D 14

```
MMM         MMM     AAAAAAAAA        IIIIIIIII    LLL
MMM         MMM     AAAAAAAAA        IIIIIIIII    LLL
MMM         MMM     AAAAAAAAA        IIIIIIIII    LLL
MMMMMM   MMMMMM  AAA         AAA        III       LLL
MMMMMM   MMMMMM  AAA         AAA        III       LLL
MMMMMM   MMMMMM  AAA         AAA        III       LLL
MMM   MMM   MMM  AAA         AAA        III       LLL
MMM   MMM   MMM  AAA         AAA        III       LLL
MMM   MMM   MMM  AAA         AAA        III       LLL
MMM         MMM  AAA         AAA        III       LLL
MMM         MMM  AAA         AAA        III       LLL
MMM         MMM  AAA         AAA        III       LLL
MMM         MMM  AAAAAAAAAAAAAAA        III       LLL
MMM         MMM  AAAAAAAAAAAAAAA        III       LLL
MMM         MMM  AAAAAAAAAAAAAAA        III       LLL
MMM         MMM  AAA         AAA        III       LLL
MMM         MMM  AAA         AAA        III       LLL
MMM         MMM  AAA         AAA        III       LLL
MMM         MMM  AAA         AAA     IIIIIIIII    LLLLLLLLLLLLLLL
MMM         MMM  AAA         AAA     IIIIIIIII    LLLLLLLLLLLLLLL
MMM         MMM  AAA         AAA     IIIIIIIII    LLLLLLLLLLLLLLL
```

N 11

```
**FILE**ID**NETSUBS




NN      NN  EEEEEEEEEE  TTTTTTTTTT    SSSSSSSS  UU      UU  BBPBBBBB      SSSSSSSS
NN      NN  EEEEEEEEEE  TTTTTTTTTT    SSSSSSSS  UU      UU  BBBBBBBB      SSSSSSSS
NN      NN  EE              TT      SS          UU      UU  BB      BB  SS
NN      NN  EE              TT      SS          UU      UU  BB      BB  SS
NNNN    NN  EE              TT      SS          UU      UU  BB      BB  SS
NNNN    NN  EE              TT      SS          UU      UU  BB      BB  SS
NN  NN  NN  EEEEEEEE        TT        SSSSSS    UU      UU  BBBBBBBB      SSSSSS
NN  NN  NN  EEEEEEEE        TT        SSSSSS    UU      UU  BBBBBBBB      SSSSSS
NN    NNNN  EE              TT              SS  UU      UU  BB      BB          SS
NN    NNNN  EE              TT              SS  UU      UU  BB      BB          SS
NN      NN  EE              TT              SS  UU      UU  BB      BB          SS      ....
NN      NN  EE              TT              SS  UU      UU  BB      BB          SS      ....
NN      NN  EEEEEEEEEE      TT      SSSSSSSS    UUUUUUUUUU  BBBBBBBB    SSSSSSSS        ....
NN      NN  EEEEEEEEEE      TT      SSSSSSSS    UUUUUUUUUU  BBBBBBBB    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
```

MA
VO

;

```
0 MODULE MAIL$NETSUBS (
0               IDENT = 'V04-000'
0               ) =
1 BEGIN
1
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1 !
1
1 !++
1 ! FACILITY:      VAX/VMS MAIL UTILITY
1 !
1 ! ABSTRACT:      Subroutines to speak to networks
1 !
1 ! ENVIRONMENT:  NATIVE/USER MODE
1 !
1 ! AUTHOR: Benn Schreiber, CREATION DATE: 10-Jul-1983
1 !
1 ! MODIFIED BY:
1 !
1 !
1 !       V03-015 ROP0030         Robert Posniak          24-JUL-1984
1 !               Allow VFC format files to be sent in
1 !               block mode.
1 !
1 !       V03-014 ROP0012         Robert Posniak          27-JUN-1984
1 !               Only send in block mode if input file has
1 !               variable length records.  Add check of
1 !               nodename for foreign protocol address
1 !               already exists test.
1 !
1 !       V03-013 ROP0001          Robert Posniak          24-MAY-1984
1 !               Check for oversized record when sending in record
1 !               mode.
1 !
1 !       V03-012 BLS0311         Benn Schreiber          1-MAY-1984
```

```
1 !                     Don't send 0-address to $qio.
1 !
1 !     V03-011 BLS0292         Benn Schreiber          29-MAR-1984
1 !                     Correct handling of alternate protocol per problems
1 !                     reported by Peter Lipman.  Complete attachment for MR.
1 !
1 !     V03-010 BLS0280         Benn Schreiber           4-MAR-1984
1 !                     Report errors in mail$get_input better.
1 !
1 !     V03-009 BLS0272         Benn Schreiber          18-FEB-1984 13:36:59
1 !                     Complete alternate protocol hooks. Use LIB$FIND_IMAGE_SYMBOL
1 !
1 !     V03-008 BLS0263         Benn Schreiber           4-FEB-1984
1 !                     Separate sending 0-end-of-username out into a routine
1 !                     so that slave mails that timeout on usernames don't timeout.
1 !
1 !     V03-007 BLS0255         Benn Schreiber          28-Dec-1983
1 !                     Convert to global flags.  Add routine to check addressee
1 !                     already in list.  If createlink is called for node already
1 !                     known dead, resignal the error for network master.  Insist
1 !                     on getting an ncb back in the mailbox.  mail$get_input now
1 !                     supports optional 3rd arg for output length.
1 !
1 !     V03-006 BLS0250         Benn Schreiber          12-Dec-1983
1 !                     Clear block mode flag in accept_link if error.
1 !
1 !     V03-005 BLS0246         Benn Schreiber          28-Nov-1983
1 !                     Allow ^C out of qio to access remote node.
1 !
1 !     V03-004 BLS0241         Benn Schreiber          27-Sep-1983
1 !                     Fix maxmsg and bufquo args to ASN_WTH_MBX.
1 !
1 !     V03-003 BLS0240         Benn Schreiber          15-Sep-1983
1 !                     Corrections to enable alternate net protocol.
1 !
1 !     V03-002 BLS0235         Benn Schreiber          23-Aug-1983
1 !                     Fix loop problem while searching for existing link, and
1 !                     ensure UBF set up correctly for sending messages.
1 !
1 !--
```

```
;    99          0098  1 !
;   100          0099  1 ! INCLUDE FILES
;   101          0100  1 !
;   102          0101  1     LIBRARY     'SYS$LIBRARY:STARLET';
;   103          0102  1     REQUIRE     'SRC$:MAILREQ';
;   104          0248  1     LIBRARY     'LIB$:MAILDEF';
;   105          0249  1
;   106          0250  1 EXTERNAL ROUTINE
;   107          0251  1     LIB$ASN_WTH_MBX,                      !Assign channel with mailbox
;   108          0252  1     LIB$GET_VM,                           !Allocate dynamic memory
;   109          0253  1     LIB$PUT_OUTPUT,                       !Output to SYS$OUTPUT
;   110          0254  1     LIB$SCOPY_R_DX,                       !String copy
;   111          0255  1     MAIL$ENABLE_CTRLC,                    !Enable main ctrl/c handling
;   112          0256  1     MAIL$DISABLE_CTRLC,                   ! and disable it
;   113          0257  1     MAIL$READ_ERROR_TEXT,                 !Read error text from slave and signal
;   114          0258  1     SMG$READ_COMPOSED_LINE,               !SMG input routine
;   115          0259  1     SYS$FAOL,                             !Formatted ascii
;   116          0260  1     LIB$FIND_IMAGE_SYMBOL,                !Image activate and return address
;   117          0261  1     UTIL$REPORT_IO_ERROR;                 !Report io error
;   118          0262  1
;   119          0263  1 EXTERNAL
;   120          0264  1     MAIL$SD_LNM_FILE_DEV,                 ! 'LNM$FILE_DEV'
;   121          0265  1     MAIL$G_CNCT : $BBLOCK,                !Static cnct for inbound connects
;   122          0266  1     MAIL$Q_ATTDESC : $BBLOCK,             !Descriptor of attachment file spec
;   123          0267  1     MAIL$Q_INPTRAN : $BBLOCK,             !Descriptor of SYS$NET translation
;   124          0268  1     MAIL$Q_PROTOCOL : $BBLOCK,            !Descriptor of protocol if alt input
;   125          0269  1     MAIL$L_SMG_KEYTABLE,                  !SMG keytable index
;   126          0270  1     MAIL$L_SMG_KEYBOARD,                  !SMG keyboard index
;   127          0271  1     MAIL$W_TTCHAN : WORD,                 !Channel for terminal i/o
;   128          0272  1     MAIL$GL_SYSFLAGS : $BBLOCK,           !System-wide control flags
;   129          0273  1     MAIL$GL_FLAGS : $BBLOCK;              !control flags
;   130          0274  1
;   131          0275  1 EXTERNAL LITERAL
;   132          0276  1     SMG$_EOF;                             !End of file from SMG$ routines
;   133          0277  1
;   134          0278  1 OWN
;   135          0279  1     LINK_CHAN,                            !Channel for inbound logical link
;   136          0280  1     LINK_TFRADR,                          !Transfer address for alt prot. inb.
;   137          0281  1     LINK_CONTEXT,                         ! and it's context
;   138          0282  1     NETMBX_CHAN;                          !Network mailbox channel
;   139          0283  1
;   140          0284  1 GLOBAL
;   141          0285  1     MAIL$L_MBXBUF : LONG INITIAL(32),     !Size of mailbox buffer
;   142          0286  1     MAIL$L_MBXQUO : LONG INITIAL(96);     !Mailbox quota (3*mbxbuf)
;   143          0287  1
;   144          0288  1 BIND
;   145          0289  1     PROT_DESC = $DESCRIPTOR('MAIL$PROTOCOL') : $BBLOCK,          !..routine name
;   146          0290  1     X25_DESC = $DESCRIPTOR('PSIMAIL') : $BBLOCK,                 !X25 image
;   147          0291  1     NETACP_DESC = $DESCRIPTOR('_NET:') : $BBLOCK,         !For speaking to netacp
;   148          0292  1     LINK_DESC = $DESCRIPTOR('SYS$NET') : $BBLOCK,         !Logical we look for
;   149          0293  1     OBJECT_DESC = $DESCRIPTOR('::"MAIL=') : $BBLOCK,      !Remote mail object
;   150          0294  1     PREFIX_DESC = $DESCRIPTOR('MAIL$PROTOCOL_') : $BBLOCK,
;   151          0295  1     SD_MAJOR = $DESCRIPTOR('MAIL$C_PROT_MAJOR'),
;   152          0296  1     SD_MINOR = $DESCRIPTOR('MAIL$C_PROT_MINOR');
;   153          0297  1
;   154          0298  1 GLOBAL BIND
;   155          0299  1     MAIL$Q_OBJDESC = OBJECT_DESC;         !For debugging private object type
;   156          0300  1 !
;   157          0301  1 ! Define shared messages
;   158          0302  1 !
;   159        P 0303  1 $SHR_MSGDEF(MAIL,126,LOCAL,
;   160          0304  1         (READERR,ERROR));
```

```
1 GLOBAL ROUTINE MAIL$ADDR_EXISTS(PROT_DESC,NODE_DESC,USER_DESC,ADRLST) =
1 !+++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       Check whether the named addressee is already in the list.
1 !       Return true if found, false if not.
1 !
1 ! Inputs:
1 !
1 !       prot_desc = address of protocol descriptor
1 !       node_desc = address of nodename descriptor
1 !       user_desc = address of username descriptor
1 !       adrlst = address of address list listhead
1 !---
2 BEGIN
2 MAP
2     PROT_DESC : REF $BBLOCK,
2     NODE_DESC : REF $BBLOCK,
2     USER_DESC : REF $BBLOCK,
2     ADRLST : REF VECTOR[2,LONG];
2
2 LOCAL
2     DESC : VECTOR[2,LONG],
2     ADR : REF $BBLOCK,
2     LNK : REF $BBLOCK;
2
2 ADR = .ADRLST[0];
2 !
2 ! Loop through the addressee list
2 !
2 WHILE .ADR NEQ ADRLST[0]
3 DO BEGIN
3 !
3 ! First check the username
3 !
3     IF CH$EQL(.USER_DESC[DSC$W_LENGTH],.USER_DESC[DSC$A_POINTER],
3               .ADR[ADR_B_NAM[NG],ADR[ADR_T_NAME])
4     THEN BEGIN
4         LNK = .ADR[ADR_L_LLNK];
4 !
4 ! If protocol and node are 0, and this entry has no LNK pointer, then
4 ! this is a match
4 !
5         IF (.PROT_DESC[DSC$W_LENGTH] EQL 0)
5             AND (.NODE_DESC[DSC$W_LENGTH] EQL 0)
5             AND (.LNK EQL 0)
4             THEN RETURN TRUE;
4         IF .PROT_DESC[DSC$W_LENGTH] EQL 0
5         THEN BEGIN
5 !
5 ! Same nodename is a match
5 !
6             IF (.LNK NEQ 0)
6                 AND (.LNK[LNK_B_PNLEN] EQL 0)
5                 THEN IF CH$EQ[(.NODE_DESC[DSC$W_LENGTH],.NODE_DESC[DSC$A_POINTER],
5                         .LNK[LNK_B_NODLEN],LNK[[NK_T_NODE])
5                 THEN RETURN TRUE;
```

```
;  219            0362  5                 END
;  220            0363  5                 !
;  221            0364  5                 ! If foreign protocol, check protocol name and node name
;  222            0365  5                 !
;  223            0366  4               ELSE IF (.LNK NEQ 0) AND (.LNK[LNK_B_PNLEN] NEQ 0) THEN
;  224            0367  4                       IF CH$EQL(.PROT_DESC[DSC$W_LENGTH],
;  225            0368  4                           .PROT_DESC[DSC$A_POINTER],.LNK[LNK_B_PNLEN],LNK[LNK_T_PNAM])
;  226            0369  4                     AND CH$EQL(.NODE_DESC[DSC$W_LENGTH],.NODE_DESC[DSC$A_POINTER],
;  227            0370  4                             .LNK[LNK_B_NODLEN],LNK[LNK_T_NODE])
;  228            0371  4                                                             THEN RETURN TRUE;
;  229            0372  3               END;
;  230            0373  3           ADR = .ADR[ADR_L_FLINK];                        !Move along the list
;  231            0374  2           END;
;  232            0375  2 RETURN FALSE
;  233            0376  1 END;


                                                                        .TITLE  MAIL$NETSUBS
                                                                        .IDENT  \V04-000\

                                                                        .PSECT  $CODE$,NOWRT,2

     4C  4F  43  4F  54  4F  52  50  24  4C  49  41  4D  00000 P.AAB:   .ASCII  \MAIL$PROTOCOL\                                ;
                                                         0000D          .BLKB   3
                                               0000000D  00010 P.AAA:   .LONG   13                                             ;
                                               00000000' 00014          .ADDRESS P.AAB                                         ;
                             4C  49  41  4D  49  53  50  00018 P.AAD:   .ASCII  \PSIMAIL\                                      ;
                                                         0001F          .BLKB   1
                                               00000007  00020 P.AAC:   .LONG   7                                              ;
                                               00000000' 00024          .ADDRESS P.AAD                                         ;
                                     3A  54  45  4E  5F  00028 P.AAF:   .ASCII  \_NET:\                                        ;
                                                         0002D          .BLKB   3
                                               00000005  00030 P.AAE:   .LONG   5                                              ;
                                               00000000' 00034          .ADDRESS P.AAF                                         ;
                             54  45  4E  24  53  59  53  00038 P.AAH:   .ASCII  \SYS$NET\                                      ;
                                                         0003F          .BLKB   1
                                               00000007  00040 P.AAG:   .LONG   7                                              ;
                                               00000000' 00044          .ADDRESS P.AAH                                         ;
                         3D  4C  49  41  4D  22  3A  3A  00048 P.AAJ:   .ASCII  \::"MAIL=\                                     ;
                                               00000008  00050 P.AAI:   .LONG   8                                              ;
                                               00000000' 00054          .ADDRESS P.AAJ                                         ;
 5F  4C  4F  43  4F  54  4F  52  50  24  4C  49  41  4D  00058 P.AAL:   .ASCII  \MAIL$PROTOCOL_\                               ;
                                                         00066          .BLKB   2
                                               0000000E  00068 P.AAK:   .LONG   14                                             ;
                                               00000000' 0006C          .ADDRESS P.AAL                                         ;
4A  41  4D  5F  54  4F  52  50  5F  43  24  4C  49  41  4D  00070 P.AAN:   .ASCII  \MAIL$C_PROT_MAJOR\                            ;
                                                     52  4F  0007F                                                              ;
                                                         00081          .BLKB   3
                                               00000011  00084 P.AAM:   .LONG   17                                             ;
                                               00000000' 00088          .ADDRESS P.AAN                                         ;
4E  49  4D  5F  54  4F  52  50  5F  43  24  4C  49  41  4D  0008C P.AAP:   .ASCII  \MAIL$C_PROT_MINOR\                            ;
                                                     52  4F  0009B                                                              ;
                                                         0009D          .BLKB   3
                                               00000011  000A0 P.AAO:   .LONG   17                                             ;
                                               00000000' 000A4          .ADDRESS P.AAP                                         ;

                                                                        .PSECT  $OWN$,NOEXE,2
```

```
                                                              00000 LINK_CHAN:
                                                                            .BLKB   4
                                                              00004 LINK_TFRADR:
                                                                            .BLKB   4
                                                              00008 LINK_CONTEXT:
                                                                            .BLKB   4
                                                              0000C NETMBX_CHAN:
                                                                            .BLKB   4

                                                                            .PSECT  $GLOBAL$,NOEXE,2

                                                    00000020  00000 MAIL$L_MBXBUF::
                                                                            .LONG   32                                          ;
                                                    00000060  00004 MAIL$L_MBXQUO::
                                                                            .LONG   96                                          ;

                                                                    PROT_DESC=          P.AAA
                                                                    X25_DESC=           P.AAC
                                                                    NETACP_DESC=        P.AAE
                                                                    LINK_DESC=          P.AAG
                                                                    OBJECT_DESC=        P.AAI
                                                                    PREFIX_DESC=        P.AAK
                                                                    SD_MAJOR=           P.AAM
                                                                    SD_MINOR=           P.AAO
                                                                    MAIL$Q_OBJDESC==     P.AAI
                                                                            .EXTRN  LIB$ASN_WTH_MBX
                                                                            .EXTRN  LIB$GET_VM, LIB$PUT_OUTPUT
                                                                            .EXTRN  LIB$SCOPY_R_DX, MAIL$ENABLE_CTRLC
                                                                            .EXTRN  MAIL$DISABLE_CTRLC
                                                                            .EXTRN  MAIL$READ_ERROR_TEXT
                                                                            .EXTRN  SMG$READ_COMPOSED_LINE
                                                                            .EXTRN  SYS$FAOL, LIB$FIND_IMAGE_SYMBOL
                                                                            .EXTRN  UTIL$REPORT_IO_ERROR
                                                                            .EXTRN  MAIL$SD_LNM_FILE_DEV
                                                                            .EXTRN  MAIL$G_CNCT, MAIL$Q_ATTDESC
                                                                            .EXTRN  MAIL$Q_INPTRAN, MAIL$Q_PROTOCOL
                                                                            .EXTRN  MAIL$L_SMG_KEYTABLE
                                                                            .EXTRN  MAIL$L_SMG_KEYBOARD
                                                                            .EXTRN  MAIL$W_TTCHAN, MAIL$GL_SYSFLAGS
                                                                            .EXTRN  MAIL$GL_FLAGS, SMG$_EOF

                                                                            .PSECT  $CODE$,NOWRT,2

                                                        007C  00000         .ENTRY  MAIL$ADDR_EXISTS, Save R2,R3,R4,R5,R6               ; 0305
                                        5E             08  C2  00002         SUBL2   #8, SP                                              ;
                                        55        10   BC  D0  00005         MOVL    @ADRLST, ADR                                        ; 0331
                                        56        0C   AC  D0  00009         MOVL    USER_DESC, R6                                       ; 0340
                                    10  AC             55  D1  0000D 1$:     CMPL    ADR, ADRLST                                         ; 0335
                                                       66  13  00011         BEQL    7$                                                  ;
                                        50        1D   A5  9A  00013         MOVZBL  29(ADR), R0                                         ; 0341
            50            00        04  B6        CC   BC  2D  00017         CMPC5   @USER_DESC, @4(R6), #0, R0, 30(ADR)                 ;
                                                  1E   A5      0001E                                                                     ;
                                                       52  12  00020         BNEQ    6$                                                  ;
                                        54        08   A5  D0  00022         MOVL    8(ADR), LNK                                         ; 0343
                                        51        04   AC  D0  00026         MOVL    PROT_DESC, R1                                       ; 0348
                                                       50  D4  0002A         CLRL    R0                                                  ;
```

```
                                            61  B5 0002C          TSTW    (R1)                            ;
                                            0B  12 0002E          BNEQ    2$                              ;
                                            50  D6 00030          INCL    R0                              ;
                                          08  BC  B5 00032          TSTW    @NODE_DESC                      ; 0349
                                            04  12 00035          BNEQ    2$                              ;
                                            54  D5 00037          TSTL    LNK                             ; 0350
                                            35  13 00039          BEQL    5$                              ;
                                 09         50  E9 0003B 2$:      BLBC    R0, 3$                          ; 0357
                                            54  D5 0003E          TSTL    LNK                             ;
                                            32  13 00040          BEQL    6$                              ;
                                        4F  A4  95 00042          TSTB    79(LNK)                         ; 0358
                                            15  11 00045          BRB     4$                              ;
                                            54  D5 00047 3$:      TSTL    LNK                             ; 0366
                                            29  13 00049          BEQL    6$                              ;
                                        4F  A4  95 0004B          TSTB    79(LNK)                         ;
                                            24  13 0004E          BEQL    6$                              ;
                                 50     4F  A4  9A 00050          MOVZBL  79(LNK), R0                     ; 0368
         50         00      04   B1         61  2D 00054          CMPC5   (R1), @4(R1), #0, R0, 80(LNK)   ;
                                        50  A4     0005A                                                  ;
                                            16  12 0005C 4$:      BNEQ    6$                              ;
                                 50     08  AC  D0 0005E          MOVL    NODE_DESC, R0                   ; 0369
                                 51     2F  A4  9A 00062          MOVZBL  47(LNK), R1                     ; 0370
         51         00      04   B0         60  2D 00066          CMPC5   (R0), @4(R0), #0, R1, 48(LNK)   ;
                                        30  A4     0006C                                                  ;
                                            04  12 0006E          BNEQ    6$                              ;
                                 50         01  D0 00070 5$:      MOVL    #1, R0                          ; 0371
                                                04 00073          RET                                     ;
                                 55         65  D0 00074 6$:      MOVL    (ADR), ADR                      ; 0373
                                            94  11 00077          BRB     1$                              ; 0335
                                            50  D4 00079 7$:      CLRL    R0                              ; 0375
                                                04 0007B          RET                                     ; 0376

; Routine Size:  124 bytes,    Routine Base:  $CODE$ + 00A8
;  235          0377  1 ROUTINE CTRLCAST (LNKDESC) =
;  236          0378  1 !+++
;  237          0379  1 ! FUNCTIONAL DESCRIPTION:
;  238          0380  1 !
