it in the NPARSE data area.
;  343          0338  2 !
;  344          0339  2
;  345          0340  2 IF .mom$gl_logmask [dbg$c_nparse]
;  346          0341  2 THEN
;  347          0342  2     npa$gl_logmask = 1
;  348          0343  2 ELSE
;  349          0344  2     npa$gl_logmask = 0;
;  350          0345  2
;  351          0346  1 END;                                    ! of MOM_INIT_LOGGING


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

                        47  4F  4C  24  4C  4D  4E  00014 P.AAE:  .ASCII  \NML$LOG\                       ;
                                                    0001B       .BLKB   1
                                            00000007    0001C P.AAD:  .LONG   7                         ;
                                            00000000'   00020         .ADDRESS P.AAE                    ;


                                                        .PSECT  $CODE$,NOWRT 2

                                                000C 00000 MOM_INIT_LOGGING:
                                                           .WORD   Save R2,R3                           ; 0316
                                    53 00000000G    00  9E 00002           MOVAB   MOM$GL_LOGMASK, R3   ;
                                    52 00000000G    00  9E 00009           MOVAB   NPA$GL_LOGMASK, R2   ;
                                                    53  DD 00010           PUSHL   R3                   ; 0334
                                       00000000'    00  9F 00012           PUSHAB  P.AAD                ;
                            00000000G   00          02  FB 00018           CALLS   #2, MOM$TRNLOGNUM    ;
                        04              63          02  E1 0001F           BBC     #2, MOM$GL_LOGMASK, 1$ ; 0340
                                        62          01  D0 00023           MOVL    #1, NPA$GL_LOGMASK     ; 0342
                                                    04  00026           RET                            ;
```

```
                                                62  D4 00027 1$:     CLRL    NPA$GL_LOGMASK                          ; 0344
                                                    04 00029         RET                                             ; 0376

; Routine Size:  42 bytes,    Routine Base:  $CODE$ + 0146
```

```
;  353          0347  1 %SBTTL 'MOM_GET_NICE_MSG   Get the NICE service command message'
;  354          0348  1 ROUTINE MOM_GET_NICE_MSG =
;  355          0349  1
;  356          0350  1 !++
;  357          0351  1 ! FUNCTIONAL DESCRIPTION:
;  358          0352  1 !
;  359          0353  1 !       This routine is called by MOM$MAIN when MOM is processing a
;  360          0354  1 !       maintenance operation which was requested via NCP.  The routine
;  361          0355  1 !       gets the NCP NICE message from the mailbox where it was put
;  362          0356  1 !       by NML before it started MOM.
;  363          0357  1 !
;  364          0358  1 ! IMPLICIT INPUTS:
;  365          0359  1 !       NML$MOMMBX - translates to name of mailbox that NML wrote the
;  366          0360  1 !               NICE message to before spawning MOM.
;  367          0361  1 !--
;  368          0362  2 BEGIN
;  369          0363  2
;  370          0364  2 LOCAL
;  371          0365  2     status,                                 ! Temporary status
;  372          0366  2     iosb: $iosb;
;  373          0367  2
;  374          0368  2 !
;  375          0369  2 ! Assign a channel to NML$MOMMBX, and, if I got it, read NML's message
;  376          0370  2 ! from it.  The first 3 bytes of the message will contain the network
;  377          0371  2 ! management version of the NCP originating the message.
;  378          0372  2 !
;  379        P 0373  2 status = $ASSIGN (DEVNAM = %ASCID 'NML$MOM_MBX',
;  380          0374  2                   CHAN = mom$w_mbx_chan);
;  381          0375  2 IF .status THEN
;  382          0376  3     BEGIN
;  383        P 0377  3     status = $QIOW (
;  384        P 0378  3                 FUNC = io$_readvblk,            ! Read NML's mailbox
;  385        P 0379  3                 CHAN = .mom$w_mbx_chan,         ! Mailbox channel
;  386        P 0380  3                 IOSB = iosb,
;  387        P 0381  3                 P1 = mom$ab_nml_mailbox_buffer, ! Buffer for NML info
;  388          0382  3                 P2 = mom$k_nml_mbx_buf_len);    ! Length of mailbox buffer
;  389          0383  3     IF .status THEN
;  390          0384  3         status = .iosb [ios$w_status];
;  391          0385  3     !
;  392          0386  3     ! If got the NICE message OK, save the length.  Compensate for the 3
;  393          0387  3     ! bytes of NCP management version at the beginning of the buffer.
;  394          0388  3     !
;  395          0389  3     IF .status THEN
;  396          0390  4         BEGIN
;  397          0391  4         mom$gl_nice_rcv_msg_len = .iosb [ios$w_count] - 3;
;  398          0392  4         mom$debug_msg (dbg$c_netio,             ! Log message transmitted
;  399          0393  4                        mom$ab_nice_rcv_buf,
;  400          0394  4                        .mom$gl_nice_rcv_msg_len,
;  401          0395  4                        %ASCID 'MOM NICE message received from NML');
;  402          0396  3         END;
;  403          0397  2     END;
;  404          0398  2 RETURN .status;
;  405          0399  1 END;                                    ! of MOM_GET_NICE_MSG


