```
NNN         NNN  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT     AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPPP
NNN         NNN  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT     AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPPP
NNN         NNN  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT     AAAAAAAAA        CCCCCCCCCCCC  PPPPPPPPPPPP
NNN         NNN  EEE                    TTT        AAA         AAA  CCC              PPP         PPP
NNN         NNN  EEE                    TTT        AAA         AAA  CCC              PPP         PPP
NNN         NNN  EEE                    TTT        AAA         AAA  CCC              PPP         PPP
NNNNNN      NNN  EEE                    TTT        AAA         AAA  CCC              PPP         PPP
NNNNNN      NNN  EEE                    TTT        AAA         AAA  CCC              PPP         PPP
NNNNNN      NNN  EEE                    TTT        AAA         AAA  CCC              PPP         PPP
NNN   NNN   NNN  EEEEEEEEEEEE           TTT        AAA         AAA  CCC              PPPPPPPPPPPP
NNN   NNN   NNN  EEEEEEEEEEEE           TTT        AAA         AAA  CCC              PPPPPPPPPPPP
NNN   NNN   NNN  EEEEEEEEEEEE           TTT        AAA         AAA  CCC              PPPPPPPPPPPP
NNN      NNNNNN  EEE                    TTT        AAAAAAAAAAAAAAA  CCC              PPP
NNN      NNNNNN  EEE                    TTT        AAAAAAAAAAAAAAA  CCC              PPP
NNN      NNNNNN  EEE                    TTT        AAAAAAAAAAAAAAA  CCC              PPP
NNN         NNN  EEE                    TTT        AAA         AAA  CCC              PPP
NNN         NNN  EEE                    TTT        AAA         AAA  CCC              PPP
NNN         NNN  EEE                    TTT        AAA         AAA  CCC              PPP
NNN         NNN  EEEEEEEEEEEEEEE        TTT        AAA         AAA     CCCCCCCCCCCC  PPP
NNN         NNN  EEEEEEEEEEEEEEE        TTT        AAA         AAA     CCCCCCCCCCCC  PPP
NNN         NNN  EEEEEEEEEEEEEEE        TTT        AAA         AAA     CCCCCCCCCCCC  PPP
```