;  239          0381  1 !       Entered when a CTRL/C is detected while attempting connect to
;  240          0382  1 !       remote node.
;  241          0383  1 !---
;  242          0384  2 BEGIN
;  243          0385  2 MAP
;  244          0386  2     LNKDESC : REF $BBLOCK;
;  245          0387  2
;  246          0388  2 LOCAL
;  247          0389  2     DESC : VECTOR[2,LONG];
;  248          0390  2
;  249          0391  2 !
;  250          0392  2 ! Cancel network access qio, then fix up ctrl/c handler
;  251          0393  2 !
;  252          0394  2 $CANCEL(CHAN=.LNKDESC[LNK_W_CHAN]);
;  253          0395  2 LNKDESC[LNK_V_DEAD] = TRUE;
;  254          0396  2
;  255          0397  2 IF .MAIL$GL_FLAGS[MAIF_V_ITERM]
;  256          0398  3 THEN BEGIN
;  257          0399  3     $CANCEL(CHAN=.MAIL$W_TTCHAN);          !Cancel our ctrl/c ast
;  258          0400  3     MAIL$ENABLE_CTRLC();                   !and enable main one
;  259          0401  2     END;
;  260          0402  2
;  261          0403  2 DESC[0] = .LNKDESC[LNK_B_NODLEN];
;  262          0404  2 DESC[1] = LNKDESC[LNK_T_NODE];
;  263          0405  2 SIGNAL(MAIL$_CONABORT,1,DESC,MAIL$_SENDABORT);  !Signal and unwind
;  264          0406  2
;  265          0407  2 RETURN 1
;  266          0408  1 END;


                                                           .EXTRN  SYS$CANCEL

                                   000C 00000 CTRLCAST:
                                                           .WORD   Save R2,R3                                 ; 0377
                  53 00000000G   00   9E 00002             MOVAB   SYS$CANCEL, R3                             ;
                  5E             08   C2 00009             SUBL2   #8, SP                                     ;
                  52        04   AC   D0 0000C             MOVL    LNKDESC, R2                                ; 0394
                  7E        2C   A2   3C 00010             MOVZWL  44(R2), -(SP)                              ;
                  63             01   FB 00014             CALLS   #1, SYS$CANCEL                             ;
             2E   A2             02   88 00017             BISB2   #2, 46(R2)                                 ; 0395
   11 00000000G   00             02   E1 0001B             BBC     #2, MAIL$GL_FLAGS, 1$                      ; 0397
                  7E 00000000G   00   3C 00023             MOVZWL  MAIL$W_TTCHAN, -(SP)                       ; 0399
                  63             01   FB 0002A             CALLS   #1, SYS$CANCEL                             ;
      00000000G   00             00   FB 0002D             CALLS   #0, MAIL$ENABLE_CTRLC                      ; 0400
                  6E        2F   A2   9A 00034   1$:       MOVZBL  47(R2), DESC                               ; 0403
         04       AE        30   A2   9E 00038             MOVAB   48(R2), DESC+4                             ; 0404
                     007E805A    8F   DD 0003D             PUSHL   #8290394                                   ; 0405
                            04   AE   9F 00043             PUSHAB  DESC                                       ;
                                 01   DD 00046             PUSHL   #1                                         ;
                     007E8112    8F   DD 00048             PUSHL   #8290578                                   ;
      00000000G   00             04   FB 0004E             CALLS   #4, LIB$SIGNAL                             ;
                  50             01   D0 00055             MOVL    #1, R0                                     ; 0407
```

```
                                      04 00058          RET                                                  ; 0408

; Routine Size:  89 bytes,     Routine Base:  $CODE$ + 0124
;  268          0409  1 ROUTINE SEND_STRING (DESC) =
;  269          0410  1 !++
;  270          0411  1 ! FUNCTIONAL DESCRIPTION:
;  271          0412  1 !
;  272          0413  1 !       Write string to SYS$OUTPUT
;  273          0414  1 !
;  274          0415  1 !--
;  275          0416  2 BEGIN
;  276          0417  2
;  277          0418  2 LIB$PUT_OUTPUT(.DESC);
;  278          0419  2 RETURN 0
;  279          0420  1 END;



                                        0000 00000 SEND_STRING:
                                                              .WORD   Save nothing                              ; 0409
                                  04   AC  DD 00002           PUSHL   DESC                                      ; 0418
                    00000000G  00      01  FB 00005           CALLS   #1, LIB$PUT_OUTPUT                        ;
                                       50  D4 0000C           CLRL    R0                                        ; 0419
                                           04 0000E           RET                                               ; 0420

; Routine Size:  15 bytes,    Routine Base:  $CODE$ + 017D
```

```
      0421  1 GLOBAL ROUTINE MAIL$PRUNW_HANDLER (SIGARG,MECHARG) =
      0422  1 !++
      0423  1 ! FUNCTIONAL DESCRIPTION:
      0424  1 !
      0425  1 !        General handler to print message w/putmsg and then unwind if
      0426  1 !        the signal is MAIL$_CONABORT
      0427  1 !--
      0428  2 BEGIN
      0429  2 MAP
      0430  2     SIGARG : REF $BBLOCK,
      0431  2     MECHARG : REF $BBLOCK;
      0432  2
      0433  2 BIND
      0434  2     SIGNAME = SIGARG[CHF$L_SIG_NAME] : $BBLOCK;
      0435  2
      0436  2 IF .SIGNAME EQL SS$_UNWIND
      0437  2     THEN RETURN SS$_CONTINUE;
      0438  2
      0439  2 IF .SIGNAME NEQ MAIL$_CONABORT
      0440  2     THEN RETURN SS$_RESIGNAL;
      0441  2
      0442  2 IF NOT .SIGNAME
      0443  3 THEN BEGIN
      0444  3     MECHARG[CHF$L_MCH_SAVR0] = .SIGNAME;
      0445  3     SIGARG[CHF$L_SIG_ARGS] = .SIGARG[CHF$L_SIG_ARGS] - 2;
    P 0446  3     $PUTMSG(MSGVEC=SIGARG[CHF$L_SIG_ARGS],
      0447  3             ACTRTN = SEND_STRING);
      0448  3     SIGARG[CHF$L_SIG_ARGS] = .SIGARG[CHF$L_SIG_ARGS] + 2;
      0449  3     SIGNAME[STS$V_SEVERITY] = STS$K_WARNING;
      0450  2     END;
      0451  2
      0452  2 SETUNWIND();
      0453  2 RETURN 0
      0454  1 END;


                                                            .EXTRN   SYS$PUTMSG

                                    0004 00000              .ENTRY   MAIL$PRUNW_HANDLER, Save R2              ; 0421
               52         04    AC    D0 00002              MOVL     SIGARG, R2                               ; 0434
   00000920    8F         04    A2    D1 0C006              CMPL     4(R2), #2336                             ; 0436
                                04    12 0000E              BNEQ     1$                                       ;
               50               01    D0 00010              MOVL     #1, R0                                   ; 0437
                                      04 00013              RET                                               ;
   007E8112    8F         04    A2    D1 00014   1$:        CMPL     4(R2), #8290578                          ; 0439
                                06    13 0001C              BEQL     2$                                       ;
               50       0918    8F    3C 0001E              MOVZWL   #2328, R0                                ; 0440
                                      04 00023              RET                                               ;
               21         04    A2    E8 00024   2$:        BLBS     4(R2), 3$                                ; 0442
               50         08    AC    D0 00028              MOVL     MECHARG, R0                              ; 0444
         0C    A0         04    A2    D0 0002C              MOVL     4(R2), 12(R0)                            ;
               62               02    C2 00031              SUBL2    #2, (R2)                                 ; 0445
                                7E    7C 00034              CLRQ     -(SP)                                    ; 0447
                          B8    AF    9F 00036              PUSHAB   SEND_STRING                              ;
                                52    DD 00039              PUSHL    R2                                       ;
   00000000G   00               04    FB 0003B              CALLS    #4, SYS$PUTMSG                           ;
```

```
                                62            02  C0 00042          ADDL2   #2, (R2)                                    ; 0448
                           04   A2            07  8A 00045          BICB2   #7, 4(R2)                                   ; 0449
                                              7E  7C 00049 3$:      CLRQ    -(SP)                                       ; 0452
                    00000000G   00            02  FB 0004B          CALLS   #2, SYS$UNWIND                              ;
                                              50  D4 00052          CLRL    R0                                          ; 0453
                                                  04 00054          RET                                                 ; 0454

; Routine Size:  85 bytes,    Routine Base:  $CODE$ + 018C
;  316          0455   1 ROUTINE ACCESS_NODE(LNKDESC, CNCTDESC, ALTOBJ_DESC) =
;  317          0456   1 !+++
;  318          0457   1 ! FUNCTIONAL DESCRIPTION:
;  319          0458   1 !
;  320          0459   1 !       Perform the access qio.
;  321          0460   1 !
;  322          0461   1 ! INPUTS:
;  323          0462   1 !
;  324          0463   1 !       lnkdesc = address of lnk descriptor block
;  325          0464   1 !       cnctdesc = address of cnct block
;  326          0465   1 !
;  327          0466   1 !--
;  328          0467   2 BEGIN
;  329          0468   2
;  330          0469   2 MAP
;  331          0470   2     LNKDESC : REF $BBLOCK,
;  332          0471   2     CNCTDESC : REF $BBLOCK,
;  333          0472   2     ALTOBJ_DESC : REF $BBLOCK;
;  334          0473   2
;  335          0474   2 BUILTIN
;  336          0475   2     NULLPARAMETER;
;  337          0476   2
;  338          0477   2 LOCAL
;  339          0478   2     STATUS,
;  340          0479   2     DESC : VECTOR[2,LONG],
;  341          0480   2     CNFREC : $BBLOCK[CNF_C_LENGTH],
;  342          0481   2     PTR : REF VECTOR[,BYTE],
;  343          0482   2     PTR1 : REF VECTOR[,BYTE],
;  344          0483   2     IOSB : VECTOR[4,WORD];
;  345          0484   2
;  346          0485   2 BIND
;  347          0486   2     TMPBUF = MAIL$G_CNCT[CNCT_T_BUFFER] : $BBLOCK,
;  348          0487   2     TMPWORD = TMPBUF : VECTOR[,WORD],
;  349          0488   2     TMPBYTE = TMPBUF : VECTOR[,BYTE];
;  350          0489   2
;  351          0490   2 IF .MAIL$GL_FLAGS[MAIF_V_ITERM]
;  352          0491   2     AND .MAIL$W_TTCHAN NEQ 0
;  353          0492   3 THEN BEGIN
;  354          0493   3     MAIL$DISABLE_CTRLC();
;  355        P 0494   3     IF_ERR($QIOW(CHAN=.MAIL$W_TTCHAN,
;  356        P 0495   3                  FUNC=IO$_SETMODE OR IO$M_CTRLCAST,
;  357        P 0496   3                  IOSB=IOSB,
;  358        P 0497   3                  P1=CTRLCAST,
;  359        P 0498   3                  P2=.LNKDESC),
;  360          0499   3             SIGNAL(.STATUS));
;  361          0500   3     IF NOT .IOSB[0]
;  362          0501   3         THEN SIGNAL(.IOSB[0]);
;  363          0502   2     END;
;  364          0503   2 !
;  365          0504   2 ! Set up configuration record
;  366          0505   2 !
;  367          0506   2 CNFREC[CNF_B_VERSION] = CNF_C_VERS;
;  368          0507   2 CNFREC[CNF_B_ECO] = CNF_C_ECO;
;  369          0508   2 CNFREC[CNF_B_CUSTECO] = 0;
;  370          0509   2 CNFREC[CNF_B_OS] = CNF_C_VAXVMS;
;  371          0510   2 CNFREC[CNF_L_OPTIONS] = 0;
;  372          0511   2 CNFREC[CNF_B_RFM] = .CNCTDESC[CNCT_B_FILRFM]; !Record format
;  373          0512  2 CNFREC[CNF_B_RAT] = .CNCTDESC[CNCT_B_FILRAT]; ! and attributes
;  374          0513  2 !
;  375          0514  2 ! We want to send in block mode only if the input file has var len records
;  376          0515  2 ! or VFC format
;  377          0516  2 !
;  378          0517  2 CNFREC[CNF_L_IOMODE] = 0;
;  379          0518  3 IF ((.CNFREC[CNF_B_RFM] EQL FAB$C_VAR) OR (.CNFREC[CNF_B_RFM] EQL FAB$C_VFC))
;  380          0519  2     THEN CNFREC[CNF_L_IOMODE] = CNF_M_BLKSEND;
;  381          0520  2 CNFREC[CNF_B_SPARE1] = 0;
;  382          0521  2 CNFREC[CNF_B_SPARE2] = 0;
;  383          0522  2 !
;  384          0523  2 ! Set up the ncb.  the format is:
;  385          0524  2 !       NODE::"MAIL=/<word of 0><count><'count' bytes><16 - 'count' 0's>"
;  386          0525  2 !
;  387          0526  2 PTR = CH$MOVE(.LNKDESC[LNK_B_NODLEN],LNKDESC[LNK_T_NODE],TMPBUF);
;  388          0527  2 IF NULLPARAMETER(3)
;  389          0528  2 THEN PTR = CH$MOVE(.OBJECT_DESC[DSC$W_LENGTH],.OBJECT_DESC[DSC$A_POINTER],.PTR)
;  390          0529  2 ELSE PTR = CH$MOVE(.ALTOBJ_DESC[DSC$W_LENGTH],
;  391          0530  2                     .ALTOBJ_DESC[DSC$A_POINTER],.PTR);
;  392          0531  2 PTR[0] = %C'/';
;  393          0532  2 PTR = PTR[1];
;  394          0533  2 PTR[0] = PTR[1] = 0;     !Create word of 0
;  395          0534  2 PTR = PTR[2];
;  396          0535  2 PTR[0] = CNF_C_LENGTH;  !Set length of configuration data
;  397          0536  2 PTR = PTR[1];
;  398          0537  2 PTR = CH$MOVE(CNF_C_LENGTH,CNFREC,.PTR); !move configuration data
;  399        L 0538  2 %IF 16-CNF_C_LENGTH GTRU 0
;  400        U 0539  2 %THEN
;  401        U 0540  2     PTR = CH$FILL(0,16-CNF_C_LENGTH,.PTR);              !Fill rest with 0s'
;  402          0541  2 %FI
;  403          0542  2 PTR[0] = %C'"';                                     !And a closing quotes
;  404          0543  2 PTR = PTR[1];
;  405          0544  2 DESC[0] = .PTR - TMPBUF;                            !Create descriptor of NCB
;  406          0545  2 DESC[1] = TMPBUF;
;  407          0546  2 !
;  408          0547  2 ! Do Access qio
;  409          0548  2 !
;  410        P 0549  2 STATUS = $QIOW(FUNC=IO$_ACCESS,
;  411        P 0550  2                CHAN=.LNKDESC[LNK_W_CHAN],
;  412        P 0551  2                IOSB=IOSB,
;  413          0552  2                P2=DESC);
;  414          0553  2
;  415          0554  2 IF .STATUS
;  416          0555  2     THEN STATUS = .IOSB[0];
;  417          0556  2
;  418          0557  2 IF .MAIL$GL_FLAGS[MAIF_V_ITERM]
;  419          0558  2     AND .MAIL$W_TTCHAN NEQ 0
;  420          0559  3 THEN BEGIN
;  421          0560  3     $CANCEL(CHAN=.MAIL$W_TTCHAN);       !Cancel our ctrl/c ast
;  422          0561  3     MAIL$ENABLE_CTRLC();                !and enable main one
;  423          0562  2     END;
;  424          0563  2
;  425          0564  2 RETURN .STATUS
;  426          0565  1 END;
```

```
                                                      .EXTRN  SYS$QIOW

                              0FFC 00000 ACCESS_NODE:
                                                      .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11                  ; 0455
             5B 00000000G  00   9E 00002              MOVAB   LIB$SIGNAL, R11                                       ;
             5A 00000000G  00   9E 00009              MOVAB   SYS$QIOW, R10                                         ;
             59 00000000G  00   9E 00010              MOVAB   TMPBUF, R9                                            ;
             58      FF28  CF   9E 00017              MOVAB   CTRLCAST, R8                                          ;
             57 00000000G  00   9E 0001C              MOVAB   MAIL$W_TTCHAN, R7                                     ;
             5E            20   C2 00023              SUBL2   #32, SP                                               ;
 37 00000000G  00          02   E1 00026              BBC     #2, MAIL$GL_FLAGS, 2$                                 ; 0490
                           67   B5 0002E              TSTW    MAIL$W_TTCHAN                                         ; 0491
                           33   13 00030              BEQL    2$                                                    ;
    00000000G  00          00   FB 00032              CALLS   #0, MAIL$DISABLE_CTRLC                                ; 0493
                           7E   7C 00039              CLRQ    -(SP)                                                 ; 0499
                           7E   7C 0003B              CLRQ    -(SP)                                                 ;
                     04    AC   DD 0003D              PUSHL   LNKDESC                                               ;
                           58   DD 00040              PUSHL   R8                                                    ;
                           7E   7C 00042              CLRQ    -(SP)                                                 ;
                     20    AE   9F 00044              PUSHAB  IOSB                                                  ;
             7E    0123    8F   3C 00047              MOVZWL  #291, -(SP)                                           ;
             7E            67   3C 0004C              MOVZWL  MAIL$W_TTCHAN, -(SP)                                  ;
                           7E   D4 0004F              CLRL    -(SP)                                                 ;
             6A            0C   FB 00051              CALLS   #12, SYS$QIOW                                         ;
             05            50   E8 00054              BLBS    STATUS, 1$                                            ;
                           50   DD 00057              PUSHL   STATUS                                                ;
             6B            01   FB 00059              CALLS   #1, LIB$SIGNAL                                        ;
             06            6E   E8 0005C 1$:          BLBS    IOSB, 2$                                              ; 0500
             7E            6E   3C 0005F              MOVZWL  IOSB, -(SP)                                           ; 0501
             6B            01   FB 00062              CALLS   #1, LIB$SIGNAL                                        ;
        08   AE 07000003   8F   D0 00065 2$:          MOVL    #117440515, CNFREC                                    ; 0506
             50            08   AC D0 0006D           MOVL    CNCTDESC, R0                                          ; 0511
        14   AE      0081  C0   90 00071              MOVB    129(R0), CNFREC+12                                    ;
        15   AE      0080  C0   90 00077              MOVB    128(R0), CNFREC+13                                    ; 0512
                       0C  AE   7C 0007D              CLRQ    CNFREC+4                                              ; 0510
        02             14  AE   91 00080              CMPB    CNFREC+12, #2                                         ; 0518
                           06   13 00084              BEQL    3$                                                    ;
        03             14  AE   91 00086              CMPB    CNFREC+12, #3                                         ;
                           04   12 0008A              BNEQ    4$                                                    ;
        10   AE            01   D0 0008C 3$:          MOVL    #1, CNFREC+8                                          ; 0519
                       16  AE   B4 00090 4$:          CLRW    CNFREC+14                                             ; 0520
             56        04  AC   D0 00093              MOVL    LNKDESC, R6                                           ; 0526
             50        2F  A6   9A 00097              MOVZBL  47(R6), R0                                            ;
 69     30   A6            50   28 0009B              MOVC3   R0, 48(R6), TMPBUF                                    ;
             03            6C   91 000A0              CMPB    (AP), #3                                              ; 0527
                           05   1F 000A3              BLSSU   5$                                                    ;
                       0C  AC   D5 000A5              TSTL    12(AP)                                                ;
                           0D   12 000A8              BNEQ    6$                                                    ;
             50      FDC5  CF   D0 000AA 5$:          MOVL    OBJECT_DESC+4, R0                                     ; 0528
 63          60      FDBC  CF   28 000AF              MOVC3   OBJECT_DESC, (R0), (PTR)                              ;
                           09   11 000B5              BRB     7$                                                    ;
             50        0C  AC   D0 000B7 6$:          MOVL    ALTOBJ_DESC, R0                                       ; 0529
 63     04   B0            60   28 000BB              MOVC3   (R0), @4(R0), (PTR)                                   ; 0530
             83            2F   90 000C0 7$:          MOVB    #47, (PTR)+                                           ; 0531
                           83   B4 000C3              CLRW    (PTR)+                                                ; 0533
             83            10   90 000C5              MOVB    #16, (PTR)+                                           ; 0535
 63     08   AE            10   28 000C8              MOVC3   #16, CNFREC, (PTR)                                    ; 0537
```

```
                        83          22  90 000CD          MOVB     #34, (PTR)+                          ; 0542
                        50          69  9E 000D0          MOVAB    TMPBUF, R0                           ; 0544
      18    AE          53          50  C3 000D3          SUBL3    R0, PTR, DESC                        ;
                  1C    AE          69  9E 000D8          MOVAB    TMPBUF, DESC+4                       ; 0545
                                    7E  7C 000DC          CLRQ     -(SP)                                ; 0552
                                    7E  7C 000DE          CLRQ     -(SP)                                ;
                              28    AE  9F 000E0          PUSHAB   DESC                                 ;
                                    7E  7C 000E3          CLRQ     -(SP)                                ;
                                    7E  D4 000E5          CLRL     -(SP)                                ;
                              20    AE  9F 000E7          PUSHAB   IOSB                                 ;
                                    32  DD 000EA          PUSHL    #50                                  ;
                        7E    2C    A6  3C 000EC          MOVZWL   44(R6), -(SP)                        ;
                                    7E  D4 000F0          CLRL     -(SP)                                ;
                        6A          0C  FB 000F2          CALLS    #12, SYS$QIOW                        ;
                        52          50  D0 000F5          MOVL     R0, STATUS                           ;
                        03          52  E9 000F8          BLBC     STATUS, 8$                           ; 0554
                        52          6E  3C 000FB          MOVZWL   IOSB, STATUS                         ; 0555
         15 00000000G   00          02  E1 000FE 8$:      BBC      #2, MAIL$GL_FLAGS, 9$                ; 0557
                                    67  B5 00106          TSTW     MAIL$W_TTCHAN                        ; 0558
                                    11  13 00108          BEQL     9$                                   ;
                        7E          67  3C 0010A          MOVZWL   MAIL$W_TTCHAN, -(SP)                 ; 0560
                00000000G   00      01  FB 0010D          CALLS    #1, SYS$CANCEL                       ;
                00000000G   00      00  FB 00114          CALLS    #0, MAIL$ENABLE_CTRLC                ; 0561
                        50          52  D0 0011B 9$:      MOVL     STATUS, R0                           ; 0564
                                        04 0011E          RET                                           ; 0565

; Routine Size:  287 bytes,     Routine Base:  $CODE$ + 01E1
;  428          0566  1 ROUTINE CHECK_PROTOCOL_VERSION (IMAGE_DESC) =
;  429          0567  1 !++
;  430          0568  1 ! FUNCTIONAL DESCRIPTION:
;  431          0569  1 !
