```
EEEEEEEEEEEEEEE  VVV         VVV   LLL
EEEEEEEEEEEEEEE  VVV         VVV   LLL
EEEEEEEEEEEEEEE  VVV         VVV   LLL
EEE              VVV         VVV   LLL
EEE              VVV         VVV   LLL
EEE              VVV         VVV   LLL
EEE              VVV         VVV   LLL
EEE              VVV         VVV   LLL
EEE              VVV         VVV   LLL
EEEEEEEEEEEE     VVV         VVV   LLL
EEEEEEEEEEEE     VVV         VVV   LLL
EEEEEEEEEEEE     VVV         VVV   LLL
EEE              VVV         VVV   LLL
EEE              VVV         VVV   LLL
EEE              VVV         VVV   LLL
EEE                 VVV   VVV      LLL
EEE                 VVV   VVV      LLL
EEE                 VVV   VVV      LLL
EEEEEEEEEEEEEEE        VVV         LLLLLLLLLLLLLLL
EEEEEEEEEEEEEEE        VVV         LLLLLLLLLLLLLLL
EEEEEEEEEEEEEEE        VVV         LLLLLLLLLLLLLLL
```

```
_$25

Valu
----
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
0200
```

FILEID**EVLSHOW

```
EEEEEEEEEE  VV      VV  LL          SSSSSSSS  HH      HH    OOOOOO    WW      WW
EEEEEEEEEE  VV      VV  LL          SSSSSSSS  HH      HH    OOOOOO    WW      WW
EE          VV      VV  LL        SS          HH      HH  OO      OO  WW      WW
EE          VV      VV  LL        SS          HH      HH  OO      OO  WW      WW
EE          VV      VV  LL        SS          HH      HH  OO      OO  WW      WW
EE          VV      VV  LL        SS          HH      HH  OO      OO  WW      WW
EEEEEEEE    VV      VV  LL          SSSSSS    HHHHHHHHHH  OO      OO  WW      WW
EEEEEEEE    VV      VV  LL          SSSSSS    HHHHHHHHHH  OO      OO  WW      WW
EE          VV      VV  LL                SS  HH      HH  OO      OO  WW  WW  WW
EE          VV      VV  LL                SS  HH      HH  OO      OO  WW  WW  WW
EE            VV  VV    LL                SS  HH      HH  OO      OO  WWWW  WWWW      ....
EE            VV  VV    LL                SS  HH      HH  OO      OO  WWWW  WWWW      ....
EEEEEEEEEE      VV      LLLLLLLLLL  SSSSSSSS  HH      HH    OOOOOO    WW      WW      ....
EEEEEEEEEE      VV      LLLLLLLLLL  SSSSSSSS  HH      HH    OOOOOO    WW      WW      ....

LL              IIIIII    SSSSSSSS
LL              IIIIII    SSSSSSSS
LL                II    SS
LL                II    SS
LL                II    SS
LL                II    SS
LL                II      SSSSSS
LL                II      SSSSSS
LL                II            SS
LL                II            SS
LL                II            SS
LL                II            SS
LLLLLLLLLL      IIIIII    SSSSSSSS
LLLLLLLLLL      IIIIII    SSSSSSSS
```

```
;  1     0001  0 %TITLE  'Routines to Obtain Data from NETACP'
;  2     0002  0 MODULE EVLSHOW (IDENT = 'V04-000') =
;  3     0003  1 BEGIN
;  4     0004  1
;  5     0005  1 !
;  6     0006  1 !*****************************************************************************
;  7     0007  1 !*                                                                           *
;  8     0008  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;  9     0009  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
; 10     0010  1 !*  ALL RIGHTS RESERVED.                                                     *
; 11     0011  1 !*                                                                           *
; 12     0012  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
; 13     0013  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
; 14     0014  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
; 15     0015  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
; 16     0016  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY  *
; 17     0017  1 !*  TRANSFERRED.                                                             *
; 18     0018  1 !*                                                                           *
; 19     0019  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
; 20     0020  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
; 21     0021  1 !*  CORPORATION.                                                             *
; 22     0022  1 !*                                                                           *
; 23     0023  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
; 24     0024  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
; 25     0025  1 !*                                                                           *
; 26     0026  1 !*                                                                           *
; 27     0027  1 !*****************************************************************************
; 28     0028  1 !
; 29     0029  1
; 30     0030  1 !++
; 31     0031  1 ! FACILITY:     DECnet Event Logging (EVL)
; 32     0032  1 !
; 33     0033  1 ! ABSTRACT:
; 34     0034  1 !