Ps
NE
NE
NE
NE
$R

```
**FILE**ID**SERVER                                   F 5                                                    NE
                                                                                                             VC
```

```
  SSSSSSSS  EEEEEEEEEE  RRRRRRRR    VV      VV  EEEEEEEEEE  RRRRRRRR
  SSSSSSSS  EEEEEEEEEE  RRRRRRRR    VV      VV  EEEEEEEEEE  RRRRRRRR
SS          EE          RR      RR  VV      VV  EE          RR      RR
SS          EE          RR      RR  VV      VV  EE          RR      RR
SS          EE          RR      RR  VV      VV  EE          RR      RR
SS          EE          RR      RR  VV      VV  EE          RR      RR
  SSSSSS    EEEEEEEE    RRRRRRRR    VV      VV  EEEEEEEE    RRRRRRRR
  SSSSSS    EEEEEEEE    RRRRRRRR    VV      VV  EEEEEEEE    RRRRRRRR
        SS  EE          RR  RR      VV      VV  EE          RR  RR
        SS  EE          RR  RR      VV      VV  EE          RR  RR
        SS  EE          RR    RR      VV  VV    EE          RR    RR        ....
        SS  EE          RR    RR      VV  VV    EE          RR    RR        ....
SSSSSSSS    EEEEEEEEEE  RR      RR      VV      EEEEEEEEEE  RR      RR      ....
SSSSSSSS    EEEEEEEEEE  RR      RR      VV      EEEEEEEEEE  RR      RR      ....

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
```

```
0 MODULE network_server (IDENT = 'V04-000',
0                         MAIN = network_server,
0                         ADDRESSING_MODE(EXTERNAL=GENERAL)) =
1 BEGIN
1
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1
1 !++
1 ! FACILITY:  DECnet
1 !
1 ! ABSTRACT:
1 !
1 !       This program is used to enable a process to wait for an incoming
1 !       DECnet logical link connection, and then accept the logical link
1 !       request by invoking the correct procedure using CLI CHAIN.  This
1 !       is used to allow a single process to handle many logical link
1 !       requests, and reduce the overhead involved in process creation.
1 !
1 ! ENVIRONMENT:
1 !
1 !       VAX/VMS operating system. unprivileged user mode,
1 !
1 ! AUTHOR:  Tim Halvorsen, June 1982
1 !
1 ! Modified by:
1 !
1 !       V03-004 PRB0337         Paul Beck       27-Jun-1984  16:33
1 !               Change default timeout from 1 minute to 5 minutes.
1 !
1 !       V003    TMH0003         Tim Halvorsen   07-Apr-1983
1 !               Add support for direct execution of an object image,
1 !               if the object filespec contains an explicit ".EXE".
1 !
1 !       V002    TMH0002         Tim Halvorsen   24-Feb-1983
1 !               Add support for EPIDs by using the IPID returned
```

```
          0058  1 !              by DECLSERV to index the SPI database, rather than
          0059  1 !              using the EPID returned by GETJPI.
          0060  1 !
          0061  1 !       V001   TMH0001          Tim Halvorsen    7-Feb-1983
          0062  1 !              Add code to display where each connect request comes
          0063  1 !              from (by displaying the NCB), so that .LOG files can
          0064  1 !              be more easily read.
          0065  1 !--
          0066  1
          0067  1 !
          0068  1 ! Include files
          0069  1 !
          0070  1
          0071  1 LIBRARY 'SYS$LIBRARY:STARLET';         ! VAX/VMS common definitions
          0072  1
          0073  1 LIBRARY 'SHRLIB$:NET';                 ! NETACP control QIO definitions
```

```
!
! Table of contents
!

FORWARD ROUTINE
    network_server,                          ! Main routine
    timeout_ast:          NOVALUE,           ! Timeout AST
    issue_mailbox_read:   NOVALUE,           ! Issue network mailbox read
    net_interrupt:        NOVALUE,           ! Mailbox attention AST
    fao_buffer;                              ! Invoke FAO and return descriptor

!
! Literals
!

LITERAL
    true = 1,
    false = 0;

!
! Macros
!

MACRO
    fao(string) =
        fao_buffer(%ASCID string
        %IF %LENGTH GTR 1 %THEN ,%REMAINING %FI)%,

    write_line(string) =
        LIB$PUT_OUTPUT(fao(string
        %IF %LENGTH GTR 1 %THEN ,%REMAINING %FI))%,

    signal_if_error(command) =
        BEGIN
        LOCAL
            status;

        status = command;
        IF NOT .status
        THEN
            BEGIN
            SIGNAL(.status);
            RETURN .status OR sts$m_inhib_msg;
            END;
        END%;

!
! Own storage
!

LITERAL
    mbx_maxmsg = 128;                        ! Maximum size of mailbox message

OWN
    net_channel: WORD,                       ! Channel to ACP
    mbx_channel: WORD,                       ! Channel to assoc. mailbox
    mbx_message: VECTOR [mbx_maxmsg,BYTE],   ! Mailbox input buffer
```

```
1      mbx_iosb:   $BBLOCK [8];                 ! I/O status block for mailbox
1
1  !
1  ! External routines
1  !
1
1  EXTERNAL ROUTINE
1      lib$asn_wth_mbx,                          ! Assign with assoc. mailbox
1      lib$set_logical,                          ! Define supervisor mode logical name
1      lib$run_program,                          ! Chain to another program
1      lib$do_command,                           ! Chain a CLI command string
1      lib$put_output,                           ! Write to SYS$OUTPUT
1      str$concat;                               ! Concatenate strings together
```

```
1 ROUTINE network_server =
1
1 !---
1 !
1 !       This routine is the entry point to the program
1 !
1 ! Inputs:
1 !
1 !       None
1 !
1 ! Outputs:
1 !
1 !       Routine value = status code
1 !---
1
2 BEGIN
2
2 LOCAL
2     nfb:        $BBLOCK [nfb$c_length+20*4],    ! Network function block
2                                                 ! (room for 20 field requests)
2     nfb_desc:   VECTOR [2]                  ! Descriptor of NFB
2                 INITIAL(nfb$c_length + 3*4),
2     iosb:       $BBLOCK [8],                ! I/O status block
2     time_buf:   VECTOR [128,BYTE],          ! Buffer for timeout specifier
2     time_desc:  VECTOR [2]                  ! Descriptor of timeout specifier
2                 INITIAL(128),
2     delta_time: VECTOR [2],                 ! Binary time quadword
2     buffer:     VECTOR [64],                ! Return buffer
2     buffer_desc: VECTOR [2]                 ! Descriptor of above buffer
2                 INITIAL(256),
2     keys:       $BBLOCK [4+4+nfb$c_ctx_size], ! Buffer for search key & context
2     key_desc:   VECTOR [2]                  ! Descriptor of above buffer
2                 INITIAL(4+4+nfb$c_ctx_size),
2     ptr:        REF $BBLOCK,                ! Pointer into return buffer
2     cmd_desc:   $BBLOCK [8]                 ! Command string
2                 PRESET ([dsc$b_class] = dsc$k_class_d,
2                         [dsc$w_length] = 0,
2                         [dsc$a_pointer] = 0),
2     ncb_desc:   VECTOR [2],                 ! Descriptor of NCB
2     ascii_ncb_desc: VECTOR [2],             ! Descriptor of ASCII portion of NCB
2     filespec:   VECTOR [2],                 ! Descriptor of procedure filespec
2     prcnam:     VECTOR [2],                 ! Descriptor of process name
2     ipid,                                   ! Our IPID
2     epid,                                   ! Our EPID
2     item_list:  $BBLOCK [10*4]
2                 PRESET ([0,0,16,0] = 4,
2                         [2,0,16,0] = jpi$_pid,
2                         [8,0,32,0] = 0,
2                         [12,0,32,0] = 0),
2     status;
2
2 BIND
2     default_time = %ASCID '0 00:05:00': $BBLOCK;
2
2 !
2 ! Initialize some stack local variables with dynamic pointers
2 !
```

```
;  203          0201  2
;  204          0202  2 nfb_desc [1] = nfb;
;  205          0203  2 time_desc [1] = time_buf;
;  206          0204  2 buffer_desc [1] = buffer;
;  207          0205  2 key_desc [1] = keys;
;  208          0206  2 item_list [4,0,32,0] = epid;
;  209          0207  2
;  210          0208  2 !
;  211          0209  2 ! Get our own EPID for later lookup of our server parameters
;  212          0210  2 !
;  213          0211  2
;  214        P 0212  2 signal_if_error(
;  215          0213  2     $GETJPI(ITMLST = item_list));       ! Get our EPID
;  216          0214  2
;  217          0215  2 !
;  218          0216  2 ! Assign a channel to the network ACP
;  219          0217  2 !
;  220          0218  2
;  221        P 0219  2 signal_if_error(
;  222        P 0220  2     LIB$ASN_WTH_MBX(%ASCID '_NET:',     ! Assign channel to NETACP
;  223        P 0221  2                     0,0,                ! mailbox MAXMSG,BUFQUO
;  224        P 0222  2                     net_channel,        ! Channel to NETACP
;  225          0223  2                     mbx_channel));      ! Channel to mailbox
;  226          0224  2
;  227          0225  2 !
;  228          0226  2 ! Issue a read on the associated mailbox, so that we can receive
;  229          0227  2 ! notification of network broadcast messages.  This is done so that
;  230          0228  2 ! we can detect the network shutting down.
;  231          0229  2 !
;  232          0230  2
;  233          0231  2 issue_mailbox_read();                   ! Issue mailbox read
;  234          0232  2
;  235          0233  2 !
;  236          0234  2 ! Set our process name to something which indicates that we are a network
;  237          0235  2 ! server waiting for work.  This has the effect of wiping out the previous
;  238          0236  2 ! process name set by the previous connect to this process.
;  239          0237  2 !
;  240          0238  2
;  241          0239  2 prcnam [0] = .buffer_desc [0];          ! Make descriptor of scratch buffer
;  242          0240  2 prcnam [1] = .buffer_desc [1];
;  243          0241  2
;  244        P 0242  2 $FAO(%ASCID 'SERVER_!XW',              ! Generate a unique process name
;  245        P 0243  2         prcnam,                         ! Output buffer descriptor
;  246        P 0244  2         prcnam [0],                     ! Place to return length
;  247          0245  2         .epid);                         ! Use last 4 digits of EPID
;  248          0246  2
;  249          0247  2 $SETPRN(PRCNAM = prcnam);               ! Set our process name
;  250          0248  2                                         ! (ignore any errors)
;  251          0249  2
;  252          0250  2 !
;  253          0251  2 ! Schedule a timer, so that if the following QIO does not complete within
;  254          0252  2 ! a reasonable amount of time, we can go away (since there was no work to do).
;  255          0253  2 !
;  256          0254  2