;  432          0570  1 !       Ensure that the symbols MAIL$C_PROT_MAJOR and MAIL$C_PROT_MINOR
;  433          0571  1 !       are defined, and that they have acceptable values
;  434          0572  1 !
;  435          0573  1 !--
;  436          0574  2 BEGIN
;  437          0575  2
;  438          0576  2 LOCAL
;  439          0577  2     MAJOR_P,
;  440          0578  2     MINOR_P;
;  441          0579  2
;  442        P 0580  2 IF_ERR(LIB$FIND_IMAGE_SYMBOL(.IMAGE_DESC,SD_MAJOR,MAJOR_P);,
;  443          0581  2         RETURN .STATUS);
;  444          0582  2 IF .MAJOR_P NEQ 1
;  445          0583  2     THEN RETURN SIGNAL(MAIL$_IVPROTVAL,3,
;  446          0584  2                 SD_MAJOR,.MAJOR_P,.IMAGE_DESC);
;  447          0585  2
;  448        P 0586  2 IF_ERR(LIB$FIND_IMAGE_SYMBOL(.IMAGE_DESC,SD_MINOR,MINOR_P);,
;  449          0587  2         RETURN .STATUS);
;  450          0588  2
;  451          0589  2 IF .MINOR_P NEQ 1
;  452          0590  2     THEN RETURN SIGNAL(MAIL$_IVPROTVAL,3,
;  453          0591  2                 SD_MINOR,.MINOR_P,.IMAGE_DESC);
;  454          0592  2
;  455          0593  2 RETURN TRUE
;  456          0594  1 END;


                                  000C 00000 CHECK_PROTOCOL_VERSION:
                                                  .WORD   Save R2,R3                               ; 0566
         53 00000000G  00  9E 00002              MOVAB   LIB$FIND_IMAGE_SYMBOL, R3                ;
         52       FD77  CF  9E 00009              MOVAB   SD_MAJOR, R2                             ;
         5E             08  C2 0000E              SUBL2   #8, SP                                   ;
                  4004  8F  BB 00011              PUSHR   #^M<R2,SP>                               ; 0581
                    04  AC  DD 00015              PUSHL   IMAGE_DESC                               ;
         63             03  FB 00018              CALLS   #3, LIB$FIND_IMAGE_SYMBOL                ;
         40             50  E9 0001B              BLBC    STATUS, 4$                               ;
         01             6E  D1 0001E              CMPL    MAJOR_P, #1                              ; 0582
                        0A  13 00021              BEQL    1$                                       ;
                    04  AC  DD 00023              PUSHL   IMAGE_DESC                               ; 0584
                    04  AE  DD 00026              PUSHL   MAJOR_P                                  ;
                        52  DD 00029              PUSHL   R2                                       ; 0583
                        1E  11 0002B              BRB     2$                                       ;
                    04  AE  9F 0002D 1$:          PUSHAB  MINOR_P                                  ; 0587
                    1C  A2  9F 00030              PUSHAB  SD_MINOR                                 ;
                    04  AC  DD 00033              PUSHL   IMAGE_DESC                               ;
         63             03  FB 00036              CALLS   #3, LIB$FIND_IMAGE_SYMBOL                ;
         22             50  E9 00039              BLBC    STATUS, 4$                               ;
         01         04  AE  D1 0003C              CMPL    MINOR_P, #1                              ; 0589
                        19  13 00040              BEQL    3$                                       ;
                    04  AC  DD 00042              PUSHL   IMAGE_DESC                               ; 0591
```

```
                                  08    AE  DD 00045          PUSHL    MINOR_P                                        ;
                                  1C    A2  9F 00048          PUSHAB   SD_MINOR                                       ; 0590
                                        03  DD 0004B 2$:      PUSHL    #3                                             ;
                            007E8132    8F  DD 0004D          PUSHL    #8290610                                       ;
                   00000000G  00        05  FB 00053          CALLS    #5, LIB$SIGNAL                                 ;
                                            04 0005A          RET                                                     ;
                          50            01  D0 0005B 3$:      MOVL     #1, R0                                         ; 0593
                                            04 0005E 4$:      RET                                                     ; 0594

; Routine Size:  95 bytes,     Routine Base:  $CODE$ + 0300
;  458          0595  1 ROUTINE TRY_CONNECT(LNKDESC,CNCTDESC,ALTOBJ_DESC) =
;  459          0596  1 !++
;  460          0597  1 !
;  461          0598  1 ! Try to connect with the remote node, ensuring that a connect
;  462          0599  1 ! confirm message is received.
;  463          0600  1 !
;  464          0601  1 !--
;  465          0602  2 BEGIN
;  466          0603  2 MAP
;  467          0604  2     LNKDESC : REF $BBLOCK,
;  468          0605  2     CNCTDESC : REF $BBLOCK;
;  469          0606  2
;  470          0607  2 BUILTIN
;  471          0608  2     NULLPARAMETER;
;  472          0609  2
;  473          0610  2 LOCAL
;  474          0611  2     STATUS,
;  475          0612  2     OBJPTR,
;  476          0613  2     PTR : REF VECTOR[,BYTE],
;  477          0614  2     PTR1 : REF $BBLOCK;
;  478          0615  2
;  479          0616  2 BIND
;  480          0617  2     TMPBUF = MAIL$G_CNCT[CNCT_T_BUFFER] : $BBLOCK,
;  481          0618  2     TMPWORD = TMPBUF : VECTOR[,WORD],
;  482          0619  2     TMPBYTE = TMPBUF : VECTOR[,BYTE],
;  483          0620  2     QIOSB = LNKDESC[LNK_Q_IOSB] : VECTOR[,WORD];
;  484          0621  2
;  485          0622  2 OBJPTR = 0;
;  466          0623  2 IF NOT NULLPARAMETER(3)
;  487          0624  2     THEN OBJPTR = .ALTOBJ_DESC;
;  488          0625  2
;  489          0626  2 INCRU I FROM 1 TO 5
;  490          0627  3 DO BEGIN
;  491          0628  3     !
;  492          0629  3     ! Try up to 5 times to access the remote node.  The extra times
;  493          0630  3     ! are done in the instance that the connect was made but we
;  494          0631  3     ! failed to read the mailbox.
;  495          0632  3     !
;  496          0633  4     IF NOT (STATUS = ACCESS_NODE(.LNKDESC,.CNCTDESC,.OBJPTR))
;  497          0634  3         THEN EXITLOOP;
;  498          0635  3     !
;  499          0636  3     ! Read the mailbox to get the connect confirm message
;  500          0637  3     !
;  501        P 0638  4     IF (STATUS = $QIOW(CHAN=.LNKDESC[LNK_W_MBXCHAN],
;  502        P 0639  4                 FUNC=IO$_READVBLK,
;  503        P 0640  4                 IOSB=QIOSB,
;  504        P 0641  4                 P1=TMPBUF,
;  505          0642  4                 P2=.MAIL$L_MBXBUF))
;  506          0643  4             AND (STATUS = .QIOSB[0])
;  507          0644  4             AND (.TMPWORD[0] EQL MSG$_CONFIRM)  !ensure it's a connect confirm
;  508          0645  4     THEN BEGIN
;  509          0646  4         PTR1 = TMPBYTE[4] + .TMPBYTE[4] + 2;
;  510          0647  4         PTR = .PTR1 - 1;
;  511          0648  4         !
;  512          0649  4         ! See if receiver is up to block mode transfer.  Assume 1 block
;  513          0650  4         ! transfers for now.
;  514          0651  4         !
;   515            0652  5             IF (.PTR[0] EQL CNF_C_LENGTH)
;   516            0653  4                 AND NOT .PTR1[CNF_V_BLKSEND]
;   517            0654  5                 AND (.PTR1[CNF_B_VERSION] GEQU CNF_C_VERS)
;   518            0655  5                     THEN IF (.(PTR1[CNF_B_ECO])<0,8,1> GEQ CNF_C_ECO)
;   519            0656  4                         THEN LNKDESC[LNK_V_BLKMODE] = .PTR1[CNF_V_BLKRECV];
;   520            0657  4             EXITLOOP;
;   521            0658  3             END;
;   522            0659  3         !
;   523            0660  3         ! We failed to read the connect confirm.  Issue a Deaccess and
;   524            0661  3         ! try again.
;   525            0662  3         !
;   526          P 0663  3         $QIOW(CHAN=.LNKDESC[LNK_W_CHAN],
;   527          P 0664  3               FUNC=IO$_DEACCESS,
;   528            0665  3               IOSB=QIOSB);
;   529            0666  3         STATUS = SS$_NODATA;
;   530            0667  2         END;          !loop
;   531            0668  2
;   532            0669  2 RETURN .STATUS
;   533            0670  1 END;




                                                  03FC 00000 TRY_CONNECT:
                                                                         .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9              ; 0595
                       59 00000000G     00   9E 00002                   MOVAB   SYS$QIOW, R9                              ;
                       58 00000000G     00   9E 00009                   MOVAB   TMPBUF, R8                                ;
                       54           04  AC   D0 00010                   MOVL    LNKDESC, R4                               ; 0620
                                        57   D4 00014                   CLRL    OBJPTR                                    ; 0622
                       03               6C   91 00016                   CMPB    (AP), #3                                  ; 0623
                                        09   1F 00019                   BLSSU   1$                                        ;
                                    0C  AC   D5 0001B                   TSTL    12(AP)                                    ;
                                        04   13 0001E                   BEQL    1$                                        ;
                       57           0C  AC   D0 00020                   MOVL    ALTOBJ_DESC, OBJPTR                       ; 0624
                       56               01   D0 00024 1$:               MOVL    #1, I                                     ; 0665
                                        57   DD 00027 2$:               PUSHL   OBJPTR                                    ; 0633
                                    08  AC   DD 00029                   PUSHL   CNCTDESC                                  ;
                                        54   DD 0002C                   PUSHL   R4                                        ;
                  FE4F  CF              03   FB 0002E                   CALLS   #3, ACCESS_NODE                           ;
                        55              50   D0 00033                   MOVL    R0, STATUS                                ;
                        5C              55   E9 00036                   BLBC    STATUS, 3$                                ;
                                        7E   7C 00039                   CLRQ    -(SP)                                     ; 0642
                                        7E   7C 0003B                   CLRQ    -(SP)                                     ;
                           00000000'    00   DD 0003D                   PUSHL   MAIL$L_MBXBUF                             ;
                                        58   DD 00043                   PUSHL   R8                                        ;
                                        7E   7C 00045                   CLRQ    -(SP)                                     ;
                                    14  A4   9F 00047                   PUSHAB  20(R4)                                    ;
                                        31   DD 0004A                   PUSHL   #49                                       ;
                        7E          2A  A4   3C 0004C                   MOVZWL  42(R4), -(SP)                             ;
                                        7E   D4 00050                   CLRL    -(SP)                                     ;
                        69              0C   FB 00052                   CALLS   #12, SYS$QIOW                             ;
                        55              50   D0 00055                   MOVL    R0, STATUS                                ;
                        3C              55   E9 00058                   BLBC    STATUS, 4$                                ;
                        55          14  A4   3C 0005B                   MOVZWL  20(R4), STATUS                            ; 0643
                        35              55   E9 0005F                   BLBC    STATUS, 4$                                ;
                        31              68   B1 00062                   CMPW    TMPWORD, #49                              ; 0644
```

```
                                          30  12 00065        BNEQ    4$                                   ;
                           50       06    A8  9E 00067        MOVAB   TMPBYTE+6, R0                        ; 0646
                           52       04    A8  9A 0006B        MOVZBL  TMPBYTE+4, PTR1                      ;
                           52             50  C0 0006F        ADDL2   R0, PTR1                             ;
                           53       FF    A2  9E 00072        MOVAB   -1(R2), PTR                          ; 0647
                           10             63  91 00076        CMPB    (PTR), #16                           ; 0652
                                          41  12 00079        BNEQ    5$                                   ;
                           3D       08    A2  E8 0007B        BLBS    8(PTR1), 5$                          ; 0653
                           03             62  91 0007F        CMPB    (PTR1), #3                           ; 0654
                                          38  1F 00082        BLSSU   5$                                   ;
                                    01    A2  95 00084        TSTB    1(PTR1)                              ; 0655
                                          33  19 00087        BLSS    5$                                   ;
        50       08    A2          01             01  EF 00089        EXTZV   #1, #1, 8(PTR1), R0                  ; 0656
   2E   A4             01          04             50  F0 0008F        INSV    R0, #4, #1, 46(R4)                   ;
                                          25  11 00095 3$:    BRB     5$                                   ; 0645
                                          7E  7C 00097 4$:    CLRQ    -(SP)                                ; 0665
                                          7E  7C 00099        CLRQ    -(SP)                                ;
                                          7E  7C 0009B        CLRQ    -(SP)                                ;
                                          7E  7C 0009D        CLRQ    -(SP)                                ;
                                    14    A4  9F 0009F        PUSHAB  20(R4)                               ;
                                          34  DD 000A2        PUSHL   #52                                  ;
                           7E       2C    A4  3C 000A4        MOVZWL  44(R4), -(SP)                        ;
                                          7E  D4 000A8        CLRL    -(SP)                                ;
                           69             0C  FB 000AA        CALLS   #12, SYS$QIOW                        ;
                           55     01AC    8F  3C 000AD        MOVZWL  #428, STATUS                         ; 0666
                                          56  D6 000B2        INCL    I                                    ; 0626
                           05             56  D1 000B4        CMPL    I, #5                                ;
                                          03  1A 000B7        BGTRU   5$                                   ;
                                        FF6B  31 000B9        BRW     2$                                   ;
                           50             55  D0 000BC 5$:    MOVL    STATUS, R0                           ; 0669
                                              04 000BF        RET                                          ; 0670

; Routine Size:  192 bytes,    Routine Base:  $CODE$ + 035F
```

```
1 ROUTINE CONNECT_LINK(LNKDESC,PROTOCOL_DESC,NODE_DESC,CNCTDESC) =
1 !+++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       Make an outbound connection with a remote node
1 !
1 ! INPUTS:
1 !
1 !       lnkdesc = address of lnk descriptor block
1 !       protocol_desc = address of protocol descriptor
1 !       node_desc = address of descriptor of node name
1 !       cnctdesc = address of cnct block for message
1 !
1 !--
2 BEGIN
2
2 MAP
2     LNKDESC : REF $BBLOCK,
2     PROTOCOL_DESC : REF $BBLOCK,
2     NODE_DESC : REF $BBLOCK,
2     CNCTDESC : REF $BBLOCK;
2
2 LOCAL
2     STATUS,
2     PTR : REF VECTOR[,BYTE],
2     PTR1 : REF $BBLOCK,
2     TRNLNMLST : $ITMLST_DECL(ITEMS=1),
2     DESC : VECTOR[2,LONG],
2     DESC_1 : VECTOR[2,LONG];
2
2 BIND
2     TMPBUF = MAIL$G_CNCT[CNCT_T_BUFFER] : $BBLOCK,
2     TMPWORD = TMPBUF : VECTOR[,WORD],
2     TMPBYTE = TMPBUF : VECTOR[,BYTE],
2     QIOSB = LNKDESC[LNK_Q_IOSB] : VECTOR[,WORD];
2
2 IF NOT .LNKDESC[LNK_V_ALTP]
3 THEN BEGIN
3 !
3 ! Use DECNET
3 !
3     !
3     ! Assign a channel to _NET.  Then, attempt to access the remote
3     ! node.
3     !
4     IF (STATUS = LIB$ASN_WTH_MBX(NETACP_DESC,
4                               MAIL$L_MBXBUF,MAIL$L_MBXQUO,LNKDESC[LNK_W_CHAN],
4                               LNKDESC[LNK_W_MBXCHAN]))
3     THEN STATUS = TRY_CONNECT(.LNKDESC,.CNCTDESC);
3     !
3     ! Check for control/c typed after we switched handlers.  unwind if
3     ! ctrl/c typed.
3     !
3     IF .MAIL$GL_FLAGS[MAIF_V_CTRLCFL]
4     THEN BEGIN
4         MAIL$GL_FLAGS[MAIF_V_CTRLCFL] = 0;
4         $DASSGN(CHAN=.LNKDESC[LNK_W_CHAN]);
```

```
  4             SIGNAL(MAIL$_CONABORT,1,DESC,MAIL$_SENDABORT); !Will unwind
  3             END;
  3         IF NOT .STATUS
  4         THEN BEGIN
  4             $DASSGN(CHAN=.LNKDESC[LNK_W_CHAN]);
  4             IF NOT .LNKDESC[LNK_V_DEAD]
  5                 THEN (SIGNAL(MAIL$_LOGLINK,1,.NODE_DESC,.STATUS);
  5                             LNKDESC[LNK_L_STS] = .STATUS)
  4                 ELSE RETURN MAIL$_LOGLINK;
  3             END;
  3         RETURN .STATUS
  3         END
  3     ELSE BEGIN
  3     !
  3     ! Alternate protocol.  Translate MAIL$PROTOCOL_pname
  3     ! If it translates, use that for the image name.  If it doesn't
  3     ! translate, use pname_MAILSHR
  3     !
  3         PTR = CH$MOVE(.PREFIX_DESC[DSC$W_LENGTH],
  3                         .PREFIX_DESC[DSC$A_POINTER],TMPBUF);
  3         PTR = CH$MOVE(.PROTOCOL_DESC[DSC$W_LENGTH],
  3                         .PROTOCOL_DESC[DSC$A_POINTER],.PTR);
  3
  3         DESC[0] = .PTR - TMPBUF;
  3         DESC[1] = TMPBUF;
P 3         $ITMLST_INIT(ITMLST=TRNLNMLST,
P 3           (ITMCOD=LNM$_STRING,BUFADR=.DESC[1],
  3                 BUFSIZ=NAM$C_MAXRSS,RETLEN=DESC));
  3
P 3         IF NOT $TRNLNM(ATTR=%REF(LNM$M_CASE_BLIND),
P 3           TABNAM=MAIL$SD_LNM_FILE_DEV,
P 3           LOGNAM=DESC,
  4           ITMLST=TRNLNMLST)
  4         THEN BEGIN
  4             PTR = CH$MOVE(.PROTOCOL_DESC[DSC$W_LENGTH],
  4                         .PROTOCOL_DESC[DSC$A_POINTER],TMPBUF);
  4             PTR = CH$MOVE(8,UPLIT('_MAILSHR'),.PTR);
  4             DESC[0] = .PTR - TMPBUF;
  4             END
  3         ELSE IF .TMPBYTE[0] EQL %C'%'
  4         THEN BEGIN
  4             !
  4             ! If it has a leading percent, then strip it off and attempt
  4             ! to connect to the resulting string. It should have the format
  4             ! node::"task=taskname", STAR::"TASK=MAILX" for instance.
  4             ! If successful, mail will speak mail-11 with the remote slave
  4             !
