```
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
```

```
  AAAAAA    DDDDDDDD    DDDDDDDD    RRRRRRRR    LL          IIIIII     SSSSSSSS  TTTTTTTTTT
  AAAAAA    DDDDDDDD    DDDDDDDD    RRRRRRRR    LL          IIIIII     SSSSSSSS  TTTTTTTTTT
AA      AA  DD      DD  DD      DD  RR      RR  LL            II     SS              TT
AA      AA  DD      DD  DD      DD  RR      RR  LL            II     SS              TT
AA      AA  DD      DD  DD      DD  RR      RR  LL            II     SS              TT
AA      AA  DD      DD  DD      DD  RR      RR  LL            II     SS              TT
AA      AA  DD      DD  DD      DD  RRRRRRRR    LL            II       SSSSSS        TT
AA      AA  DD      DD  DD      DD  RRRRRRRR    LL            II       SSSSSS        TT
AAAAAAAAAA  DD      DD  DD      DD  RR  RR      LL            II             SS      TT
AAAAAAAAAA  DD      DD  DD      DD  RR  RR      LL            II             SS      TT
AA      AA  DD      DD  DD      DD  RR    RR    LL            II             SS      TT      ....
AA      AA  DD      DD  DD      DD  RR    RR    LL            II             SS      TT      ....
AA      AA  DDDDDDDD    DDDDDDDD    RR      RR  LLLLLLLLLL  IIIIII   SSSSSSSS        TT      ....
AA      AA  DDDDDDDD    DDDDDDDD    RR      RR  LLLLLLLLLL  IIIIII   SSSSSSSS        TT      ....
```

```
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
```

Table of contents

```
0000     1 .TITLE  MAIL$ADDRLIST   ADDRESSEE LIST ROUTINES
0000     2         .IDENT  'V04-000'
0000     3 ;
0000     4 ;*****************************************************************************
0000     5 ;*                                                                           *
0000     6 ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0000     7 ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0000     8 ;*  ALL RIGHTS RESERVED.                                                     *
0000     9 ;*                                                                           *
0000    10 ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0000    11 ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    12 ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0000    13 ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0000    14 ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0000    15 ;*  TRANSFERRED.                                                             *
0000    16 ;*                                                                           *
0000    17 ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    18 ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    19 ;*  CORPORATION.                                                             *
0000    20 ;*                                                                           *
0000    21 ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0000    22 ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0000    23 ;*                                                                           *
0000    24 ;*                                                                           *
0000    25 ;*****************************************************************************
0000    26 ;
0000    27
0000    28 ;++
0000    29 ; FACILITY:     VAX/VMS MAIL UTILITY
0000    30 ;
0000    31 ; ABSTRACT:     ADDRESSEE LIST ROUTINES
0000    32 ;
0000    33 ;
0000    34 ; ENVIRONMENT:  NATIVE/USER MODE
0000    35 ;
0000    36 ; AUTHOR:       LEN KAWELL, CREATION DATE: 22-NOV-78
0000    37 ;
0000    38 ; MODIFICATION HISTORY:
0000    39 ;
0000    40 ;       V03-015 BLS0340         Benn Schreiber          10-AUG-1984
0000    41 ;               Correct handling of alternate protocol if nodename
0000    42 ;               specified.
0000    43 ;
0000    44 ;       V03-014 ROP0002         Robert Posniak          30-MAY-1984
0000    45 ;               Check for prot%""quoted@gibberish"".  Strip
0000    46 ;               quotes from gibberish.
0000    47 ;
0000    48 ;       V03-013 BLS0310         Benn Schreiber          27-APR-1984
0000    49 ;               Check for blank line before checking comment in
0000    50 ;               address list parsing.
0000    51 ;
0000    52 ;       V03-012 BLS0292         Benn Schreiber          30-MAR-1984
0000    53 ;               Correct call to free_vm to only pass two args.
0000    54 ;
0000    55 ;       V03-011 BLS0263         Benn Schreiber          22-Jan-1984
0000    56 ;               Honor terminal logical name attribute. Remove resultant
0000    57 ;               file spec substitution. Too much work required to make
```

```
0000     58 ;                  it work right now.
0000     59 ;
0000     60 ;        V03-010 BLS0261         Benn Schreiber          17-Jan-1984
0000     61 ;                Correct references to uaf$b_flags and uaf$v_disnewmail
0000     62 ;
0000     63 ;        V03-009 BLS0255         Benn Schreiber          28-Dec-1983
0000     64 ;                Substitute resultant dist. file spec if only item in
0000     65 ;                the list.  Correct errors when protocol spec used.
0000     66 ;                Add call to see if addressee already in the list.
0000     67 ;
0000     68 ;        V03-008 BLS0246         Benn Schreiber          28-Nov-1983
0000     69 ;                Fix loop detection counter reset.  Add call to see if
0000     70 ;                destination node is in current cluster and skip network
0000     71 ;                setup if so.
0000     72 ;
0000     73 ;        V03-007 BLS0235         Benn Schreiber          23-Aug-1983
0000     74 ;                Standardize file error reporting in distribution list handling.
0000     75 ;                Fix reference to UAF buffer.  Catch more junky addressee specs
0000     76 ;                at parse time.
0000     77 ;
0000     78 ;        V03-006 BLS0222         Benn Schreiber          15-May-1983
0000     79 ;                Ignore blank lines in addressee lists
0000     80 ;
0000     81 ;        V03-005 BLS0214         Benn Schreiber          27-Mar-1983
0000     82 ;                Enhance error detection.  Correct stack offset in
0000     83 ;                distribution list handling.  Set SQO when reading
0000     84 ;                distribution list.
0000     85 ;
0000     86 ;        V03-004 BLS0211         Benn Schreiber          13-Mar-1983
0000     87 ;                Extensive rework for optional protocol spec and multi-threaded
0000     88 ;                server
0000     89 ;
0000     90 ;        V03-003 BLS0184         Benn Schreiber          24-Sep-1982
0000     91 ;                If addressee is on same node, skip setting up logical
0000     92 ;                link.  This can be forced by using "__node" to use decnet
0000     93 ;
0000     94 ;        V03-002 BLS0179         Benn Schreiber          25-Jul-1982
0000     95 ;                Allow hyphens in user names.
0000     96 ;
0000     97 ;        V03-001 BLS0177         Benn Schreiber          22-Jun-1982
0000     98 ;                Allow periods in user names.  Check for recursive logical name
0000     99 ;                translation loop that does not terminate.  Implement the mail
0000    100 ;                disabled flag (UAF$V_DISMAIL in the UAF record).  Implement
0000    101 ;                MAIL logical names (honor forwarding address in
0000    102 ;                SYS$SYSTEM:VMSMAIL.DAT)
0000    103 ;--
```

```
                     0000   105          .SBTTL  DECLARATIONS
                     0000   106 ;
                     0000   107 ; INCLUDE FILES:
                     0000   108 ;
                     0000   109
                     0000   110 ;
                     0000   111 ; MACROS:
                     0000   112 ;
                     0000   113
                     0000   114 ;
                     0000   115 ; EQUATED SYMBOLS:
                     0000   116 ;
                     0000   117
00000004             0000   118 ADDR            = 4                     ;ADDRESSEE NAME DESC ADDR ARG
00000004             0000   119 FAB             = 4                     ;MESSAGE FILE FAB ADDR ARG
00000008             0000   120 RAB             = 8                     ;MESSAGE FILE RAB ADDR ARG
                     0000   121
0000002C             0000   122 COMMA           = 44                    ;ASCII COMMA
00000020             0000   123 CH_SPACE        = ^A/ /
00000009             0000   124 CH_TAB          = ^A/   /
000000FF             0000   125 MAIL$K_INBUFFSZ = 255                   ;INPUT BUFFER SIZE
                     0000   126
                     0000   127          $CHFDEF                         ;CONDITION HANDLER DEFINITIONS
                     0000   128          $FABDEF                         ;FILE ACCESS BLOCK DEFINITIONS
                     0000   129          $LNMDEF                         ;LOGICAL NAME DEFINITIONS
                     0000   130          $MAILMSGDEF                     ;MAIL MESSAGES
                     0000   131          $NAMDEF                         ;NAM BLOCK DEFINITIONS
                     0000   132          $RABDEF                         ;RECORD ACCESS BLOCK DEFINITIONS
                     0000   133          $STSDEF                         ;STATUS CODE DEFINITIONS
                     0000   134          $TPADEF                         ;TPARSE DEFINITIONS
                     0000   135          ADRDEF                          ;ADDRESSEE LIST DESCRIPTORS
                     0000   136          CNCTDEF                         ;CONNECT BLOCK OFFSETS
                     0000   137          LNKDEF                          ;LOGICAL LINK BLOCK OFFSETS
                     0000   138          MAIFDEF                         ;DEFINE MAIL GLOBAL FLAGS
                     0000   139          VMDDEF                          ;VMS MAIL FILE OFFSETS
                     0000   140
                     0000   141 ;
                     0000   142 ; DEFINE LOCAL NAMES FOR SHARED MESSAGES
                     0000   143 ;
                     0000   144          $SHR_MSGDEF     MAIL,126,LOCAL,<<READERR,ERROR>,<OPENIN,ERROR>>
                     0000   145
                     0000   146 ;
                     0000   147 ; DEFINE EXTENSION TO THE TPARSE BLOCK
                     0000   148 ;
00000024             0000   149 TPA_L_LCOUNT = TPA$L_PARAM+4             ;LOOP COUNT
00000028             0000   150 TPA_Q_ONAME = TPA_L_LCOUNT+4             ;DESCRIPTOR OF ORIGINAL NAME
00000030             0000   151 TPA_L_LLNK = TPA_Q_ONAME+8               ;LOGICAL LINK OR 0 IF LOCAL
00000034             0000   152 TPA_Q_LOGNAM = TPA_L_LLNK+4              ;DESCRIPTOR
0000003C             0000   153 TPA_Q_PNAM = TPA_Q_LOGNAM+8              ;PROTOCOL NAME DESCRIPTOR
00000044             0000   154 TPA_L_CNCT = TPA_Q_PNAM+8                ;ADDR OF CNCT DESC
00000048             0000   155 TPA_L_PTPA = TPA_L_CNCT+4                ;ADDR OF PREV. TPARSE BLOCK
0000004C             0000   156 TPA_L_SPST = TPA_L_PTPA+4                ;START OF SPEC IN STRING
00000050             0000   157 TPA_Q_UNAME = TPA_L_SPST+4               ;USERNAME DESCRIPTOR
00000058             0000   158 TPA_W_UENT = TPA_Q_ONAME+8               ;# USERS (CALLS TO ENTER_ADDR, ACTUALLY)
0000005A             0000   159 TPA_W_FLAGS = TPA_W_UENT+2               ;FLAGS
00000000             0000   160     TPA_V_TERMINAL = 0                  ;LAST LOGICAL USERNAME WAS TERMINAL
00000001             0000   161     TPA_M_TERMINAL = 1                  ;MASK FOR FLAG
```

```
00000038  0000   162 TPA_C_EXTRA = TPA_W_FLAGS+2-TPA_L_LCOUNT;LENGTH OF EXTENSION
          0000   163
          0000   164 ;
          0000   165 ; OWN STORAGE:
          0000   166 ;
          0000   167
          00000000   168         .PSECT  $PLIT$,LONG,RD,NOWRT,NOEXE
          0000   169
          0000   170 DIST_TYPE:                                  ;DISTRIBUTION LIST FILE TYPE
53 49 44 2E  0000   171         .ASCII  /.DIS/
  00000004  0004   172 DIST_TYPE_SIZ = .-DIST_TYPE
          0004   173
          0004   174 AT_STRING:
       40 0004   175         .ASCII  /@/
          0005   176
          00000000   177         .PSECT  $CODE$,LONG,RD,NOWRT,EXE
```

```
                     0000   179          .SBTTL  ADD ADDRESSEE TO ADDRESSEE LIST
                     0000   180 ;++
                     0000   181 ; FUNCTIONAL DESCRIPTION:
                     0000   182 ;
                     0000   183 ;       PARSE ADDRESSEE NAMES AND ADD TO ADDRESSEE LIST.
                     0000   184 ;
                     0000   185 ; CALLING SEQUENCE:
                     0000   186 ;       CALL MAIL$ADD_ADDR(ADDR,CNCTDESC)
                     0000   187 ;
                     0000   188 ; INPUT PARAMETERS:
                     0000   189 ;       4(AP) = ADDRESSEE NAME(S) DESC
                     0000   190 ;       8(AP) = ADDRESS OF CONNECT BLOCK DESCRIPTOR
                     0000   191 ;
                     0000   192 ;
                     0000   193 ; OUTPUT PARAMETERS:
                     0000   194 ;       NONE
                     0000   195 ;
                     0000   196 ; IMPLICIT OUTPUTS:
                     0000   197 ;       ADDRESSEE LIST ENTRIES CREATED FOR ADDRESSEE(S) AND ENTERED
                     0000   198 ;       INTO ADDRESSEE LIST.  IF AN ADDRESSEE IS LOCATED ON A REMOTE
                     0000   199 ;       SYSTEM, A LOGICAL LINK AND LOGICAL LINK LIST ENTRY ARE ALSO
                     0000   200 ;       CREATED.
                     0000   201 ;
                     0000   202 ; COMPLETION CODES:
                     0000   203 ;       R0 = SUCCESS IF ALL ADDRESSEES FOUND SUCCESSFULLY,
                     0000   204 ;          = WARNING IF AT LEAST ONE ADDRESSEE FOUND SUCCESSFULLY,
                     0000   205 ;          = ERROR IF NO ADDRESSEES FOUND OR SYNTAX ERROR IN LIST.
                     0000   206 ;
                     0000   207 ; SIDE EFFECTS:
                     0000   208 ;       IF FAILURE, ERROR IS SIGNALED.