;  257        P 0255  2 status = $TRNLOG(LOGNAM = %ASCID 'NETSERVER$TIMEOUT', ! Get timeout value
;  258        P 0256  2                 RSLBUF = time_desc,
;  259          0257  2                 RSLLEN = time_desc [0]);
;       2
;       2 IF .status NEQ ss$_normal              ! If not explicitly specified,
;       2 THEN
;       3     BEGIN
;       3     time_desc [0] = .default_time [dsc$w_length];
;       3     time_desc [1] = .default_time [dsc$a_pointer];
;       2     END;
;       2
;     P 2 signal_if_error(
;     P 2     $BINTIM(TIMBUF = time_desc,        ! Translate time specifier to binary
;       2             TIMADR = delta_time));
;       2
;     P 2 signal_if_error(
;     P 2     $SETIMR(DAYTIM = delta_time,       ! Start timer
;       2             ASTADR = timeout_ast));     ! Address of AST routine
;       2
;       2 !
;       2 ! Tell NETACP that we are available for a connect request.  The QIOW
;       2 ! will complete when a connect has been assigned to us.
;       2 !
;       2
;       2 CH$FILL(0,nfb$c_length,nfb);            ! Pre-zero NFB fields
;       2 nfb [nfb$b_fct] = nfb$c_declserv;       ! Tell NETACP we are available for work
;       2
;     P 2 status = $QIOW(FUNC = IO$_ACPCONTROL,   ! Issue control function
;     P 2                CHAN = .net_channel,
;     P 2                IOSB = iosb,
;       2                P1 = nfb_desc);          ! Address of NFB descriptor
;       2
;       2 IF NOT .status                          ! If error detected,
;       3     OR NOT (status = .iosb [0,0,16,0])
;       2 THEN
;       2     IF .status EQL ss$_abort            ! If we timed out,
;       2     THEN
;       3         BEGIN
;       3         $DASSGN(CHAN = .net_channel);   ! Deassign the ACP channel
;       3         RETURN sts$k_severe OR sts$m_inhib_msg; ! Return "fatal" from program
;       3         END
;       2     ELSE
;       3         BEGIN
;       3         SIGNAL(.status);                ! else signal the error
;       3         $DASSGN(CHAN = .net_channel);   ! Deassign the ACP channel
;       3         RETURN true;
;       3         END;
;       2
;       2 ipid = .iosb [4,0,32,0];                ! Get our IPID returned by DECLSERV
;       2
;       2 CH$FILL(0,nfb$c_length,nfb);            ! Pre-zero NFB fields
;       2
;       2 nfb [nfb$b_fct] = nfb$c_fc_show;        ! Request "show" function
;       2 nfb [nfb$b_database] = nfb$c_db_spi;    ! of server process database
;       2 nfb [nfb$l_srch_key] = nfb$c_spi_pid;   ! for our process
;       2 nfb [nfb$b_oper] = nfb$c_op_eql;        ! by checking if field EQL P2 value
;       2
;       2 CH$MOVE(4*4, UPLIT LONG(                ! Request the following fields:
;       2                nfb$c_spi_ncb,           ! Network connect block
;       2                nfb$c_spi_sfi,           ! Procedure filespec
```

```
  2                          nfb$c_spi_pnm,          ! Process name
  2                          nfb$c_endoflist),
  2                          nfb [nfb$l_fldid]);
  2
  2  keys [0,0,32,0] = 0;                             ! Zero count of fields in P4 (unused)
  2  keys [4,0,32,0] = .ipid;                         ! Search value = our IPID
  2  keys [8,0,16,0] = 0;                             ! Context area = at beginning
  2
P 2  status = $QIOW(FUNC = IO$_ACPCONTROL,            ! Issue control function
P 2                  CHAN = .net_channel,
P 2                  IOSB = iosb,
P 2                  P1 = nfb_desc,                   ! Address of NDB descriptor
P 2                  P2 = key_desc,                   ! Address of key buffer descriptor
  2                  P4 = buffer_desc);               ! Address of return buffer descriptor
  2
  2  IF NOT .status                        ! If error detected,
  3      OR NOT (status = .iosb [0,0,16,0])
  2  THEN
  3      BEGIN
  3      SIGNAL(.status);                             ! then stop looping
  3      $DASSGN(CHAN = .net_channel);                ! Deassign the ACP channel
  3      RETURN true;
  2      END;
  2
  2  ptr = buffer [0];                                ! Point to first string in buffer
  2
  2  ncb_desc [0] = .ptr [0,0,16,0];                  ! Construct descriptor of NCB
  2  ncb_desc [1] = .ptr + 2;
  2  ptr = .ptr + 2 + .ptr [0,0,16,0];                ! Skip by string in buffer
  2
  2  filespec [0] = .ptr [0,0,16,0];                  ! Construct descriptor of procedure
  2  filespec [1] = .ptr + 2;
  2  ptr = .ptr + 2 + .ptr [0,0,16,0];                ! Skip by string in buffer
  2
  2  prcnam [0] = .ptr [0,0,16,0];                    ! Construct descriptor of process name
  2  prcnam [1] = .ptr + 2;
  2  ptr = .ptr + 2 + .ptr [0,0,16,0];                ! Skip by string in buffer
  2
  2  ptr = CH$FIND_CH(.ncb_desc [0], .ncb_desc [1], '/');
  2
  2  ascii_ncb_desc [0] = .ptr - .ncb_desc [1];
  2  ascii_ncb_desc [1] = .ncb_desc [1];
  2
  2  write_line('');
  2  write_line('        -----------------------------------------------------------');
  2  write_line('');
  2  write_line('        Connect request received at !%D', 0);
  2  write_line('            from remote process !AS"', ascii_ncb_desc);
  2  write_line('            for object "!AS"', filespec);
  2  write_line('');
  2  write_line('        -----------------------------------------------------------');
  2  write_line('');
  2
P 2  signal_if_error(
  2      $SETPRN(PRCNAM = prcnam));                   ! Set our process name
  2
P 2  signal_if_error(
;  374        P 0372  2     LIB$SET_LOGICAL(%ASCID 'SYS$NET',    ! Define SYS$NET to NCB
;  375          0373  2                 ncb_desc));
;  376          0374  2
;  377          0375  2 cmd_desc [dsc$b_class] = dsc$k_class_d; ! Create dynamic string descriptor
;  378          0376  2 cmd_desc [dsc$a_pointer] = 0;           ! Indicate no dynamic string yet
;  379        P 0377  2 signal_if_error(
;  380        P 0378  2     STR$CONCAT(cmd_desc,                ! Create "@filespec" command
;  381          0379  2                 %ASCID '@',filespec));
;  382          0380  2
;  383          0381  2 IF NOT CH$FAIL(CH$FIND_SUB(           ! If .EXE found in filespec,
;  384          0382  2                 .filespec [0], .filespec [1],
;  385          0383  2                 4, UPLIT BYTE('.EXE')))
;  386          0384  2 THEN
;  387        P 0385  2     signal_if_error(
;  388          0386  3         LIB$RUN_PROGRAM(filespec))      ! Chain to program (EXIT AND CHAIN)
;  389          0387  2 ELSE
;  390        P 0388  2     signal_if_error(
;  391          0389  2         LIB$DO_COMMAND(cmd_desc));      ! Else, chain to command line
;  392          0390  2
;  393          0391  2 !
;  394          0392  2 ! Do not put any code after this point.  Both LIB$RUN_PROGRAM and
;  395          0393  2 ! LIB$DO_COMMAND do not return, then cause immediately program exit.
;  396          0394  2 ! The only way we get here is if they fail.
;  397          0395  2 !
;  398          0396  2
;  399          0397  2 RETURN true;                            ! Return successfully
;  400          0398  2
;  401          0399  1 END;
; INFO#250           L1:0245
; Referenced LOCAL symbol EPID is probably not initialized


                                                                              .TITLE  NETWORK_SERVER
                                                                              .IDENT  \V04-000\

                                                                              .PSECT  $PLIT$,NOWRT,NOEXE,2

                                                   0319  0004  00000 P.AAA:   .WORD   4, 793                          ;
                                                            00#  00004         .BYTE   0[4]                            ;
                                             00000000  00000000  00008         .LONG   0, 0                            ;
           00  00  30  30  3A  35  30  3A  30  30  20  30  00010 P.AAC:   .ASCII  \0 00:05:00\<0><0>              ;
                                                       010E000A  0001C P.AAB:   .LONG   17694730                        ;
                                                      00000000'  00020         .ADDRESS P.AAC                          ;
                           00  00  00  3A  54  45  4E  5F  00024 P.AAE:   .ASCII  \_NET:\<0><0><0>                ;
                                                       010E0005  0002C P.AAD:   .LONG   17694725                        ;
                                                      00000000'  00030         .ADDRESS P.AAE                          ;
           00  00  57  58  21  5F  52  45  56  52  45  53  00034 P.AAG:   .ASCII  \SERVER_!XW\<0><0>              ;
                                                       010E000A  00040 P.AAF:   .LONG   17694730                        ;
                                                      00000000'  00044         .ADDRESS P.AAG                          ;
4F  45  4D  49  54  24  52  45  56  52  45  53  54  45  4E  00048 P.AAI:   .ASCII  \NETSERVER$TIMEOUT\<0><0><0>    ;
                                           00  00  00  54  55  00057                                                 ;
                                                       010E0011  0005C P.AAH:   .LONG   17694737                        ;
                                                      00000000'  00060         .ADDRESS P.AAI                          ;
                     00000000  12020045  12020043  12020044  00064 P.AAJ:   .LONG   302121028, 302121027, 302121029, 0  ;
                                                                 00074 P.AAL:   .BLKB   0                               ;
                                                       010E0000  00074 P.AAK:   .LONG   17694720                        ;
                                                      00000000'  00078         .ADDRESS P.AAL                          ;
```

```
2D   2D   2D   2D   2D   2D   2D   20   20   20   20   20   20   20   20  0007C P.AAN:   .ASCII   \        --------------------------------\
2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D  0008B
                         2D   2D   2D   2D   2D   2D   2D   2D   2D   2D  0009A
2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D  000A4          .ASCII   \------------------------\
                              2D   2D   2D   2D   2D   2D   2D   2D   2D  000B3
                                                                010E0040  000BC P.AAM:   .LONG    17694784
                                                                00000000' 000C0          .ADDRESS P.AAN