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2
```

```
                00  58  42  4D  5F  4D  4F  4D  24  4C  4D  4E  00024 P.AAG:  .ASCII  \NML$MOM_MBX\<0>
                                                    010E000B  00030 P.AAF:  .LONG   17694731
                                                    00000000'  00034       .ADDRESS P.AAG
67  61  73  73  65  6D  20  45  43  49  4E  20  4D  4F  4D  00038 P.AAI:  .ASCII  \MOM NICE message received from NML\<0>
6D  6F  72  66  20  64  65  76  69  65  63  65  72  20  65  00047
                                        00  4C  4D  4E  20  00056
                                                        00  0005B       .ASCII  <0>
                                                    010E0022  0005C P.AAH:  .LONG   17694754
                                                    00000000'  00060       .ADDRESS P.AAI

                                                                           .EXTRN  SYS$ASSIGN, SYS$QIOW

                                                                           .PSECT  $CODE$,NOWRT,2

                                                        001C  00000 MOM_GET_NICE_MSG:
                                                                           .WORD   Save R2,R3,R4                                  ; 0348
                                    54 00000000'    00  9E  00002          MOVAB   MOM$W_MBX_CHAN, R4                             ;
                                    53 00000000G    00  9E  00009          MOVAB   MOM$GL_NICE_RCV_MSG_LEN, R3                    ;
                                    5E                  08  C2  00010          SUBL2   #8, SP                                     ;
                                                        7E  7C  00013          CLRQ    -(SP)                                      ; 0374
                                                        54  DD  00015          PUSHL   R4                                         ;
                                       00000000'        00  9F  00017          PUSHAB  P.AAF                                      ;
                            00000000G   00              04  FB  0001D          CALLS   #4, SYS$ASSIGN                             ;
                                    52                  50  D0  00024          MOVL    R0, STATUS                                 ;
                                    4D                  52  E9  00027          BLBC    STATUS, 1$                                 ; 0375
                                                        7E  7C  0002A          CLRQ    -(SP)                                      ; 0382
                                                        7E  7C  0002C          CLRQ    -(SP)                                      ;
                                       00000000G        8F  DD  0002E          PUSHL   #MOM$K_NML_MBX_BUF_LEN                     ;
                                       00000000G        00  9F  00034          PUSHAB  MOM$AB_NML_MAILBOX_BUFFER                  ;
                                                        7E  7C  0003A          CLRQ    -(SP)                                      ;
                                             20         AE  9F  0003C          PUSHAB  IOSB                                       ;
                                                        31  DD  0003F          PUSHL   #49                                        ;
                                    7E                  64  3C  00041          MOVZWL  MOM$W_MBX_CHAN, -(SP)                      ;
                                                        7E  D4  00044          CLRL    -(SP)                                      ;
                            00000000G   00              0C  FB  00046          CALLS   #12, SYS$QIOW                              ;
                                    52                  50  D0  0004D          MOVL    R0, STATUS                                 ;
                                    24                  52  E9  00050          BLBC    STATUS, 1$                                 ; 0383
                                    52                  6E  3C  00053          MOVZWL  IOSB, STATUS                               ; 0384
                                    1E                  52  E9  00056          BLBC    STATUS, 1$                                 ; 0389
                                    63           02     AE  3C  00059          MOVZWL  IOSB+2, MOM$GL_NICE_RCV_MSG_LEN            ; 0391
                                    63                  03  C2  0005D          SUBL2   #3, MOM$GL_NICE_RCV_MSG_LEN                ;
                                       00000000'        00  9F  00060          PUSHAB  P.AAH                                      ; 0394
                                                        63  DD  00066          PUSHL   MOM$GL_NICE_RCV_MSG_LEN                    ;
                                       00000000G        00  9F  00068          PUSHAB  MOM$AB_NICE_RCV_BUF                        ; 0392
                                                        7E  D4  0006E          CLRL    -(SP)                                      ;
                            00000000G   00              04  FB  00070          CALLS   #4, MOM$DEBUG_MSG                          ;
                                    50                  52  D0  00077 1$:      MOVL    STATUS, R0                                 ; 0398
                                                            04  0007A          RET                                                ; 0399