                     0000   209 ;
                     0000   210 ;--
                     0000   211 MAIL_ADD_ADDR:                                ;INTERNAL ROUTINE WHICH IS CALLED
             001C    0000   212          .WORD   ^M<R2,R3,R4>                ; TO RECURSIVELY ADD AN ADDRESS
                     0002   213                                              ;  (CALLED FROM WITHIN MAIL$ADD_ADDR ONLY)
     54  08 AD   D0 0002   214          MOVL    8(FP),R4                    ;GET CALLER'S TPARSE BLOCK
         24 A4   D7 0006   215          DECL    TPA_L_LCOUNT(R4)            ;DECREMENT THE COUNT
            45   14 0009   216          BGTR    60$                         ;BRANCH OF NOT LOOPING
     50  44 A4   D0 000B   217          MOVL    TPA_L_CNCT(R4),R0           ;GET CNCT BLOCK ADDRESS
            7E   7C 000F   218          CLRQ    -(SP)                       ;MAKE NULL DESCRIPTOR
         52 5E   D0 0011   219          MOVL    SP,R2                       ;SAVE ITS ADDRESS
         28 A0   9F 0014   220          PUSHAB  CNCT_Q_ADRLST(R0)           ;STACK ADDRESS OF LISTHEAD
         34 A4   9F 0017   221          PUSHAB  TPA_Q_LOGNAM(R4)            ;STACK USERNAME DESCRIPTOR ADDR
            52   DD 001A   222          PUSHL   R2                          ;USE NULL NODENAME
            52   DD 001C   223          PUSHL   R2                          ;AND NULL USERNAME
00000000'EF     04 FB 001E   224          CALLS   #4,MAIL$ADDR_EXISTS         ;SEE IF ALREADY IN LIST
         24 50  E8 0025   225          BLBS    R0,50$                      ;BRANCH IF YES
     52  34 A4   9E 0028   226          MOVAB   TPA_Q_LOGNAM(R4),R2         ;SET USERNAME DESCR ADDR IN R2
     53  44 A4   D0 002C   227          MOVL    TPA_L_CNCT(R4),R3           ;SET CNCT ADDR IN R3
           042C   30 0030   228          BSBW    CREATE_ENTRY                ;CREATE THIS ENTRY
         04 50  E9 0033   229          BLBC    R0,40$                      ;BRANCH IF FAILED
  1C A1  01  88 0036   230          BISB2   #ADR_M_NOSEND,ADR_B_FLAGS(R1) ;FLAG NO SEND
         28 A4   9F 003A   231 40$:     PUSHAB  TPA_Q_ONAME(R4)             ;STACK ADDRESS OF DESCIRPTOR
            01   DD 003D   232          PUSHL   #1                          ;STACK NUMBER OF ARGUMENTS
   007E808A 8F   DD 003F   233          PUSHL   #MAIL$_FORWLOOP             ;FORWARDING LOOP ERROR
00000000'GF     03 FB 0045   234          CALLS   #3,G^LIB$SIGNAL
         50   02 D0 004C   235 50$:     MOVL    #STS$K_ERROR,R0             ;RETURN ERROR STATUS
```

```
                     04   004F   236          RET
                53   54   D0   0050   237 60$:  MOVL    R4,R3                       ;SET R3 POINTING TO PREV. CNCT
                     54   D4   0053   238          CLRL    R4                          ;CLEAR PRESET UENT VALUE
          15 00000000'EF    08   E1   0055   239          BBC     #MAIF_V_CTRLCFL,MAIL$GL_FLAGS,ADD_ADDR ;BRANCH IF NOT CTRL/C
             50   007E8112 8F   D0   005D   240          MOVL    #MAIL$_CONABORT,R0          ;FLAG ABORTED BY CTRL/C
                     04   0064   241          RET
                          0065   242
                          0065   243 AT_LIST_ADD_ADDR:
                001C      0065   244          .WORD   ^M<R2,R3,R4>                ;ENTRY FOR DISTRIBUTION LIST RECURSION
                53   D4   0067   245          CLRL    R3                          ;CLEAR PREVIOUS FLAG
                54   01   D0   0069   246          MOVL    #1,R4                       ;SET UENT FLAG
                04   11   006C   247          BRB     ADD_ADDR
                          006E   248
                001C      006E   249          .ENTRY  MAIL$ADD_ADDR,-             ;ADD ADDRESSEE TO LIST
                          0070   250                  ^M<R2,R3,R4>                ;(ENTRY MASK)
                53   7C   0070   251          CLRQ    R3                          ;SET NO PREVIOUS CALL AND NO UENT
                          0072   252 ADD_ADDR:
             50   04 AC   D0   0072   253          MOVL    ADDR(AP),R0                 ;GET ADDRESSEE DESC ADDR
             5E   A4 AE   9E   0076   254          MOVAB   -<TPA$K_LENGTH0+TPA_C_EXTRA>(SP),SP ;CREATE TPARSE BLOCK
                52   5E   D0   007A   255          MOVL    SP,R2                       ;GET ADDR OF IT
                62   08   D0   007D   256          MOVL    #TPA$K_COUNT0,(R2)          ;SET PARAMETER COUNT
             20 A2   01   D0   0080   257          MOVL    #1,TPA$L_PARAM(R2)          ;USE PARAMETER FOR STATUS - ASSUME SUCCESS
             08 A2   60   3C   0084   258          MOVZWL  (R0),TPA$L_STRINGCNT(R2)    ;SET SIZE OF ADDRESSEE
          0C A2   04 A0   D0   0088   259          MOVL    4(R0),TPA$L_STRINGPTR(R2)   ;SET ADDR OF ADDRESSEE
                53   D5   008D   260          TSTL    R3                          ;PREVIOUS BLOCK PRESENT?
                1F   13   008F   261          BEQL    5$                          ;IF EQL NO
          24 A2   24 A3   D0   0091   262          MOVL    TPA_L_LCOUNT(R3),TPA_L_LCOUNT(R2)
          28 A2   28 A3   7D   0096   263          MOVQ    TPA_Q_ONAME(R3),TPA_Q_ONAME(R2)
          44 A2   44 A3   D0   009B   264          MOVL    TPA_L_CNCT(R3),TPA_L_CNCT(R2)
          58 A2   58 A3   B0   00A0   265          MOVW    TPA_W_UENT(R3),TPA_W_UENT(R2)
          5A A2   5A A3   B0   00A5   266          MOVW    TPA_W_FLAGS(R3),TPA_W_FLAGS(R2)
             48 A2   53   D0   00AA   267          MOVL    R3,TPA_L_PTPA(R2)           ;COPY TPARSE BLOCK ADDRESS
                1E   11   00AE   268          BRB     7$
          24 A2   0A   D0   00B0   269 5$:  MOVL    #10,TPA_L_LCOUNT(R2)        ;SET LOOP COUNT LIMIT
       28 A2   08 A2   7D   00B4   270          MOVQ    TPA$L_STRINGCNT(R2),TPA_Q_ONAME(R2);SET ORIGINAL NAME
             30 A2   D4   00B9   271          CLRL    TPA_L_LLNK(R2)              ;CLEAR LLNK
             34 A2   7C   00BC   272          CLRQ    TPA_Q_LOGNAM(R2)            ;CLEAR DESCRIPTOR
       44 A2   08 AC   D0   00BF   273          MOVL    8(AP),TPA_L_CNCT(R2)        ;SET CNCT BLOCK ADDRESS
             48 A2   D4   00C4   274          CLRL    TPA_L_PTPA(R2)              ;CLEAR PREV TPARSE BLOCK POINTER
             58 A2   D4   00C7   275          CLRL    TPA_W_UENT(R2)              ;CLEAR # ENTER ADDRS (AND FLAGS)
          58 A2   54   B0   00CA   276          MOVW    R4,TPA_W_UENT(R2)           ;SET PRESET UENT
             3C A2   7C   00CE   277 7$:  CLRQ    TPA_Q_PNAM(R2)              ;PROTOCOL NAME DESCRIPTOR
             50 A2   7C   00D1   278          CLRQ    TPA_Q_UNAME(R2)             ;CLEAR USERNAME DESCRIPTOR
                52   DD   00D4   279          PUSHL   R2
          0133'CF    01   FB   00D6   280          CALLS   #1,W^CALL_TPARSE
                1C   50   E8   00DB   281          BLBS    R0,10$                      ;BR IF SUCCESS
       007E8112 8F      50   D1   00DE   282          CMPL    R0,#MAIL$_CONABORT          ;ABORT BY ^C?
                31   13   00E5   283          BEQL    30$                         ;BRANCH IF YES
             10 A2   DF   00E7   284          PUSHAL  TPA$L_TOKENCNT(R2)          ;SET ADDR OF TOKEN DESCRIPTOR
                01   DD   00EA   285          PUSHL   #1                          ;ONE FAO ARGUMENT
          007E10FA 8F      DD   00EC   286          PUSHL   #SHR$_SYNTAX ! -            ;SET SYNTAX ERROR MESSAGE
                               00F2   287                  <MAIL$_FACILITY @ STS$V_FAC_NO> ! STS$K_ERROR
       00000000'GF      03   FB   00F2   288          CALLS   #3,G^LIB$SIGNAL             ;SIGNAL THE ERROR
                     04   00F9   289          RET                                 ;RETURN
                          00FA   290 10$:
             19 20 A2   E8   00FA   291          BLBS    TPA$L_PARAM(R2),25$         ;BR IF COMPLETE SUCCESS
                00   DD   00FE   292          PUSHL   #STS$K_WARNING              ;ASSUME AT LEAST PARTIAL SUCCESS
```

```
                51   44 A2    D0  0100    293          MOVL    TPA_L_CNCT(R2),R1       ;GET ADDR OF CNCT BLOCK
                   28 A1    9F  0104    294          PUSHAB  CNCT_Q_ADRLST(R1)       ;GET ADDR OF LISTHEAD
        000005B3'EF    01    FB  0107    295          CALLS   #1,MAIL$ANY_ADDRESSEES  ;SEE IF ANY ADDRESSEES FOUND
                   03 50    E8  010E    296          BLBS    R0,20$                  ;BRANCH IF YES
                  6E    02    D0  0111    297          MOVL    #STS$K_ERROR,(SP)       ;NO--SET ERROR
                        50 8ED0  0114    298 20$:     POPL    R0                      ;GET RETURN STATUS
                           04    0117    299 25$:     RET                             ;RETURN
                                 0118    300
                                 0118    301 ;
                                 0118    302 ; CTRL/C was typed while attempting connection
                                 0118    303 ;
                53   44 A2    D0  0118    304 30$:     MOVL    TPA_L_CNCT(R2),R3       ;GET CNCT BLOCK ADDRESS
                   30 A3    9F  011C    305          PUSHAB  CNCT_Q_LNKLST(R3)       ;STACK LOGICAL LINK LIST
        00000000'EF    01    FB  011F    306          CALLS   #1,MAIL$DELETELINK      ;DELETE ANY OUTSTANDING LINKS
                           53    DD  0126    307          PUSHL   R3                      ;STACK CNCT BLOCK ADDRESS
        000005DD'EF    01    FB  0128    308          CALLS   #1,MAIL$INI_ADDR        ;RE-INITIALIZE LIST
                  50    02    D0  012F    309          MOVL    #STS$K_ERROR,R0         ;RETURN ERROR
                           04    0132    310          RET
                                 0133    311
                                 0133    312 CALL_TPARSE:
                        0000     0133    313          .WORD   0
        6D   00000000'EF    DE  0135    314          MOVAL   MAIL$PRUNW_HANDLER,(FP)
             00000000'EF    DF  013C    315          PUSHAL  ADDR_KEY                ;SET KEYWORD TABLE ADDRESS
             00000000'EF    DF  0142    316          PUSHAL  ADDR_STATE              ;SET STATE TABLE ADDRESS
                   04 AC    DD  0148    317          PUSHL   4(AP)                   ;SET PARAMETER BLOCK ADDRESS
        00000000'GF    03    FB  014B    318          CALLS   #3,G^LIB$TPARSE         ;PARSE ADDRESSEE NAME
                           04    0152    319          RET
```

```
                                G 3
0153    321          .SBTTL  ADDRESSEE PARSING STATE TABLE
0153    322 ;++
0153    323 ;
0153    324 ; PARSE "TO:" LIST AND FILL IN LINK AND ADDRESSEE TABLES
0153    325 ;
0153    326 ;       PARSES LISTS WITH THE FOLLOWING FORMAT:
0153    327 ;
0153    328 ;               ADDRESSEE,ADDRESSEE,...,ADDRESSEE
0153    329 ;
0153    330 ;       WHERE "ADDRESSEE" HAS THE FOLLOWING FORMAT:
0153    331 ;
0153    332 ;               [NODE1::][NODE2::]...[NODEN::]USERNAME
0153    333 ;
0153    334 ;       WHERE USERNAME CAN ALSO BE A UIC/PPN FOR RSX/RSTS SYSTEMS.
0153    335 ;
0153    336 ;       LIST CAN ALSO HAVE A "@FILESPEC" AS LAST TOKEN, WHERE
0153    337 ;       THE FILE CONTAINS A LIST OF ADDRESSEES.
0153    338 ;
0153    339 ;--
0153    340          $INIT_STATE ADDR_STATE,ADDR_KEY
0153    341          $STATE  NEXT_ADDR
0153    342          $TRAN   TPA$_LAMBDA,,SET_SPST
0153    343
0153    344          $STATE
0153    345          $TRAN   TPA$_EOS,TPA$_EXIT
0153    346          $TRAN   CH_SPACE,NEXT_ADDR
0153    347          $TRAN   CH_TAB,NEXT_ADDR
0153    348          $TRAN   '!',TPA$_EXIT
0153    349          $TRAN   COMMA,NEXT_ADDR
0153    350          $TRAN   '+',NEXT_ADDR
0153    351          $TRAN   '@',DIST_PARSE
0153    352          $TRAN   '"',QDIST_PARSE
0153    353          $TRAN   !PROTOCOL_SPEC,,STORE_PROTOCOL
0153    354          $TRAN   TPA$_LAMBDA
0153    355
0153    356          $STATE
0153    357          $TRAN   !NODE1_SPEC
0153    358          $TRAN   !CHECK_QUOTE,QUOTED_SPEC
0153    359          $TRAN   !GET_SYMBOL,,STORE_LOGNAME
0153    360          $TRAN   TPA$_LAMBDA,FIND_ADDRESSEE
0153    361
0153    362          $STATE
0153    363          $TRAN   !SCAN_ADDR,SPEC_CONTINUE,STORE_UNAME
0153    364          $TRAN   TPA$_LAMBDA,FIND_ADDRESSEE
0153    365
0153    366          $STATE  SPEC_CONTINUE
0153    367          $TRAN   !CHECK_SEP,NEXT_ADDR,ENTER_ADDR
0153    368          $TRAN   TPA$_LAMBDA
0153    369 ;
0153    370 ; FIND NEXT ADDRESSEE AFTER BAD USERNAME AND REPORT ERROR ON THIS ONE
0153    371 ; EXIT WITH TPA$_EXIT BECAUSE MESSAGE WILL HAVE BEEN ISSUED ALREADY
0153    372 ;
0153    373          $STATE  FIND_ADDRESSEE
0153    374          $TRAN   TPA$_EOS
0153    375          $TRAN   '!'