; 35     0035  1 !       This module contains routines and data to obtain information
; 36     0036  1 !       from the network.
; 37     0037  1 !
; 38     0038  1 ! ENVIRONMENT:  VAX/VMS Operating System
; 39     0039  1 !
; 40     0040  1 ! AUTHOR:       Darrell Duffy   , CREATION DATE:  7-July-1980
; 41     0041  1 !
; 42     0042  1 ! MODIFIED BY:
; 43     0043  1 !
; 44     0044  1 !       V001    TMH0001         Tim Halvorsen   02-Jun-1982
; 45     0045  1 !                               Convert to use new format NETACP control QIO interface.
; 46     0046  1 !--
```

```
;    48     0047  1 %SBTTL  'Definitions'
;    49     0048  1
;    50     0049  1 !
;    51     0050  1 ! TABLE OF CONTENTS:
;    52     0051  1 !
;    53     0052  1
;    54     0053  1 FORWARD ROUTINE
;    55     0054  1         EVL$OBTAINNETCHAN       : NOVALUE,
;    56     0055  1         EVL$NETSHOW,
;    57     0056  1         EVL$INITLOCALNODE       : NOVALUE
;    58     0057  1         ;
;    59     0058  1
;    60     0059  1 !
;    61     0060  1 ! INCLUDE FILES:
;    62     0061  1 !
;    63     0062  1
;    64     0063  1 LIBRARY 'SYS$LIBRARY:STARLET';          ! VMS common definitions
;    65     0064  1 LIBRARY 'LIB$:EVLIBRARY';               ! BLISS definitions
;    66     0065  1 LIBRARY 'SHRLIB$:NET';                  ! Network ACP interface
;    67     0066  1
;    68     0067  1 !
;    69     0068  1 ! OWN STORAGE:
;    70     0069  1 !
;    71     0070  1
;    72     0071  1 GLOBAL
;    73     0072  1     EVL$GW_NETSHOCHAN   : WORD          ! Channel to network for show
;    74     0073  1     ;
;    75     0074  1
;    76     0075  1 !
;    77     0076  1 ! EXTERNAL REFERENCES:
;    78     0077  1 !
;    79     0078  1
;    80     0079  1 EXTERNAL LITERAL                        ! Error codes
;    81     0080  1         EVL$_NETASN,                    ! Netdevice could not be assigned
;    82     0081  1         EVL$_ACPSHO                     ! Error from ACP show function
;    83     0082  1         ;
;    84     0083  1
;    85     0084  1 EXTERNAL
;    86     0085  1     EVL$GT_LOCALNODE,                   ! Buffer for local node address
;    87     0086  1                                         ! and name
;    88     0087  1     EVL$GB_LOCALNODE : BYTE             ! Length of data in bytes
;    89     0088  1     ;
```

```
;    91          0089  1 %SBTTL  'EVL$OBTAINNETCHAN  Obtain a Channel to NET'
;    92          0090  1 GLOBAL ROUTINE EVL$OBTAINNETCHAN (CHANADR) :NOVALUE =
;    93          0091  1
;    94          0092  1 !++
;    95          0093  1 ! FUNCTIONAL DESCRIPTION:
;    96          0094  1 !
;    97          0095  1 !       Obtain a channel to the network.  Probably for control qio
;    98          0096  1 !       functions.  This routine performs the error signalling
;    99          0097  1 !       in case something is wrong with the network.
;   100          0098  1 !
;   101          0099  1 ! FORMAL PARAMETERS:
;   102          0100  1 !
;   103          0101  1 !       CHANADR         Address of a word to return channel
;   104          0102  1 !
;   105          0103  1 ! IMPLICIT INPUTS:
;   106          0104  1 !
;   107          0105  1 !       NONE
;   108          0106  1 !
;   109          0107  1 ! IMPLICIT OUTPUTS:
;   110          0108  1 !
;   111          0109  1 !       NONE
;   112          0110  1 !
;   113          0111  1 ! ROUTINE VALUE:
;   114          0112  1 ! COMPLETION CODES:
;   115          0113  1 !
;   116          0114  1 !       NONE
;   117          0115  1 !
;   118          0116  1 ! SIDE EFFECTS:
;   119          0117  1 !
;   120          0118  1 !       NONE
;   121          0119  1 !
;   122          0120  1 !--
;   123          0121  1
;   124          0122  2     BEGIN
;   125          0123  2
;   126          0124  2     LOCAL
;   127          0125  2         STATUS
;   128          0126  2         ;
;   129          0127  2
;   130          0128  2     IF NOT
;   131        P 0129  3         ( STATUS = $ASSIGN                        ! Obtain the channel
;   132        P 0130  3             (
;   133        P 0131  3             CHAN = .CHANADR,
;   134        P 0132  3             DEVNAM = %ASCID '_NET:'
;   135          0133  4             )
;   136          0134  3         )
;   137          0135  2     THEN
;   138          0136  2         SIGNAL_STOP (EVL$_NETASN, 0, .STATUS)   ! Signal any error loudly
;   139          0137  2
;   140          0138  2
;   141          0139  1     END;


                                                        .TITLE  EVLSHOW Routines to Obtain Data from NETACP
                                                        .IDENT  \V04-000\

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