                                                                          000C4 P.AAP:   .BLKB    0
                                                                010E0000  000C4 P.AAO:   .LONG    17694720
                                                                00000000' 000C8          .ADDRESS P.AAP
74   63   65   6E   6E   6F   43   20   20   20   20   20   20   20   20  000CC P.AAR:   .ASCII   \        Connect request received at !%D-
76   69   65   63   65   72   20   74   73   65   75   71   65   72   20  000DB                   \<0>
                         00   44   25   21   20   74   61   20   64   65  000EA
                                                                010E0027  000F4 P.AAQ:   .LONG    17694759
                                                                00000000' 000F8          .ADDRESS P.AAR
6F   72   66   20   20   20   20   20   20   20   20   20   20   20   20  000FC P.AAT:   .ASCII   \            from remote process !AS"\
73   65   63   6F   72   70   20   65   74   6F   6D   65   72   20   6D  0010B
                                             22   53   41   21   20   73  0011A
                                                                010E0024  00120 P.AAS:   .LONG    17694756
                                                                00000000' 00124          .ADDRESS P.AAT
72   6F   66   20   20   20   20   20   20   20   20   20   20   20   20  00128 P.AAV:   .ASCII   \            for object "!AS"\
          22   53   41   21   22   20   74   63   65   6A   62   6F   20  00137
                                                                010E001C  00144 P.AAU:   .LONG    17694748
                                                                00000000' 00148          .ADDRESS P.AAV
                                                                          0014C P.AAX:   .BLKB    0
                                                                010E0000  0014C P.AAW:   .LONG    17694720
                                                                00000000' 00150          .ADDRESS P.AAX
2D   2D   2D   2D   2D   2D   2D   20   20   20   20   20   20   20   20  00154 P.AAZ:   .ASCII   \        --------------------------------\
2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D  00163
                         2D   2D   2D   2D   2D   2D   2D   2D   2D   2D  00172
2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D   2D  0017C          .ASCII   \------------------------\
                              2D   2D   2D   2D   2D   2D   2D   2D   2D  0018B
                                                                010E0040  00194 P.AAY:   .LONG    17694784
                                                                00000000' 00198          .ADDRESS P.AAZ
                                                                          0019C P.ABB:   .BLKB    0
                                                                010E0000  0019C P.ABA:   .LONG    17694720
                                                                00000000' 001A0          .ADDRESS P.ABB
                                   00   54   45   4E   24   53   59   53  001A4 P.ABD:   .ASCII   \SYS$NET\<0>
                                                                010E0007  001AC P.ABC:   .LONG    17694727
                                                                00000000' 001B0          .ADDRESS P.ABD
                                                       00   00   00   40  001B4 P.ABF:   .ASCII   \@\<0><0><0>
                                                                010E0001  001B8 P.ABE:   .LONG    17694721
                                                                00000000' 001BC          .ADDRESS P.ABF
                                                       45   58   45   2E  001C0 P.ABG:   .ASCII   \.EXE\

                                                                                         .PSECT   $OWN$,NOEXE,2

                                                                          00000 NET_CHANNEL:
                                                                                         .BLKB    2
                                                                          00002 MBX_CHANNEL:
                                                                                         .BLKB    2
                                                                          00004 MBX_MESSAGE:
                                                                                         .BLKB    128
                                                                          00084 MBX_IOSB:
                                                                                         .BLKB    8

                                                                                DEFAULT_TIME=         P.AAB
                                                                .EXTRN  LIB$ASN_WTH_MBX
                                                                .EXTRN  LIB$SET_LOGICAL
                                                                .EXTRN  LIB$RUN_PROGRAM
                                                                .EXTRN  LIB$DO_COMMAND, LIB$PUT_OUTPUT
                                                                .EXTRN  STR$CONCAT, SYS$GETJPI
                                                                .EXTRN  SYS$FAO, SYS$SETPRN
                                                                .EXTRN  SYS$TRNLOG, SYS$BINTIM
                                                                .EXTRN  SYS$SETIMR, SYS$QIOW
                                                                .EXTRN  SYS$DASSGN

                                                                .PSECT  $CODE$,NOWRT,2

                                              0FFC 00000 NETWORK_SERVER:
                                                                .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11            ; 0144
                     5B       0000'   CF   9E 00002             MOVAB   NET_CHANNEL, R11                                ;
                     5A       0000V   CF   9E 00007             MOVAB   FAO_BUFFER, R10                                 ;
                     59   00000000G   00   9E 0000C             MOVAB   LIB$PUT_OUTPUT, R9                              ;
                     58       0000'   CF   9E 00013             MOVAB   P.AAA, R8                                       ;
                     5E        FD54   CE   9E 00018             MOVAB   -684(SP), SP                                    ;
              98     AD                1C   7D 0001D             MOVQ    #28, NFB_DESC                                   ; 0159
            FF08     CD          80   8F   9A 00021             MOVZBL  #128, TIME_DESC                                 ;
                               FF0C   CD   D4 00027             CLRL    TIME_DESC+4                                     ;
            00A4     CE        0100   8F   3C 0002B             MOVZWL  #256, BUFFER_DESC                               ;
                               00A8   CE   D4 00032             CLRL    BUFFER_DESC+4                                   ;
              54     AE          48   8F   9A 00036             MOVZBL  #72, KEY_DESC                                   ;
                                 58   AE   D4 0003B             CLRL    KEY_DESC+4                                      ;
              4C     AE    02000000   8F   D0 0003E             MOVL    #33554432, CMD_DESC                             ; 0181
                                 50   AE   D4 00046             CLRL    CMD_DESC+4                                      ;
    28            00             68   10   2C 00049             MOVC5   #16, P.AAA, #0, #40, ITEM_LIST                  ; 0192
                                 04   AE      0004E                                                                     ;
              9C     AD          A0   AD   9E 00050             MOVAB   NFB, NFB_DESC+4                                 ; 0202
            FF0C     CD        FF10   CD   9E 00055             MOVAB   TIME_BUF, TIME_DESC+4                           ; 0203
            00A8     CE        00AC   CE   9E 0005C             MOVAB   BUFFER, BUFFER_DESC+4                           ; 0204
              58     AE          5C   AE   9E 00063             MOVAB   KEYS, KEY_DESC+4                                ; 0205
              08     AE               6E   9E 00068             MOVAB   EPID, ITEM_LIST+4                               ; 0206
                                      7E   7C 0006C             CLRQ    -(SP)                                           ; 0213
                                      7E   D4 0006E             CLRL    -(SP)                                           ;
                                 10   AE   9F 00070             PUSHAB  ITEM_LIST                                       ;
                                      7E   7C 00073             CLRQ    -(SP)                                           ;
                                      7E   D4 00075             CLRL    -(SP)                                           ;
       00000000G     00               07   FB 00077             CALLS   #7, SYS$GETJPI                                  ;
                     52               50   D0 0007E             MOVL    R0, STATUS                                      ;
                     7A               52   E9 00081             BLBC    STATUS, 2$                                      ;
                                 02   AB   9F 00084             PUSHAB  MBX_CHANNEL                                     ; 0223
                                      5B   DD 00087             PUSHL   R11                                             ;
                                      7E   7C 00089             CLRQ    -(SP)                                           ;
                                 2C   A8   9F 0008B             PUSHAB  P.AAD                                           ;
       00000000G     00               05   FB 0008E             CALLS   #5, LIB$ASN_WTH_MBX                             ;
                     52               50   D0 00095             MOVL    R0, STATUS                                      ;
                     7C               52   E9 00098             BLBC    STATUS, 3$                                      ;
           0000V     CF               00   FB 0009B             CALLS   #0, ISSUE_MAILBOX_READ                          ; 0231