0153    376          $TRAN   COMMA
0153    377          $TRAN   '+'
```

```
0153   378          $TRAN   TPA$_ANY,FIND_ADDRESSEE
0153   379
0153   380          $STATE
0153   381          $TRAN   TPA$_LAMBDA,TPA$_EXIT,REPORT_SYNTAX_ERROR
0153   382
0153   383          $STATE  SCAN_ADDR
0153   384          $TRAN   !PROTOCOL_SPEC,SCAN_ADDR
0153   385          $TRAN   !NODE_SPEC,SCAN_ADDR
0153   386          $TRAN   !USER_SPEC,TPA$_EXIT
0153   387          $TRAN   TPA$_LAMBDA,TPA$_EXIT
0153   388
0153   389 ;
0153   390 ; DISTRIBUTION LIST PARSING
0153   391 ;
0153   392          $STATE  DIST_PARSE
0153   393          $TRAN   TPA$_FILESPEC,NEXT_ADDR,GET_DIST_LIST
0153   394 ;
0153   395 ; QUOTED DISTRIBUTION LIST PARSING
0153   396 ;
0153   397          $STATE  QDIST_PARSE
0153   398          $TRAN   '@'
0153   399          $STATE
0153   400          $TRAN   !QFILE_SPEC,TPA$_EXIT,GET_DIST_LIST
0153   401          $STATE  QFILE_SPEC
0153   402          $TRAN   !NOT_QEOS,QFILE_SPEC
0153   403          $TRAN   TPA$_LAMBDA,TPA$_EXIT
0153   404          $STATE  NOT_QEOS
0153   405          $TRAN   '"'
0153   406          $TRAN   TPA$_ANY,TPA$_EXIT
0153   407          $STATE
0153   408          $TRAN   TPA$_EOS,TPA$_FAIL
0153   409          $TRAN   TPA$_ANY,TPA$_EXIT
0153   410 ;
0153   411 ; NODE SPECIFICATION SUB-EXPRESSION PARSING
0153   412 ;
0153   413          $STATE  NODE1_SPEC                          ;FIRST NODENAME PARSING
0153   414          $TRAN   !GET_SYMBOL,NODE_SPECC,STORE_LOGNAME
0153   415          $STATE  NODE_SPEC                           ;NODENAME PARSING
0153   416          $TRAN   !GET_SYMBOL
0153   417          $STATE  NODE_SPECC                          ;NODENAME CONTINUATION PARSING
0153   418          $TRAN   !ACCESS_LIST
0153   419          $TRAN   TPA$_LAMBDA
0153   420          $STATE
0153   421          $TRAN   ':'
0153   422          $STATE
0153   423          $TRAN   ':',TPA$_EXIT
0153   424
0153   425          $STATE  ACCESS_LIST                         ;DECNET ACCESS LIST PARSING
0153   426          $TRAN   '"'
0153   427          $STATE  ACCESS_SCAN
0153   428          $TRAN   '"',TPA$_EXIT
0153   429          $TRAN   TPA$_ANY,ACCESS_SCAN
0153   430 ;
0153   431 ; USERNAME/PPN SPECIFICATION SUB-EXPRESSION PARSING
0153   432 ;
0153   433          $STATE  USER_SPEC                           ;USERNAME SPEC PARSING
0153   434          $TRAN   !GET_SYMBOL,TPA$_EXIT
```

```
                                                I  3

0153   435          $TRAN   '['
0153   436          $TRAN   '('
0153   437          $TRAN   '"',QUSER_SPEC
0153   438          $STATE                                  ;UIC/PPN USERSNAME SPEC PARSING
0153   439          $TRAN   TPA$_DECIMAL
0153   440          $STATE
0153   441          $TRAN   COMMA
0153   442          $STATE
0153   443          $TRAN   TPA$_DECIMAL
0153   444          $STATE
0153   445          $TRAN   ']',TPA$_EXIT
0153   446          $TRAN   ')',TPA$_EXIT
0153   447
0153   448          $STATE  QUSER_SPEC                      ;QUOTED USERNAME SPEC PARSING
0153   449          $TRAN   '"',TPA$_EXIT
0153   450          $TRAN   TPA$_ANY,QUSER_SPEC
0153   451
0153   452          $STATE  QUOTED_SPEC
0153   453          $TRAN   !USER_SPEC,SPEC_CONTINUE,STORE_LOGNAME
0153   454
0153   455 ;
0153   456 ; PROTOCOL NAME PARSING
0153   457 ;
0153   458          $STATE  PROTOCOL_SPEC
0153   459          $TRAN   !GET_SYMBOL
0153   460
0153   461          $STATE
0153   462          $TRAN   '"',QPROT_SPEC
0153   463          $TRAN   '%',TPA$_EXIT
0153   464
0153   465          $STATE  QPROT_SPEC
0153   466          $TRAN   '"'
0153   467          $TRAN   TPA$_ANY,QPROT_SPEC
0153   468
0153   469          $STATE
0153   470          $TRAN   '%',TPA$_EXIT
0153   471 ;
0153   472 ; SYMBOL PARSING
0153   473 ;
0153   474          $STATE  GET_SYMBOL
0153   475          $TRAN   '.',GETSYM1
0153   476          $TRAN   '-',GETSYM1
0153   477          $TRAN   TPA$_SYMBOL,GETSYM1
0153   478          $TRAN   TPA$_LAMBDA,TPA$_FAIL
0153   479
0153   480          $STATE  GETSYM1
0153   481          $TRAN   '.',GETSYM1
0153   482          $TRAN   '-',GETSYM1
0153   483          $TRAN   TPA$_SYMBOL,GETSYM1
0153   484          $TRAN   TPA$_LAMBDA,TPA$_EXIT
0153   485
0153   486 ;
0153   487 ; CHECK THAT NEXT CHARACTER IS LEGAL ADDRESSEE TERMINATOR
0153   488 ;
0153   489          $STATE  CHECK_SEP
0153   490          $TRAN   !LEGAL_SEP,TPA$_EXIT
0153   491
```

```
0153   492          $STATE  LEGAL_SEP
0153   493          $TRAN   !LEGAL_SEP1,TPA$_FAIL
0153   494          $TRAN   TPA$_LAMBDA,TPA$_EXIT
0153   495
0153   496          $STATE  LEGAL_SEP1
0153   497          $TRAN   TPA$_EOS,TPA$_FAIL
0153   498          $TRAN   '!',TPA$_FAIL
0153   499          $TRAN   COMMA,TPA$_FAIL
0153   500          $TRAN   '+',TPA$_FAIL
0153   501          $TRAN   CH_SPACE,TPA$_FAIL
0153   502          $TRAN   CH_TAB,TPA$_FAIL
0153   503          $TRAN   TPA$_LAMBDA,TPA$_EXIT
0153   504
0153   505 ;
0153   506 ;  Check to see if the next character is a quote
0153   507 ;
0153   508          $STATE  CHECK_QUOTE
0153   509          $TRAN   !QUOTE_SPEC,TPA$_EXIT
0153   510
0153   511          $STATE  QUOTE_SPEC
0153   512          $TRAN   !SPEC_QUOTE,TPA$_FAIL
0153   513          $TRAN   TPA$_LAMBDA,TPA$_EXIT
0153   514
0153   515          $STATE  SPEC_QUOTE
0153   516          $TRAN   '"',TPA$_FAIL
0153   517          $TRAN   TPA$_LAMBDA,TPA$_EXIT
0153   518
0153   519
0153   520          $END_STATE
0153   521
0153   522
00000153   523          .PSECT  $CODE$,LONG,RD,NOWRT,EXE
0153   524
```

```
                                0153    526          .SBTTL  STORE_PROTOCOL - STORE PROTOCOL NAME IN TPA BLOCK EXTENSION
                                0153    527 ;++
                                0153    528 ; ROUTINE CALLED BY TPARSE TO STORE THE PROTOCOL NAME
                                0153    529 ;
                                0153    530 ;--
                                0153    531
                                0153    532 STORE_PROTOCOL:
                          0000  0153    533          .WORD   0
                3C AC     10 AC  7D     0155    534          MOVQ    TPA$L_TOKENCNT(AP),TPA_Q_PNAM(AP)       ;COPY PROTOCOL NAME
                          3C AC  B7     015A    535          DECW    TPA_Q_PNAM(AP)                          ;DON'T COUNT THE '%'
                             04  04     015D    536          RET
                                 015E    537
                                 015E    538 ;++
                                 015E    539 ; THIS ROUTINE STORES THE ADDRESS OF THE START OF A USER SPEC
                                 015E    540 ; AND CLEARS THE DESCRIPTORS IN THE TPARSE EXTENSION BLOCK
                                 015E    541 ;--
                                 015E    542          .DEBUG  SET_SPST
                                 015E    543 SET_SPST:
                          0000   015E    544          .WORD   0
                4C AC     0C AC  D0     0160    545          MOVL    TPA$L_STRINGPTR(AP),TPA_L_SPST(AP);SAVE START
                          3C AC  D4     0165    546          CLRL    TPA_Q_PNAM(AP)          ;CLEAR PROTOCOL NAME LENGTH
                          34 AC  D4     0168    547          CLRL    TPA_Q_LOGNAM(AP)        ;CLEAR LOGNAM DESCRIPTOR
                          50 AC  D4     016B    548          CLRL    TPA_Q_UNAME(AP)         ;CLEAR USERNAME DESCRIPTOR
                             04  04     016E    549          RET
                                 016F    550 ;++
                                 016F    551 ; THIS TPARSE ACTION ROUTINE REPORTS A SYNTAX ERROR.
                                 016F    552 ;--
                                 016F    553          .DEBUG  REPORT_SYNTAX_ERROR
                                 016F    554 REPORT_SYNTAX_ERROR:
                          0000   016F    555          .WORD   0
                      4C AC      DD     0171    556          PUSHL   TPA_L_SPST(AP)          ;FORM A DESCRIPTOR
             7E       14 AC      6E     C3     0174    557          SUBL3   (SP),TPA$L_TOKENPTR(AP),-(SP) ;COMPUTE LENGTH OF STRING
                                 04     12     0179    558          BNEQ    10$             ;BRANCH IF SOMETHING THERE
                                 6E     D6     017B    559          INCL    (SP)            ;COUNT THE ILLEGAL TERMINATOR TOO
                                 07     11     017D    560          BRB     20$
                          08 AC  D5     017F    561 10$:     TSTL    TPA$L_STRINGCNT(AP)     ;DID WE GET HERE AT END OF STRING?
                                 02     13     0182    562          BEQL    20$             ;IF EQL YES
                                 6E     D7     0184    563          DECL    (SP)            ;NO--DON'T PRINT THE TERMINATOR
                                 5E     DD     0186    564 20$:     PUSHL   SP              ;STACK ADDRESS OF DESCRIPTOR
                                 01     DD     0188    565          PUSHL   #1              ;ONE FAO ARG
                    007E10FA     8F     DD     018A    566          PUSHL   #<SHR$_SYNTAX!<MAIL$_FACILITY@STS$V_FAC_NO>!STS$K_ERROR>
          00000000'GF            03     FB     0190    567          CALLS   #3,G^LIB$SIGNAL
                20 AC            00     D0     0197    568          MOVL    #STS$K_WARNING,TPA$L_PARAM(AP) ;SET WARNING STATUS
                   50            01     D0     019B    569          MOVL    #1,R0
                                 04     19E    570          RET
                                 019F    571 ;++
                                 019F    572 ; THIS TPARSE ACTION ROUTINE STORES THE USERNAME IN TPA$L_TOKENCNT/TOKENPTR
                                 019F    573 ; INTO THE TPARSE BLOCK EXTENSION
                                 019F    574 ;--
                                 019F    575 STORE_UNAME:
                          0000   019F    576          .WORD   0
                50 AC     10 AC  7D     01A1    577          MOVQ    TPA$L_TOKENCNT(AP),TPA_Q_UNAME(AP)
                             04  04     01A6    578          RET
                                 01A7    579 ;++
                                 01A7    580 ; THIS TPARSE ACTION ROUTINE STORES THE LOGICAL NAME IN TPA$L_TOKENCNT/TOKENPTR
                                 01A7    581 ; INTO THE TPARSE BLOCK EXTENSION
                                 01A7    582 ;
```

```
                                01A7   583 ;--
                                01A7   584
                                01A7   585         ASSUME  TPA$L_TOKENPTR EQ TPA$L_TOKENCNT+4
                                01A7   586
                                01A7   587 STORE_LOGNAME:
                          0000  01A7   588         .WORD   0
                 34 AC    10 AC 7D  01A9   589         MOVQ    TPA$L_TOKENCNT(AP),TPA_Q_LOGNAM(AP)     ;COPY DESCRIPTOR
                                04  01AE   590         RET
                                01AF   591 ;++
                                01AF   592 ; THIS ROUTINE RESETS TPA_L_LCOUNT WHEN A NEW ADDRESSEE IS ENCOUNTERED
                                01AF   593 ;
                                01AF   594 ;--
                                01AF   595
                                01AF   596 RESET_LCOUNT:
                      51    5C  D0  01AF   597         MOVL    AP,R1                                   ;GET PTR TO TPARSE BLOCK
                   24 A1    0A  D0  01B2   598 10$:     MOVL    #10,TPA_L_LCOUNT(R1)                    ;RESET LOOP COUNT
                   5A A1    01  8A  01B6   599         BICB2   #TPA_M_TERMINAL,TPA_W_FLAGS(R1) ;RESET TERMINAL ATTRIBUTE FLAG
                   51    48 A1  D0  01BA   600         MOVL    TPA_L_PTPA(R1),R1                       ;GO UP AND GET THEM ALL
                         F2  12  01BE   601         BNEQ    10$
                             05  01C0   602         RSB
                                01C1   603 ;++
                                01C1   604 ; ALREADY_IN_LIST - See if addressee already in list
                                01C1   605 ;
                                01C1   606 ; Inputs:
                                01C1   607 ;
                                01C1   608 ;       AP = pointer to tparse arg block
                                01C1   609 ;
                                01C1   610 ; outputs:
                                01C1   611 ;
                                01C1   612 ;       true if already entered, false if not
                                01C1   613 ;
                                01C1   614 ;       note**this routine is bsbw'd to to save code space
                                01C1   615 ;
                                01C1   616 ALREADY_IN_LIST:
                      50    44 AC  D0  01C1   617         MOVL    TPA_L_CNCT(AP),R0               ;GET CNCT BLOCK ADDRESS
                            28 A0  9F  01C5   618         PUSHAB  CNCT_Q_ADRLST(R0)               ;STACK ADDRESS OF ADDRESSEE LISTHEAD
                            50 AC  9F  01C8   619         PUSHAB  TPA_Q_UNAME(AP)                 ;STACK USERNAME DESCRIPTOR ADDR
                            34 AC  9F  01CB   620         PUSHAB  TPA_Q_LOGNAM(AP)                ;STACK NODENAME DESCR ADDR
                            3C AC  9F  01CE   621         PUSHAB  TPA_Q_PNAM(AP)                  ;STACK PROTOCOL DESCR ADDR
                00000000'EF     04  FB  01D1   622         CALLS   #4,MAIL$ADDR_EXISTS             ;SEE IF EXISTS IN ADDRESSEE LIST
                                05  01D8   623         RSB                                             ;RETURN WITH RESULT IN R0
```

```
                     01D9   625          .SBTTL  ENTER_ADDR - ENTER ADDRESSEE IN ADDRESSEE LIST
                     01D9   626 ;++
                     01D9   627 ;
                     01D9   628 ; ENTER_ADDR - ENTER ADDRESSEE IN ADDRESSEE LIST
                     01D9   629 ;
                     01D9   630 ; THIS ACTION ROUTINE IS CALLED TO ENTER AN ADDRESSEE IN THE ADDRESSEE
                     01D9   631 ; LIST.  FIRST, AN ATTEMPT IS MADE TO TRANSLATE THE FIRST SYMBOL IN THE
                     01D9   632 ; ADDRESSEE NAME.  FOR A REMOTE ADDRESSEE THE FIRST SYMBOL IS A NODE NAME,
                     01D9   633 ; OTHERWISE, IT'S JUST A LOGICAL NAME OR A USERNAME.
                     01D9   634 ;
                     01D9   635 ; IF THE SYMBOL TRANSLATES, THE TRANSLATION IS PREFIXED TO THE REST OF THE
                     01D9   636 ; ADDRESSEE STRING (IF ANY) AND THE PARSER IS RECURSIVELY CALLED TO PARSE THE
                     01D9   637 ; NEW ADDRESSEE SPEC AND ENTER THE RESULT IN THE ADDRESSEE LIST.
                     01D9   638 ;
                     01D9   639 ; IF THE SYMBOL DOESN'T TRANSLATE, AND IT'S A USERNAME, LOOK IT UP IN
                     01D9   640 ; SYS$SYSTEM:VMSMAIL.DAT TO SEE IF IT IS FORWARDED.  IF IT IS, TAKE THE
                     01D9   641 ; FORWARDING ADDRESS AND CALL THE PARSER RECURSIVELY TO PARSE THE NEW
                     01D9   642 ; ADDRESSEE SPEC.
                     01D9   643 ;
                     01D9   644 ; IF THE SYMBOL DOESN'T TRANSLATE, AND IT WAS A NODE NAME, A LOGICAL LINK
                     01D9   645 ; IS CREATED TO A SLAVE MAIL JOB AND THE REST OF THE ADDRESSEE IS PASSED
                     01D9   646 ; TO THE SLAVE MAIL JOB FOR VERIFICATION.  IF THE SYMBOL ISN'T A NODE NAME,
                     01D9   647 ; IT MUST BE A USER NAME, AND IT'S VERIFIED BY LOOKING FOR A CORRESPONDING
                     01D9   648 ; AUTHORIZATION FILE RECORD.
                     01D9   649 ;
                     01D9   650 ; THE ADDRESSEE IS THEN ENTERED INTO THE ADDRESSEE LIST FOR LATER USE
                     01D9   651 ; WHEN THE MAIL IS ACTUALLY TO BE SENT.
                     01D9   652 ;
                     01D9   653 ; INPUTS:
                     01D9   654 ;
                     01D9   655 ;       TPA_Q_LOGNAM(AP) = NODENAME/USERNAME LOGICAL NAME DESCRIPTOR
                     01D9   656 ;
                     01D9   657 ;       IF TPA_Q_LOGNAM(AP) IS A NODE NAME:
                     01D9   658 ;
                     01D9   659 ;               TPA_Q_UNAME(AP) = DESCRIPTOR OF ADDRESSEE NAME
                     01D9   660 ;
                     01D9   661 ; OUTPUTS:
                     01D9   662 ;
                     01D9   663 ;       ADDRESSEE LIST ENTRY CREATED AND ENTERED INTO ADDRESSEE LIST.
                     01D9   664 ;--
                     01D9   665 CASE_BLIND:
           02000000  01D9   666          .LONG   LNM$M_CASE_BLIND
                     01DD   667 ENTER_ADDR:                             ;ENTER  ADDRESSEE IN LIST
                01FC 01DD   668          .WORD   ^M<R2,R3,R4,R5,R6,R7,R8>;(ENTRY MASK)
              58 AC   B6 01DF   669          INCW    TPA_W_UENT(AP)              ;COUNT ANOTHER TIME THROUGH
        57    34 AC   7E 01E2   670          MOVAQ   TPA_Q_LOGNAM(AP),R7         ;GET ADDR OF LOGICAL NAME DESC
0C 5A AC      00   E0 01E6   671          BBS     #TPA_V_TERMINAL,TPA_W_FLAGS(AP),4$ ;BRANCH IF PREVIOUSLY TERMINAL
5F 8F    04 B7   91 01EB   672          CMPB    @4(R7),#^A'_'               ;IS IT PREFIXED WITH UNDERSCORE?
           08   12 01F0   673          BNEQ    41$                         ;BR IF NOT
           67   B7 01F2   674          DECW    (R7)                        ;REMOVE UNDERSCORE FROM COUNT
        04 A7   D6 01F4   675          INCL    4(R7)                       ;AND FROM ADDRESS OF LOGICAL NAME
         00F5   31 01F7   676 4$:       BRW     CHECK_ADDR                  ;SKIP TRANSLATION
                01FA   677 ;
                01FA   678 ; SEE IF NODENAME/USERNAME IS PREFIXED WITH A QUOTE IF SO
                01FA   679 ; STRIP QUOTES FROM FIRST AND LAST POSITIONS IN STRING
                01FA   680 ;
     22 04 B7   91 01FA   681 41$:     CMPB    @4(R7),#^A/"/               ;IS IT PREFIXED WITH A QUOTE
```

```
                     06   12  01FE    682          BNEQ    5$                              ;NO, THEN BRANCH
                 67  02   A2  0200    683          SUBW2   #2,(R7)                         ;REMOVE QUOTES FROM COUNT
                   04 A7   D6  0203    684          INCL    4(R7)                           ;SKIP OVER FIRST QUOTE
                           0206    685 ;
                           0206    686 ; SEE IF NODENAME/USERNAME TRANSLATES AS A LOGICAL NAME. IF IT IS, USE
                           0206    687 ; TRANSLATION AND RECURSIVELY CALL ADD_ADDR.
                           0206    688 ;
              5E   FF01    CE   9E  0206    689 5$:      MOVAB   -LNM$C_NAMLENGTH(SP),SP;CREATE TRANSLATION BUFFER
                           5E   DD  020B    690          PUSHL   SP                      ;CREATE BUFFER DESCRIPTOR
                7E   FF    8F   9A  020D    691          MOVZBL  #LNM$C_NAMLENGTH,-(SP)  ;...
                     56    5E   D0  0211    692          MOVL    SP,R6                   ;GET ADDRESS OF DESC
                           7E   D4  0214    693          CLRL    -(SP)                   ;CREATE LONGWORD TO GET ATTRIBUTES
                     55    5E   D0  0216    694          MOVL    SP,R5                   ;SAVE THE ADDRESS
                           7E   D4  0219    695          CLRL    -(SP)                   ;CREATE TRNLNM ITEM LIST
                           7E   D4  021B    696          CLRL    -(SP)                   ; NO RETURN ADDRESS
                           55   DD  021D    697          PUSHL   R5                      ; ADDRESS TO RETURN ATTRIBUTES
                00030004   8F   DD  021F    698          PUSHL   #<LNM$_ATTRIBUTES@16>!4 ;STACK BUFFER LENGTH AND ITEM
                           56   DD  0225    699          PUSHL   R6                      ; RETURN LENGTH ADDRESS
                     04    A6   DD  0227    700          PUSHL   4(R6)                   ; BUFFER ADDRESS
                     50    66   3C  022A    701          MOVZWL  (R6),R0                 ; GET BUFFER LENGTH
          7E   50   00020000   8F   C1  022D    702          ADDL3   #<LNM$_STRING@16>,R0,-(SP) ;FINISH ITEM LIST
                     51    5E   D0  0235    703          MOVL    SP,R1                   ;SAVE ADDRESS
                           0238    704          $TRNLNM_S ATTR=CASE_BLIND,-
                           0238    705                    TABNAM=MAIL$SD_LNM_FILE_DEV,-
                           0238    706                    LOGNAM=(R7),-
                           0238    707                    ITMLST=(R1)
                        71 50   E9  024E    708          BLBC    R0,20$                  ;BR IF FAILURE
          50   3C AC     50 AC   A9  0251    709          BISW3   TPA_Q_UNAME(AP),TPA_Q_PNAM(AP),R0 ;WAS THERE MORE THAN USERNAM?
                           0B   12  0257    710          BNEQ    6$                      ;BRANCH IF YES
                65   0200   8F   B3  0259    711          BITW    #LNM$M_TERMINAL,(R5)    ;NO--WAS THIS TERMINAL?
                           04   13  025E    712          BEQL    6$                      ;IF EQL NO
                   5A AC    01   88  0260    713          BISB2   #TPA_M_TERMINAL,TPA_W_FLAGS(AP) ;YES--FLAG FOR NEXT TIME
                           56   DD  0264    714 6$:      PUSHL   R6                      ;STACK ADDRESS OF DESCRIPTOR
00000000'EF                01   FB  0266    715          CALLS   #1,MAIL$UPCASE_Q        ;MAKE SURE IT'S UPPER CASE
 50    50 AC                66   C1  026D    716 7$:      ADDL3   (R6),TPA_Q_UNAME(AP),R0 ;COMPUTE SIZE OF NEW ADDRESSEE
       51    3C AC          3C   3C  0272    717          MOVZWL  TPA_Q_PNAM(AP),R1       ;GET PROTOCOL SIZE
                           05   13  0276    718          BEQL    71$                     ;BRANCH IF NONE
             50    51      C0   0278    719          ADDL2   R1,R0                   ;INCLUDE PROTOCOL SIZE
                           50   D6  027B    720          INCL    R0                      ; AND ALLOW DELIMITER (%)
             5E    50      C2   027D    721 71$:     SUBL    R0,SP                   ;CREATE NEW ADDRESSEE BUFFER
                           5E   DD  0280    722          PUSHL   SP                      ;CREATE BUFFER DESCRIPTOR
                           50   DD  0282    723          PUSHL   R0                      ;...
          53    04 AE      D0   0284    724          MOVL    4(SP),R3                ;GET BUFFER ADDRESS
                3C AC      B5   0288    725          TSTW    TPA_Q_PNAM(AP)          ;PROTOCOL SPECIFIED?
                           09   13  028B    726          BEQL    75$                     ;IF EQL NO
                3C AC      28   028D    727          MOVC    TPA_Q_PNAM(AP),-        ;COPY PROTOCOL SPEC FIRST
          63    40 BC            0290    728                  @TPA_Q_PNAM+4(AP),(R3)
                   83      90   0293    729          MOVB    #^A/%/,(R3)+            ;STORE DELIMITER
 63    04 B6                66   28  0296    730 75$:     MOVC    (R6),@4(R6),(R3)        ;MOVE TRANSLATION TO BUFFER
                50 AC      28   029B    731          MOVC    TPA_Q_UNAME(AP),-       ;MOVE REST OF OLD ADDRESSEE TO BUFFER
          63    54 BC            029E    732                  @TPA_Q_UNAME+4(AP),(R3)
                50 AC      B5   02A1    733          TSTW    TPA_Q_ONAME(AP)         ;NODENAME SPECIFIED ALSO?
                           05   13  02A4    734          BEQL    76$                     ;IF EQL NO, LOGNAM ALREADY SET UP
    34 AC      50 AC      7D   02A6    735          MOVQ    TPA_Q_UNAME(AP),TPA_Q_LOGNAM(AP) ;POSITION USERNAME DESC.
    28 AC      34 AC      7D   02AB    736 76$:     MOVQ    TPA_Q_LOGNAM(AP),TPA_Q_ONAME(AP) ;AND SET ORIGINAL NAME
                           5E   DD  02B0    737          PUSHL   SP                      ;SET ADDRESS OF ADDRESSEE DESC
     FD49 CF               01   FB  02B2    738          CALLS   #1,MAIL_ADD_ADDR        ;ADD NEW ADDRESSEE TO LIST
```

```
                07 50    E8  02B7   739          BLBS    R0,10$                          ;BR IF SUCCESS
             20 AC 50    D0  02BA   740 8$:      MOVL    R0,TPA$L_PARAM(AP)              ;SAVE STATUS
                50 01    D0  02BE   741          MOVL    #1,R0                           ;ALWAYS SET SUCCESS
                         04  02C1   742 10$:     RET                                     ;
                             02C2   743 ;
                             02C2   744 ; SEE IF USERNAME EXISTS IN SYS$SYSTEM:VMSMAIL.DAT, AND IF SO,
                             02C2   745 ; FEED FORWARDING ADDRESS BACK INTO MAIL_ADD_ADDR
                             02C2   746 ;
      50   3C AC   50 AC    A9  02C2   747 20$:     BISW3   TPA_Q_UNAME(AP),TPA_Q_PNAM(AP),R0 ;WAS THIS A NODE NAME?