  4             DESC[0] = .DESC[0] - 1;
  4             DESC[1] = .DESC[1] + 1;
  4             DESC_1[0] = .DESC[0];
  4             DESC_1[1] = .DESC[1];
  4             IF NOT CH$FAIL(PTR = CH$FIND_CH(.DESC_1[0],.DESC_1[1],%C':'))
  4                     THEN DESC_1[0] = .PTR - .DESC_1[1];
  4             DESC_1[0] = MINU(.DESC_1[0],LNK_S_NODE);        !Descriptor of node name
  4             CH$MOVE(.DESC_1[0],.DESC_1[1],LNKDESC[LNK_T_NODE]); !Also put in lnkdesc
  4             DESC_1[1] = LNKDESC[LNK_T_NODE];
  4             LNKDESC[LNK_V_ALTP] = FALSE;
```

```
;  649          0785  4           LNKDESC[LNK_B_NODLEN] = 0;
;  650          0786  5           IF (STATUS = LIB$ASN_WTH_MBX(NETACP_DESC,
;  651          0787  5                                   MAIL$L_MBXBUF,MAIL$L_MBXQUO,LNKDESC[LNK_W_CHAN],
;  652          0788  5                                   LNKDESC[LNK_W_MBXCHAN]))
;  653          0789  4               THEN STATUS = TRY_CONNECT(.LNKDESC,.CNCTDESC,DESC);
;  654          0790  4           LNKDESC[LNK_B_NODLEN] = .DESC_1[0];
;  655          0791  4           IF NOT .STATUS
;  656          0792  5           THEN BEGIN
;  657          0793  5               $DASSGN(CHAN=.LNKDESC[LNK_W_CHAN]);
;  658          0794  5               IF NOT .LNKDESC[LNK_V_DEAD]
;  659          0795  6                   THEN (SIGNAL(MAIL$_LOGLINK,1,DESC_1,.STATUS);
;  660          0796  6                                   LNKDESC[LNK_L_STS] = .STATUS)
;  661          0797  5                   ELSE RETURN MAIL$_LOGLINK;
;  662          0798  4               END;
;  663          0799  4           RETURN .STATUS;
;  664          0800  3           END;
;  665          0801  3       DESC_1[0] = .PREFIX_DESC[DSC$W_LENGTH] - 1;
;  666          0802  3       DESC_1[1] = .PREFIX_DESC[DSC$A_POINTER];
;  667        P 0803  3       IF_ERR(LIB$FIND_IMAGE_SYMBOL(DESC,DESC_1,LNKDESC[LNK_L_TFRADR]);,
;  668          0804  3                   RETURN .STATUS);
;  669        P 0805  3       IF_ERR(CHECK_PROTOCOL_VERSION(DESC);,
;  670          0806  3                   RETURN .STATUS);
;  671          0807  3       RETURN (.LNKDESC[LNK_L_TFRADR])(LNKDESC[LNK_L_CONTEXT],
;  672          0808  3                                       LNK_C_OUT_CONNECT,
;  673          0809  3                                       .PROTOCOL_DESC,
;  674          0810  3                                       .NODE_DESC,
;  675          0811  3                                       MAIL$_LOGLINK,
;  676          0812  3                                       .CNCTDESC[CNCT_B_FILRAT],
;  677          0813  3                                       .CNCTDESC[CNCT_B_FILRFM],
;  678          0814  3                                       .MAIL$GL_SYSFLAGS<16,16,0>,
;  679          0815  4                                       (IF .MAI[$GL_FLAGS[MAIF_V_ATTACHMENT]
;  680          0816  4                                               THEN MAIL$Q_ATTDESC
;  681          0817  3                                               ELSE 0));
;  682          0818  2       END;
;  683          0819  1   END;


                                            0041F          .BLKB    1
        52  48  53  4C  49  41  4D  5F  00420 P.AAQ:   .ASCII   \_MAILSHR\                              ;

                                                           .EXTRN   SYS$DASSGN, SYS$TRNLNM

                                     0FFC 00000 CONNECT_LINK:
                                                           .WORD    Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11   ; 0671
                      5B 00000000'   00  9E 00002          MOVAB    MAIL$L_MBXQUO, R11                     ;
                      5A 00000000G   00  9E 00009          MOVAB    TMPBUF, R10                            ;
                      59      FBF4   CF  9E 00010          MOVAB    NETACP_DESC, R9                        ;
                      5E             24  C2 00015          SUBL2    #36, SP                                ;
                      56        04   AC  D0 00018          MOVL     LNKDESC, R6                            ; 0705
        79        2E  A6             02  E0 0001C          BBS      #2, 46(R6), 5$                         ; 0707
                                2A   A6  9F 00021          PUSHAB   42(R6)                                 ; 0718
                                2C   A6  9F 00024          PUSHAB   44(R6)                                 ; 0717
                                     5B  DD 00027          PUSHL    R11                                    ; 0716
                                FC   AB  9F 00029          PUSHAB   MAIL$L_MBXBUF                          ;
                                     59  DD 0002C          PUSHL    R9                                     ;
        00000000G  00                05  FB 0002E          CALLS    #5, LIB$ASN_WTH_MBX                    ; 0718
```

```
                 58             50 D0 00035      MOVL     R0, STATUS                            ;
                 0D             58 E9 00038      BLBC     STATUS, 1$                            ;
                           10   AC DD 0003B      PUSHL    CNCTDESC                              ; 0719
                                56 DD 0003E      PUSHL    R6                                    ;
            FEF2 CF             02 FB 00040      CALLS    #2, TRY_CONNECT                       ;
                 58             50 D0 00045      MOVL     R0, STATUS                            ;
                 2A 00000000G   00 E9 00048 1$:  BLBC     MAIL$GL_FLAGS+1, 2$                   ; 0724
       00000000G 00             01 8A 0004F      BICB2    #1, MAIL$GL_FLAGS+1                   ; 0726
                 7E        2C   A6 3C 00056      MOVZWL   44(R6), -(SP)                         ; 0727
       00000000G 00             01 FB 0005A      CALLS    #1, SYS$DASSGN                        ;
                     007E805A   8F DD 00061      PUSHL    #8290394                              ; 0728
                           10   AE 9F 00067      PUSHAB   DESC                                  ;
                                01 DD 0006A      PUSHL    #1                                    ;
                     007E8112   8F DD 0006C      PUSHL    #8290578                              ;
       00000000G 00             04 FB 00072      CALLS    #4, LIB$SIGNAL                        ;
                 03             58 E9 00079 2$:  BLBC     STATUS, 3$                            ; 0730
                              0132 31 0007C      BRW      14$                                   ;
                 7E        2C   A6 3C 0007F 3$:  MOVZWL   44(R6), -(SP)                         ; 0732
       00000000G 00             01 FB 00083      CALLS    #1, SYS$DASSGN                        ;
03            2E A6             01 E1 0008A      BBC      #1, 46(R6), 4$                        ; 0733
                              0117 31 0008F      BRW      13$                                   ;
                                58 DD 00092 4$:  PUSHL    STATUS                                ; 0734
                           0C   AC DD 00094      PUSHL    NODE_DESC                             ;
                              00FA 31 00097      BRW      12$                                   ;
                 50        3C   A9 D0 0009A 5$:  MOVL     PREFIX_DESC+4, R0                     ; 0747
    6A           60        38   A9 28 0009E      MOVC3    PREFIX_DESC, (R0), TMPBUF             ; 0746
                 57        08   AC D0 000A3      MOVL     PROTOCOL_DESC, R7                     ; 0748
    63        04 B7             67 28 000A7      MOVC3    (R7), @4(R7), (PTR)                   ; 0749
                 50             6A 9E 000AC      MOVAB    TMPBUF, R0                            ; 0751
0C  AE           53             50 C3 000AF      SUBL3    R0, PTR, DESC                         ;
              10 AE             6A 9E 000B4      MOVAB    TMPBUF, DESC+4                        ; 0752
                 50        14   AE 9E 000B8      MOVAB    TRNLNMLST, $$ITMBLKPTR                ; 0755
                 80  000200FF   8F D0 000BC      MOVL     #131327, ($$ITMBLKPTR)+               ;
                 80        10   AE D0 000C3      MOVL     DESC+4, ($$ITMBLKPTR)+                ;
                 80        0C   AE 9E 000C7      MOVAB    DESC, ($$ITMBLKPTR)+                  ;
                                80 D4 000CB      CLRL     ($$ITMBLKPTR)+                        ;
                           14   AE 9F 000CD      PUSHAB   TRNLNMLST                             ; 0760
                                7E D4 000D0      CLRL     -(SP)                                 ;
                           14   AE 9F 000D2      PUSHAB   DESC                                  ;
                    00000000G   00 9F 000D5      PUSHAB   MAIL$SD_LNM_FILE_DEV                  ;
              10 AE  02000000   8F D0 000DB      MOVL     #33554432, 16(SP)                     ;
                           10   AE 9F 000E3      PUSHAB   16(SP)                                ;
       00000000G 00             05 FB 000E6      CALLS    #5, SYS$TRNLNM                        ;
                 16             50 E8 000ED      BLBS     R0, 7$                                ;
    6A        04 B7             67 28 000F0      MOVC3    (R7), @4(R7), TMPBUF                  ; 0762
    63      03F0 C9             08 28 000F5      MOVC3    #8, P.AAQ, (PTR)                      ; 0764
                 50             6A 9E 000FB      MOVAB    TMPBUF, R0                            ; 0765
0C  AE           53             50 C3 000FE      SUBL3    R0, PTR, DESC                         ;
                              00AF 31 00103 6$:  BRW      15$                                   ; 0757
                 25             6A 91 00106 7$:  CMPB     TMPBYTE, #37                          ; 0767
                                F8 12 00109      BNEQ     6$                                    ;
                           0C   AE D7 0010B      DECL     DESC                                  ; 0775
                           10   AE D6 0010E      INCL     DESC+4                                ; 0776
              04 AE        0C   AE 7D 00111      MOVQ     DESC, DESC_1                          ; 0777
08  BE        04 AE             3A 3A 00116      LOCC     #58, DESC_T, @DESC_1+4                ; 0779
                                02 12 0011C      BNEQ     8$                                    ;
                                51 D4 0011E      CLRL     R1                                    ;
```

```
                        53            51  D0 00120 8$:     MOVL     R1, PTR                                 ;
                                      06  13 00123         BEQL     9$                                      ;
    04    AE            53        08  AE  C3 00125         SUBL3    DESC_1+4, PTR, DESC_1                   ; 0780
                        50        04  AE  D0 0012B 9$:     MOVL     DESC_1, R0                              ; 0781
                        1F            50  D1 0012F         CMPL     R0, #31                                 ;
                                      03  1B 00132         BLEQU    10$                                     ;
                        50            1F  D0 00134         MOVL     #31, R0                                 ;
                    04  AE            50  D0 00137 10$:    MOVL     R0, DESC_1                              ;
    30    A6        08  BE        04  AE  28 0013B         MOVC3    DESC_1, @DESC_1+4, 48(R6)               ; 0782
                    08  AE        30  A6  9E 00142         MOVAB    48(R6), DESC_1+4                        ; 0783
                    2E  A6      FF04  8F  AA 00147         BICW2    #65284, 46(R6)                          ; 0785
                                  2A  A6  9F 0014D         PUSHAB   42(R6)                                  ; 0788
                                  2C  A6  9F 00150         PUSHAB   44(R6)                                  ; 0787
                                      5B  DD 00153         PUSHL    R11                                     ; 0786
                                  FC  AB  9F 00155         PUSHAB   MAIL$L_MBXBUF                           ;
                                      59  DD 00158         PUSHL    R9                                      ;
             00000000G  00            05  FB 0015A         CALLS    #5, LIB$ASN_WTH_MBX                     ; 0788
                        58            50  D0 00161         MOVL     R0, STATUS                              ;
                        10            58  E9 00164         BLBC     STATUS, 11$                             ;
                                  0C  AE  9F 00167         PUSHAB   DESC                                    ; 0789
                                  10  AC  DD 0016A         PUSHL    CNCTDESC                                ;
                                      56  DD 0016D         PUSHL    R6                                      ;
                  FDC3  CF            03  FB 0016F         CALLS    #3, TRY_CONNECT                         ;
                        58            50  D0 00174         MOVL     R0, STATUS                              ;
                    2F  A6        04  AE  90 00177 11$:    MOVB     DESC_1, 47(R6)                          ; 0790
                        32            58  E8 0017C         BLBS     STATUS, 14$                             ; 0791
                        7E        2C  A6  3C 0017F         MOVZWL   44(R6), -(SP)                           ; 0793
             00000000G  00            01  FB 00183         CALLS    #1, SYS$DASSGN                          ;
          1A        2E  A6            01  E0 0018A         BBS      #1, 46(R6), 13$                         ; 0794
                                      58  DD 0018F         PUSHL    STATUS                                  ; 0795
                                  08  AE  9F 00191         PUSHAB   DESC_1                                  ;
                                      01  DD 00194 12$:    PUSHL    #1                                      ;
                            007E802A  8F  DD 00196         PUSHL    #8290346                                ;
             00000000G  00            04  FB 0019C         CALLS    #4, LIB$SIGNAL                          ;
                    1C  A6            58  D0 001A3         MOVL     STATUS, 28(R6)                          ; 0796
                                      08  11 001A7         BRB      14$                                     ;
                        50  007E802A  8F  D0 001A9 13$:    MOVL     #8290346, R0                            ; 0797
                                          04 001B0         RET                                              ;
                        50            58  D0 001B1 14$:    MOVL     STATUS, R0                              ; 0799
                                          04 001B4         RET                                              ;
                    04  AE        38  A9  3C 001B5 15$:    MOVZWL   PREFIX_DESC, DESC_1                     ; 0801
                                  04  AE  D7 001BA         DECL     DESC_1                                  ;
                    08  AE        3C  A9  D0 001BD         MOVL     PREFIX_DESC+4, DESC_1+4                 ; 0802
                                  10  A6  9F 001C2         PUSHAB   16(R6)                                  ; 0804
                                  08  AE  9F 001C5         PUSHAB   DESC_1                                  ;
                                  14  AE  9F 001C8         PUSHAB   DESC                                    ;
             00000000G  00            03  FB 001CB         CALLS    #3, LIB$FIND_IMAGE_SYMBOL               ;
                        49            50  E9 001D2         BLBC     STATUS, 18$                             ;
                                  0C  AE  9F 001D5         PUSHAB   DESC                                    ; 0806
                  FCFB  CF            01  FB 001D8         CALLS    #1, CHECK_PROTOCOL_VERSION              ;
                        3E            50  E9 001DD         BLBC     STATUS, 18$                             ;
          0B 00000000G  00            03  E1 001E0         BBC      #3, MAIL$GL_FLAGS+2, 16$                ; 0815
                        50 00000000G  00  9E 001E8         MOVAB    MAIL$Q_ATTDESC, R0                      ;
                                      50  DD 001EF         PUSHL    R0                                      ;
                                      02  11 001F1         BRB      17$                                     ;
                                      7E  D4 001F3 16$:    CLRL     -(SP)                                   ;
                        7E 00000000G  00  3C 001F5 17$:    MOVZWL   MAIL$GL_SYSFLAGS+2, -(SP)               ; 0814
                    50        10   AC  D0 001FC              MOVL     CNCTDESC, R0                              ; 0813
                    7E      0081   C0  9A 00200              MOVZBL   129(R0), -(SP)                            ;
                    7E      0080   C0  9A 00205              MOVZBL   128(R0), -(SP)                            ; 0812
                        007E802A   8F  DD 0020A              PUSHL    #8290346                                  ; 0807
                              0C   AC  DD 00210              PUSHL    NODE_DESC                                 ; 0810
                                   57  DD 00213              PUSHL    R7                                        ; 0809
                                   7E  D4 00215              CLRL     -(SP)                                     ; 0807
                              0C   A6  9F 00217              PUSHAB   12(R6)                                    ;
               10   B6                 09  FB 0021A              CALLS    #9, @16(R6)                               ;
                                       04 0021E 18$:         RET                                                ; 0819

; Routine Size:  543 bytes,    Routine Base:  $CODE$ + 0428
;  685        0820  1 GLOBAL ROUTINE MAIL$CREATELINK (PROTOCOL_DESC,NODE_DESC,CNCTDESC,RETADR) =
;  686        0821  1 !+++
;  687        0822  1 ! FUNCTIONAL DESCRIPTION:
;  688        0823  1 !
;  689        0824  1 !       This routine is called to create a logical link to the
;  690        0825  1 !       specified node. First, the existing logical link list is
;  691        0826  1 !       searched to see if a link to that node already exists.  If
;  692        0827  1 !       it does, then the address of the list entry is returned.
;  693        0828  1 !       If a link does not exist, one is assigned and a logical link
;  694        0829  1 !       list entry is created, entered in the list, and the address returned.
;  695        0830  1 !
;  696        0831  1 ! INPUTS:
;  697        0832  1 !
;  698        0833  1 !       protocol_desc = address of descriptor of protocol, 0 implies DECnet
;  699        0834  1 !       node_desc = address of descriptor of node name
;  700        0835  1 !       cnctdesc = address of cnct block
;  701        0836  1 !       retadr = address of longword to return logical link list entry address
;  702        0837  1 !
;  703        0838  1 !---
;  704        0839  2 BEGIN
;  705        0840  2
;  706        0841  2 MAP
;  707        0842  2     PROTOCOL_DESC : REF $BBLOCK,
;  708        0843  2     NODE_DESC : REF $BBLOCK,
;  709        0844  2     CNCTDESC : REF $BBLOCK,
;  710        0845  2     RETADR : REF VECTOR[,LONG];
;  711        0846  2
;  712        0847  2 BUILTIN
;  713        0848  2     INSQUE;
;  714        0849  2
;  715        0850  2 LOCAL
;  716        0851  2     STATUS,
;  717        0852  2     PTR : REF $BBLOCK;
;  718        0853  2
;  719        0854  2 BIND
;  720        0855  2     LNKLST = CNCTDESC[CNCT_Q_LNKLST] : VECTOR[,LONG];
;  721        0856  2
;  722        0857  2 PTR = .LNKLST[0];
;  723        0858  2 !
;  724        0859  2 ! See if link already exists
;  725        0860  2 !
;  726        0861  2 WHILE .PTR NEQ LNKLST[0]
;  727        0862  3 DO BEGIN
;  728        0863  3     IF CH$EQL(.NODE_DESC[DSC$W_LENGTH],.NODE_DESC[DSC$A_POINTER],
;  729        0864  3             .PTR[LNK_B_NODLEN],PTR[LNK_T_NODE])
;  730        0865  5     THEN IF ((.PTR[LNK_B_PNLEN] EQL 0)          !Check protocol spec match
;  731        0866  4             AND (.PROTOCOL_DESC[DSC$W_LENGTH] EQL 0))
;  732        0867  3             OR CH$EQL(.PROTOCOL_DESC[DSC$W_LENGTH],
;  733        0868  3                     .PROTOCOL_DESC[DSC$A_POINTER],
;  734        0869  3                     .PTR[LNK_B_PNLEN],PTR[LNK_T_PNAM])
;  735        0870  4         THEN BEGIN
;  736        0871  4             RETADR[0] = .PTR;                   !Return address of found lnk
;  737        0872  4             IF .PTR[LNK_V_DEAD]
;  738        0873  4                 AND .MAIL$GL_FLAGS[MAIF_V_NETJOB] !Only signal if net slave
;  739        0874  4                 THEN SIGNAL(MAIL$_LOGLINK,1,.NODE_DESC,.PTR[LNK_L_STS]);
;  740        0875  4             RETURN (NOT .PTR[LNK_V_DEAD]);      ! and whether it's dead or not
;  741        0876  3             END;
;  742          0877  3       PTR = .PTR[LNK_L_FLINK];                           !Next block
;  743          0878  2       END;
;  744          0879  2  !
;  745          0880  2  ! Not found.  Create logical link list entry
;  746          0881  2  !
;  747        P 0882  2  IF_ERR(LIB$GET_VM(%REF(.PROTOCOL_DESC[DSC$W_LENGTH]+LNK_C_LENGTH),PTR);,
;  748        P 0883  2          SIGNAL(.STATUS);
;  749          0884  2          RETURN .STATUS);
;  750          0885  2
;  751          0886  2  !
;  752          0887  2  ! Insert into the list
;  753          0888  2  !
;  754          0889  2  CH$FILL(0,LNK_C_LENGTH,.PTR);
;  755          0890  2  INSQUE(.PTR,LNK[ST]);
;  756          0891  2  PTR[LNK_B_NODLEN] = .NODE_DESC[DSC$W_LENGTH];
;  757          0892  2  CH$MOVE(.PTR[LNK_B_NODLEN],.NODE_DESC[DSC$A_POINTER],PTR[LNK_T_NODE]);
;  758          0893  2  !
;  759          0894  2  ! Copy protocol name if passed.  Set ALTP flag
;  760          0895  2  !
;  761          0896  2  IF (PTR[LNK_B_PNLEN] = .PROTOCOL_DESC[DSC$W_LENGTH]) NEQ 0
;  762          0897  3  THEN BEGIN
;  763          0898  3      CH$MOVE(.PTR[LNK_B_PNLEN],.PROTOCOL_DESC[DSC$A_POINTER],
;  764          0899  3                                  PTR[LNK_T_PNAM]);
;  765          0900  3      PTR[LNK_V_ALTP] = TRUE;
;  766          0901  2      END;
;  767          0902  2  !
;  768          0903  2  ! Create logical link to slave mail
;  769          0904  2  !