```
                    00  00  00  3A  54  45  4E  5F  00000 P.AAB:  .ASCII  \_NET:\<0><0><0>                          ;
                                    010E0005        00008 P.AAA:  .LONG   17694725                                ;
                                    00000000'       0000C         .ADDRESS P.AAB                                  ;

                                                                  .PSECT  $GLOBAL$,NOEXE,2

                                                  00000 EVL$GW_NETSHOCHAN::
                                                                  .BLKB   2

                                                                  .EXTRN  EVL$_NETASN, EVL$_ACPSHO
                                                                  .EXTRN  EVL$GT_LOCALNODE
                                                                  .EXTRN  EVL$GB_LOCALNODE
                                                                  .EXTRN  SYS$ASSIGN

                                                                  .PSECT  $CODE$,NOWRT,2

                                                      0000 00000          .ENTRY  EVL$OBTAINNETCHAN, Save nothing   ; 0090
                                                   7E   7C 00002          CLRQ    -(SP)                             ; 0133
                                              04   AC   DD 00004          PUSHL   CHANADR                           ;
                                            0000'  CF   9F 00007          PUSHAB  P.AAA                             ;
                    00000000G   00             04   FB 0000B          CALLS   #4, SYS$ASSIGN                    ;
                                11             50   E8 00012          BLBS    STATUS, 1$                        ;
                                               50   DD 00015          PUSHL   STATUS                            ; 0136
                                               7E   D4 00017          CLRL    -(SP)                             ;
                                      00000000G 8F   DD 00019          PUSHL   #EVL$_NETASN                      ;
                    00000000G   00             03   FB 0001F          CALLS   #3, LIB$STOP                      ;
                                                   04 00026 1$:       RET                                       ; 0139

; Routine Size:  39 bytes,    Routine Base:  $CODE$ + 0000
```

```
;   143     0140  1 %SBTTL  'EVL$NETSHOW  Perform a Net Show QIO'
;   144     0141  1 GLOBAL ROUTINE EVL$NETSHOW (DATABASE, SEARCHID, SEARCHVAL,
;   145     0142  1         CONTEXT, FIELDS, FIELDSADR, RTNBFR, RTNLEN) =
;   146     0143  1
;   147     0144  1 !++
;   148     0145  1 ! FUNCTIONAL DESCRIPTION:
;   149     0146  1 !
;   150     0147  1 !       Perform a net show qio function.  The nfb and related structures
;   151     0148  1 !       are built from the parameter list of the routine and the return
;   152     0149  1 !       length and buffer are returned in the specified areas.
;   153     0150  1 !
;   154     0151  1 ! FORMAL PARAMETERS:
;   155     0152  1 !
;   156     0153  1 !       DATABASE                NFB$C_DB_LNI, OBI, NDI, CRI, PLI, EVI...
;   157     0154  1 !       SEARCHID                Field id of search key
;   158     0155  1 !       SEARCHVAL               Address of search key
;   159     0156  1 !       CONTEXT                 Address of context buffer (NFB$C_CTX_SIZE bytes)
;   160     0157  1 !                               updated to the current position on exit
;   161     0158  1 !       FIELDS                  Number of fields in fields list
;   162     0159  1 !       FIELDSADR               Address of list of fields id's
;   163     0160  1 !       RTNBFR                  Address of descriptor of buffer
;   164     0161  1 !       RTNLEN (Optional)       Address of longword to return bytes in buffer
;   165     0162  1 !
;   166     0163  1 ! IMPLICIT INPUTS:
;   167     0164  1 !
;   168     0165  1 !       EVL$GW_NETSHOCHAN       Channel to use to perform function
;   169     0166  1 !
;   170     0167  1 ! ROUTINE VALUE:
;   171     0168  1 !
;   172     0169  1 !       Status of SS$_NORMAL or SS$_ENDOFFILE
;   173     0170  1 !--
;   174     0171  1
;   175     0172  2 BEGIN
;   176     0173  2
;   177     0174  2 BUILTIN
;   178     0175  2     NULLPARAMETER;                          ! True if parameter unspecified
;   179     0176  2
;   180     0177  2 MAP
;   181     0178  2     SEARCHID:   BBLOCK,                     ! Get at subfields of longword
;   182     0179  2     FIELDSADR:  REF VECTOR;                 ! Vector of field ids
;   183     0180  2
;   184     0181  2 LITERAL
;   185     0182  2     MAXFIELDS = 20;                         ! Max number of field ids
;   186     0183  2
;   187     0184  2 LOCAL
;   188     0185  2     NFB:                                    ! Network Function Block
;   189     0186  2         BBLOCK [NFB$C_LENGTH + MAXFIELDS*4],
;   190     0187  2     NFBDSC:     VECTOR [2],                 ! Descriptor of same
;   191     0188  2     KEY:        VECTOR [128, BYTE],         ! Key block
;   192     0189  2     KEYDSC:     VECTOR [2],                 ! and its descriptor
;   193     0190  2     RTNLENGTH:  WORD,                       ! Return length of buffer
;   194     0191  2     IOSB:       BBLOCK [IOSB$C_SIZE],       ! Iosb for use here
;   195     0192  2     PTR:        REF VECTOR,                 ! Pointer to something
;   196     0193  2     NUMFIELDS,                              ! Number of fields
;   197     0194  2     STATUS;                                 ! Status return
;   198     0195  2
;   199     0196  2 CH$FILL(0, NFB$C_LENGTH, NFB);              ! Pre-zero NFB fields
```