                             02C8   748                                                ;(OR A PROTOCOL?)
                   25 12     02C8   749          BNEQ    CHECK_ADDR                      ;IF NEQ YES. DON'T BOTHER LOOKING
                08 AE 9F     02CA   750          PUSHAB  8(SP)                           ;STACK OUTPUT BUFFER ADDR
                   67 9F     02CD   751          PUSHAB  (R7)                            ;STACK DESCRIPTOR FOR NAME
             0000'CF   02    FB  02CF   752          CALLS   #2,W^MAIL$CHECK_FORW            ;SEE IF USER IS FORWARDING
                18 50 E9     02D4   753          BLBC    R0,CHECK_ADDR                   ;BRANCH IF NO FORWARDING
          66   4B AE 9A     02D7   754          MOVZBL  VMD_B_FWDLNG+8(SP),(R6) ;SETUP DESCRIPTOR FOR NEW DESTINATION
       04 A6   4C AE 9E     02DB   755          MOVAB   VMD_T_FWDNAM+8(SP),4(R6); (ASSUME R6 STILL POINTS AT BUFFER)
                   8B 11     02E0   756          BRB     7$                              ;PROCESS NEW DESTINATION
                             02E2   757          .ENABLE LOCAL_BLOCK
                             02E2   758 CHECK_PROT:
                50 AC B5     02E2   759          TSTW    TPA_Q_UNAME(AP)                 ;WAS NODENAME ALSO SPECIFIED?
                   51 12     02E5   760          BNEQ    8$                              ;IF NEQ YES
             50 AC 67 7D     02E7   761          MOVQ    (R7),TPA_Q_UNAME(AP)            ;NO, SET USERNAME
                   67 D4     02EB   762          CLRL    (R7)                            ;AND ZERO NODENAME LENGTH
                   49 11     02ED   763          BRB     8$                              ; CONTINUE
                             02EF   764 ;
                             02EF   765 ; CHECK IF ADDRESSEE EXISTS
                             02EF   766 ;
                             02EF   767 CHECK_ADDR:                                  ;
          53   44 AC D0      02EF   768          MOVL    TPA_L_CNCT(AP),R3               ;GET CNCT BLOCK ADDRESS
               3C AC B5      02F3   769          TSTW    TPA_Q_PNAM(AP)                  ;PROTOCOL SPECIFIED?
                  EA 12      02F6   770          BNEQ    CHECK_PROT                      ;IF NEQ YES
               50 AC B5      02F8   771          TSTW    TPA_Q_UNAME(AP)                 ;WAS NODENAME SPECIFIED?
                  06 12      02FB   772          BNEQ    5$                              ;IF NEQ YES
                00C5 31      02FD   773          BRW     10$                             ;NO--GO HANDLE LOCAL USER
                00CB 31      0300   774 1$:     BRW     15$                             ;BRANCH AID
                             0303   775 ;
                             0303   776 ; IF DESTINATION NODE IS SAME AS OUR NODE, THEN IGNORE AND SEND
                             0303   777 ; REST OF ADDRESSEE NAME BACK INTO ADD_ADDR
                             0303   778 ;
          51   00000000'EF  9E  0303   779 5$:     MOVAB   MAIL$Q_SYSNODE,R1               ;POINT TO OUR NODE NAME DESCRIPTOR
04 B1  61   00   04 B7    67  2D  030A   780          CMPC5   (R7),@4(R7),#0,(R1),@4(R1);DEST. NODE SAME AS OUR NODE?
                   18 12      0312   781          BNEQ    7$                              ;BRANCH IF NOT
             7E   50 AC 7D     0314   782 6$:     MOVQ    TPA_Q_UNAME(AP),-(SP)   ;YES--STACK DESCRIPTOR OF ADDRESSEE
                  5E DD        0318   783          PUSHL   SP                              ;STACK ADDRESS OF DESCRIPTOR
          34 AC   50 AC 7D     031A   784          MOVQ    TPA_Q_UNAME(AP),TPA_Q_LOGNAM(AP) ;SET IN CASE FORWARD ERROR
          28 AC   50 AC 7D     031F   785          MOVQ    TPA_Q_UNAME(AP),TPA_Q_ONAME(AP) ;SET ORIGINAL NAME
           FCD7 CF   01 FB     0324   786          CALLS   #1,MAIL_ADD_ADDR                ;RECURSE TO PEEL OFF NEXT NODE NAME
                00C8 31        0329   787          BRW     20$                             ;GO RETURN
                  57 DD        032C   788 7$:     PUSHL   R7                              ;STACK NODE DESCRIPTOR ADDRESS
          00000000'EF   01 FB  032E   789          CALLS   #1,MAIL$CHK_CLUNODES    ;SEE IF SENDING TO NODE IN OUR CLUSTER
                DC 50 E8       0335   790          BLBS    R0,6$                           ;BRANCH IF YES
                 FE86 30       0338   791 8$:     BSBW    ALREADY_IN_LIST                 ;SEE IF THIS ONE ALREADY ENTERED
                C2 50 E8       033B   792          BLBS    R0,1$                           ;IF SO, ALL DONE
                30 AC DF       033E   793          PUSHAL  TPA_L_LLNK(AP)          ;SET ADDRESS OF RETURN ADDRESS BUFFER
             53   44 AC D0     0341   794          MOVL    TPA_L_CNCT(AP),R3       ;REFETCH CNCT BLOCK BECAUSE OF CMPC
                  53 DD        0345   795          PUSHL   R3                              ;STACK CNCT BLOCK ADDRESS
```

```
                67   7F  0347   796          PUSHAQ  (R7)                            ;SET ADDRESS OF NODENAME DESCRIPTOR
             3C AC   7F  0349   797          PUSHAQ  TPA_Q_PNAM(AP)                  ;STACK PROTOCOL NAME ADDRESS
00000000'EF    04   FB  034C   798          CALLS   #4,MAIL$CREATELINK              ;CREATE A LOGICAL LINK TO THE NODE
             59 50   E9  0353   799          BLBC    R0,9$                           ;BRANCH IF FAILURE
          52   30 AC   D0  0356   800          MOVL    TPA_L_LLNK(AP),R2               ;GET THE LNK BLOCK ADDRESS
       30 2E A2   02   E0  035A   801          BBS     #LNK_V_ALTP,LNK_B_FLAGS(R2),81$ ;BRANCH IF ALTERNATE PROTOCOL
             3C AC   B5  035F   802          TSTW    TPA_Q_PNAM(AP)                  ;NO--WAS PROTOCOL SPECIFIED?
                2B   13  0362   803          BEQL    81$                             ;IF EQL NO
                        0364   804 ;
                        0364   805 ; SPECIAL "%NODNAME::"TASK=" CONNECTION.  WE MUST RECOMBINE THE NODE
                        0364   806 ; AND USERNAME
                        0364   807 ;
          5E    FF01 CE   9E  0364   808          MOVAB   -LNM$C_NAMLENGTH(SP),SP ;ALLOCATE A BUFFER
                   58   5E  D0  0369   809          MOVL    SP,R8                   ;SAVE ITS ADDRESS
                   3C   BB  036C   810          PUSHR   #^M<R2,R3,R4,R5>        ;SAVE SOME REGISTERS
    68    04 B7    67   28  036E   811          MOVC3   (R7),@4(R7),(R8)        ;COPY THE NODE NAME
                   67   B5  0373   812          TSTW    (R7)                    ;WAS THERE A NODE NAME?
                   05   13  0375   813          BEQL    801$                    ;IF EQL NO
          83    3A3A 8F   B0  0377   814          MOVW    #^A/::/,(R3)+           ;TERMINATE THE NODE NAME
   63    54 BC    50   AC  28  037C   815 801$:    MOVC3   TPA_Q_UNAME(AP),@TPA_Q_UNAME+4(AP),(R3) ;AND THE USERNAME
      50 AC    53   58  C3  0382   816          SUBL3   R8,R3,TPA_Q_UNAME(AP)   ;CREATE NEW DESCRIPTOR OF USERNAME
            54 AC   58  D0  0387   817          MOVL    R8,TPA_Q_UNAME+4(AP)
                   67   7C  038B   818          CLRQ    (R7)                    ;AND CLEAR THE NODENAME
                   3C   BA  038D   819          POPR    #^M<R2,R3,R4,R5>        ;RESTORE REGISTERS
                08 A3   9F  038F   820 81$:     PUSHAB  CNCT_Q_FROMDESC(R3)     ;STACK ADDR OF FROM DESCRIPTOR
                30 AC   DD  0392   821          PUSHL   TPA_L_LLNK(AP)          ;STACK LOGICAL LINK ENTRY
00000000'EF        02   FB  0395   822          CALLS   #2,MAIL$NET_FROM        ;SEND "FROM" STRING
                10 50   E9  039C   823          BLBC    R0,9$                   ;BRANCH IF FAILURE
                50 AC   DF  039F   824          PUSHAL  TPA_Q_UNAME(AP)         ;SET ADDRESS OF ADDRESSEE DESC
                30 AC   DD  03A2   825          PUSHL   TPA_L_LLNK(AP)          ;SET ADDRESS OF LOGICAL LINK ENTRY
00000000'EF        02   FB  03A5   826          CALLS   #2,MAIL$NET_ADDR        ;CHECK IF ADDRESSEE EXISTS
                77 50   E8  03AC   827          BLBS    R0,CREATE_ADDR          ;BR IF SUCCESS
                50   DD  03AF   828 9$:      PUSHL   R0                      ;SAVE ERROR CODE
          52    50 AC   9E  03B1   829          MOVAB   TPA_Q_UNAME(AP),R2      ;GET USERNAME DESCR ADDR
                 00A7   30  03B5   830          BSBW    CREATE_ENTRY            ;ALLOCATE USERNAME BLOCK
                04 50   E9  03B8   831          BLBC    R0,91$                  ;BRANCH IF ERROR
          1C A1    01   88  03BB   832          BISB2   #ADR_M_NOSEND,ADR_B_FLAGS(R1) ;SET NOSEND TO THIS USER
                   50 8ED0  03BF   833 91$:     POPL    R0                      ;RESTORE ERROR CODE
                 002F   31  03C2   834 92$:     BRW     20$                     ;
                        03C5   835
          50 AC    67   7D  03C5   836 10$:     MOVQ    (R7),TPA_Q_UNAME(AP)    ;MOVE USERNAME DESC TO COMMON DESC
                   67   7C  03C9   837          CLRQ    (R7)                    ;CLEAR NODE NAME
                 FDF3   30  03CB   838          BSBW    ALREADY_IN_LIST         ;SEE IF ALREADY ENTERED
                27 50   E8  03CE   839 15$:     BLBS    R0,25$                  ;IF SO, ALL DONE
                30 AC   D4  03D1   840          CLRL    TPA_L_LLNK(AP)          ;LOCAL ADDRESSEE
                50 AC   7F  03D4   841          PUSHAQ  TPA_Q_UNAME(AP)         ;SET ADDRESS OF USERNAME DESC
00000000'EF        01   FB  03D7   842          CALLS   #1,MAIL$GET_USER        ;CHECK IF USER EXISTS
                13 50   E9  03DE   843          BLBC    R0,20$                  ;BRANCH IF USER DOES NOT EXIST
50    00000000'EF      D0  03E1   844          MOVL    MAIL$T_UAF_BUFF,R0      ;GET BUFFER ADDRESS
      00000000'8F      E0  03E8   845          BBS     #UAF$V_NOMAIL,-         ;BRANCH IF MAIL DISABLED
       0A 0000'C0          03EE   846                  UAF$L_FLAGS(R0),20$
                32   11  03F2   847          BRB     CREATE_ADDR             ;NOW ENTER IN ADDRESSEE LIST
                        03F4   848 20$:
          20 AC    50   D0  03F4   849          MOVL    R0,TPA$L_PARAM(AP)      ;SAVE FAILURE STATUS
             50    01   D0  03F8   850 25$:     MOVL    #1,R0                   ;SET SUCCESS TO CONTINUE PARSE
                   04   03FB   851          RET                             ;
                        03FC   852 ;
```

```
                          03FC   853 ; USER IS DISABLED FROM RECEIVING MAIL
                          03FC   854 ;
                    50 AC   9F    03FC   855 30$:    PUSHAB  TPA_Q_UNAME(AP)          ;STACK ADDRESS OF NAME DESCRIPTOR
                       01   DD    03FF   856         PUSHL   #1                       ;STACK COUNT OF ARGUMENTS
                007E8082 8F   DD    0401   857         PUSHL   #MAIL$_USERDSABL         ;SIGNAL THAT USER CANNOT RECEIVE MAIL
          00000000'GF    03   FB    0407   858         CALLS   #3,G^LIB$SIGNAL
                52 50 AC   9E    040E   859         MOVAB   TPA_Q_UNAME(AP),R2       ;GET USERNAME DESCR ADDR
                  004A     30    0412   860         BSBW    CREATE_ENTRY             ;ALLOCATE USERNAME BLOCK
                04 50      E9    0415   861         BLBC    R0,35$                   ;BRANCH IF ERROR
              1C A1    01   88    0418   862         BISB2   #ADR_M_NOSEND,ADR_B_FLAGS(R1) ;SET NOSEND TO THIS USER
          50  007E8082 8F   D0    041C   863 35$:    MOVL    #MAIL$_USERDSABL,R0      ;SET RETURN STATUS
                   FFCE    31    0423   864         BRW     20$                      ;GO RETURN
                           0426   865         .DISABLE LOCAL_BLOCK
                           0426   866 ;
                           0426   867 ; CREATE ADDRESSEE LIST ENTRY FOR ADDRESSEE
                           0426   868 ;
                           0426   869 CREATE_ADDR:                                  ;CREATE ADDRESSEE LIST ENTRY
                52 50 AC   9E    0426   870         MOVAB   TPA_Q_UNAME(AP),R2       ;GET ADDR OF USERNAME DESCRIPTOR
                  0032     30    042A   871         BSBW    CREATE_ENTRY             ;CREATE THE ADR LIST ENTRY
                1C 50      E9    042D   872         BLBC    R0,10$                   ;BR IF FAILURE
              52    51     D0    0430   873         MOVL    R1,R2                    ;GET ADDR OF NEW ENTRY
                20 AC      D0    0433   874         MOVL    TPA_L_LLNK(AP),-         ;SET ADDR OF LOGICAL LINK ENTRY
                08 A2            0436   875                 ADR_L_LLNK(R2)           ;
                38 A3      D6    0438   876         INCL    CNCT_L_NADDRS(R3)        ;COUNT ANOTHER ACTUAL ADDRESSEE
                08 A2      DD    043B   877         PUSHL   ADR_L_LLNK(R2)           ;STACK LOGICAL LINK ENTRY
          00000000'EF    01   FB    043E   878         CALLS   #1,MAIL$INCLINKADR       ;INCREMENT LINK ADDRESSEE COUNT IF NET
                 FD67    30    0445   879         BSBW    RESET_LCOUNT             ;RESET LOOP COUNTER