;  770          0905  2  RETADR[0] = .PTR;
;  771          0906  2  STATUS = CONNECT_LINK(.PTR,.PROTOCOL_DESC,.NODE_DESC,.CNCTDESC);
;  772          0907  2  $DASSGN(CHAN=.PTR[LNK_W_MBXCHAN]);                 !Deassign mailbox now
;  773          0908  2  IF NOT .STATUS
;  774          0909  3      THEN BEGIN
;  775          0910  3          PTR[LNK_W_CHAN] = 0;
;  776          0911  3          PTR[LNK_V_DEAD] = TRUE;
;  777          0912  2          END;
;  778          0913  2
;  779          0914  2  RETURN .STATUS
;  780          0915  2
;  781          0916  1  END;



                                         03FC 00000          .ENTRY  MAIL$CREATELINK, Save R2,R3,R4,R5,R6,R7,R8,-; 0820
                                                                     R9                                           ;
                        59 00000000G  00  9E 00002          MOVAB   LIB$SIGNAL, R9                               ;
                        5E            08  C2 00009          SUBL2   #8, SP                                       ;
            56     0C   AC            30  C1 0000C          ADDL3   #48, CNCTDESC, R6                            ; 0855
                   04   AE            66  D0 00011          MOVL    (R6), PTR                                    ; 0857
                        55      08    AC  D0 00015          MOVL    NODE_DESC, R5                                ; 0863
                        54      04    AE  D0 00019 1$:      MOVL    PTR, R4                                      ; 0861
                        56            54  D1 0001D          CMPL    R4, R6                                       ;
                                      5D  13 00020          BEQL    6$                                           ;
                        50      2F    A4  9A 00022          MOVZBL  47(R4), R0                                   ; 0864
      50           00   04  B5  08    BC  2D 00026          CMPC5   @NODE_DESC, @4(R5), #0, R0, 48(R4)           ;
```

```
                                                      30    A4     0002D                                                          ;
                                                            48  12 0002F      BNEQ    5$                                          ;
                                                      4F    A4  95 00031      TSTB    79(R4)                                      ; 0865
                                                            05  12 00034      BNEQ    2$                                          ;
                                                      04    BC  B5 00036      TSTW    @PROTOCOL_DESC                              ; 0866
                                                            12  13 00039      BEQL    3$                                          ;
                                        50            04    AC  D0 0003B 2$:  MOVL    PROTOCOL_DESC, R0                           ; 0867
                                        51            4F    A4  9A 0003F      MOVZBL  79(R4), R1                                  ; 0869
        51            00          04    B0                  60  2D 00043      CMPC5   (R0), @4(R0), #0, R1, 80(R4)                ;
                                                      50    A4     00049                                                          ;
                                                            2C  12 0004B      BNEQ    5$                                          ;
                                  10    BC                  54  D0 0004D 3$:  MOVL    R4, @RETADR                                 ; 0871
                      19          2E    A4                  01  E1 00051      BBC     #1, 46(R4), 4$                              ; 0872
                      11   00000000G    00                  01  E1 00056      BBC     #1, MAIL$GL_FLAGS, 4$                       ; 0873
                                                      1C    A4  DD 0005E      PUSHL   28(R4)                                      ; 0874
                                                      08    AC  DD 00061      PUSHL   NODE_DESC                                   ;
                                                            01  DD 00064      PUSHL   #1                                          ;
                                                48  007E802A    8F  DD 00066      PUSHL   #8290346                                ;
                                        69                  04  FB 0006C      CALLS   #4, LIB$SIGNAL                              ;
        50      2E    A4                01                  01  EF 0006F 4$:  EXTZV   #1, #1, 46(R4), R0                          ; 0875
                                        50                  50  D2 00075      MCOML   R0, R0                                      ;
                                                                04 00078      RET                                                 ;
                                  04    AE                  64  D0 00079 5$:  MOVL    (R4), PTR                                   ; 0877
                                                            9A  11 0007D      BRB     1$                                          ; 0861
                                                      04    AE  9F 0007F 6$:  PUSHAB  PTR                                         ; 0884
                                  04    AE            04    BC  3C 00082      MOVZWL  @PROTOCOL_DESC, 4(SP)                       ;
                                  04    AE      00000050    8F  C0 00087      ADDL2   #80, 4(SP)                                  ;
                                                      04    AE  9F 0008F      PUSHAB  4(SP)                                       ;
                           00000000G    00                  02  FB 00092      CALLS   #2, LIB$GET_VM                              ;
                                        52                  50  D0 00099      MOVL    R0, STATUS                                  ;
                                        07                  52  E8 0009C      BLBS    STATUS, 7$                                  ;
                                        52                  52  DD 0009F      PUSHL   STATUS                                      ;
                                        69                  01  FB 000A1      CALLS   #1, LIB$SIGNAL                              ;
                                                            68  11 000A4      BRB     9$                                          ;
0050    8F            00                6E                  00  2C 000A6 7$:  MOVC5   #0, (SP), #0, #80, @PTR                     ; 0889
                                                      04    BE     000AD                                                          ;
                                        66            04    BE  0E 000AF      INSQUE  @PTR, (R6)                                  ; 0890
                                        56            04    AE  D0 000B3      MOVL    PTR, R6                                     ; 0891
                                        58            08    AC  D0 000B7      MOVL    NODE_DESC, R8                               ;
                                  2F    A6                  68  90 000BB      MOVB    (R8), 47(R6)                                ;
                                        50            2F    A6  9A 000BF      MOVZBL  47(R6), R0                                  ; 0892
                30    A6          04    B8                  50  28 000C3      MOVC3   R0, @4(R8), 48(R6)                          ;
                                        57            04    AC  D0 000C9      MOVL    PROTOCOL_DESC, R7                           ; 0896
                                        50                  67  3C 000CD      MOVZWL  (R7), R0                                    ;
                                  4F    A6                  50  90 000D0      MOVB    R0, 79(R6)                                  ;
                                                            50  D5 000D4      TSTL    R0                                          ;
                                                            0E  13 000D6      BEQL    8$                                          ;
                                        50            4F    A6  9A 000D8      MOVZBL  79(R6), R0                                  ; 0898
                50    A6          04    B7                  50  28 000DC      MOVC3   R0, @4(R7), 80(R6)                          ; 0899
                                  2E    A6                  04  88 000E2      BISB2   #4, 46(R6)                                  ; 0900
                                  10    BC                  56  D0 000E6 8$:  MOVL    R6, @RETADR                                 ; 0905
                                                      0C    AC  DD 000EA      PUSHL   CNCTDESC                                    ; 0906
                                                    01C0    8F  BB 000ED      PUSHR   #^M<R6,R7,R8>                               ;
                                FCEB    CF                  04  FB 000F1      CALLS   #4, CONNECT_LINK                            ;
                                        52                  50  D0 000F6      MOVL    R0, STATUS                                  ;
                                        7E            2A    A6  3C 000F9      MOVZWL  42(R6), -(SP)                               ; 0907
                           00000000G    00                  01  FB 000FD      CALLS   #1, SYS$DASSGN                              ;
                07             52 E8 00104          BLBS    STATUS, 9$                      ; 0908
                         2C    A6 B4 00107          CLRW    44(R6)                          ; 0910
          2E    A6             02 88 0010A          BISB2   #2, 46(R6)                      ; 0911
                50             52 D0 0010E 9$:      MOVL    STATUS, R0                      ; 0914
                                  04 00111          RET                                     ; 0916

; Routine Size:  274 bytes,    Routine Base:  $CODE$ + 0647
;  783          0917  1 ROUTINE WRITE_SLAVE(LNKDESC,OUT_DESC) =
;  784          0918  1 !++
;  785          0919  1 ! FUNCTIONAL DESCRIPTION:
;  786          0920  1 !
;  787          0921  1 !       Write a record to the remote node
;  788          0922  1 !
;  789          0923  1 ! Inputs:
;  790          0924  1 !
;  791          0925  1 !       lnkdesc = address of descriptor of lnk block
;  792          0926  1 !       out_desc = address of descriptor of record to write
;  793          0927  1 !
;  794          0928  1 !
;  795          0929  1 !       Errors are signalled as well as returned.
;  796          0930  1 !--
;  797          0931  2 BEGIN
;  798          0932  2
;  799          0933  2 MAP
;  800          0934  2     LNKDESC : REF $BBLOCK,
;  801          0935  2     OUT_DESC : REF $BBLOCK;
;  802          0936  2
;  803          0937  2 LOCAL
;  804          0938  2     STATUS;
;  805          0939  2
;  806          0940  2 BIND
;  807          0941  2     QIOSB = LNKDESC[LNK_Q_IOSB] : VECTOR[,WORD];
;  808          0942  2
;  809          0943  2 IF .LNKDESC[LNK_V_DEAD]
;  810          0944  2     THEN RETURN FALSE;
;  811          0945  2
;  812        P 0946  2 STATUS = $QIOW(CHAN=.LNKDESC[LNK_W_CHAN],
;  813        P 0947  2                 FUNC=IO$_WRITEVBLK,
;  814        P 0948  2                 IOSB=LNKDESC[LNK_Q_IOSB],
;  815        P 0949  2                 P1=(IF .OUT_DESC[DSC$A_POINTER] NEQ 0
;  816        P 0950  2                         THEN .OUT_DESC[DSC$A_POINTER]
;  817        P 0951  2                         ELSE OUT_DESC),
;  818          0952  2                 P2=.OUT_DESC[DSC$W_LENGTH]);
;  819          0953  2
;  820          0954  2 IF .STATUS
;  821          0955  2     THEN STATUS = .QIOSB[0];
;  822          0956  2
;  823          0957  2 IF NOT .STATUS
;  824          0958  3 THEN BEGIN
;  825          0959  3     SIGNAL(((SS$_PROTOCOL AND NOT STS$M_SEVERITY)
;  826          0960  3                     OR STS$K_ERROR OR MAIL$V_FACILITY),0,.STATUS);
;  827          0961  3     LNKDESC[LNK_V_DEAD] = TRUE;
;  828          0962  2     END;
;  829          0963  2
;  830          0964  2 RETURN .STATUS
;  831          0965  1 END;



                                               000C 00000 WRITE_SLAVE:
                                                                   .WORD    Save R2,R3                         ; 0917
                        52        04   AC  D0 00002                MOVL     LNKDESC, R2                        ; 0941
```

```
                           55        2E   A2             01 E0 00006              BBS      #1, 46(R2), 5$                ; 0943
                                                         7E 7C 0000B              CLRQ     -(SP)                         ; 0952
                                                         7E 7C 0000D              CLRQ     -(SP)                         ;
                                          50        08   AC D0 0000F              MOVL     OUT_DESC, R0                  ;
                                          7E             60 3C 00013              MOVZWL   (R0), -(SP)                   ;
                                                    04   A0 D5 00016              TSTL     4(R0)                         ;
                                                         05 13 00019              BEQL     1$                            ;
                                                    04   A0 DD 0001B              PUSHL    4(R0)                         ;
                                                         06 11 0001E              BRB      2$                            ;
                                          50        08   AC 9E 00020 1$:          MOVAB    OUT_DESC, R0                  ;
                                                         50 DD 00024              PUSHL    R0                            ;
                                                         7E 7C 00026 2$:          CLRQ     -(SP)                         ;
                                                    14   A2 9F 00028              PUSHAB   20(R2)                        ;
                                                         30 DD 0002B              PUSHL    #48                           ;
                                          7E        2C   A2 3C 0002D              MOVZWL   44(R2), -(SP)                 ;
                                                         7E D4 00031              CLRL     -(SP)                         ;
                              00000000G   00             0C FB 00033              CALLS    #12, SYS$QIOW                 ;
                                          53             50 D0 0003A              MOVL     R0, STATUS                    ;
                                          07             53 E9 0003D              BLBC     STATUS, 3$                    ; 0954
                                          53        14   A2 3C 00040              MOVZWL   20(R2), STATUS                ; 0955
                                          15             53 E8 00044              BLBS     STATUS, 4$                    ; 0957
                                                         53 DD 00047 3$:          PUSHL    STATUS                        ; 0960
                                                         7E D4 00049              CLRL     -(SP)                         ; 0959
                                             007E2072    8F DD 0004B              PUSHL    #8265842                      ;
                              00000000G   00             03 FB 00051              CALLS    #3, LIB$SIGNAL                ;
                                     2E   A2             02 88 00058              BISB2    #2, 46(R2)                    ; 0961
                                          50             53 D0 0005C 4$:          MOVL     STATUS, R0                    ; 0964
                                                            04 0005F              RET                                    ;
                                                         50 D4 00060 5$:          CLRL     R0                            ; 0965
                                                            04 00062              RET                                    ;

; Routine Size:  99 bytes,    Routine Base:  $CODE$ + 0759
;  833          0966  1 ROUTINE READ_SLAVE(LNKDESC,IN_DESC) =
;  834          0967  1 !++
;  835          0968  1 ! FUNCTIONAL DESCRIPTION:
;  836          0969  1 !
;  837          0970  1 !        Read a record from the remote node
;  838          0971  1 !
;  839          0972  1 ! Inputs:
;  840          0973  1 !
;  841          0974  1 !        lnkdesc = address of lnk block for node
;  842          0975  1 !        in_desc = address of descriptor of buffer
;  843          0976  1 !                length is modified in place to reflect amount actually read
;  844          0977  1 !
;  845          0978  1 !        Errors are signalled as well as returned
;  846          0979  1 !--
;  847          0980  2 BEGIN
;  848          0981  2
;  849          0982  2 MAP
;  850          0983  2     LNKDESC : REF $BBLOCK,
;  851          0984  2     IN_DESC : REF $BBLOCK;
;  852          0985  2 BIND
;  853          0986  2     QIOSB = LNKDESC[LNK_Q_IOSB] : VECTOR[,WORD];
;  854          0987  2
;  855          0988  2 LOCAL
;  856          0989  2     STATUS;
;  857          0990  2
;  858          0991  2 IF .LNKDESC[LNK_V_DEAD]
;  859          0992  2     THEN RETURN FALSE;
;  860          0993  2
;  861        P 0994  2 STATUS = $QIOW(CHAN=.LNKDESC[LNK_W_CHAN],
;  862        P 0995  2                 FUNC=IO$_READVBLK,
;  863        P 0996  2                 IOSB=LNKDESC[LNK_Q_IOSB],
;  864        P 0997  2                 P1=.IN_DESC[DSC$A_POINTER],
;  865          0998  2                 P2=.IN_DESC[DSC$W_LENGTH]);
;  866          0999  2
;  867          1000  2 IN_DESC[DSC$W_LENGTH] = .QIOSB[1];
;  868          1001  2
;  869          1002  2 IF .STATUS
;  870          1003  2     THEN STATUS = .QIOSB[0];
;  871          1004  2
;  872          1005  2 IF NOT .STATUS
;  873          1006  3 THEN BEGIN
;  874          1007  4     SIGNAL((SS$_PROTOCOL AND NOT STS$M_SEVERITY
;  875          1008  3                     OR STS$K_ERROR OR MAIL$V_FACILITY),0,.STATUS);
;  876          1009  3     LNKDESC[LNK_V_DEAD] = TRUE;
;  877          1010  2     END;
;  878          1011  2
;  879          1012  2 RETURN .STATUS
;  880          1013  2
;  881          1014  1 END;



                                             003C 00000 READ_SLAVE:
                                                                .WORD   Save R2,R3,R4,R5                        ; 0966
                           52       04   AC  D0 00002           MOVL    LNKDESC, R2                             ; 0986
```

```
                        55          14   A2  9E 00006           MOVAB   20(R2), R5                          ;
               4B       2E          A2   01  E0 0000A           BBS     #1, 46(R2), 3$                      ; 0991
                                         7E  7C 0000F           CLRQ    -(SP)                               ; 0998
                                         7E  7C 00011           CLRQ    -(SP)                               ;
                        53          08   AC  D0 00013           MOVL    IN_DESC, R3                         ;
                        7E               63  3C 00017           MOVZWL  (R3), -(SP)                         ;
                                    04   A3  DD 0001A           PUSHL   4(R3)                               ;
                                         7E  7C 0001D           CLRQ    -(SP)                               ;
                                    14   A2  9F 0001F           PUSHAB  20(R2)                              ;
                                         31  DD 00022           PUSHL   #49                                 ;
                        7E          2C   A2  3C 00024           MOVZWL  44(R2), -(SP)                       ;
                                         7E  D4 00028           CLRL    -(SP)                               ;
            00000000G   00               0C  FB 0002A           CALLS   #12, SYS$QIOW                       ;
                        54               50  D0 00031           MOVL    R0, STATUS                          ;
                        63          02   A5  B0 00034           MOVW    2(R5), (R3)                         ; 1000
                        06               54  E9 00038           BLBC    STATUS, 1$                          ; 1002
                        54               65  3C 0003B           MOVZWL  (R5), STATUS                        ; 1003
                        15               54  E8 0003E           BLBS    STATUS, 2$                          ; 1005
                                         54  DD 00041 1$:       PUSHL   STATUS                              ; 1008
                                         7E  D4 00043           CLRL    -(SP)                               ; 1007
                              007E2072   8F  DD 00045           PUSHL   #8265842                            ;
            00000000G   00               03  FB 0004B           CALLS   #3, LIB$SIGNAL                      ;
                        2E          A2   02  88 00052           BISB2   #2, 46(R2)                          ; 1009
                        50               54  D0 00056 2$:       MOVL    STATUS, R0                          ; 1012
                                             04 00059           RET                                         ;
                                         50  D4 0005A 3$:       CLRL    R0                                  ; 1014
                                             04 0005C           RET                                         ;

; Routine Size:  93 bytes,    Routine Base:  $CODE$ + 07BC
;  883         1015  1 ROUTINE CHECK_SLAVE_STATUS(LNKDESC) =
;  884         1016  1 !++
;  885         1017  1 ! FUNCTIONAL DESCRIPTION:
;  886         1018  1 !
;  887         1019  1 !       Reads a response from the remote node
;  888         1020  1 !
;  889         1021  1 ! Inputs:
;  890         1022  1 !
;  891         1023  1 !       lnkdesc = address of lnk descriptor for node
;  892         1024  1 !
;  893         1025  1 ! Read from the node, and treat the first 4 bytes as a longword value,
;  894         1026  1 ! indicating success or failure.  If failure, then read and print the
;  895         1027  1 ! error text to follow
;  896         1028  1 !
;  897         1029  1 !--
;  898         1030  2 BEGIN
;  899         1031  2
;  900         1032  2 MAP
;  901         1033  2     LNKDESC : REF $BBLOCK;
;  902         1034  2
;  903         1035  2 LOCAL
;  904         1036  2     STATUS,
;  905         1037  2     DESC : VECTOR[2,LONG],
;  906         1038  2     TMPBUF : $BBLOCK[MAIL$K_INBUFFSZ];
;  907         1039  2
;  908         1040  2 BIND
;  909         1041  2     TMPVEC = TMPBUF : VECTOR[,LONG];
;  910         1042  2
;  911         1043  2 DESC[0] = MAIL$K_INBUFFSZ;
;  912         1044  2 DESC[1] = TMPBUF;
;  913       P 1045  2 IF_ERR(READ_SLAVE(.LNKDESC,DESC);,
;  914         1046  2         RETURN .STATUS);
;  915         1047  2
;  916         1048  2 !
;  917         1049  2 ! Check the first longword read.  If lbs, then return success.
;  918         1050  2 ! Otherwise, call routine to read error text from remote node (until
;  919         1051  2 ! 1 byte record of 0) and then signal it
;  920         1052  2 !
;  921         1053  4 RETURN (IF (STATUS = .TMPVEC[0])
;  922         1054  3             THEN TRUE
;  923         1055  4             ELSE (MAIL$READ_ERROR_TEXT(.LNKDESC,READ_SLAVE);
;  924         1056  3                         .STATUS))
;  925         1057  1 END;



                                   000C 00000 CHECK_SLAVE_STATUS:
                                                        .WORD   Save R2,R3                                  ; 1015
                  53        9E   AF   9E 00002          MOVAB   READ_SLAVE, R3                              ;
                  5E      FDF8   CE   9E 00006          MOVAB   -520(SP), SP                                ;
            F8    AD      0200   8F   3C 0000B          MOVZWL  #512, DESC                                  ; 1043
            FC    AD             6E   9E 00011          MOVAB   TMPBUF, DESC+4                              ; 1044
                            F8   AD   9F 00015          PUSHAB  DESC                                        ; 1046
                            04   AC   DD 00018          PUSHL   LNKDESC                                     ;
                  63             02   FB 0001B          CALLS   #2, READ_SLAVE                              ;
```

```
                      1A              50  E9 0001E         BLBC    STATUS, 3$                              ;
                      52              6E  D0 00021         MOVL    TMPVEC, STATUS                          ; 1053
                      05              52  E9 00024         BLBC    STATUS, 1$                              ;
                      52              01  D0 00027         MOVL    #1, R2                                  ;
                                      0C  11 0002A         BRB     2$                                      ;
                                      53  DD 0002C 1$:     PUSHL   R3                                      ; 1055
                               04     AC  DD 0002E         PUSHL   LNKDESC                                 ;
            00000000G  00             02  FB 00031         CALLS   #2, MAIL$READ_ERROR_TEXT                ;
                       50             52  D0 00038 2$:     MOVL    R2, R0                                  ; 1053
                                          04 0003B 3$:     RET                                             ; 1057

; Routine Size:  60 bytes,    Routine Base:  $CODE$ + 0819
;  927         1058  1 ROUTINE WRITE_CHECK_SLAVE(LNKDESC,OUT_DESC) =
;  928         1059  1 !++
;  929         1060  1 ! FUNCTIONAL DESCRIPTION:
;  930         1061  1 !
;  931         1062  1 !        Write a record to the remote node, and then check the
;  932         1063  1 !        response sent back
;  933         1064  1 !
;  934         1065  1 ! Inputs:
;  935         1066  1 !
;  936         1067  1 !        lnkdesc = address of lnk descriptor
;  937         1068  1 !        outdesc = address of descriptor of record to send
;  938         1069  1 !
;  939         1070  1 ! The record is written to the remote node.  A response is read. If
;  940         1071  1 ! not success, the error text is read and signalled.
;  941         1072  1 !
;  942         1073  1 !--
;  943         1074  2 BEGIN
;  944         1075  2
;  945         1076  2 MAP
;  946         1077  2     LNKDESC : REF $BBLOCK,
;  947         1078  2     OUT_DESC : REF $BBLOCK;
;  948         1079  2
;  949         1080  2 BUILTIN
;  950         1081  2     CALLG,AP;
;  951         1082  2
;  952         1083  2 LOCAL
;  953         1084  2     STATUS;
;  954         1085  2
;  955         1086  3 IF NOT (STATUS = CALLG(.AP,WRITE_SLAVE))
;  956         1087  2     THEN RETURN .STATUS
;  957         1088  2     ELSE RETURN CHECK_SLAVE_STATUS(.LNKDESC)
;  958         1089  1 END;



                                          0000 00000 WRITE_CHECK_SLAVE:
                                                           .WORD   Save nothing                                ; 1058
                 FEFD   CF             6C  FA 00002        CALLG   (AP), WRITE_SLAVE                           ; 1086
                        51             50  D0 00007        MOVL    R0, STATUS                                  ;
                        04             50  E8 0000A        BLBS    R0, 1$                                      ;
                        50             51  D0 0000D        MOVL    STATUS, R0                                  ; 1088
                                           04 00010        RET                                                 ;
                                 04    AC  DD 00011 1$:    PUSHL   LNKDESC                                     ;
                   AC   AF             01  FB 00014        CALLS   #1, CHECK_SLAVE_STATUS                      ;
                                           04 00018        RET                                                 ; 1089

; Routine Size:  25 bytes,    Routine Base:  $CODE$ + 0855
;  960        1090  1 GLOBAL ROUTINE MAIL$NET_FROM(LNKDESC,SENDER_DESC) =
;  961        1091  1 !++
;  962        1092  1 ! FUNCTIONAL DESCRIPTION:
;  963        1093  1 !
;  964        1094  1 !       Send the sender's name to a remote node
;  965        1095  1 !
;  966        1096  1 ! Inputs:
;  967        1097  1 !
;  968        1098  1 !       lnkdesc = address of lnk descriptor
;  969        1099  1 !       sender_desc = address of descriptor of sender's name
;  970        1100  1 !