; Routine Size:  123 bytes,    Routine Base:  $CODE$ + 0170
```

```
;  407          0400  1 %SBTTL 'MOM$SEND   Send NICE response to caller'
;  408          0401  1
;  409          0402  1 GLOBAL ROUTINE MOM$SEND (BUF_ADDRESS, BUF_LENGTH) =
;  410          0403  1
;  411          0404  1 !++
;  412          0405  1 !
;  413          0406  1 ! Function:
;  414          0407  1 !       The routine writes the NICE status messages back to NML's
;  415          0408  1 !       mailbox.  NML will then read the mailbox and forward the response
;  416          0409  1 !       to NCP. The status messages give the completion status of the
;  417          0410  1 !       requested maintenance operation.
;  418          0411  1 !
;  419          0412  1 ! Inputs:
;  420          0413  1 !
;  421          0414  1 !       buf_address     Address of the buffer to be transmitted.
;  422          0415  1 !       buf_length      Length of the buffer in bytes.
;  423          0416  1 !
;  424          0417  1 ! Outputs:
;  425          0418  1 !
;  426          0419  1 !       Returns success.  Errors are signalled.
;  427          0420  1 !--
;  428          0421  1
;  429          0422  2 BEGIN
;  430          0423  2
;  431          0424  2 LOCAL
;  432          0425  2     status,
;  433          0426  2     iosb:       $iosb;
;  434          0427  2
;  435          0428  2 mom$debug_msg(dbg$c_netio,               ! Log message transmitted
;  436          0429  2                .buf_address,
;  437          0430  2                .buf_length,
;  438          0431  2                %ASCID 'MOM NICE message sent to NML');
;  439          0432  2
;  440        P 0433  2 status = $QIOW (
;  441        P 0434  2                 FUNC = io$_writevblk OR io$m_now,! Write to NML's mailbox
;  442        P 0435  2                 CHAN = .mom$w_mbx_chan,         ! Mailbox channel
;  443        P 0436  2                 IOSB = iosb,
;  444        P 0437  2                 P1 = .buf_address,              ! Buffer for NICE message
;  445          0438  2                 P2 = .buf_length);              ! Length of NICE buffer
;  446          0439  2 !
;  447          0440  2 ! If the QIO to the mailbox was successful then get look at the IOSB.
;  448          0441  2 !
;  449          0442  2
;  450          0443  2 IF .status THEN
;  451          0444  2     status = .iosb [ios$w_status];
;  452          0445  2
;  453          0446  2 !
;  454          0447  2 ! If status is bad then signal the error.
;  455          0448  2 !
;  456          0449  2
;  457          0450  2 IF NOT .status THEN
;  458          0451  2     SIGNAL_STOP (.status);
;  459          0452  2
;  460          0453  2
;  461          0454  2 RETURN true;                            ! Return successful
;  462          0455  2
;  463          0456  1 END;                        ! of MOM$SEND
```

```
                                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