              50   01     D0    0448   880         MOVL    #1,R0                    ;SET SUCCESS
                           04    044B   881         RET                              ;
                           044C   882 10$:
                      50   DD    044C   883         PUSHL   R0                       ;SET FAILURE STATUS
                      50   DD    044E   884         PUSHL   R0                       ;SAVE STATUS
          00000000'GF    01   FB    0450   885         CALLS   #1,G^LIB$SIGNAL          ;SIGNAL THE FAILURE
                20 AC    8E   D0    0457   886         MOVL    (SP)+,TPA$L_PARAM(AP)    ;SAVE STATUS
                50       01   D0    045B   887         MOVL    #1,R0                    ;SET SUCCESS TO CONTINUE PARSE
                           04    045E   888         RET                              ;
                           045F   889
                           045F   890 ;
                           045F   891 ; ALLOCATE ENTRY AND LINK INTO LIST
                           045F   892 ;
                           045F   893 ; R2 = ADDR OF DESCRIPTOR OF USERNAME
                           045F   894 ; R3 = ADDR OF CNCT BLOCK
                           045F   895 ;
                           045F   896 CREATE_ENTRY:
                01FC 8F    BB    045F   897         PUSHR   #^M<R2,R3,R4,R5,R6,R7,R8>
                57   52    7D    0463   898         MOVQ    R2,R7                    ;SAVE USERNAME DESCR. AND CNCT ADDR
                7E   67    3C    0466   899         MOVZWL  (R7),-(SP)               ;GET LENGTH OF NAME
                6E   1E    C0    0469   900         ADDL2   #ADR_C_LENGTH,(SP)       ;COMPUTE SIZE OF ENTRY
                     7E    DF    046C   901         PUSHAL  -(SP)                    ;SET RETURN ADDR BUFFER
                   08 AE   DF    046E   902         PUSHAL  8(SP)                    ;STACK ADDR OF SIZE
          00000000'GF    02   FB    0471   903         CALLS   #2,G^LIB$GET_VM          ;ALLOCATE IT
                20 50      E9    0478   904         BLBC    R0,10$                   ;BRANCH IF ERROR
                56   8E    D0    047B   905         MOVL    (SP)+,R6                 ;GET ADDR OF NEW ENTRY
                     50 8ED0    047E   906         POPL    R0
66   1E   00    6E    00    2C    0481   907         MOVC5   #0,(SP),#0,#ADR_C_LENGTH,(R6) ;ZERO THE ENTRY
     1E A6   04 B7    67    28    0487   908         MOVC3   (R7),@4(R7),ADR_T_NAME(R6);COPY USERNAME INTO BLOCK
             1D A6    67    90    048D   909         MOVB    (R7),ADR_B_NAMLNG(R6)    ;SET ADDRESSEE NAME LENGTH
```

```
                2C B8    66   0E  0491   910          INSQUE  (R6),@CNCT_Q_ADRLST+4(R8) ;INSERT AT END OF LIST
                   51    56   D0  0495   911          MOVL    R6,R1                     ;SET ADDR OF BLOCK TO RETURN
                   50    01   D0  0498   912          MOVL    #1,R0                     ;SET RETURN STATUS
                   01FC  8F   BA  049B   913 10$:     POPR    #^M<R2,R3,R4,R5,R6,R7,R8>;RESTORE REGISTERS
                              05  049F   914          RSB
```

```
                                                F 4
                                                .SBTTL  GET_DIST_LIST - GET DISTRIBUTION LIST
                                                ;++
                                                ;
                                                ; GET_DIST_LIST - GET DISTRIBUTION LIST ACTION ROUTINE
                                                ;
                                                ; THIS ACTION ROUTINE IS CALLED TO OPEN AND READ A DISTRIBUTION LIST FILE.
                                                ;
                                                ; INPUTS:
                                                ;
                                                ;       TPA$L_TOKENCNT(AP) = SIZE OF FILE SPECIFICATION.
                                                ;       TPA$L_TOKENPTR(AP) = ADDR OF FILE SPECIFICATION.
                                                ;
                                                ; OUTPUTS:
                                                ;
                                                ;       EACH ADDRESSEE IS READ FROM THE DISTRIBUTION LIST FILE AND
                                                ;       ADDED TO THE ADDRESSEE LIST BY RECURSIVELY CALLING ADD_ADDR.
                                                ;       IF THIS IS THE FIRST (AND HENCE ONLY) ADDRESSEE ITEM IN THE
                                                ;       LIST, WE WILL MODIFY THE TO LIST TO CONTAIN THE FULL FILE SPEC.
                                                ;--
                                                GET_DIST_LIST:                          ;GET DISTRIBUTION LIST
                                        0FFC    .WORD   ^M<R2,R3,R4,R5,R6,R7,R8,R9,R10,R11> ;(ENTRY MASK)
                                                ;
                                                ; CREATE A FAB AND A RAB
                                                ;
                                        9E      MOVAB   -FAB$K_BLN(SP),R6       ;ALLOCATE FAB
                                        9E      MOVAB   -RAB$K_BLN(R6),R7       ;ALLOCATE RAB
                                        9E      MOVAB   -NAM$K_BLN(R7),R8       ;ALLOCATE NAM
                                        9E      MOVAB   -NAM$C_MAXRSS(R8),R9    ;AND A BUFFER
                                        D0      MOVL    R9,SP                   ;ALLOCATE THE SPACE
                                        2C      MOVC5   #0,(SP),#0,-
                                                #FAB$K_BLN+RAB$K_BLN+NAM$K_BLN+NAM$C_MAXRSS,(SP) ;ZERO FAB,RAB,NAM
                                                ASSUME  FAB$B_BLN EQ FAB$B_BID+1
                                        B0      MOVW    #FAB$K_BLN@8!FAB$C_BID,- ;SET BLOCK LENGTH AND ID FIELDS
                                                FAB$B_BID(R6)
                                        88      BISB    #FAB$M_GET,FAB$B_FAC(R6);GET ACCESS ONLY
                                        C8      BISL    #FAB$M_SQO,FAB$L_FOP(R6) ;SET SEQUENTIAL ACCESS ONLY
                                        D0      MOVL    R8,FAB$L_NAM(R6)        ;SET NAM ADDRESS
                                        D0      MOVL    #MAIL$_OPENIN,FAB$L_CTX(R6) ;SET ERROR CODE
                                                ASSUME  RAB$B_BLN EQ RAB$B_BID+1
                                        B0      MOVW    #RAB$K_BLN@8!RAB$C_BID,- ;SET BLOCK LENGTH AND ID FIELDS
                                                RAB$B_BID(R7)
                                                ASSUME  RAB$C_SEQ EQ 0
                                        D0      MOVL    R6,RAB$L_FAB(R7)        ;SET FAB ADDRESS
                                        D0      MOVL    #MAIL$_OPENIN,RAB$L_CTX(R7)
                                                ASSUME  NAM$B_BLN EQ NAM$B_BID+1
                                        B0      MOVW    #NAM$K_BLN@8!NAM$C_BID,-;SET UP NAM BLOCK
                                                NAM$B_BID(R8)
                                        D0      MOVL    R9,NAM$L_RSA(R8)
                                        D0      MOVL    R9,NAM$L_ESA(R8)
                                        90      MOVB    #NAM$C_MAXRSS,NAM$B_RSS(R8)
                                        90      MOVB    #NAM$C_MAXRSS,NAM$B_ESS(R8)
                                        9E      MOVAB   -MAIL$K_INBUFFSZ(SP),SP ;ALLOCATE USER BUFFER
                                        B0      MOVW    #MAIL$K_INBUFFSZ,RAB$W_USZ(R7) ;SET USER BUFFER SIZE
                                        D0      MOVL    SP,RAB$L_UBF(R7)        ;SET USER BUFFER ADDRESS
                                                ;
                                                ; OPEN THE DISTRIBUTION LIST FILE
                                                ;
```

```
                34 A6    10 AC    90  0512   973          MOVB    TPA$L_TOKENCNT(AP),FAB$B_FNS(R6) ;SET FILE NAME SIZE
                2C A6    14 AC    D0  0517   974          MOVL    TPA$L_TOKENPTR(AP),FAB$L_FNA(R6) ;SET FILE NAME ADDRESS
          30 A6    00000000'EF    9E  051C   975          MOVAB   DIST_TYPE,FAB$L_DNA(R6) ;SET DEFAULT FILE NAME ADDRESS
                35 A6    04     90  0524   976          MOVB    #DIST_TYPE_SIZ,FAB$B_DNS(R6) ;SET DEFAULT FILE NAME SIZE
                                     0528   977          $OPEN   FAB=(R6),ERR=UTIL$REPORT_IO_ERROR       ;OPEN THE FILE
                         12 50    E9  0537   978          BLBC    R0,5$                           ;BR IF FAILURE
                                     053A   979          $CONNECT RAB=(R7),ERR=UTIL$REPORT_IO_ERROR      ;CONNECT THE RAB
                         03 50    E8  0549   980          BLBS    R0,10$                          ;BRANCH ON SUCCESS
                          004A    31  054C   981 5$:      BRW     DIST_ERROR
                                     054F   982 ;
                                     054F   983 ; IF THIS IS THE ONLY ITEM IN THE TO LIST, THEN MODIFY IT TO CONTAIN
                                     054F   984 ; RESULTANT FILE SPEC
                                     054F   985 ;
                                     054F   986 10$:;    TSTW    TPA_W_UENT(AP)                  ;ANYTHING ENTERED?
                                     054F   987 ;        BNEQ    20$                             ;YES--TOO BAD
                                     054F   988 ;        PUSHL   R6                              ;STACK FAB ADDRESS
                                     054F   989 ;        CALLS   #1,G^UTIL$GETFILENAME           ;GET FILE SPEC FOR FILE
                                     054F   990 ;        PUSHAB  AT_STRING                       ;CREATE DESCRIPTOR
                                     054F   991 ;        PUSHL   #1                              ;OF '@'
                                     054F   992 ;        PUSHL   R0                              ;STACK FILE DESCRIPTOR ADDRESS
                                     054F   993 ;        PUSHAB  4(SP)                           ;PREFIX IT WITH '@'
                                     054F   994 ;        PUSHAB  MAIL$Q_TO                       ;AND THE TO-LIST ADDRESS
                                     054F   995 ;        CALLS   #3,G^STR$CONCAT                 ;MAKE A NEW TO-LIST
                                     054F   996 ;
                                     054F   997 ; READ ALL THE ADDRESSEES AND ADD THEM TO THE ADDRESSEE LIST
                                     054F   998 ;
          18 A7    007E10B2 8F    D0  054F   999 20$:     MOVL    #MAIL$_READERR,RAB$L_CTX(R7)
                             7E    7C  0557  1000          CLRQ    -(SP)                           ;CREATE ADDRESSEE DESC BUFFER
                                     0559  1001 DIST_LOOP:
                                     0559  1002          $GET    RAB=(R7),ERR=UTIL$REPORT_IO_ERROR ;GET A RECORD
                    2E 50    E9  0568  1003          BLBC    R0,DIST_ERROR                   ;BR IF EOF OR ERROR
                6E    22 A7    B0  056B  1004          MOVW    RAB$W_RSZ(R7),(SP)              ;SET RECORD SIZE IN DESCRIPTOR
                       E8    13  056F  1005          BEQLU   DIST_LOOP                       ;SKIP BLANK LINES
                   28 B7    21  91  0571  1006          CMPB    #^A"!",@RAB$L_RBF(R7)           ;IS RECORD A COMMENT?
                       E2    13  0575  1007          BEQLU   DIST_LOOP                       ;BR IF YES - IGNORE IT
             04 AE    28 A7    D0  0577  1008          MOVL    RAB$L_RBF(R7),4(SP)             ;SET RECORD ADDRESS IN DESCRIPTOR
                       5E    DD  057C  1009          PUSHL   SP                              ;SET ADDRESS OF DESCRIPTOR
          00000000'EF    01    FB  057E  1010          CALLS   #1,MAIL$UPCASE_Q                ;CONVERT RECORD TO UPPER CASE
                    44 AC    DD  0585  1011          PUSHL   TPA_L_CNCT(AP)                  ;STACK CNCT BLOCK ADDRESS
                    04 AE    9F  0588  1012          PUSHAB  4(SP)                           ;SET ADDRESS OF DESCRIPTOR
             FAD5 CF    02    FB  058B  1013          CALLS   #2,AT_LIST_ADD_ADDR             ;ADD THE ADDRESSEE TO THE LIST
                   C6 50    E8  0590  1014          BLBS    R0,DIST_LOOP                    ;BR IF SUCCESS
                20 AC    50    D0  0593  1015          MOVL    R0,TPA$L_PARAM(AP)              ;SAVE FAILURE STATUS
                       C0    11  0597  1016          BRB     DIST_LOOP                       ;
                                     0599  1017 ;
                                     0599  1018 ; ERROR ACCESSING THE FILE
                                     0599  1019 ;
                                     0599  1020 DIST_ERROR:
          00000000'8F    50    D1  0599  1021          CMPL    R0,#RMS$_EOF                    ;END OF FILE?