;  971        1101  1 !--
;  972        1102  1
;  973        1103  2 BEGIN
;  974        1104  2
;  975        1105  2 MAP
;  976        1106  2     LNKDESC : REF $BBLOCK,
;  977        1107  2     SENDER_DESC : REF $BBLOCK;
;  978        1108  2
;  979        1109  2 LOCAL
;  980        1110  2     DESC : VECTOR[2,LONG],
;  981        1111  2     STATUS;
;  982        1112  2
;  983        1113  2 BUILTIN
;  984        1114  2     CALLG,AP;
;  985        1115  2
;  986        1116  2 IF .LNKDESC[LNK_V_DEAD]
;  987        1117  2     OR .LNKDESC[LNK_V_FSENT]
;  988        1118  2     THEN RETURN TRUE;
;  989        1119  2
;  990        1120  2 IF .LNKDESC[LNK_V_ALTP]
;  991        1121  3 THEN BEGIN
;  992        1122  3     DESC[0] = .LNKDESC[LNK_B_NODLEN];
;  993        1123  3     DESC[1] = LNKDESC[LNK_T_NODE];
;  994        1124  4     STATUS = (IF .LNKDESC[LNK_L_TFRADR] NEQ 0
;  995        1125  4                     THEN (.LNKDESC[LNK_L_TFRADR])(LNKDESC[LNK_L_CONTEXT],
;  996        1126  4                                                   LNK_C_OUT_SENDER,
;  997        1127  4                                                   DESC,
;  998        1128  4                                                   .SENDER_DESC)
;  999        1129  4                     ELSE TRUE)
; 1000        1130  3     END
; 1001        1131  2 ELSE STATUS = CALLG(.AP,WRITE_SLAVE);
; 1002        1132  2
; 1003        1133  2 LNKDESC[LNK_V_FSENT] = TRUE;
; 1004        1134  2 RETURN .STATUS
; 1005        1135  2
; 1006        1136  1 END;



                                        0000 00000        .ENTRY  MAIL$NET_FROM, Save nothing                  ; 1090
                 5E                08   C2 00002          SUBL2   #8, SP                                       ;
                 50          04    AC   D0 00005          MOVL    LNKDESC, R0                                  ; 1116
     05          2E    A0          01   E0 00009          BBS     #1, 46(R0), 1$                               ;
     04          2E    A0          03   E1 0000E          BBC     #3, 46(R0), 2$                               ; 1117
```

```
                  50          01 D0 00013 1$:    MOVL    #1, R0                         ; 1118
                              04 00016           RET                                    ;
   24      2E     A0          02 E1 00017 2$:    BBC     #2, 46(R0), 4$                 ; 1120
                  6E    2F    AC 9A 0001C        MOVZBL  47(R0), DESC                   ; 1122
           04     AE    30    A0 9E 00020        MOVAB   48(R0), DESC+4                 ; 1123
                        10    A0 D5 00025        TSTL    16(R0)                         ; 1124
                              11 13 00028        BEQL    3$                             ;
                        08    AC DD 0002A        PUSHL   SENDER_DESC                    ; 1128
                        04    AE 9F 0002D        PUSHAB  DESC                           ; 1125
                              01 DD 00030        PUSHL   #1                             ;
                        0C    A0 9F 00032        PUSHAB  12(R0)                         ;
           10     B0          04 FB 00035        CALLS   #4, @16(R0)                    ;
                              0A 11 00039        BRB     5$                             ;
                  50          01 D0 0003B 3$:    MOVL    #1, STATUS                     ; 1124
                              05 11 0003E        BRB     5$                             ;
         FEA6     CF          6C FA 00040 4$:    CALLG   (AP), WRITE_SLAVE              ; 1131
                  51    04    AC D0 00045 5$:    MOVL    LNKDESC, R1                    ; 1133
           2E     A1          08 88 00049        BISB2   #8, 46(R1)                     ;
                              04 0004D           RET                                    ; 1136

; Routine Size:  78 bytes,    Routine Base:  $CODE$ + 086E
; 1008          1137  1 GLOBAL ROUTINE MAIL$NET_ADDR(LNKDESC,ADDR_DESC) =
; 1009          1138  1 !++
; 1010          1139  1 ! FUNCTIONAL DESCRIPTION:
; 1011          1140  1 !
; 1012          1141  1 !       Check that an addressee exists on a remote node
; 1013          1142  1 !
; 1014          1143  1 ! Inputs:
; 1015          1144  1 !
; 1016          1145  1 !       lnkdesc = address of lnk descriptor for node
; 1017          1146  1 !       addr_desc = address of descriptor of addressee
; 1018          1147  1 !
; 1019          1148  1 ! Returns true if addressee exists, false if not
; 1020          1149  1 !
; 1021          1150  1 !--
; 1022          1151  2 BEGIN
; 1023          1152  2
; 1024          1153  2 MAP
; 1025          1154  2     LNKDESC : REF $BBLOCK,
; 1026          1155  2     ADDR_DESC : REF $BBLOCK;
; 1027          1156  2
; 1028          1157  2 LOCAL
; 1029          1158  2     DESC : VECTOR[2,LONG];
; 1030          1159  2 BUILTIN
; 1031          1160  2     CALLG,AP;
; 1032          1161  2
; 1033          1162  2 IF .LNKDESC[LNK_V_DEAD]
; 1034          1163  2     THEN RETURN FALSE;
; 1035          1164  2
; 1036          1165  2 IF .LNKDESC[LNK_V_ALTP]
; 1037          1166  3 THEN BEGIN
; 1038          1167  3     DESC[0] = .LNKDESC[LNK_B_NODLEN];
; 1039          1168  3     DESC[1] = LNKDESC[LNK_T_NODE];
; 1040          1169  4     RETURN (IF .LNKDESC[LNK_L_TFRADR] EQL 0
; 1041          1170  4                     THEN FALSE
; 1042          1171  4                     ELSE (.LNKDESC[LNK_L_TFRADR])(LNKDESC[LNK_L_CONTEXT],
; 1043          1172  4                                                   LNK_C_OUT_CRUSER,
; 1044          1173  4                                                   DESC,
; 1045          1174  4                                                   .ADDR_DESC,
; 1046          1175  4                                                   MAIL$READ_ERROR_TEXT))
; 1047          1176  3     END
; 1048          1177  2 ELSE IF .LNKDESC[LNK_W_CHAN] EQL 0
; 1049          1178  2         THEN RETURN FALSE
; 1050          1179  2         ELSE RETURN CALLG(.AP,WRITE_CHECK_SLAVE)
; 1051          1180  1 END;



                                0000 00000        .ENTRY  MAIL$NET_ADDR, Save nothing          : 1137
                5E          08    C2 00002        SUBL2   #8, SP                               :
                50    04    AC    D0 00005        MOVL    LNKDESC, R0                          : 1162
   34     2E    A0          01    E0 00009        BBS     #1, 46(R0), 2$                       :
   24     2E    A0          02    E1 0000E        BBC     #2, 46(R0), 1$                       : 1165
                6E    2F    A0    9A 00013        MOVZBL  47(R0), DESC                         : 1167
          04    AE    30    A0    9E 00017        MOVAB   48(R0), DESC+4                       : 1168
                      10    A0    D5 0001C        TSTL    16(R0)                               : 1169
```

```
                                       21  13 0001F            BEQL    2$                                             ;
                           00000000G   00  9F 00021            PUSHAB  MAIL$READ_ERROR_TEXT                           ; 1171
                                  08   AC  DD 00027            PUSHL   ADDR_DESC                                      ; 1174
                                  08   AE  9F 0002A            PUSHAB  DESC                                           ; 1171
                                       02  DD 0002D            PUSHL   #2                                             ;
                                  0C   A0  9F 0002F            PUSHAB  12(R0)                                         ;
                  10   B0              05  FB 00032            CALLS   #5, @16(R0)                                    ;
                                           04 00036            RET                                                    ; 1177
                                  2C   A0  B5 00037 1$:        TSTW    44(R0)                                         ;
                                       06  13 0003A            BEQL    2$                                             ;
                FF58   CF              6C  FA 0003C            CALLG   (AP), WRITE_CHECK_SLAVE                        ; 1179
                                           04 00041            RET                                                    ; 1177
                                       50  D4 00042 2$:        CLRL    R0                                             ; 1180
                                           04 00044            RET                                                    ;

; Routine Size:  69 bytes,    Routine Base:  $CODE$ + 08BC
```

```
  1 ROUTINE SEND_MESSAGE(LNKDESC,CNCTDESC) =
  1 !++
  1 ! FUNCTIONAL DESCRIPTION:
  1 !
  1 !         Send text of message to remote node
  1 !
  1 ! Inputs:
  1 !
  1 !         lnkdesc = address of lnk descriptor for remote node
  1 !         cnctdesc = address of cnct descriptor for message
  1 !
  1 !--
  2 BEGIN
  2
  2 MAP
  2     LNKDESC : REF $BBLOCK,
  2     CNCTDESC : REF $BBLOCK;
  2
  2 BIND
  2     RAB = CNCTDESC[CNCT_T_RAB] : $BBLOCK;
  2
  2 LOCAL
  2     STATUS,
  2     DESC : VECTOR[2,LONG];
  2
  2 RAB[RAB$W_USZ] = MAIL$K_INBUFFSZ;
  2 RAB[RAB$L_UBF] = CNCTDESC[CNCT_T_BUFFER];          !Ensure User buffer is right
  2 !
  2 ! ensure rab is connected for BIO if sending in block mode
  2 !
  2 IF .LNKDESC[LNK_V_BLKMODE]
  3 THEN BEGIN
  3     IF NOT .RAB[RAB$V_BIO]
  4     THEN BEGIN
  4         $DISCONNECT(RAB=RAB,ERR=UTIL$REPORT_IO_ERROR);
  4         RAB[RAB$V_BIO] = TRUE;
P 4         IF_ERR($CONNECT(RAB=RAB,ERR=UTIL$REPORT_IO_ERROR);,
  4                 RETURN .STATUS);
  3         END;
  3     !
  3     ! Read from file and write to node until errors or end
  3     !
  3     WHILE (STATUS=$READ(RAB=RAB,ERR=UTIL$REPORT_IO_ERROR)) NEQ RMS$_EOF
  4     DO BEGIN
  4         IF NOT .STATUS THEN RETURN .STATUS;
  4         DESC[0] = .RAB[RAB$W_RSZ];
  4         DESC[1] = .RAB[RAB$L_RBF];
P 4         IF_ERR(WRITE_SLAVE(.[NKDESC,DESC);,
  4                         RETURN .STATUS);
  4         END
  3     END
  3 !
  3 ! Do it with records if we have to.  Make sure rab is connected for
  3 ! record i/o
  3 !
  3 ELSE BEGIN
  3     IF .RAB[RAB$V_BIO]
```

```
; 1110          1238  4     THEN BEGIN
; 1111          1239  4         $DISCONNECT(RAB=RAB,ERR=UTIL$REPORT_IO_ERROR);
; 1112          1240  4         RAB[RAB$V_BIO] = FALSE;
; 1113        P 1241  4         IF_ERR($CONNECT(RAB=RAB,ERR=UTIL$REPORT_IO_ERROR);,
; 1114          1242  4                 RETURN .STATUS);
; 1115          1243  3         END;
; 1116          1244  3     WHILE (STATUS=$GET(RAB=RAB,ERR=UTIL$REPORT_IO_ERROR)) NEQ RMS$_ECF
; 1117          1245  4     DO BEGIN
; 1118          1246  4         IF NOT .STATUS THEN RETURN .STATUS;
; 1119          1247  4         IF .RAB[RAB$W_RSZ] GTRU 255 THEN
; 1120          1248  4                 RETURN SIGNAL(RMS$_RTB,.RAB[RAB$W_RSZ]);
; 1121          1249  4         DESC[0] = .RAB[RAB$W_RSZ];
; 1122          1250  4         DESC[1] = .RAB[RAB$L_RBF];
; 1123          1251  6         IF NOT ((.RAB[RAB$W_RSZ] EQL 1)         !Don't send 1-byte records of 0
; 1124          1252  5             AND (.(.RAB[RAB$L_RBF])<0,8> EQL 0))        ! because they break protocol
; 1125        P 1253  4                 THEN IF_ERR(WRITE_SLAVE(.LNKDESC,DESC);,
; 1126          1254  4                         RETURN .STATUS);
; 1127          1255  3         END;
; 1128          1256  2     END;
; 1129          1257  2 DESC[0] = 1;                    !Make a descriptor
; 1130          1258  2 DESC[1] = DESC[0] + 2;          ! Describing 1 byte of 0
; 1131          1259  2 RETURN WRITE_SLAVE(.LNKDESC,DESC)       !Send the 1 byte of 0 and return
; 1132          1260  1 END;



                                                                          .EXTRN  SYS$DISCONNECT, SYS$CONNECT
                                                                          .EXTRN  SYS$READ, SYS$GET

                                                       01FC 00000 SEND_MESSAGE:
                                                                          .WORD   Save R2,R3,R4,R5,R6,R7,R8                   ; 1181
                                       58 00000000G  00  9E 00002         MOVAB   SYS$CONNECT, R8                             ;
                                       57 00000000G  00  9E 00009         MOVAB   SYS$DISCONNECT, R7                          ;
                                       56      FE44   CF  9E 00010         MOVAB   WRITE_SLAVE, R6                             ;
                                       55 00000000G  00  9E 00015         MOVAB   UTIL$REPORT_IO_ERROR, R5                    ;
                                       5E            08  C2 0001C         SUBL2   #8, SP                                      ;
                         52       08   AC 00000286   8F  C1 0001F         ADDL3   #646, CNCTDESC, R2                          ; 1200
                                  20   A2     0200   8F  B0 00028         MOVW    #512, 32(R2)                                ; 1206
                    24   A2       08   AC 00000086   8F  C1 0002E         ADDL3   #134, CNCTDESC, 36(R2)                      ; 1207
                                       54       04   AC  D0 00038         MOVL    LNKDESC, R4                                 ; 1211
                         41       2E   A4            04  E1 0003C         BBC     #4, 46(R4), 3$                              ;
                         10       05   A2            03  E0 00041         BBS     #3, 5(R2), 1$                               ; 1213
                                                     24  BB 00046         PUSHR   #^M<R2,R5>                                  ; 1215
                                       67            02  FB 00048         CALLS   #2, SYS$DISCONNECT                          ;
                                  05   A2            08  88 0004B         BISB2   #8, 5(R2)                                   ; 1216
                                                     24  BB 0004F         PUSHR   #^M<R2,R5>                                  ; 1218
                                       68            02  FB 00051         CALLS   #2, SYS$CONNECT                             ;
                                                     28  11 00054         BRB     2$                                          ;
                                                     24  BB 00056 1$:     PUSHR   #^M<R2,R5>                                  ; 1223
                            00000000G  00            02  FB 00058         CALLS   #2, SYS$READ                                ;
                                       53            50  D0 0005F         MOVL    R0, STATUS                                  ;
                            0001827A   8F            53  D1 00062         CMPL    STATUS, #98938                              ;
                                                     3F  13 00069         BEQL    5$                                          ;
                                       41            53  E9 0006B         BLBC    STATUS, 6$                                  ; 1225
                                       6E       22   A2  3C 0006E         MOVZWL  34(R2), DESC                                ; 1226
                                  04   AE       28   A2  D0 00072         MOVL    40(R2), DESC+4                              ; 1227
                                              4010   8F  BB 00077         PUSHR   #^M<R4,SP>                                  ; 1229
                                  66               02 FB 0007B          CALLS   #2, WRITE_SLAVE                  ;
                                  D5               50 E8 0007E 2$:      BLBS    STATUS, 1$                       ;
                                                   04 00081             RET                                      ;
                 10          05   A2               03 E1 00082 3$:      BBC     #3, 5(R2), 4$                    ; 1237
                                                   24 BB 00087          PUSHR   #^M<R2,R5>                       ; 1239
                                  67               02 FB 00089          CALLS   #2, SYS$DISCONNECT               ;
                             05   A2               08 8A 0008C          BICB2   #8, 5(R2)                        ; 1240
                                                   24 BB 00090          PUSHR   #^M<R2,R5>                       ; 1242
                                  68               02 FB 00092          CALLS   #2, SYS$CONNECT                  ;
                                                   51 11 00095          BRB     10$                              ;
                                                   24 BB 00097 4$:      PUSHR   #^M<R2,R5>                       ; 1244
                      00000000G   00               02 FB 00099          CALLS   #2, SYS$GET                      ;
                                  53               50 D0 000A0          MOVL    R0, STATUS                       ;
                      0001827A    8F               53 D1 000A3          CMPL    STATUS, #98938                   ;
                                                   40 13 000AA 5$:      BEQL    11$                              ;
                                  04               53 E8 000AC          BLBS    STATUS, 7$                       ; 1246
                                  50               53 D0 000AF 6$:      MOVL    STATUS, R0                       ;
                                                   04 000B2             RET                                      ;
                        00FF      8F         22    A2 B1 000B3 7$:      CMPW    34(R2), #255                     ; 1247
                                                   12 1B 000B9          BLEQU   8$                               ;
                                  7E         22    A2 3C 000BB          MOVZWL  34(R2), -(SP)                    ; 1248
                                       000181A8    8F DD 000BF          PUSHL   #98728                           ;
                      00000000G   00               02 FB 000C5          CALLS   #2, LIB$SIGNAL                   ;
                                                   04 000CC             RET                                      ;
                                  6E         22    A2 3C 000CD 8$:      MOVZWL  34(R2), DESC                     ; 1249
                             04   AE         28    A2 D0 000D1          MOVL    40(R2), DESC+4                   ; 1250
                                  01         22    A2 B1 000D6          CMPW    34(R2), #1                       ; 1251
                                                   05 12 000DA          BNEQ    9$                               ;
                                             28    B2 95 000DC          TSTB    @40(R2)                          ; 1252
                                                   B6 13 000DF          BEQL    4$                               ;
                                           4010    8F BB 000E1 9$:      PUSHR   #^M<R4,SP>                       ; 1254
                                  66               02 FB 000E5          CALLS   #2, WRITE_SLAVE                  ;
                                  AC               50 E8 000E8 10$:     BLBS    STATUS, 4$                       ;
                                                   04 000EB             RET                                      ;
                                  6E               01 D0 000EC 11$:     MOVL    #1, DESC                         ; 1257
                             04   AE         02    AE 9E 000EF          MOVAB   DESC+2, DESC+4                   ; 1258
                                           4010    8F BB 000F4          PUSHR   #^M<R4,SP>                       ; 1259
                                  66               02 FB 000F8          CALLS   #2, WRITE_SLAVE                  ;
                                                   04 000FB             RET                                      ; 1260

; Routine Size:  252 bytes,    Routine Base:  $CODE$ + 0901
```

```
1 GLOBAL ROUTINE MAIL$NET_END_USERS(CNCTDESC) : NOVALUE =
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       Send the end of username flag (byte of 0) and the to-list
1 !       to all the remote nodes that are described by cnctdesc.
1 !
1 ! Inputs:
1 !
1 !       cnctdesc = address of cnct descriptor
1 !
1 !--
2 BEGIN
2 MAP
2     CNCTDESC : REF $BBLOCK;
2
2 LOCAL
2     DESC : VECTOR[2,LONG],
2     LNKDESC : REF $BBLOCK;
2
2 !
2 ! Form a descriptor of a byte of 0
2 !
2 DESC[0] = 1;
2 DESC[1] = DESC[0] + 2;
2 LNKDESC = .(CNCTDESC[CNCT_Q_LNKLST])<0,32,0>;
2 WHILE .LNKDESC NEQ CNCTDESC[CNCT_Q_LNKLST]
3 DO BEGIN
3     IF NOT .LNKDESC[LNK_V_ALTP]               !If sending with decnet
4     THEN BEGIN
4         IF WRITE_SLAVE(.LNKDESC,DESC)    !Send the 1 byte of 0
4             THEN WRITE_SLAVE(.LNKDESC,CNCTDESC[CNCT_Q_TODESC]); !send "to" list
4         END
4     ELSE BEGIN
4         LOCAL
4             NDESC : VECTOR[2,LONG];
4         !
4         ! Send with alternate protocol
4         !
4         IF .LNKDESC[LNK_L_TFRADR] NEQ 0
5         THEN BEGIN
5             NDESC[0] = .LNKDESC[LNK_B_NODLEN];
5             NDESC[1] = LNKDESC[LNK_T_NODE];
5             IF (.LNKDESC[LNK_L_TFRADR])(LNKDESC[LNK_L_CONTEXT],
5                                         LNK_C_OUT_CKUSER,
5                                         NDESC,
5                                         DESC,
5                                         MAIL$READ_ERROR_TEXT)
5                 THEN (.LNKDESC[LNK_L_TFRADR])(LNKDESC[LNK_L_CONTEXT],
5                                         LNK_C_OUT_TO,
5                                         NDESC,
5                                         CNCTDESC[CNCT_Q_TODESC]);
4             END;
3         END;
3     LNKDESC = .LNKDESC[LNK_L_FLINK];
2     END;
2 RETURN;
```

```
; 1191          1318  1 END;




                                         001C 00000          .ENTRY  MAIL$NET_END_USERS, Save R2,R3,R4            ; 1261
                   54        FD56     CF   9E 00002          MOVAB   WRITE_SLAVE, R4                              ;
                   5E                 10   C2 00007          SUBL2   #16, SP                                      ;
              08   AE                 01   D0 0000A          MOVL    #1, DESC                                     ; 1284
              0C   AE          0A     AE   9E 0000E          MOVAB   DESC+2, DESC+4                               ; 1285
                   53          04     AC   D0 00013          MOVL    CNCTDESC, R3                                 ; 1286
                   52          30     A3   D0 00017          MOVL    48(R3), LNKDESC                              ;
                   50          30     A3   9E 0001B 1$:      MOVAB   48(R3), R0                                   ; 1287
                   50                 52   D1 0001F          CMPL    LNKDESC, R0                                  ;
                                      54   13 00022          BEQL    4$                                           ;
        15    2E   A2                 02   E0 00024          BBS     #2, 46(LNKDESC), 2$                          ; 1289
                               08     AE   9F 00029          PUSHAB  DESC                                         ; 1291
                                      52   DD 0002C          PUSHL   LNKDESC                                      ;
                   64                 02   FB 0002E          CALLS   #2, WRITE_SLAVE                              ;
                   3F                 50   E9 00031          BLBC    R0, 3$                                       ;
                               10     A3   9F 00034          PUSHAB  16(R3)                                       ; 1292
                                      52   DD 00037          PUSHL   LNKDESC                                      ;
                   64                 02   FB 00039          CALLS   #2, WRITE_SLAVE                              ;
                                      35   11 0003C          BRB     3$                                           ; 1289
                               10     A2   D5 0003E 2$:      TSTL    16(LNKDESC)                                  ; 1300
                                      30   13 00041          BEQL    3$                                           ;
                   6E          2F     A2   9A 00043          MOVZBL  47(LNKDESC), NDESC                           ; 1302
              04   AE          30     A2   9E 00047          MOVAB   48(R2), NDESC+4                              ; 1303
                        00000000G     00   9F 0004C          PUSHAB  MAIL$READ_ERROR_TEXT                         ; 1304
                               0C     AE   9F 00052          PUSHAB  DESC                                         ;
                               08     AE   9F 00055          PUSHAB  NDESC                                        ;
                                      02   DD 00058          PUSHL   #2                                           ;
                               0C     A2   9F 0005A          PUSHAB  12(LNKDESC)                                  ;
              10   B2                 05   FB 0005D          CALLS   #5, @16(LNKDESC)                             ;
                   0F                 50   E9 00061          BLBC    R0, 3$                                       ;
                               10     A3   9F 00064          PUSHAB  16(R3)                                       ; 1312
                               04     AE   9F 00067          PUSHAB  NDESC                                        ; 1309
                                      03   DD 0006A          PUSHL   #3                                           ; 1312
                               0C     A2   9F 0006C          PUSHAB  12(LNKDESC)                                  ; 1309
              10   B2                 04   FB 0006F          CALLS   #4, @16(LNKDESC)                             ; 1312
                   52                 62   D0 00073 3$:      MOVL    (LNKDESC), LNKDESC                           ; 1315
                                      A3   11 00076          BRB     1$                                           ; 1287
                                           04 00078 4$:      RET                                                  ; 1318

; Routine Size:  121 bytes,    Routine Base:  $CODE$ + 09FD
```

```
   1 GLOBAL ROUTINE MAIL$NET_SEND(ADRDESC,CNCTDESC) =
   1 !++
   1 ! FUNCTIONAL DESCRIPTION:
   1 !