              2C     AE        00A4   CE   7D 000A0             MOVQ    BUFFER_DESC, PRCNAM                             ; 0239
                                      6E   DD 000A6             PUSHL   EPID                                            ; 0245
                                 30   AE   9F 000A8             PUSHAB  PRCNAM                                          ;
                                 34   AE   9F 000AB             PUSHAB  PRCNAM                                          ;
                                 40   A8   9F 000AE             PUSHAB  P.AAF                                           ;
       00000000G     00               04   FB 000B1             CALLS   #4, SYS$FAO                                     ;
```

```
                                       2C  AE  9F 000B8          PUSHAB  PRCNAM                                  : 0247
                 00000000G  00             01  FB 000BB          CALLS   #1, SYS$SETPRN                          :
                                           7E  7C 000C2          CLRQ    -(SP)                                   : 0257
                                           7E  D4 000C4          CLRL    -(SP)                                   :
                                     FF08  CD  9F 000C6          PUSHAB  TIME_DESC                               :
                                     FF08  CD  9F 000CA          PUSHAB  TIME_DESC                               :
                                       5C  A8  9F 000CE          PUSHAB  P.AAH                                   :
                 00000000G  00             06  FB 000D1          CALLS   #6, SYS$TRNLOG                          :
                            56             50  D0 000D8          MOVL    R0, STATUS                              :
                            01             56  D1 000DB          CMPL    STATUS, #1                              : 0259
                            0C             13 000DE              BEQL    1$                                      :
             FF08  CD              1C  A8  3C 000E0              MOVZWL  DEFAULT_TIME, TIME_DESC                 : 0262
             FF0C  CD              20  A8  D0 000E6              MOVL    DEFAULT_TIME+4, TIME_DESC+4             : 0263
                                     FF00  CD  9F 000EC 1$:      PUSHAB  DELTA_TIME                              : 0268
                                     FF08  CD  9F 000F0          PUSHAB  TIME_DESC                               :
                 00000000G  00             02  FB 000F4          CALLS   #2, SYS$BINTIM                          :
                            52             50  D0 000FB          MOVL    R0, STATUS                              :
                            16             52  E9 000FE 2$:      BLBC    STATUS, 3$                              :
                                           7E  D4 00101          CLRL    -(SP)                                   : 0272
                                    0000V  CF  9F 00103          PUSHAB  TIMEOUT_AST                             :
                                     FF00  CD  9F 00107          PUSHAB  DELTA_TIME                              :
                                           7E  D4 0010B          CLRL    -(SP)                                   :
                 00000000G  00             04  FB 0010D          CALLS   #4, SYS$SETIMR                          :
                            52             50  D0 00114          MOVL    R0, STATUS                              :
                            03             52  E8 00117 3$:      BLBS    STATUS, 4$                              :
                                         01EE  31 0011A          BRW     13$                                     :
   10            00         6E             00  2C 0011D 4$:      MOVC5   #0, (SP), #0, #16, NFB                  : 0279
                                       A0  AD     00122                                                          :
                     A0     AD             17  90 00124          MOVB    #23, NFB                                : 0280
                                           7E  7C 00128          CLRQ    -(SP)                                   : 0285
                                           7E  7C 0012A          CLRQ    -(SP)                                   :
                                           7E  D4 0012C          CLRL    -(SP)                                   :
                                       98  AD  9F 0012E          PUSHAB  NFB_DESC                                :
                                           7E  7C 00131          CLRQ    -(SP)                                   :
                                       90  AD  9F 00133          PUSHAB  IOSB                                    :
                                           38  DD 00136          PUSHL   #56                                     :
                            7E             6B  3C 00138          MOVZWL  NET_CHANNEL, -(SP)                      :
                                           7E  D4 0013B          CLRL    -(SP)                                   :
                 00000000G  00             0C  FB 0013D          CALLS   #12, SYS$QIOW                           :
                            56             50  D0 00144          MOVL    R0, STATUS                              :
                            07             56  E9 00147          BLBC    STATUS, 5$                              : 0287
                            56         90  AD  3C 0014A          MOVZWL  IOSB, STATUS                            : 0288
                            17             56  E8 0014E          BLBS    STATUS, 6$                              :
                            2C             56  D1 00151 5$:      CMPL    STATUS, #44                             : 0290
                                           6E  12 00154          BNEQ    7$                                      :
                            7E             6B  3C 00156          MOVZWL  NET_CHANNEL, -(SP)                      : 0293
                 00000000G  00             01  FB 00159          CALLS   #1, SYS$DASSGN                          :
                            50 10000004    8F  D0 00160          MOVL    #268435460, R0                          : 0297
                                           04     00167          RET                                             :
                            57         94  AD  D0 00168 6$:      MOVL    IOSB+4, IPID                            : 0303
   10            00         6E             00  2C 0016C          MOVC5   #0, (SP), #0, #16, NFB                  : 0305
                                       A0  AD     00171                                                          :
                     A0     AD             22  90 00173          MOVB    #34, NFB                                : 0307
                     A2     AD             12  90 00177          MOVB    #18, NFB+2                              : 0308
                     A4     AD 12010010    8F  D0 0017B          MOVL    #302055440, NFB+4                       : 0309
                                       A3  AD  94 00183          CLRB    NFB+3                                   : 0310
            B0   AD         64  A8         10  28 00186          MOVC3   #16, P.AAJ, NFB+16                      : 0317
```

```
NETWORK_SERVER                                          F 6
V04-000                                                 16-Sep-1984 01:39:23    VAX-11 Bliss-32 V4.0-742          Page  13
                                                        14-Sep-1984 12:49:31    [NETACP.SRC]SERVER.B32;1                (3)

                                     5C    AE D4 0018C        CLRL    KEYS                                  ; 0319
                      60  AE               57 D0 0018F        MOVL    IPID, KEYS+4                          ; 0320
                                     64    AE B4 00193        CLRW    KEYS+8                                ; 0321
                                           7E 7C 00196        CLRQ    -(SP)                                 ; 0328
                                   00AC    CE 9F 00198        PUSHAB  BUFFER_DESC                           ;
                                           7E D4 0019C        CLRL    -(SP)                                 ;
                                     64    AE 9F 0019E        PUSHAB  KEY_DESC                              ;
                                     98    AD 9F 001A1        PUSHAB  NFB_DESC                              ;
                                           7E 7C 001A4        CLRQ    -(SP)                                 ;
                                     90    AD 9F 001A6        PUSHAB  IOSB                                  ;
                                           38 DD 001A9        PUSHL   #56                                   ;
                          7E               6B 3C 001AB        MOVZWL  NET_CHANNEL, -(SP)                    ;
                                           7E D4 001AE        CLRL    -(SP)                                 ;
               00000000G  00               0C FB 001B0        CALLS   #12, SYS$QIOW                         ;
                          56               50 D0 001B7        MOVL    R0, STATUS                            ;
                          07               56 E9 001BA        BLBC    STATUS, 7$                            ; 0330
                          56         90    AD 3C 001BD        MOVZWL  IOSB, STATUS                          ; 0331
                          16               56 E8 001C1        BLBS    STATUS, 8$                            ;
                                           56 DD 001C4 7$:    PUSHL   STATUS                                ; 0334
               00000000G  00               01 FB 001C6        CALLS   #1, LIB$SIGNAL                        ;
                          7E               6B 3C 001CD        MOVZWL  NET_CHANNEL, -(SP)                    ; 0335
               00000000G  00               01 FB 001D0        CALLS   #1, SYS$DASSGN                        ;
                                         0143 31 001D7        BRW     14$                                   ; 0336
                          51       00AC    CE 9E 001DA 8$:    MOVAB   BUFFER, PTR                           ; 0339
                          50               61 3C 001DF        MOVZWL  (PTR), R0                             ; 0341
                      44  AE               50 D0 001E2        MOVL    R0, NCB_DESC                          ;