                         04    13  05A0  1022          BEQL    DIST_DONE                       ;YES, OK
                20 AC    50    D0  05A2  1023          MOVL    R0,TPA$L_PARAM(AP)              ;NO, SET CODE
                                     05A6  1024 DIST_DONE:                                   ;
                                     05A6  1025          $CLOSE  (R6)                            ;CLOSE THE FILE
                    50    01    D0  05AF  1026          MOVL    #1,R0                           ;SET SUCCESS FOR PARSER
                                 04  05B2  1027          RET
                                     05B3  1028
```

```
                                                H 4
                     05B3  1030          .SBTTL  MAIL$ANY_ADDRESSEES - SEE IF ANY ADDRESSEES FOUND
                     05B3  1031 ;++
                     05B3  1032 ; FUNCTIONAL DESCRIPTION:
                     05B3  1033 ;
                     05B3  1034 ;       LOOK THROUGH THE ADDRESSEE LIST AND SEE IF ANY FOUND
                     05B3  1035 ;       THAT WE WILL BE SENDING TO
                     05B3  1036 ;
                     05B3  1037 ; INPUTS:
                     05B3  1038 ;
                     05B3  1039 ;       4(AP)   ADDRESS OF ADDRESSEE LISTHEAD
                     05B3  1040 ;
                     05B3  1041 ; ROUTINE VALUE:
                     05B3  1042 ;
                     05B3  1043 ;       TRUE = THERE IS AT LEAST ONE
                     05B3  1044 ;       FALSE = NONE
                     05B3  1045 ;
                     05B3  1046 ;--
                     05B3  1047
                000C 05B3  1048          .ENTRY  MAIL$ANY_ADDRESSEES,^M<R2,R3>
     51   04 AC     D0   05B5  1049          MOVL    4(AP),R1                        ;GET ADDRESSEE LISTHEAD
          52   61   D0   05B9  1050          MOVL    (R1),R2                         ;GET FIRST ENTRY
          51   52   D1   05BC  1051 10$:     CMPL    R2,R1                           ;SEE IF DONE
               15   13   05BF  1052          BEQL    30$                             ;BRANCH IF YES
0B 1C A2       00   E0   05C1  1053          BBS     #ADR_V_NOSEND,ADR_B_FLAGS(R2),20$ ;BRANCH IF NOSEND SET
     50   08 A2   D0   05C6  1054          MOVL    ADR_C_CLNK(R2),R0               ;GET LOGICAL LINK DESCRIPTOR
               0D   13   05CA  1055          BEQL    40$                             ;BRANCH IF NONE--WE FOUND ONE
08 2E A0       01   E1   05CC  1056          BBC     #LNK_V_DEAD,LNK_B_FLAGS(R0),40$ ;BRANCH IF NOT DEAD--FOUND ONE
          52   62   D0   05D1  1057 20$:     MOVL    ADR_C_FLINK(R2),R2              ;GET NEXT ADDRESSEE
               E6   11   05D4  1058          BRB     10$                             ;AND CONTINUE
               50   D4   05D6  1059 30$:     CLRL    R0                              ;NONE FOUND
                    04   05D8  1060          RET
          50   01   D0   05D9  1061 40$:     MOVL    #1,R0
                    04   05DC  1062          RET
```

```
                     05DD  1064          .SBTTL  INITIALIZE ADDRESSEE LIST
                     05DD  1065 ;++
                     05DD  1066 ; FUNCTIONAL DESCRIPTION:
                     05DD  1067 ;
                     05DD  1068 ;       INITIALIZE ADDRESSEE LIST BY REMOVING ANY OLD ENTRIES FROM
                     05DD  1069 ;       THE LIST.
                     05DD  1070 ;
                     05DD  1071 ; CALLING SEQUENCE:
                     05DD  1072 ;       CALL MAIL$INI_ADDR(CNCTDESC)
                     05DD  1073 ;
                     05DD  1074 ; INPUT PARAMETERS:
                     05DD  1075 ;       CNCTDESC = ADDRESS OF THE CONNECT DESCRIPTOR.  IF THIS IS THE FIRST
                     05DD  1076 ;                  CALL, THE ADRLST MUST BE INITIALIZED.
                     05DD  1077 ;
                     05DD  1078 ; IMPLICIT INPUTS:
                     05DD  1079 ;       NONE
                     05DD  1080 ;
                     05DD  1081 ; OUTPUT PARAMETERS:
                     05DD  1082 ;       NONE
                     05DD  1083 ;
                     05DD  1084 ; IMPLICIT OUTPUTS:
                     05DD  1085 ;       NONE
                     05DD  1086 ;
                     05DD  1087 ; COMPLETION CODES:
                     05DD  1088 ;       NONE
                     05DD  1089 ;
                     05DD  1090 ; SIDE EFFECTS:
                     05DD  1091 ;       ADDRESSEE AND LOGICAL LINK LIST ENTRIES REMOVED
                     05DD  1092 ;       AND DEALLOCATED.
                     05DD  1093 ;
                     05DD  1094 ;--
                000C 05DD  1095          .ENTRY  MAIL$INI_ADDR,-         ;INITIALIZE ADDRESSEE LIST
                     05DF  1096                  ^M<R2,R3>               ;(ENTRY MASK)
          53   04 AC   D0  05DF  1097          MOVL    4(AP),R3            ;GET ADDRESS OF ADDRESSEE LIST
               38 A3   D4  05E3  1098          CLRL    CNCT_L_NADDRS(R3)   ;INIT NUMBER OF ACTUAL ADDRESSEES
          53   28 A3   9E  05E6  1099          MOVAB   CNCT_Q_ADRLST(R3),R3 ;POINT TO THE ADDRESSEE LIST HEAD
          52   00 B3   0F  05EA  1100 10$:     REMQUE  @(R3),R2            ;GET NEXT ENTRY
                    04   1D  05EE  1101          BVS     20$                 ;BR IF NONE
                    03   10  05F0  1102          BSBB    DEALLOC_ENTRY       ;DEALLOCATE THE ENTRY
                    F6   11  05F2  1103          BRB     10$                 ;
                    04       05F4  1104 20$:     RET
                             05F5  1105 ;
                             05F5  1106 ; DEALLOCATE AN ADDRESSEE LIST ENTRY
                             05F5  1107 ;
                             05F5  1108 DEALLOC_ENTRY:
                    52   DD  05F5  1109          PUSHL   R2                  ;SET ADDR OF ENTRY
          7E   1D   A2   9A  05F7  1110          MOVZBL  ADR_B_NAMLNG(R2),-(SP) ;STACK LENGTH OF NAME
               6E   1E   C0  05FB  1111          ADDL2   #ADR_C_LENGTH,(SP)  ;COMPUTE SIZE OF BLOCK
                    04   AE  DF  05FE  1112          PUSHAL  4(SP)               ;SET ADDR OF ADDR OF ENTRY
                    04   AE  DF  0601  1113          PUSHAL  4(SP)               ;SET ADDR OF SIZE OF ENTRY
00000000'GF         02       FB  0604  1114          CALLS   #2,G^LIB$FREE_VM    ;DEALLOCATE ENTRY
               5E   08       C0  060B  1115          ADDL2   #8,SP               ;CLEAR TWO LONGWORDS FROM STACK
                    05       060E  1116          RSB                         ;
                             060F  1117
                             060F  1118          .END
```

```
$$$CNT                  = 00000003
$$$FLG                  = FFFFFFFF
$$$KEY                  = FFFFFFFF
$$$KFG                  = FFFFFFFF
$$$MOD                  = 00000000
$$.TMP1                 = 00000001
$$.TMP2                 = 00000066
$$GBL                   = 00000000
$$KEYTAB                = 00000000 R      05
ACCESS_LIST               000000D0 R      04
ACCESS_SCAN               000000D2 R      04
ADDR                    = 00000004
ADDR_KEY                  00000000 RG     05
ADDR_STATE                00000000 RG     04
ADD_ADDR                  00000072 R      03
ADRDEF                  = 00000000
ADR_B_FLAGS             = 0000001C
ADR_B_NAMLNG            = 0000001D
ADR_C_LENGTH            = 0000001E
ADR_L_BLINK             = 00000004
ADR_L_FLINK             = 00000000
ADR_L_LLNK              = 00000008
ADR_L_SPARES            = 0000000C
ADR_M_NOSEND            = 00000001
ADR_S_ADRDEF            = 0000001E
ADR_S_SPARES            = 00000010
ADR_T_NAME              = 0000001E
ADR_V_NOSEND            = 00000000
ALREADY_IN_LIST           000001C1 R      03
AT_LIST_ADD_ADDR          00000065 R      03
AT_STRING                 00000004 R      02
CALL_TPARSE               00000133 R      03
CASE_BLIND                000001D9 R      03
CHECK_ADDR                000002EF R      03
CHECK_PROT                000002E2 R      03
CHECK_QUOTE               0000016A R      04
CHECK_SEP                 0000013E R      04
CH_SPACE                = 00000020
CH_TAB                  = 00000009
CNCTDEF                 = 00000000
CNCT_B_FILRAT           = 00000080
CNCT_B_FILRFM           = 00000081
CNCT_B_FLAGS            = 00000084
CNCT_B_TRAN             = 00000085
CNCT_C_BUFSIZ           = 00000200
CNCT_C_GETADDR          = 00000001
CNCT_C_GETMSG           = 00000004
CNCT_C_GETSENDER        = 00000000
CNCT_C_GETSUBJ          = 00000003
CNCT_C_GETTO            = 00000002
CNCT_C_LENGTH           = 00000286
CNCT_L_BLINK            = 00000004
CNCT_L_FILERR           = 00000048
CNCT_L_FILESTS          = 0000004C
CNCT_L_FILESTV          = 00000050
CNCT_L_FLINK            = 00000000
CNCT_L_MSGBLKS          = 0000003C
CNCT_L_NADDRS           = 00000038
CNCT_L_NRECS            = 00000064
CNCT_L_SPARES           = 00000068
CNCT_M_BLKMODE          = 00000004
CNCT_M_DISCONNECT       = 00000001
CNCT_M_SQ               = 00000002
CNCT_Q_ADRLST           = 00000028
CNCT_Q_CCDESC           = 00000020
CNCT_Q_ESDESC           = 00000054
CNCT_Q_FROMDESC         = 00000008
CNCT_Q_IOSB             = 00000040
CNCT_Q_LNKLST           = 00000030
CNCT_Q_SENDQ            = 0000005C
CNCT_Q_SUBJDESC         = 00000018
CNCT_Q_TODESC           = 00000010
CNCT_S_ADRLST           = 00000008
CNCT_S_BUFFER           = 00000200
CNCT_S_CCDESC           = 00000008
CNCT_S_CNCTDEF          = 00000286
CNCT_S_ESDESC           = 00000008
CNCT_S_FROMDESC         = 00000008
CNCT_S_IOSB             = 00000008
CNCT_S_LNKLST           = 00000008
CNCT_S_SENDQ            = 00000008
CNCT_S_SPARES           = 00000018
CNCT_S_SUBJDESC         = 00000008
CNCT_S_TODESC           = 00000008
CNCT_T_BUFFER           = 00000086
CNCT_T_RAB              = 00000286
CNCT_V_BLKMODE          = 00000002
CNCT_V_DISCONNECT       = 00000000
CNCT_V_SQ               = 00000001
CNCT_W_CHAN             = 00000082
CNCT_W_IOSB1            = 00000040
CNCT_W_IOSB2            = 00000042
CNCT_W_IOSB3            = 00000044
CNCT_W_IOSB4            = 00000046
COMMA                   = 0000002C
CREATE_ADDR               00000426 R      03
CREATE_ENTRY              0000045F R      03
DEALLOC_ENTRY             000005F5 R      03
DIST_DONE                 000005A6 R      03
DIST_ERROR                00000599 R      03
DIST_LOOP                 00000559 R      03
DIST_PARSE                0000008A R      04
DIST_TYPE                 00000000 R      02
DIST_TYPE_SIZ           = 00000004
ENTER_ADDR                000001DD R      03
FAB                     = 00000004
FAB$B_BID               = 00000000
FAB$B_BLN               = 00000001
FAB$B_DNS               = 00000035
FAB$B_FAC               = 00000016
FAB$B_FNS               = 00000034
FAB$C_BID               = 00000003
FAB$K_BLN               = 00000050
FAB$L_CTX               = 00000018
```

```
FAB$L_DNA                       =  00000030
FAB$L_FNA                       =  0000002C
FAB$L_FOP                       =  00000004
FAB$L_NAM                       =  00000028
FAB$M_GET                       =  00000002
FAB$M_SQO                       =  00000040
FIND_ADDRESSEE                     00000060  R      04
GETSYM1                            0000012E  R      04
GET_DIST_LIST                      000004A0  R      03
GET_SYMBOL                         0000011E  R      04
LEGAL_SEP                          00000144  R      04
LEGAL_SEP1                         0000014E  R      04
LIB$FREE_VM                        ********    X    03
LIB$GET_VM                         ********    X    03
LIB$SIGNAL                         ********    X    03
LIB$TPARSE                         ********    X    03
LNKDEF                          =  00000000
LNK_B_FLAGS                     =  0000002E
LNK_B_NODLEN                    =  0000002F
LNK_B_PNLEN                     =  0000004F
LNK_C_IN_CKUSER                 =  0000000A
LNK_C_IN_CONNECT                =  00000008
LNK_C_IN_FILE                   =  0000000D
LNK_C_IN_SENDER                 =  00000009
LNK_C_IN_SUBJ                   =  0000000C
LNK_C_IN_TO                     =  0000000B
LNK_C_IO_READ                   =  0000000E