   1 !       Send a message to the remote node.  The complete message is only sent
   1 !       the first time.  After the message is sent, and each additional call
   1 !       for a particular node, only the slave status is checked for each
   1 !       addressee.
   1 !
   1 ! Inputs:
   1 !
   1 !       adrdesc = address of addressee descriptor
   1 !       cnctdesc = address of cnct descriptor
   1 !
   1 !--
   2 BEGIN
   2
   2 MAP
   2     ADRDESC : REF $BBLOCK,
   2     CNCTDESC : REF $BBLOCK;
   2
   2 BIND
   2     LNKDESC = ADRDESC[ADR_L_LLNK] : REF $BBLOCK,
   2     SUBJDESC = CNCTDESC[CNCT_Q_SUBJDESC] : $BBLOCK;
   2
   2 LOCAL
   2     UDESC : VECTOR[2,LONG],
   2     NDESC : VECTOR[2,LONG],
   2     DESC : VECTOR[2,LONG];
   2
   2 IF .LNKDESC[LNK_V_DEAD]
   2     THEN RETURN FALSE;
   2
   2 !
   2 ! If the message hasn't been sent to this node yet, then
   2 ! send it now
   2 !
   2 NDESC[0] = .LNKDESC[LNK_B_NODLEN];
   2 NDESC[1] = LNKDESC[LNK_T_NODE];
   2 UDESC[0] = .ADRDESC[ADR_B_NAMLNG];
   2 UDESC[1] = ADRDESC[ADR_T_NAME];
   2 IF NOT .LNKDESC[LNK_V_MSGSNT]
   3 THEN BEGIN
   3     DESC[0] = .SUBJDESC[DSC$W_LENGTH];
   3     IF .DESC[0] NEQ 0
   3         THEN DESC[1] = .SUBJDESC[DSC$A_POINTER]
   3         ELSE DESC[1] = DESC[0];
   3     IF NOT .LNKDESC[LNK_V_ALTP]          !If sending with decnet
   4     THEN BEGIN
 P 4         IF_ERR(WRITE_SLAVE(.LNKDESC,DESC);,
   4                                     RETURN .STATUS);
   4         !
   4         ! Now send text of message
   4         !
 P 4         IF_ERR(SEND_MESSAGE(.LNKDESC,.CNCTDESC);,
   4                 RETURN .STATUS);
   4         LNKDESC[LNK_V_MSGSNT] = TRUE;
```

```
; 1250          1376  4           END
; 1251          1377  4       ELSE BEGIN
; 1252          1378  4           !
; 1253          1379  4           ! Send with alternate protocol
; 1254          1380  4           !
; 1255          1381  4           IF .LNKDESC[LNK_L_TFRADR] EQL 0
; 1256          1382  4               THEN RETURN TRUE;
; 1257        P 1383  4           IF_ERR((.LNKDESC[LNK_L_TFRADR])(LNKDESC[LNK_L_CONTEXT],
; 1258        P 1384  4                                           LNK_C_OUT_SUBJ,
; 1259        P 1385  4                                           NDESC,
; 1260        P 1386  4                                           DESC);,
; 1261          1387  4                   RETURN .STATUS);
; 1262        P 1388  4           IF_ERR((.LNKDESC[LNK_L_TFRADR])(LNKDESC[LNK_L_CONTEXT],
; 1263        P 1389  4                                           LNK_C_OUT_FILE,
; 1264        P 1390  4                                           NDESC,
; 1265        P 1391  4                                           CNCTDESC[CNCT_T_RAB],
; 1266        P 1392  4                                           UTIL$REPORT_IO_ERROR);,
; 1267          1393  4                   RETURN .STATUS);
; 1268          1394  4           LNKDESC[LNK_V_MSGSNT] = TRUE;
; 1269          1395  3           END;
; 1270          1396  2       END;
; 1271          1397  2   !
; 1272          1398  2   ! See how the send went to this user
; 1273          1399  2   !
; 1274          1400  3 RETURN (IF NOT .LNKDESC[LNK_V_ALTP]
; 1275          1401  3             THEN CHECK_SLAVE_STATUS(.LNKDESC)
; 1276          1402  3             ELSE (.LNKDESC[LNK_L_TFRADR])(LNKDESC[LNK_L_CONTEXT],
; 1277          1403  3                                           LNK_C_OUT_CRSEND,
; 1278          1404  3                                           NDESC,
; 1279          1405  3                                           UDESC,
; 1280          1406  3                                           MAIL$READ_ERROR_TEXT))
; 1281          1407  1 END;



                            001C 00000          .ENTRY  MAIL$NET_SEND, Save R2,R3,R4                ; 1319
          54   FCDD   CF   9E 00002             MOVAB   WRITE_SLAVE, R4                             ;
          5E          18   C2 00007             SUBL2   #24, SP                                     ;
          50     04   AC   D0 0000A             MOVL    ADRDESC, R0                                 ; 1341
 51   08  AC          18   C1 0000E             ADDL3   #24, CNCTDESC, R1                           ; 1342
          52     08   A0   D0 00013             MOVL    8(R0), R2                                   ; 1349
          53     2E   A2   9E 00017             MOVAB   46(R2), R3                                  ;
 03       63          01   E1 0001B             BBC     #1, (R3), 1$                                ;
                    00A0   31 0001F             BRW     9$                                          ;
      08  AE     2F   A2   9A 00022 1$:         MOVZBL  47(R2), NDESC                               ; 1356
      0C  AE     30   A2   9E 00027             MOVAB   48(R2), NDESC+4                             ; 1357
      10  AE     1D   A0   9A 0002C             MOVZBL  29(R0), UDESC                               ; 1358
      14  AE     1E   A0   9E 00031             MOVAB   30(R0), UDESC+4                             ; 1359
          67          63   E8 00036             BLBS    (R3), 7$                                    ; 1360
          6E          61   3C 00039             MOVZWL  (R1), DESC                                  ; 1362
                      07   13 0003C             BEQL    2$                                          ; 1363
      04  AE     04   A1   D0 0003E             MOVL    4(R1), DESC+4                               ; 1364
                      04   11 00043             BRB     3$                                          ;
      04  AE          6E   9E 00045 2$:         MOVAB   DESC, DESC+4                                ; 1365
 18       63          02   E0 00049 3$:         BBS     #2, (R3), 4$                                ; 1366
```

```
                      4004  8F  BB 0004D       PUSHR   #^M<R2,SP>                    ; 1369
              64            02  FB 00051       CALLS   #2, WRITE_SLAVE               ;
              6D            50  E9 00054       BLBC    STATUS, 10$                   ;
                        08  AC  DD 00057       PUSHL   CNCTDESC                      ; 1374
                            52  DD 0005A       PUSHL   R2                            ;
        01A8  C4            02  FB 0005C       CALLS   #2, SEND_MESSAGE              ;
              39            50  E8 00061       BLBS    STATUS, 6$                    ;
                                04 00064       RET                                   ; 1375
                        10  A2  D5 00065 4$:   TSTL    16(R2)                        ; 1381
                            04  12 00068       BNEQ    5$                            ;
              50            01  D0 0006A       MOVL    #1, R0                        ; 1382
                                04 0006D       RET                                   ;
                            5E  DD 0006E 5$:   PUSHL   SP                            ; 1387
                        0C  AE  9F 00070       PUSHAB  NDESC                         ;
                            04  DD 00073       PUSHL   #4                            ;
                        0C  A2  9F 00075       PUSHAB  12(R2)                        ;
          10  B2            04  FB 00078       CALLS   #4, @16(R2)                   ;
              45            50  E9 0007C       BLBC    STATUS, 10$                   ;
                 00000000G  00  9F 0007F       PUSHAB  UTIL$REPORT_IO_ERROR          ; 1393
  7E      08  AC  00000286  8F  C1 00085       ADDL3   #646, CNCTDESC, -(SP)         ;
                        10  AE  9F 0008E       PUSHAB  NDESC                         ;
                            05  DD 00091       PUSHL   #5                            ;
                        0C  A2  9F 00093       PUSHAB  12(R2)                        ;
          10  B2            05  FB 00096       CALLS   #5, @16(R2)                   ;
              27            50  E9 0009A       BLBC    STATUS, 10$                   ;
              63            01  88 0009D 6$:   BISB2   #1, (R3)                      ; 1394
  08          63            02  E0 000A0 7$:   BBS     #2, (R3), 8$                  ; 1400
                            52  DD 000A4       PUSHL   R2                            ; 1401
        00C0  C4            01  FB 000A6       CALLS   #1, CHECK_SLAVE_STATUS        ;
                                04 000AB       RET                                   ;
                 00000000G  00  9F 000AC 8$:   PUSHAB  MAIL$READ_ERROR_TEXT          ; 1402
                        14  AE  9F 000B2       PUSHAB  UDESC                         ;
                        10  AE  9F 000B5       PUSHAB  NDESC                         ;
                            06  DD 000B8       PUSHL   #6                            ;
                        0C  A2  9F 000BA       PUSHAB  12(R2)                        ;
          10  B2            05  FB 000BD       CALLS   #5, @16(R2)                   ;
                                04 000C1       RET                                   ; 1400
                            50  D4 000C2 9$:   CLRL    R0                            ; 1407
                                04 000C4 10$:  RET                                   ;

; Routine Size:  197 bytes,    Routine Base:  $CODE$ + 0A76
; 1283          1408  1 GLOBAL ROUTINE MAIL$READ_FOREIGN_FILE(OUTRAB) =
; 1284          1409  1 !++
; 1285          1410  1 ! FUNCTIONAL DESCRIPTION:
; 1286          1411  1 !
; 1287          1412  1 !       Calls a foreign net protocol routine to read message text
; 1288          1413  1 !       from the remote node and store it in the output file
; 1289          1414  1 !
; 1290          1415  1 !--
; 1291          1416  2 BEGIN
; 1292          1417  2 MAP
; 1293          1418  2     OUTRAB : $BBLOCK;
; 1294          1419  2
; 1295          1420  2 RETURN (.LINK_TFRADR)(LINK_CONTEXT,LNK_C_IN_FILE,0,.OUTRAB,
; 1296          1421  2                         UTIL$REPORT_IO_ERROR)
; 1297          1422  1 END;



                                  0000 00000          .ENTRY  MAIL$READ_FOREIGN_FILE, Save nothing              ; 1408
                 50 00000000'  00  D0 00002          MOVL    LINK_TFRADR, R0                                   ; 1420
                    00000000G  00  9F 00009          PUSHAB  UTIL$REPORT_IO_ERROR                              ;
                           04  AC  DD 0000F          PUSHL   OUTRAB                                            ;
                 7E            0D  7D 00012          MOVQ    #13, -(SP)                                        ;
                    00000000'  00  9F 00015          PUSHAB  LINK_CONTEXT                                      ;
                 60            05  FB 0001B          CALLS   #5, (R0)                                          ;
                                   04 0001E          RET                                                       ; 1422

; Routine Size:  31 bytes,    Routine Base:  $CODE$ + 0B3B
```

```
1 ROUTINE ACCEPT_LINK =
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       Accept a connection from a remote node
1 !
1 ! Inputs:
1 !
1 !       none
1 !
1 ! Implicit inputs:
1 !
1 !       network server data base and own storage
1 !--
2 BEGIN
2 LOCAL
2     STATUS,
2     RMSRAT,
2     RMSRFM,
2     PFLAGS,
2     PTR : REF VECTOR[,BYTE],
2     PTR1 : REF $BBLOCK,
2     LEN,
2     QIOSB : VECTOR[4,WORD];
2
2 LINK_CHAN = 0;
2 MAIL$G_CNCT[CNCT_B_FILRAT] = FAB$M_CR;
2 MAIL$G_CNCT[CNCT_B_FILRFM] = FAB$C_VAR;
2 IF .MAIL$Q_PROTOCOL[DSC$W_LENGTH] EQL 0
3 THEN BEGIN
3     !
3     ! See if SYS$NET translated is a DECnet NCB.  If so, decode the
3     ! NCB and store in CNCT
3     !
3     IF NOT CH$FAIL(PTR = CH$FIND_CH(.MAIL$Q_INPTRAN[DSC$W_LENGTH],
3                         .MAIL$Q_INPTRAN[DSC$A_POINTER],%C'/'))
4     THEN BEGIN
4         LEN = .PTR - .MAIL$Q_INPTRAN[DSC$A_POINTER] - 4;        !"/", word, cnt count
4         PTR = .PTR + 3;                                        !Skip to cnf count
4         PTR1 = PTR[1];                                         !PTR1 points to cnfdata
4         IF .LEN-CNF_C_LENGTH GEQU 0
4             AND .PTR[0] EQL CNF_C_LENGTH
4             AND .PTR1[CNF_B_VERSION] GEQU CNF_C_VERS
4             AND .PTR1[CNF_B_ECO] GEQU CNF_C_ECO
5         THEN BEGIN
5             !
5             ! It seems to be a valid CNF from another MAIL.  Store away the
5             !  info and modify accordingly
5             !
5             PTR1[CNF_V_PFXSEND] = 0;            !Clear his bit
5             PTR1[CNF_V_CCSEND] = 0;             !Clear his bit
5             PTR1[CNF_B_VERSION] = CNF_C_VERS;   !Send back our protocol ver
5             PTR1[CNF_B_ECO] = CNF_C_ECO;        ! and eco level
5             IF .PTR1[CNF_V_BLKSEND]             !Sending block mode?
6             THEN BEGIN
6                 PTR1[CNF_V_BLKSEND] = 0;        !Clear his send bit
6                 PTR1[CNF_V_BLKRECV] = 1;        !Set my receive bit
```

```
; 1356          1480  6                 MAIL$G_CNCT[CNCT_B_FILRFM] = .PTR1[CNF_B_RFM];
; 1357          1481  6                 MAIL$G_CNCT[CNCT_B_FILRAT] = .PTR1[CNF_B_RAT];
; 1358          1482  6                 MAIL$G_CNCT[CNCT_V_BLKMODE] = TRUE;
; 1359          1483  6                 PTR1[CNF_B_RFM] = 1;                !Will send 1 block at a time
; 1360          1484  5                 END;
; 1361          1485  4             END;
; 1362          1486  3         END;
; 1363          1487  3     !
; 1364          1488  3     ! Assign a channel to _NET:.  Then attempt to accept the connection.
; 1365          1489  3     ! If that fails, then give up.
; 1366          1490  3     !
; 1367          1491  4     IF (STATUS = LIB$ASN_WTH_MBX(NETACP_DESC,
; 1368          1492  4                             MAIL$L_MBXBUF,MAIL$L_MBXQUO,LINK_CHAN,
; 1369          1493  4                             NETMBX_CHAN))
; 1370        P 1494  5         AND ((STATUS = $QIOW(FUNC=IO$_ACCESS,
; 1371        P 1495  5                         CHAN=.LINK_CHAN,
; 1372        P 1496  5                         IOSB=QIOSB,
; 1373          1497  5                         P2=MAIL$Q_INPTRAN))
; 1374          1498  4             AND (STATUS = .QIOSB[0]))
; 1375          1499  4     THEN ($DASSGN(CHAN=.NETMBX_CHAN); RETURN TRUE)             !All done if DECnet
; 1376          1500  4     ELSE BEGIN
; 1377          1501  4         $DASSGN(CHAN=.NETMBX_CHAN);
; 1378          1502  4         MAIL$G_CNCT[CNCT_V_B[KMODE] = FALSE;
; 1379          1503  4         RETURN .STATUS;
; 1380          1504  3         END;
; 1381          1505  3     END
; 1382          1506  3 ELSE BEGIN
; 1383          1507  3     !
; 1384          1508  3     ! The /protocol qualifier was used in starting up inbound network
; 1385          1509  3     ! mail.  Merge in the specified file and use it.
; 1386          1510  3     !
; 1387          1511  3     PFLAGS = 0;
; 1388        P 1512  3     IF_ERR(LIB$FIND_IMAGE_SYMBOL(MAIL$Q_PROTOCOL,PROT_DESC,LINK_TFRADR);,
; 1389          1513  3                 RETURN .STATUS);
; 1390        P 1514  3     IF_ERR(CHECK_PROTOCOL_VERSION(MAIL$Q_PROTOCOL);,
; 1391          1515  3                 RETURN .STATUS);
; 1392        P 1516  3     IF_ERR((.LINK_TFRADR)(LINK_CONTEXT,
; 1393        P 1517  3                                         LNK_C_IN_CONNECT,
; 1394        P 1518  3                                         MAI[$Q_INPTRAN,
; 1395        P 1519  3                                         RMSRAT,RMSRFM,
; 1396        P 1520  3                                         .MAIL$GL_SYSFLAGS<16,16,0>,
; 1397        P 1521  3                                         MAIL$Q_PROTOCOL,
; 1398        P 1522  3                                         PFLAGS);,
; 1399          1523  3                 RETURN .STATUS);
; 1400          1524  3     LINK_CHAN = MAIL$GL_FLAGS[MAIF_V_ALTP] = 1;
; 1401          1525  3     MAIL$G_CNCT[CNCT_B_FILRFM] = .RMSRFM;
; 1402          1526  3     MAIL$G_CNCT[CNCT_B_FILRAT] = .RMSRAT;
; 1403          1527  3     MAIL$G[_FLAGS[MAIF_V_SERVERLOOP] = .PFLAGS<0,1,0>;
; 1404          1528  3     RETURN TRUE
; 1405          1529  2     END;
; 1406          1530  1 END;




                                                01FC 00000 ACCEPT_LINK:
                                                  .WORD    Save R2,R3,R4,R5,R6,R7,R8                ; 1423
               58 00000000G  00  9E 00002         MOVAB    SYS$DASSGN, R8                           ;
               57      F4C9  CF  9E 00009         MOVAB    NETACP_DESC, R7                          ;
               56 00000000G  00  9E 0000E         MOVAB    MAIL$Q_INPTRAN, R6                       ;
               55 00000000G  00  9E 00015         MOVAB    MAIL$Q_PROTOCOL, R5                      ;
               54 00000000G  00  9E 0001C         MOVAB    MAIL$G_CNCT+128, R4                      ;
               53 00000000'  00  9E 00023         MOVAB    LINK_CHAN, R3                            ;
               5E            14  C2 0002A         SUBL2    #20, SP                                  ;
                             63  D4 0002D         CLRL     LINK_CHAN                                ; 1448
               64      0202  8F  B0 0002F         MOVW     #514, MAIL$G_CNCT+128                    ; 1449
                             65  B5 00034         TSTW     MAIL$Q_PROTOCOL                          ; 1451
                             03  13 00036         BEQL     1$                                       ;
                           00AE  31 00038         BRW      5$                                       ;
               52        04  A6  D0 0003B 1$:     MOVL     MAIL$Q_INPTRAN+4, R2                     ; 1458
62             66            2F  3A 0003F         LOCC     #47, MAIL$Q_INPTRAN, (R2)                ; 1457
                             02  12 00043         BNEQ     2$                                       ;
                             51  D4 00045         CLRL     R1                                       ;
               50            51  D0 00047 2$:     MOVL     R1, PTR                                  ;
                             40  13 0004A         BEQL     3$                                       ; 1458
51             50            52  C3 0004C         SUBL3    R2, PTR, R1                              ; 1460
               52        FC  A1  9E 00050         MOVAB    -4(R1), LEN                              ;
               50            03  C0 00054         ADDL2    #3, PTR                                  ; 1461
               51        01  A0  9E 00057         MOVAB    1(R0), PTR1                              ; 1462
               52            10  C2 0005B         SUBL2    #16, R2                                  ; 1463
               10            60  91 0005E         CMPB     (PTR), #16                               ; 1464
                             29  12 00061         BNEQ     3$                                       ;
               03            61  91 00063         CMPB     (PTR1), #3                               ; 1465
                             24  1F 00066         BLSSU    3$                                       ;
               50        08  A1  9E 00068         MOVAB    8(PTR1), R0                              ; 1472
               60            14  8A 0006C         BICB2    #20, (R0)                                ; 1473
               61            03  B0 0006F         MOVW     #3, (PTR1)                               ; 1474
               17            60  E9 00072         BLBC     (R0), 3$                                 ; 1476
               60            01  8A 00075         BICB2    #1, (R0)                                 ; 1478
               60            02  88 00078         BISB2    #2, (R0)                                 ; 1479
           01  A4        0C  A1  90 0007B         MOVB     12(PTR1), MAIL$G_CNCT+129                ; 1480
               64        0D  A1  90 00080         MOVB     13(PTR1), MAIL$G_CNCT+128                ; 1481
           04  A4            04  88 00084         BISB2    #4, MAIL$G_CNCT+132                      ; 1482
           0C  A1            01  90 00088         MOVB     #1, 12(PTR1)                             ; 1483
                         0C  A3  9F 0008C 3$:     PUSHAB   NETMBX_CHAN                              ; 1491
                             53  DD 0008F         PUSHL    R3                                       ;
                  00000000'  00  9F 00091         PUSHAB   MAIL$L_MBXQUO                            ;
                  00000000'  00  9F 00097         PUSHAB   MAIL$L_MBXBUF                            ;
                             57  DD 0009D         PUSHL    R7                                       ;
    00000000G  00            05  FB 0009F         CALLS    #5, LIB$ASN_WTH_MBX                      ;
               52            50  D0 000A6         MOVL     R0, STATUS                               ;
               2F            52  E9 000A9         BLBC     STATUS, 4$                               ;
                             7E  7C 000AC         CLRQ     -(SP)                                    ; 1497
                             7E  7C 000AE         CLRQ     -(SP)                                    ;
                             56  DD 000B0         PUSHL    R6                                       ;
                             7E  7C 000B2         CLRQ     -(SP)                                    ;
                             7E  D4 000B4         CLRL     -(SP)                                    ;
                         2C  AE  9F 000B6         PUSHAB   QIOSB                                    ;
                             32  DD 000B9         PUSHL    #50                                      ;
                             63  DD 000BB         PUSHL    LINK_CHAN                                ;
                             7E  D4 000BD         CLRL     -(SP)                                    ;
    00000000G  00            0C  FB 000BF         CALLS    #12, SYS$QIOW                            ;
               52            50  D0 000C6         MOVL     R0, STATUS                               ;
```

```
                                        0F            52  E9 000C9              BLBC    STATUS, 4$                                   ;
                                        52       0C   AE  3C 0C0CC              MOVZWL  QIOSB, STATUS                                ; 1498
                                        08            52  E9 000D0              BLBC    STATUS, 4$                                   ;
                                                 0C   A3  DD 000D3              PUSHL   NETMBX_CHAN                                  ; 1499
                                        68            01  FB 000D6              CALLS   #1, SYS$DASSGN                               ;
                                                      6A  11 000D9              BRB     6$                                           ; 1500
                                                 0C   A3  DD 000DB 4$:          PUSHL   NETMBX_CHAN                                  ; 1501
                                        68            01  FB 000DE              CALLS   #1, SYS$DASSGN                               ;
                                  04    A4            04  8A 000E1              BICB2   #4, MAIL$G_CNCT+132                          ; 1502
                                        50            52  D0 000E5              MOVL    STATUS, R0                                   ; 1503
                                                          04 000E8              RET                                                  ; 1506
                                                      6E  D4 000E9 5$:          CLRL    PFLAGS                                       ; 1511
                                                 04   A3  9F 000EB              PUSHAB  LINK_TFRADR                                  ; 1513
                                                 E0   A7  9F 000EE              PUSHAB  PROT_DESC                                    ;
                                                      55  DD 000F1              PUSHL   R5                                           ;
                            00000000G   00            03  FB 000F3              CALLS   #3, LIB$FIND_IMAGE_SYMBOL                    ;
                                        4B            50  E9 000FA              BLBC    STATUS, 7$                                   ;
                                                      55  DD 000FD              PUSHL   R5                                           ; 1515
                                F6A2    CF            01  FB 000FF              CALLS   #1, CHECK_PROTOCOL_VERSION                   ;
                                        41            50  E9 00104              BLBC    STATUS, 7$                                   ;
                                        50       04   A3  D0 00107              MOVL    LINK_TFRADR, R0                              ; 1523
                                               4020   8F  BB 0010B              PUSHR   #^M<R5,SP>                                   ;
                                        7E 00000000G  00  3C 0010F              MOVZWL  MAIL$GL_SYSFLAGS+2, -(SP)                    ;
                                                 10   AE  9F 00116              PUSHAB  RMSRFM                                       ;
                                                 18   AE  9F 00119              PUSHAB  RMSRAT                                       ;
                                                      56  DD 0011C              PUSHL   R6                                           ;
                                                      08  DD 0011E              PUSHL   #8                                           ;
                                                 08   A3  9F 00120              PUSHAB  LINK_CONTEXT                                 ;
                                        60            08  FB 00123              CALLS   #8, (R0)                                     ;
                                        1F            50  E9 00126              BLBC    STATUS, 7$                                   ;
                            00000000G   00            04  88 00129              BISB2   #4, MAIL$GL_FLAGS+1                          ; 1524
                                        63            01  D0 00130              MOVL    #1, LINK_CHAN                                ;
                                  01    A4       04   AE  90 00133              MOVB    RMSRFM, MAIL$G_CNCT+129                      ; 1525
                                        64       08   AE  90 00138              MOVB    RMSRAT, MAIL$G_CNCT+128                      ; 1526
00000000G  00            01             02            6E  F0 0013C              INSV    PFLAGS, #2, #1, MAIL$GL_FLAGS+2              ; 1527
                                        50            01  D0 00145 6$:          MOVL    #1, R0                                       ; 1528
                                                          04 00148 7$:          RET                                                  ; 1530

; Routine Size:  329 bytes,    Routine Base:  $CODE$ + 0B5A
; 1408          1531   1 GLOBAL ROUTINE MAIL$GET_INPUT (OUT_DESC,PROMPT_DESC,OUTLEN) =
; 1409          1532   1 !++
; 1410          1533   1 ! FUNCTIONAL DESCRIPTION:
; 1411          1534   1 !