                      48  AE         02    A1 9E 001E6        MOVAB   2(R1), NCB_DESC+4                     ; 0342
                          51         02  A041 9E 001EB        MOVAB   2(R0)[PTR], PTR                       ; 0343
                          50               61 3C 001F0        MOVZWL  (PTR), R0                             ; 0345
                      34  AE               50 D0 001F3        MOVL    R0, FILESPEC                          ;
                      38  AE         02    A1 9E 001F7        MOVAB   2(R1), FILESPEC+4                     ; 0346
                          51         02  A041 9E 001FC        MOVAB   2(R0)[PTR], PTR                       ; 0347
                          50               61 3C 00201        MOVZWL  (PTR), R0                             ; 0349
                      2C  AE               50 D0 00204        MOVL    R0, PRCNAM                            ;
                      30  AE         02    A1 9E 00208        MOVAB   2(R1), PRCNAM+4                       ; 0350
                          51         02  A041 9E 0020D        MOVAB   2(R0)[PTR], PTR                       ; 0351
  48   BE             44  AE               2F 3A 00212        LOCC    #47, NCB_DESC, @NCB_DESC+4            ; 0353
                                           02 12 00218        BNEQ    9$                                    ;
                                           51 D4 0021A        CLRL    R1                                    ;
  3C   AE             51             48    AE C3 0021C 9$:    SUBL3   NCB_DESC+4, PTR, ASCII_NCB_DESC       ; 0355
                      40  AE         48    AE D0 00222        MOVL    NCB_DESC+4, ASCII_NCB_DESC+4          ; 0356
                                     74    A8 9F 00227        PUSHAB  P.AAK                                 ; 0358
                          6A               01 FB 0022A        CALLS   #1, FAO_BUFFER                        ;
                                           50 DD 0022D        PUSHL   R0                                    ;
                          69               01 FB 0022F        CALLS   #1, LIB$PUT_OUTPUT                    ;
                                   00BC    C8 9F 00232        PUSHAB  P.AAM                                 ; 0359
                          6A               01 FB 00236        CALLS   #1, FAO_BUFFER                        ;
                                           50 DD 00239        PUSHL   R0                                    ;
                          69               01 FB 0023B        CALLS   #1, LIB$PUT_OUTPUT                    ;
                                   00C4    C8 9F 0023E        PUSHAB  P.AAO                                 ; 0360
                          6A               01 FB 00242        CALLS   #1, FAO_BUFFER                        ;
                                           50 DD 00245        PUSHL   R0                                    ;
                          69               01 FB 00247        CALLS   #1, LIB$PUT_OUTPUT                    ;
                                           7E D4 0024A        CLRL    -(SP)                                 ; 0361
                                   00F4    C8 9F 0024C        PUSHAB  P.AAQ                                 ;
                          6A               02 FB 00250        CALLS   #2, FAO_BUFFER                        ;
                                           50 DD 00253        PUSHL   R0                                    ;
```

```
                                 69             01  FB 00255            CALLS   #1, LIB$PUT_OUTPUT                      :
                                            3C  AE  9F 00258            PUSHAB  ASCII_NCB_DESC                          : 0362
                                          0120  C8  9F 0025B            PUSHAB  P.AAS                                   :
                                 6A             02  FB 0025F            CALLS   #2, FAO_BUFFER                          :
                                 50             DD  00262               PUSHL   R0                                      :
                                 69             01  FB 00264            CALLS   #1, LIB$PUT_OUTPUT                      :
                                            34  AE  9F 00267            PUSHAB  FILESPEC                                : 0363
                                          0144  C8  9F 0026A            PUSHAB  P.AAU                                   :
                                 6A             02  FB 0026E            CALLS   #2, FAO_BUFFER                          :
                                                50  DD 00271            PUSHL   R0                                      :
                                 69             01  FB 00273            CALLS   #1, LIB$PUT_OUTPUT                      :
                                          014C  C8  9F 00276            PUSHAB  P.AAW                                   : 0364
                                 6A             01  FB 0027A            CALLS   #1, FAO_BUFFER                          :
                                                50  DD 0027D            PUSHL   R0                                      :
                                 69             01  FB 0027F            CALLS   #1, LIB$PUT_OUTPUT                      :
                                          0194  C8  9F 00282            PUSHAB  P.AAY                                   : 0365
                                 6A             01  FB 00286            CALLS   #1, FAO_BUFFER                          :
                                                50  DD 00289            PUSHL   R0                                      :
                                 69             01  FB 0028B            CALLS   #1, LIB$PUT_OUTPUT                      :
                                          019C  C8  9F 0028E            PUSHAB  P.ABA                                   : 0366
                                 6A             01  FB 00292            CALLS   #1, FAO_BUFFER                          :
                                                50  DD 00295            PUSHL   R0                                      :
                                 69             01  FB 00297            CALLS   #1, LIB$PUT_OUTPUT                      :
                                            2C  AE  9F 0029A            PUSHAB  PRCNAM                                  : 0369
                       00000000G 00             01  FB 0029D            CALLS   #1, SYS$SETPRN                          :
                                 52             50  D0 002A4            MOVL    R0, STATUS                              :
                                 61             52  E9 002A7            BLBC    STATUS, 13$                             :
                                            44  AE  9F 002AA            PUSHAB  NCB_DESC                                : 0373
                                          01AC  C8  9F 002AD            PUSHAB  P.ABC                                   :
                       00000000G 00             02  FB 002B1            CALLS   #2, LIB$SET_LOGICAL                     :
                                 52             50  D0 002B8            MOVL    R0, STATUS                              :
                                 4D             52  E9 002BB            BLBC    STATUS, 13$                             :
                              4F AE             02  90 002BE            MOVB    #2, CMD_DESC+3                          : 0375
                                            50  AE  D4 002C2            CLRL    CMD_DESC+4                              : 0376
                                            34  AE  9F 002C5            PUSHAB  FILESPEC                                : 0379
                                          01B8  C8  9F 002C8            PUSHAB  P.ABE                                   :
                                            54  AE  9F 002CC            PUSHAB  CMD_DESC                                :
                       00000000G 00             03  FB 002CF            CALLS   #3, STR$CONCAT                          :
                                 52             50  D0 002D6            MOVL    R0, STATUS                              :
                                 2F             52  E9 002D9            BLBC    STATUS, 13$                             :
   38  BE      34  AE      01C0  C8             04  39 002DC            MATCHC  #4, P.ABG, FILESPEC, @FILESPEC+4        : 0383
                                                03  13 002E5            BEQL    10$                                     :
                                 53             04  D0 002E7            MOVL    #4, R3                                  :
                                 53             04  C2 002EA  10$:      SUBL2   #4, R3                                  :
                                                0C  13 002ED            BEQL    11$                                     :
                                            34  AE  9F 002EF            PUSHAB  FILESPEC                                : 0386
                       00000000G 00             01  FB 002F2            CALLS   #1, LIB$RUN_PROGRAM                     :
                                                0A  11 002F9            BRB     12$                                     :
                                            4C  AE  9F 002FB  11$:      PUSHAB  CMD_DESC                                : 0389
                       00000000G 00             01  FB 002FE            CALLS   #1, LIB$DO_COMMAND                      :
                                 52             50  D0 00305  12$:      MOVL    R0, STATUS                              :
                                 12             52  E8 00308            BLBS    STATUS, 14$                             :
                                                52  DD 0030B  13$:      PUSHL   STATUS                                  :
                       00000000G 00             01  FB 0030D            CALLS   #1, LIB$SIGNAL                          :
                    50           52 10000000    8F  C9 00314            BISL3   #268435456, STATUS, R0                  :
                                                    04 0031C            RET                                             :
                                 50             01  D0 0031D  14$:      MOVL    #1, R0                                  : 0397
```