LNK_C_IO_WRITE                  =  0000000F
LNK_C_LENGTH                    =  00000050
LNK_C_OUT_CKSEND                =  00000006
LNK_C_OUT_CKUSER                =  00000002
LNK_C_OUT_CONNECT               =  00000000
LNK_C_OUT_DEACCESS              =  00000007
LNK_C_OUT_FILE                  =  00000005
LNK_C_OUT_SENDER                =  00000001
LNK_C_OUT_SUBJ                  =  00000004
LNK_C_OUT_TO                    =  00000003
LNK_L_ADRCNT                    =  00000008
LNK_L_BLINK                     =  00000004
LNK_L_CONTEXT                   =  0000000C
LNK_L_FLINK                     =  00000000
LNK_L_SPARES                    =  00000020
LNK_L_STS                       =  0000001C
LNK_L_TFRADR                    =  00000010
LNK_M_ALTP                      =  00000004
LNK_M_BLKMODE                   =  00000010
LNK_M_DEAD                      =  00000002
LNK_M_FSENT                     =  00000008
LNK_M_MSGSNT                    =  00000001
LNK_Q_IOSB                      =  00000014
LNK_S_IOSB                      =  00000008
LNK_S_LNKDEF                    =  00000050
LNK_S_NODE                      =  0000001F
LNK_S_SPARES                    =  00000008
LNK_T_NODE                      =  00000030
LNK_T_PNAM                      =  00000050
LNK_V_ALTP                      =  00000002
LNK_V_BLKMODE                   =  00000004
LNK_V_DEAD                      =  00000001
LNK_V_FSENT                     =  00000003
LNK_V_MSGSNT                    =  00000000
LNK_W_CHAN                      =  0000002C
LNK_W_MBXCHAN                   =  0000002A
LNK_W_SPAREW                    =  00000028
LNM$C_NAMLENGTH                 =  000000FF
LNM$M_CASE_BLIND                =  02000000
LNM$M_TERMINAL                  =  00000200
LNM$_ATTRIBUTES                 =  00000003
LNM$_STRING                     =  00000002
MAIFDEF                         =  00000000
MAIF_L_FLAGS                    =  00000000
MAIF_M_ALTP                     =  00000400
MAIF_M_ATTACHMENT               =  00080000
MAIF_M_BATCH                    =  00000200
MAIF_M_CALEDT                   =  00008000
MAIF_M_CAPTIVE                  =  00004000
MAIF_M_CTRLCFL                  =  00000100
MAIF_M_DFMSG                    =  00001000
MAIF_M_EDTEDT                   =  00010000
MAIF_M_EDTXYZ                   =  00020000
MAIF_M_FILOPN                   =  00000800
MAIF_M_ISAM                     =  00000001
MAIF_M_ITERM                    =  00000004
MAIF_M_NETJOB                   =  00000002
MAIF_M_OTERM                    =  00000008
MAIF_M_RDHOLD                   =  00000020
MAIF_M_READ_FL                  =  00000040
MAIF_M_RECL                     =  00000080
MAIF_M_SCOPE                    =  00000010
MAIF_M_SERVERLOOP               =  00040000
MAIF_M_USEREDIT                 =  00002000
MAIF_S_MAIFDEF                  =  00000004
MAIF_V_ALTP                     =  0000000A
MAIF_V_ATTACHMENT               =  00000013
MAIF_V_BATCH                    =  00000009
MAIF_V_CALEDT                   =  0000000F
MAIF_V_CAPTIVE                  =  0000000E
MAIF_V_CTRLCFL                  =  00000008
MAIF_V_DFMSG                    =  0000000C
MAIF_V_EDTEDT                   =  00000010
MAIF_V_EDTXYZ                   =  00000011
MAIF_V_FILOPN                   =  0000000B
MAIF_V_ISAM                     =  00000000
MAIF_V_ITERM                    =  00000002
MAIF_V_NETJOB                   =  00000001
MAIF_V_OTERM                    =  00000003
MAIF_V_RDHOLD                   =  00000005
MAIF_V_READ_FL                  =  00000006
MAIF_V_RECL                     =  00000007
MAIF_V_SCOPE                    =  00000004
MAIF_V_SERVERLOOP               =  00000012
MAIF_V_USEREDIT                 =  0000000D
MAIL$ADDR_EXISTS                   ********    X    03
MAIL$ADD_ADDR                      0000006E  RG    03
```

```
MAIL$ANY_ADDRESSEES             000005B3 RG    03
MAIL$CHECK_FORW                 ******** X     03
MAIL$CHK_CLUNODES               ******** X     03
MAIL$CREATELINK                 ******** X     03
MAIL$DELETELINK                 ******** X     03
MAIL$GET_USER                   ******** X     03
MAIL$GL_FLAGS                   ******** X     03
MAIL$INCLINKADR                 ******** X     03
MAIL$INI_ADDR                   000005DD RG    03
MAIL$K_INBUFFSZ               = 000000FF
MAIL$NET_ADDR                   ******** X     03
MAIL$NET_FROM                   ******** X     03
MAIL$PRUNW_HANDLER              ******** X     03
MAIL$Q_SYSNODE                  ******** X     03
MAIL$SD_LNM_FILE_DEV            ******** X     03
MAIL$T_DAF_BUFF                 ******** X     03
MAIL$UPCASE_Q                   ******** X     03
MAIL$_CONABORT                = 007E8112
MAIL$_FACILITY                = 0000007E
MAIL$_FORWLOOP                = 007E808A
MAIL$_OPENIN                  = 007E109A
MAIL$_READERR                 = 007E10B2
MAIL$_USERDSABL               = 007E8082
MAIL_ADD_ADDR                   00000000 R     03
NAM$B_BID                     = 00000000
NAM$B_BLN                     = 00000001
NAM$B_ESS                     = 0000000A
NAM$B_RSS                     = 00000002
NAM$C_BID                     = 00000002
NAM$C_MAXRSS                  = 000000FF
NAM$K_BLN                     = 00000060
NAM$L_ESA                     = 0000000C
NAM$L_RSA                     = 00000004
NEXT_ADDR                       00000000 R     04
NODET_SPEC                      000000B6 R     04
NODE_SPEC                       000000C0 R     04
NODE_SPECC                      000000C4 R     04
NOT_QEOS                        000000A8 R     04
PROTOCOL_SPEC                   00000108 R     04
QDIST_PARSE                     00000092 R     04
QFILE_SPEC                      0000009E R     04
QPROT_SPEC                      00000114 R     04
QUOTED_SPEC                     000000FE R     04
QUOTE_SPEC                      00000170 R     04
QUSER_SPEC                      000000F6 R     04
RAB                           = 00000008
RAB$B_BID                     = 00000000
RAB$B_BLN                     = 00000001
RAB$C_BID                     = 00000001
RAB$C_SEQ                     = 00000000
RAB$K_BLN                     = 00000044
RAB$L_CTX                     = 00000018
RAB$L_FAB                     = 0000003C
RAB$L_RBF                     = 00000028
RAB$L_UBF                     = 00000024
RAB$W_RSZ                     = 00000022
RAB$W_USZ                     = 00000020
REPORT_SYNTAX_ERROR             0000016F R   D  03
RESET_LCOUNT                    000001AF R      03
RMS$_EOF                        ********   X    03
SCAN_ADDR                       00000074 R      04
SET_SPST                        0000015E R   D  03
SHR$K_SHRDEF                  = 00000001
SHR$_OPENIN                   = 00001098
SHR$_READERR                  = 000010B0
SHR$_SYNTAX                   = 000010F8
SPEC_CONTINUE                   00000054 R      04
SPEC_QUOTE                      00C0017A R      04
STORE_LOGNAME                   000001A7 R      03
STORE_PROTOCOL                  00000153 R      03
STORE_UNAME                     0000019F R      03
STS$K_ERROR                   = 00000002
STS$K_WARNING                 = 00000000
STS$V_FAC_NO                  = 00000010
SYS$CLOSE                       ********  GX    03
SYS$CONNECT                     ********  GX    03
SYS$GET                         ********  GX    03
SYS$OPEN                        ********  GX    03
SYS$TRNLNM                      ********  GX    03
TPA$K_COUNTO                  = 00000008
TPA$K_LENGTHO                 = 00000024
TPA$L_PARAM                   = 00000020
TPA$L_STRINGCNT               = 00000008
TPA$L_STRINGPTR               = 0000000C
TPA$L_TOKENCNT                = 00000010
TPA$L_TOKENPTR                = 00000014
TPA$_ALPHA                    = 000001EE
TPA$_ANY                      = 000001ED
TPA$_BLANK                    = 000001F2
TPA$_DECIMAL                  = 000001F3
TPA$_DIGIT                    = 000001EF
TPA$_EOS                      = 000001F7
TPA$_EXIT                     = FFFFFFFF
TPA$_FAIL                     = FFFFFFFE
TPA$_FILESPEC                 = 000001EA
TPA$_HEX                      = 000001F5
TPA$_IDENT                    = 000001EC
TPA$_KEYWORD                  = 00000100
TPA$_LAMBDA                   = 000001F6
TPA$_MAXKEY                   = 000000DC
TPA$_OCTAL                    = 000001F4
TPA$_STRING                   = 000001F0
TPA$_SUBXPR                   = 000001F8
TPA$_SYMBOL                   = 000001F1
TPA$_UIC                      = 000001EB
TPA_C_EXTRA                   = 00000038
TPA_L_CNCT                    = 00000044
TPA_L_LCOUNT                  = 00000024
TPA_L_LLNK                    = 00000030
TPA_L_PTPA                    = 00000048
TPA_L_SPST                    = 0000004C
TPA_M_TERMINAL                = 00000001
TPA_Q_LOGNAM                  = 00000034
TPA_Q_ONAME                   = 00000028
```

```
TPA_Q_PNAM                      = 0000003C
TPA_Q_UNAME                     = 00000050
TPA_V_TERMINAL                  = 00000000
TPA_W_FLAGS                     = 0000005A
TPA_W_UENT                      = 00000058
UAF$L_FLAGS                       ********    X   03
UAF$V_NOMAIL                      ********    X   03
USER_SPEC                         000000DA R      04
UTIL$REPORT_IO_ERROR              ********    X   03
VMDDEF                          = 00000000
VMD_B_DIRLNG                    = 00000041
VMD_B_FNMLNG                    = 00000042
VMD_B_FWDLNG                    = 00000043
VMD_B_SPARE1                    = 0000003F
VMD_C_LENGTH                    = 00000044
VMD_C_USERNAME                  = 0000001F
VMD_K_LENGTH                    = 00000044
VMD_K_USERNAME                  = 0000001F
VMD_L_SPARE                     = 00000023
VMD_M_NOAUTOPURGE               = 00000004
VMD_M_SELFREPLY                 = 00000002
VMD_M_SELFSEND                  = 00000001
VMD_S_SPARE                     = 0000001C
VMD_S_SPARE1                    = 00000002
VMD_S_USERNAME                  = 0000001F
VMD_S_VMDDEF                    = 00000044
VMD_T_FWDNAM                    = 00000044
VMD_T_USERNAME                  = 00000000
VMD_V_NOAUTOPURGE               = 00000002
VMD_V_SELFREPLY                 = 00000001
VMD_V_SELFSEND                  = 00000000
VMD_W_FLAGS                     = 0000001F
VMD_W_MAIL                      = 00000021
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
$PLIT$	00000005 (5.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD NOWRT NOVEC LONG
$CODE$	0000060F (1551.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC LONG
_LIB$STATE$	00000182 (386.)	04 (4.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC BYTE
_LIB$KEY0$	00000000 (0.)	05 (5.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC WORD
_LIB$KEY1$	00000000 (0.)	06 (6.)	PIC USR CON REL LCL SHR EXE RD NOWRT NOVEC WORD

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	37	00:00:00.06	00:00:02.43
Command processing	123	00:00:00.48	00:00:04.11
Pass 1	583	00:00:21.05	00:01:29.41

Symbol table sort	0	00:00:01.04	00:00:03.58
Pass 2	210	00:00:04.93	00:00:22.80
Symbol table output	45	00:00:00.22	00:00:00.76
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	1002	00:00:27.80	00:02:03.11

The working set limit was 2100 pages.
168433 bytes (329 pages) of virtual memory were used to buffer the intermediate code.
There were 60 pages of symbol table space allocated to hold 1096 non-local and 50 local symbols.
1118 source lines were read in Pass 1, producing 41 object records in Pass 2.
55 pages of virtual memory were used to define 42 macros.

+----------------------------+
! Macro library statistics !
+----------------------------+

Macro library name	Macros defined
_255DUA28:[MAIL.OBJ]MAILMAC.MLB;1	6
_255DUA28:[SYS.OBJ]LIB.MLB;1	0
_255DUA28:[SYSLIB]STARLET.MLB;2	26
TOTALS (all libraries)	32

1483 GETS were required to define 32 macros.

There were no errors, warnings or information messages.

```
MACRO/LIS=LIS$:ADDRLIST/OBJ=OBJ$:ADDRLIST MSRC$:ADDRLIST/UPDATE=(ENH$:ADDRLIST)+EXECML$/LIB+LIB$:MAILMAC/LIB
```

CMDS
LIS

DIRECTORY
LIS

SPAWNCMD
CLD

MAILREQ
REQ

EDIT
LIS

GETUSER
LIS

MAILDEF
SDL

ADDRLIST
LIS

GETNEW
LIS