67  61  73  73  65  6D  20  45  43  49  4E  20  4D  4F  4D  00064 P.AAK:   .ASCII  \MOM NICE message sent to NML\                    ;
        4C  4D  4E  20  6F  74  20  74  6E  65  73  20  65  00073                                                                 ;
                                                    010E001C  00080 P.AAJ:   .LONG   17694748                                    ;
                                                    00000000'  00084         .ADDRESS P.AAK                                     ;


                                                                                .PSECT  $CODE$,NOWRT,2

                                                        0000  00000             .ENTRY  MOM$SEND, Save nothing                    ; 0402
                                                5E        08  C2  00002         SUBL2   #8, SP                                   ;
                                      00000000'   00  9F  00005                 PUSHAB  P.AAJ                                    ; 0430
                                                7E    04  AC  7D  00008         MOVQ    BUF_ADDRESS, -(SP)                       ; 0429
                                                      7E  D4  0000F             CLRL    -(SP)                                    ; 0428
                                00000000G     00        04  FB  00011         CALLS   #4, MOM$DEBUG_MSG                        ;
                                                      7E  7C  00018             CLRQ    -(SP)                                    ; 0438
                                                      7E  7C  0001A             CLRQ    -(SP)                                    ;
                                                7E    04  AC  7D  0001C         MOVQ    BUF_ADDRESS, -(SP)                       ;
                                                      7E  7C  00020             CLRQ    -(SP)                                    ;
                                                      20  AE  9F  00022         PUSHAB  IOSB                                     ;
                                                7E    70  8F  9A  00025         MOVZBL  #112, -(SP)                              ;
                                                7E  00000000'   00  3C  00029   MOVZWL  MOM$W_MBX_CHAN, -(SP)                    ;
                                                      7E  D4  0003(             CLRL    -(SP)                                    ;
                                00000000G     00        0C  FB  00032         CALLS   #12, SYS$QIOW                            ;
                                                06        50  E9  00039         BLBC    STATUS, 1$                               ; 0443
                                                50        6E  3C  0003C         MOVZWL  IOSB, STATUS                             ; 0444
                                                09        50  E8  0003F         BLBS    STATUS, 2$                               ; 0450
                                                          50  DD  00042 1$:     PUSHL   STATUS                                   ; 0451
                                00000000G     00        01  FB  00044         CALLS   #1, LIB$STOP                             ;
                                                50        01  D0  00048 2$:     MOVL    #1, R0                                   ; 0454
                                                          04  0004E             RET                                              ; 0456

; Routine Size:  79 bytes,    Routine Base:  $CODE$ + 01EB

;  464          0457  1
;  465          0458  1 END
;  466          0459  1
;  467          0460  0 ELUDOM


                                                                                .EXTRN  LIB$STOP

;                                   PSECT SUMMARY
;
;       Name                            Bytes                           Attributes
;
;  $OWN$                                   2  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                                570  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                                136  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
```

```
;                                Library Statistics
;
;                                        --------- Symbols ---------      Pages       Processing
;       File                             Total     Loaded    Percent      Mapped      Time
;
;   _$255$DUA28:[MOM.OBJ]MOMLIB.L32;1      194         26         13          21        00:00.1
;   _$255$DUA28:[SHRLIB]NMALIBRY.L32;1     887          5          0          47        00:00.3
;   _$255$DUA28:[SHRLIB]NET.L32;1         1279          0          0          63        00:00.3
;   _$255$DUA28:[SYSLIB]STARLET.L32;1     9776         10          0         581        00:03.3
;
;
;                                    COMMAND QUALIFIERS
;
;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:MOMMAIN/OBJ=OBJ$:MOMMAIN MSRC$:MOMMAIN/UPDATE=(ENH$:MOMMAIN)
;
; Size:          570 code + 138 data bytes
; Run Time:          00:15.1
; Elapsed Time:      00:33.4
; Lines/CPU Min:      1829
; Lexemes/CPU-Min: 14457
; Memory Used:  99 pages
; Compilation Complete
```

MOMMSG
LIS

MOMMAIN
LIS

MOMPARSE
LIS

MOMSERSTA
LIS

MOMMOPSTA
LIS

MOMSERVIC
LIS

MOMMOPLIO
LIS

MOMTESSTA
LIS

MOMRSXDEF
LIS

MOMSUBS
LIS