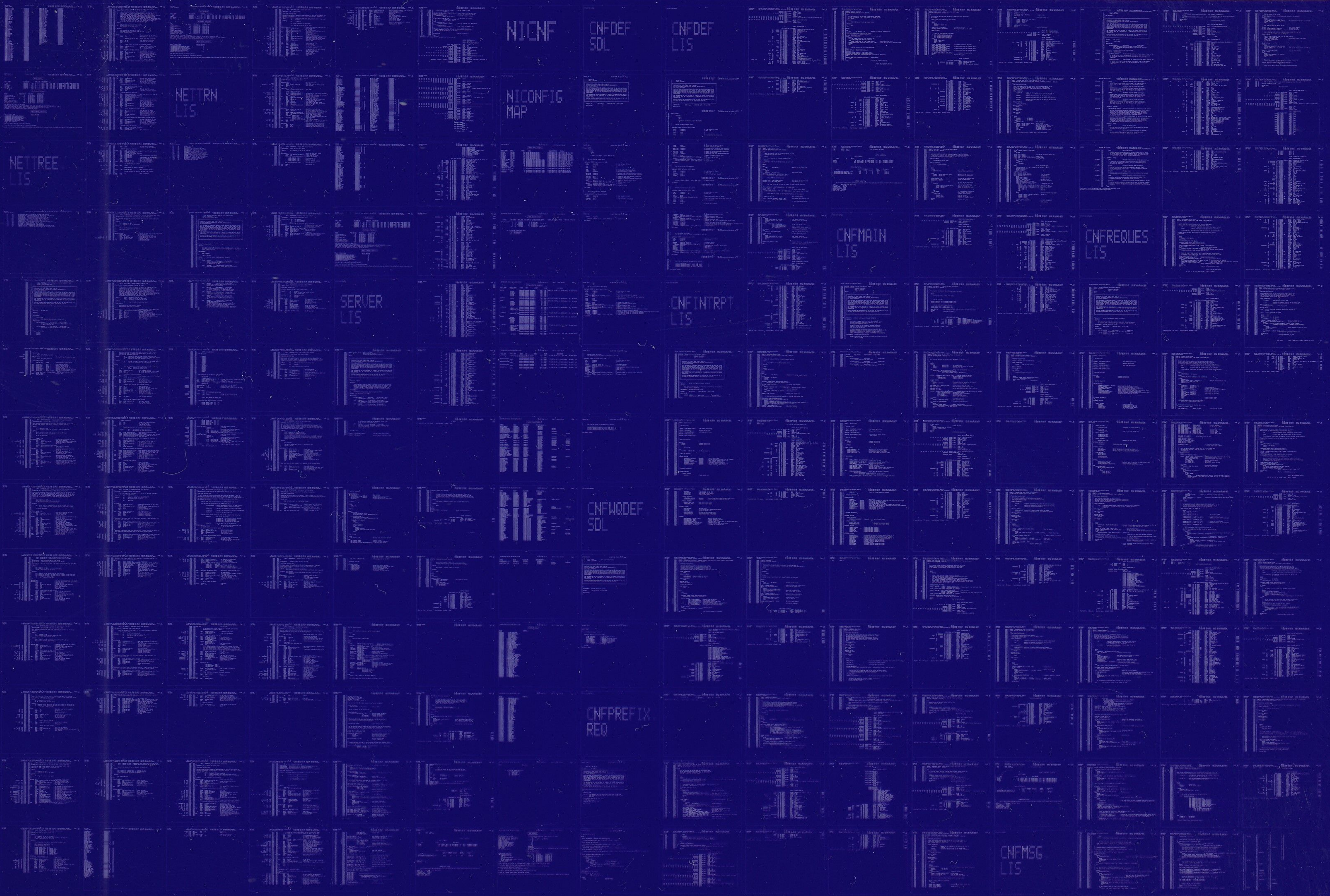
```
                                             04 00320         RET                                          ; 0399

; Routine Size:  801 bytes,    Routine Base:  $CODE$ + 0000
;  403          0400    1 ROUTINE timeout_ast: NOVALUE =
;  404          0401    1
;  405          0402    1 !--
;  406          0403    1 !
;  407          0404    1 !       This AST is called when our timer has expired.  Since the
;  408          0405    1 !       DCLSERV QIO has not completed in the required amount of time,
;  409          0406    1 !       we assume that there are no more requests to be handled by this
;  410          0407    1 !       process, and we go away.  This is done by cancelling the DECLSERV
;  411          0408    1 !       QIO.
;  412          0409    1 !
;  413          0410    1 ! Inputs:
;  414          0411    1 !
;  415          0412    1 !       net_channel = Network channel which has DECLSERV pending.
;  416          0413    1 !
;  417          0414    1 ! Outputs:
;  418          0415    1 !
;  419          0416    1 !       None
;  420          0417    1 !---
;  421          0418    1
;  422          0419    2 BEGIN
;  423          0420    2
;  424          0421    2 $CANCEL(CHAN = .net_channel);           ! Cancel the DECLSERV QIO
;  425          0422    2
;  426          0423    1 END;


                                                                  .EXTRN  SYS$CANCEL

                                              0000 00000 TIMEOUT_AST:
                                                                  .WORD   Save nothing                           ; 0400
                         7E     0000'  CF  3C 00002              MOVZWL  NET_CHANNEL, -(SP)                     ; 0421
                00000000G  00          01  FB 00007              CALLS   #1, SYS$CANCEL                         ;
                                           04 0000E              RET                                            ; 0423

; Routine Size:  15 bytes,    Routine Base:  $CODE$ + 0321
;  428          0424  1 ROUTINE issue_mailbox_read: NOVALUE =
;  429          0425  1
;  430          0426  1 !---
;  431          0427  1 !
;  432          0428  1 !       Issue an asynchronous QIO on the associated mailbox
;  433          0429  1 !       for the network channel waiting for broadcast messages.
;  434          0430  1 !
;  435          0431  1 ! Inputs:
;  436          0432  1 !
;  437          0433  1 !       mbx_channel = Channel number for mailbox
;  438          0434  1 !
;  439          0435  1 ! Outputs:
;  440          0436  1 !
;  441          0437  1 !       None
;  442          0438  1 !---
;  443          0439  1
;  444          0440  2 BEGIN
;  445          0441  2
;  446          0442  2 LOCAL
;  447          0443  2     status;
;  448          0444  2
;  449        P 0445  2 signal_if_error(
;  450        P 0446  2     $QIO(FUNC = IO$_READVBLK,           ! Issue read on mailbox
;  451        P 0447  2         CHAN = .mbx_channel,
;  452        P 0448  2         EFN = 1,
;  453        P 0449  2         IOSB = mbx_iosb,
;  454        P 0450  2         ASTADR = net_interrupt,
;  455        P 0451  2         P1 = mbx_message,
;  456          0452  2         P2 = mbx_maxmsg));
;  457          0453  2
;  458          0454  1 END;


                                                            .EXTRN  SYS$QIO

                                          0004 00000 ISSUE_MAILBOX_READ:
                                                            .WORD   Save R2                             : 0424
                                      7E   7C 00002         CLRQ    -(SP)                               : 0452
                                      7E   7C 00004         CLRQ    -(SP)                               :
                       7E       80    8F   9A 00006         MOVZBL  #128, -(SP)                         :
                              0000'   CF   9F 0000A         PUSHAB  MBX_MESSAGE                         :
                                      7E   D4 0000E         CLRL    -(SP)                               :
                              0000V   CF   9F 00010         PUSHAB  NET_INTERRUPT                       :
                              0000'   CF   9F 00014         PUSHAB  MBX_IOSB                            :
                                      31   DD 00018         PUSHL   #49                                 :
                       7E     0000'   CF   3C 0001A         MOVZWL  MBX_CHANNEL, -(SP)                  :
                                      01   DD 0001F         PUSHL   #1                                  :
              00000000G  00           0C   FB 00021         CALLS   #12, SYS$QIO                        :
                         52           50   D0 00028         MOVL    R0, STATUS                          :
                         09           52   E8 0002B         BLBS    STATUS, 1$                          :
                                      52   DD 0002E         PUSHL   STATUS                              :
              00000000G  00           01   FB 00030         CALLS   #1, LIB$SIGNAL                      :
                                      04 00037 1$:          RET                                         : 0454

; Routine Size:  56 bytes,    Routine Base:  $CODE$ + 0330
```

```
_S

Va
--
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
00
```

```
0455  1 ROUTINE net_interrupt: NOVALUE =
0456  1
0457  1 !---
0458  1 !
0459  1 !       This AST routine is called when the outstanding QIO
0460  1 !       on the associated mailbox completes.  If the interrupt
0461  1 !       indicates that the network is going down, then make us
0462  1 !       go away by canceling any I/O on the network channel
0463  1 !       (most likely a pending DECLSERV).
0464  1 !
0465  1 ! Inputs:
0466  1 !
0467  1 !       mbx_message = Mailbox message
0468  1 !       net_channel = Channel to network ACP
0469  1 !
0470  1 ! Outputs:
0471  1 !
0472  1 !       None
0473  1 !---
0474  1
0475  2 BEGIN
0476  2
0477  2 IF .mbx_message [0] EQL msg$_netshut     ! If network shutting down,