; 1412          1535   1 !       If non-network, read from SYS$INPUT.  If network, read from
; 1413          1536   1 !       network link
; 1414          1537   1 !
; 1415          1538   1 ! Inputs:
; 1416          1539   1 !
; 1417          1540   1 !       out_desc = address of dynamic desciptor for output string
; 1418          1541   1 !       prompt_desc = address of prompt descriptor
; 1419          1542   1 !
; 1420          1543   1 !--
; 1421          1544   2 BEGIN
; 1422          1545   2
; 1423          1546   2 MAP
; 1424          1547   2     OUTLEN : REF VECTOR[,WORD];
; 1425          1548   2
; 1426          1549   2 BUILTIN
; 1427          1550   2     NULLPARAMETER;
; 1428          1551   2
; 1429          1552   2 LOCAL
; 1430          1553   2     TEMPLEN : WORD,
; 1431          1554   2     STATUS;
; 1432          1555   2
; 1433          1556   2 BIND
; 1434          1557   2     QIOSB = MAIL$G_CNCT[CNCT_Q_IOSB] : VECTOR[,WORD];
; 1435          1558   2
; 1436          1559   2 IF .MAIL$GL_FLAGS[MAIF_V_NETJOB]
; 1437          1560   3 THEN BEGIN
; 1438          1561   3     !
; 1439          1562   3     ! Accept the link if it hasn't been already.
; 1440          1563   3     !
; 1441          1564   3     IF .LINK_CHAN EQL 0
; 1442        P 1565   3         THEN IF_ERR(ACCEPT_LINK();,
; 1443          1566   3                         RETURN .STATUS);
; 1444          1567   3     IF NOT .MAIL$GL_FLAGS[MAIF_V_ALTP]
; 1445          1568   4     THEN BEGIN
; 1446          1569   4         !
; 1447          1570   4         ! For decnet, read the buffer.  Then copy to the output buffer
; 1448          1571   4         !
; 1449        P 1572   4         STATUS = $QIOW(CHAN=.LINK_CHAN,
; 1450        P 1573   4                         FUNC=IO$_READVBLK,
; 1451        P 1574   4                         IOSB=QIOSB,
; 1452        P 1575   4                         P1=MAIL$G_CNCT[CNCT_T_BUFFER],
; 1453          1576   4                         P2=MAIL$K_INBUFFSZ);
; 1454          1577   4         IF .STATUS
; 1455          1578   4             THEN STATUS = .QIOSB[0];
; 1456          1579   4         IF NOT .STATUS
; 1457          1580   4             THEN RETURN .STATUS;
; 1458          1581   4         LIB$SCOPY_R_DX(QIOSB[1],MAIL$G_CNCT[CNCT_T_BUFFER],.OUT_DESC);
; 1459          1582   4         RETURN TRUE
; 1460          1583   4         END
; 1461          1584   4     !
; 1462          1585   4     ! For foreign net, let it's routine do the copy, too
; 1463          1586   4     !
; 1464          1587   3     ELSE RETURN (.LINK_TFRADR)(LINK_CONTEXT,.PROMPT_DESC,.OUT_DESC);
; 1465          1588  3     END
; 1466          1589  3 ELSE BEGIN
; 1467          1590  3     !
; 1468          1591  3     ! Not network job.
; 1469          1592  3     !
; 1470          1593  3     STATUS = SMG$READ_COMPOSED_LINE(MAIL$L_SMG_KEYBOARD,
; 1471          1594  3                                                   MAIL$L_SMG_KEYTABLE,
; 1472          1595  3                                                   .OUT_DESC,.PROMPT_DESC,TEMPLEN);
; 1473          1596  3     IF .STATUS EQL SMG$_EOF
; 1474          1597  3         THEN STATUS = RMS$_EOF;
; 1475          1598  4     IF (.STATUS EQL RMS$_TNS)
; 1476          1599  4         OR (.STATUS EQL SS$_DATAOVERUN)
; 1477          1600  3         THEN STATUS = SS$_NORMAL;
; 1478          1601  3     IF .MAIL$GL_FLAGS[MAIF_V_CTRLCFL]
; 1479          1602  4         THEN (STATUS = RMS$_EOF;
; 1480          1603  3                  MAIL$GL_FLAGS[MAIF_V_CTRLCFL] = 0);
; 1481          1604  3     IF NOT NULLPARAMETER(3)
; 1482          1605  3         AND .STATUS
; 1483          1606  3         THEN OUTLEN[0] = .TEMPLEN;
; 1484          1607  3     IF NOT .STATUS
; 1485          1608  4         AND (.STATUS NEQ RMS$_EOF)
; 1486          1609  3         THEN SIGNAL(MAIL$_READERR,1,MAIL$Q_INPTRAN,.STATUS);
; 1487          1610  3     RETURN .STATUS
; 1488          1611  2     END;
; 1489          1612  2
; 1490          1613  1 END;
```

```
                                                        003C 00000          .ENTRY  MAIL$GET_INPUT, Save R2,R3,R4,R5            ; 1531
                                        55 00000000G  00  9E 00002          MOVAB   MAIL$GL_FLAGS, R5                           ;
                                        54 00000000'  00  9E 00009          MOVAB   LINK_CHAN, R4                               ;
                                        53 00000000G  00  9E 00010          MOVAB   MAIL$G_CNCT+134, R3                         ;
                                        5E            04  C2 00017          SUBL2   #4, SP                                      ;
                         63             65            01  E1 0001A          BBC     #1, MAIL$GL_FLAGS, 5$                       ; 1559
                                                      64  D5 0001E          TSTL    LINK_CHAN                                   ; 1564
                                                      09  12 00020          BNEQ    1$                                          ;
                                 FE90   CF            00  FB 00022          CALLS   #0, ACCEPT_LINK                             ; 1566
                                        01            50  E8 00027          STATUS, 1$                                          ;
                                                          04 0002A          RET                                                 ;
                         40        01   A5            02  E0 0002B 1$:      BBS     #2, MAIL$GL_FLAGS+1, 4$                     ; 1567
                                                      7E  7C 00030          CLRQ    -(SP)                                       ; 1576
                                                      7E  7C 00032          CLRQ    -(SP)                                       ;
                                        7E     0200   8F  3C 00034          MOVZWL  #512, -(SP)                                 ;
                                                      53  DD 00039          PUSHL   R3                                          ;
                                                      7E  7C 0003B          CLRQ    -(SP)                                       ;
                                                 BA   A3  9F 0003D          PUSHAB  QIOSB                                       ;
                                                      31  DD 00040          PUSHL   #49                                         ;
                                                      64  DD 00042          PUSHL   LINK_CHAN                                   ;
                                                      7E  D4 00044          CLRL    -(SP)                                       ;
                             00000000G  00            0C  FB 00046          CALLS   #12, SYS$QIOW                               ;
                                        52            50  D0 0004D          MOVL    R0, STATUS                                  ;
                                        04            52  E9 00050          BLBC    STATUS, 2$                                  ; 1577
                                        52       BA   A3  3C 00053          MOVZWL  QIOSB, STATUS                               ; 1578
                                        03            52  E8 00057 2$:      BLBS    STATUS, 3$                                  ; 1579
```

```
                                                   00A8 31 0005A         BRW     12$                                       ;
                                               04     AC DD 0005D 3$:     PUSHL   OUT_DESC                                  ;   1581
                                                      53 DD 00060         PUSHL   R3                                        ;
                                               BC     A3 9F 00062         PUSHAB  QIOSB+2                                   ;
                            00000000G 00              03 FB 00065         CALLS   #3, LIB$SCOPY_R_DX                        ;
                                      50              01 D0 0006C         MOVL    #1, R0                                    ;   1587
                                                         04 0006F         RET                                               ;
                                      50       04     A4 D0 00070 4$:     MOVL    LINK_TFRADR, R0                           ;
                                               04     AC DD 00074         PUSHL   OUT_DESC                                  ;
                                               08     AC DD 00077         PUSHL   PROMPT_DESC                               ;
                                               08     A4 9F 0007A         PUSHAB  LINK_CONTEXT                              ;
                                      60              03 FB 0007D         CALLS   #3, (R0)                                  ;
                                                         04 00080         RET                                               ;   1589
                                                      5E DD 00081 5$:     PUSHL   SP                                        ;   1593
                                      7E       04     AC 7D 00083         MOVQ    OUT_DESC, -(SP)                           ;   1595
                                         00000000G    00 9F 00087         PUSHAB  MAIL$L_SMG_KEYTABLE                       ;   1593
                                         00000000G    00 9F 0008D         PUSHAB  MAIL$L_SMG_KEYBOARD                       ;
                            00000000G 00              05 FB 00093         CALLS   #5, SMG$READ_COMPOSED_LINE                ;
                                      52              50 D0 0009A         MOVL    R0, STATUS                                ;
                            00000000G 8F              52 D1 0009D         CMPL    STATUS, #SMG$_EOF                         ;   1596
                                                      07 12 000A4         BNEQ    6$                                        ;
                                      52 0001827A     8F D0 000A6         MOVL    #98938, STATUS                            ;   1597
                            000181B8  8F              52 D1 000AD 6$:     CMPL    STATUS, #98744                            ;   1598
                                                      09 13 000B4         BEQL    7$                                        ;
                            00000838  8F              52 D1 000B6         CMPL    STATUS, #2104                             ;   1599
                                                      03 12 000BD         BNEQ    8$                                        ;
                                      52              01 D0 000BF 7$:     MOVL    #1, STATUS                                ;   1600
                                      0B       01     A5 E9 000C2 8$:     BLBC    MAIL$GL_FLAGS+1, 9$                       ;   1601
                                      52 0001827A     8F D0 000C6         MOVL    #98938, STATUS                            ;   1602
                                  01  A5              01 8A 000CD         BICB2   #1, MAIL$GL_FLAGS+1                       ;   1603
                                      03              6C 91 000D1 9$:     CMPB    (AP), #3                                  ;   1604
                                                      0C 1F 000D4         BLSSU   10$                                       ;
                                               0C     AC D5 000D6         TSTL    12(AP)                                    ;
                                                      07 13 000D9         BEQL    10$                                       ;
                                      07              52 E9 000DB         BLBC    STATUS, 11$                               ;   1605
                                  0C  BC              6E B0 000DE         MOVW    TEMPLEN, @OUTLEN                          ;   1606
                                      20              52 E8 000E2 10$:    BLBS    STATUS, 12$                               ;   1607
                            0001827A  8F              52 D1 000E5 11$:    CMPL    STATUS, #98938                            ;   1608
                                                      17 13 000EC         BEQL    12$                                       ;
                                                      52 DD 000EE         PUSHL   STATUS                                    ;   1609
                                         00000000G    00 9F 000F0         PUSHAB  MAIL$Q_INPTRAN                            ;
                                                      01 DD 000F6         PUSHL   #1                                        ;
                                         007E10B2     8F DD 000F8         PUSHL   #8261810                                  ;
                            00000000G 00              04 FB 000FE         CALLS   #4, LIB$SIGNAL                            ;
                                      50              52 D0 00105 12$:    MOVL    STATUS, R0                                ;   1610
                                                         04 00108         RET                                               ;   1613

; Routine Size:  265 bytes,    Routine Base:  $CODE$ + 0CA3
```

```
  1 GLOBAL ROUTINE MAIL$PUT_OUTPUT(BUFDESC,FAOARGS) =
  1 !++
  1 ! FUNCTIONAL DESCRIPTION:
  1 !
  1 !       Write a record to sys$output (or sys$net if network server)
  1 !
  1 ! Inputs:
  1 !
  1 !       bufdesc = address of string to output or fao control string
  1 !       faoargs = start of fao args if bufdesc is an fao control string
  1 !               for fao strings which take no args, use a 0 for faoargs
  1 !
  1 ! If 2 or more arguments are passed, bufdesc is assumed to be an fao control
  1 ! string, and is processed as such
  1 !
  1 !--
  2 BEGIN
  2 BUILTIN
  2     ACTUALCOUNT;
  2
  2 LOCAL
  2     TMPBUF : $BBLOCK[MAIL$K_INBUFFSZ],
  2     STATUS,
  2     QIOSB : VECTOR[4,WORD],
  2     OUTDESC : REF $BBLOCK,
  2     DESC : VECTOR[2,LONG];
  2
  2 OUTDESC = .BUFDESC;
  2 IF ACTUALCOUNT() GEQU 2
  3 THEN BEGIN
  3     DESC[0] = MAIL$K_INBUFFSZ;
  3     DESC[1] = TMPBUF;
P 3     $FAOL(CTRSTR=.OUTDESC,OUTLEN=DESC,
  3         OUTBUF=DESC,PRMLST=FAOARGS);
  3     OUTDESC = DESC;
  2     END;
  2 IF NOT .MAIL$GL_FLAGS[MAIF_V_NETJOB]
  2 THEN RETURN LIB$PUT_OUTPUT(.OUTDESC)
  3 ELSE BEGIN
  3     IF .LINK_CHAN EQL 0
P 3         THEN IF_ERR(ACCEPT_LINK();,
  3                         RETURN .STATUS);
  3     IF NOT .MAIL$GL_FLAGS[MAIF_V_ALTP]
  4     THEN BEGIN
P 4         STATUS = $QIOW(CHAN=.LINK_CHAN,
P 4                         FUNC=IO$_WRITEVBLK,
P 4                         IOSB=QIOSB,
P 4                         P1=.OUTDESC[DSC$A_POINTER],
  4                         P2=.OUTDESC[DSC$W_LENGTH]);
  4         IF .STATUS
  4             THEN STATUS = .QIOSB[0];
  4         RETURN .STATUS
  4         END
  3     ELSE RETURN (.LINK_TFRADR)(LINK_CONTEXT,LNK_C_IO_WRITE,.OUTDESC);
  2     END;
  1 END;
```

```
                                                     000C 00000        .ENTRY   MAIL$PUT_OUTPUT, Save R2,R3             ; 1614
                                     53 00000000'  00  9E 00002        MOVAB    LINK_CHAN, R3                           ;
                                     5E      FDF0  CE  9E 00009        MOVAB    -528(SP), SP                            ;
                                     52        04  AC  D0 0000E        MOVL     BUFDESC, OUTDESC                        ; 1641
                                     02            6C  91 00012        CMPB     (AP), #2                                ; 1642
                                                   1F  1F 00015        BLSSU    1$                                      ;
                                     6E      0200  8F  3C 00017        MOVZWL   #512, DESC                              ; 1644
                                  04 AE        10  AE  9E 0001C        MOVAB    TMPBUF, DESC+4                          ; 1645
                                               08  AC  9F 00021        PUSHAB   FAOARGS                                 ; 1647
                                               04  AE  9F 00024        PUSHAB   DESC                                    ;
                                               08  AE  9F 00027        PUSHAB   DESC                                    ;
                                                   52  DD 0002A        PUSHL    OUTDESC                                 ;
                           00000000G 00            04  FB 0002C        CALLS    #4, SYS$FAOL                            ;
                                     52            6E  9E 00033        MOVAB    DESC, OUTDESC                           ; 1648
                        0A 00000000G 00            01  E0 00036 1$:    BBS      #1, MAIL$GL_FLAGS, 2$                   ; 1650
                                                   52  DD 0003E        PUSHL    OUTDESC                                 ; 1651
                           00000000G 00            01  FB 00040        CALLS    #1, LIB$PUT_OUTPUT                      ;
                                                       04 00047        RET                                              ; 1652
                                                   63  D5 00048 2$:    TSTL     LINK_CHAN                               ; 1653
                                                   08  12 0004A        BNEQ     3$                                      ;
                                FD5D CF            00  FB 0004C        CALLS    #0, ACCEPT_LINK                         ; 1655
                                     3A            50  E9 00051        BLBC     STATUS, 5$                              ;
                        24 00000000G 00            02  E0 00054 3$:    BBS      #2, MAIL$GL_FLAGS+1, 4$                 ; 1656
                                                   7E  7C 0005C        CLRQ     -(SP)                                   ; 1662
                                                   7E  7C 0005E        CLRQ     -(SP)                                   ;
                                     7E            62  3C 00060        MOVZWL   (OUTDESC), -(SP)                        ;
                                               04  A2  DD 00063        PUSHL    4(OUTDESC)                              ;
                                                   7E  7C 00066        CLRQ     -(SP)                                   ;
                                               28  AE  9F 00068        PUSHAB   QIOSB                                   ;
                                                   30  DD 0006B        PUSHL    #48                                     ;
                                                   63  DD 0006D        PUSHL    LINK_CHAN                               ;
                                                   7E  D4 0006F        CLRL     -(SP)                                   ;
                           00000000G 00            0C  FB 00071        CALLS    #12, SYS$QIOW                           ;
                                     13            50  E9 00078        BLBC     STATUS, 5$                              ; 1663
                                     50        08  AE  3C 0007B        MOVZWL   QIOSB, STATUS                           ; 1664
                                                       04 0007F        RET                                              ; 1667
                                     50        04  A3  D0 00080 4$:    MOVL     LINK_TFRADR, R0                         ;
                                                   52  DD 00084        PUSHL    OUTDESC                                 ;
                                                   0F  DD 00086        PUSHL    #15                                     ;
                                               08  A3  9F 00088        PUSHAB   LINK_CONTEXT                            ;
                                     60            03  FB 0008B        CALLS    #3, (R0)                                ;
                                                       04 0008E 5$:    RET                                              ; 1669

; Routine Size:  143 bytes,    Routine Base:  $CODE$ + 0DAC
; 1549          1670  0 END ELUDOM



                                                                        .EXTRN  LIB$SIGNAL, SYS$UNWIND

;                                         PSECT SUMMARY
;
;       Name                        Bytes                        Attributes
;
;   $OWN$                                 16  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;   $GLOBAL$                               8  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;   $CODE$                              3643  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)




;                                  Library Statistics
;
;                                               -------- Symbols --------       Pages        Processing
;       File                                    Total    Loaded   Percent       Mapped       Time
;
;   _$255$DUA28:[SYSLIB]STARLET.L32;1            9776        74         0         581          00:00.8
;   _$255$DUA28:[MAIL.OBJ]MAILDEF.L32;1           457        71        15          26          00:00.2




;                                         COMMAND QUALIFIERS

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

; Size:           3466 code + 201 data bytes
; Run Time:          00:42.5
; Elapsed Time:      02:41.0
; Lines/CPU Min:      2357
; Lexemes/CPU-Min: 36069
; Memory Used:  225 pages
; Compilation Complete
```

0230 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

MAILMSG
LIS

MAILCMDS
LIS

NOTIFY
LIS

MSGSUBS
LIS

NETSUBS
LIS