0478  2 THEN
0479  3     BEGIN
0480  3     $DASSGN(CHAN = .net_channel);        ! Cancel any pending DECLSERV I/O
0481  3     net_channel = 0;                     ! Mark channel no longer active
0482  3     RETURN;                              ! Do not re-issue mailbox read
0483  2     END;
0484  2
0485  2 issue_mailbox_read();                   ! Issue another read on mailbox
0486  2
0487  1 END;



                                       0000 00000 NET_INTERRUPT:
                                                       .WORD    Save nothing                                    ; 0455
                  3B       0000'  CF   91 00002        CMPB     MBX_MESSAGE, #59                                ; 0477
                                  11   12 00007        BNEQ     1$                                              ;
                  7E       0000'  CF   3C 00009        MOVZWL   NET_CHANNEL, -(SP)                              ; 0480
        00000000G 00              01   FB 0000E        CALLS    #1, SYS$DASSGN                                  ;
                           0000'  CF   B4 00015        CLRW     NET_CHANNEL                                     ; 0481
                                       04 00019        RET                                                      ; 0479
              AA  AF              00   FB 0001A 1$:    CALLS    #0, ISSUE_MAILBOX_READ                          ; 0485
                                       04 0001E        RET                                                      ; 0487

; Routine Size:  31 bytes,    Routine Base:  $CODE$ + 0368
;  494          0488  1 ROUTINE fao_buffer (ctrstr,args) =
;  495          0489  2 BEGIN
;  496          0490  2
;  497          0491  2 !---
;  498          0492  2 !
;  499          0493  2 !       This routine passes an ascii string through the FAO
;  500          0494  2 !       system service with any number of specified parameters.
;  501          0495  2 !
;  502          0496  2 !---
;  503          0497  2
;  504          0498  2 OWN
;  505          0499  2     desc :      VECTOR[2],            ! Result descriptor
;  506          0500  2     buf :       VECTOR[512,BYTE];     ! Output buffer
;  507          0501  2
;  508          0502  2 MAP
;  509          0503  2     ctrstr :    REF VECTOR[2],
;  510          0504  2     args :      VECTOR[4];
;  511          0505  2
;  512          0506  2 desc[0] = 512;                        ! Set up result descriptor
;  513          0507  2 desc[1] = buf;
;  514          0508  2 $faol(ctrstr=.ctrstr,outlen=desc,outbuf=desc,prmlst=args);
;  515          0509  2 RETURN desc;
;  516          0510  1 END;


                                                                   .PSECT  $OWN$,NOEXE,2

                                                0008C DESC:        .BLKB   8
                                                00094 BUF:         .BLKB   512

                                                                   .EXTRN  SYS$FAOL

                                                                   .PSECT  $CODE$,NOWRT,2

                                           0004 00000 FAO_BUFFER:
                                                                   .WORD   Save R2                                   ; 0488
                          52     0000'  CF   9E 00002              MOVAB   DESC, R2                                  ;
                          62     0200   8F   3C 00007              MOVZWL  #512, DESC                                ; 0506
                     04   A2       08   A2   9E 0000C              MOVAB   BUF, DESC+4                               ; 0507
                                   08   AC   9F 00011              PUSHAB  ARGS                                      ; 0508
                                        52   DD 00014              PUSHL   R2                                        ;
                                        52   DD 00016              PUSHL   R2                                        ;
                                   04   AC   DD 00018              PUSHL   CTRSTR                                    ;
                 00000000G  00          04   FB 0001B              CALLS   #4, SYS$FAOL                              ;
                          50            62   9E 00022              MOVAB   DESC, R0                                  ; 0509
                                             04 00025              RET                                               ; 0510

; Routine Size:  38 bytes,    Routine Base:  $CODE$ + 0387
;  518          0511  1 END
;  519          0512  0 ELUDOM



                                                                  .EXTRN  LIB$SIGNAL

;                                      PSECT SUMMARY
;
;       Name                      Bytes                      Attributes
;
;  $OWN$                               660  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                              452  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                              941  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                                Library Statistics
;
;                                             ---- --- Symbols --------       Pages       Processing
;       File                                  Tot l    Loaded   Percent       Mapped      Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1           9776        26         0         581         00:01.0
;  _$255$DUA28:[SHRLIB]NET.L32;1               1279        16         1          63         00:00.9



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


;                                        COMMAND QUALIFIERS

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

; Size:          941 code + 1112 data bytes
; Run Time:         00:19.8
; Elapsed Time:     00:38.3
; Lines/CPU Min:     1554
; Lexemes/CPU-Min: 22615
; Memory Used:  252 pages
; Compilation Complete
```

0279 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



NICNF

CNFDEF
SDL

CNFDEF
LIS

NETTRN
LIS

NICONFIG
MAP

NETTREE
LIS

CNFMAIN
LIS

CNFREQUES
LIS

SERVER
LIS

CNFINTRPT
LIS

CNFWQDEF
SDL

CNFPREFIX
REQ

CNFMSG
LIS