```
;   200     0197  2 NFB [NFB$B_FCT] = NFB$C_FC_SHOW;                    ! Build function code
;   201     0198  2 NFB [NFB$B_DATABASE] = .DATABASE;                   ! and parameter code of nfb
;   202     0199  2
;   203     0200  2 PTR = KEY+4;                                        ! Build the key block
;   204     0201  2 NFB [NFB$L_SRCH_KEY] = .SEARCHID;                   ! Search key first
;   205     0202  2 IF .SEARCHVAL NEQ 0                                 ! If there is one
;   206     0203  2 THEN
;   207     0204  3     BEGIN
;   208     0205  3     PTR = CH$MOVE(                                  ! Move it in
;   209     0206  4             (IF .SEARCHID [NFB$V_TYP] EQL NFB$C_TYP_STR ! Special case the string
;   210     0207  4             THEN .(.SEARCHVAL) <0, 16> + 2
;   211     0208  3             ELSE 4),
;   212     0209  3             .SEARCHVAL, .PTR);                      ! Copy the data
;   213     0210  2     END;
;   214     0211  2
;   215     0212  2 IF .CONTEXT NEQ 0                                   ! If there is a context area,
;   216     0213  2 THEN
;   217     0214  2     PTR = CH$MOVE(                                  ! Copy it as before
;   218     0215  2             .(.CONTEXT) <0, 16> + 2,
;   219     0216  2             .CONTEXT, .PTR)
;   220     0217  2 ELSE
;   221     0218  2     NFB [NFB$V_NOCTX] = TRUE;                       ! If not, indicate no context
;   222     0219  2
;   223     0220  2 KEYDSC [0] = .PTR - KEY;                            ! Build the key descriptor
;   224     0221  2 KEYDSC [1] = KEY;
;   225     0222  2 IF .CONTEXT NEQ 0                                   ! If updating current position,
;   226     0223  2 THEN
;   227     0224  2     KEYDSC [0] = MAXU(.KEYDSC [0], 4+NFB$C_CTX_SIZE); ! Make at least this big
;   228     0225  2
;   229     0226  2 NUMFIELDS = MIN (MAXFIELDS, .FIELDS);               ! Adjust number of fields
;   230     0227  2 PTR = NFB [NFB$L_FLDID];                            ! Set pointer into NFB
;   231     0228  2 INCRU I FROM 0 TO .NUMFIELDS-1                      ! Copy the field id's
;   232     0229  2 DO
;   233     0230  2     PTR [.I] = .FIELDSADR [.I];
;   234     0231  2
;   235     0232  2 NFBDSC [0] = $BYTEOFFSET(NFB$L_FLDID) + 4*.NUMFIELDS; ! Build NFB descriptor
;   236     0233  2 NFBDSC [1] = NFB;
;   237     0234  2
;   238   P 0235  2 STATUS = $QIOW(                                     ! Perform the qio
;   239   P 0236  2                 EFN = EVL$C_SYNCH_EFN,
;   240   P 0237  2                 CHAN = .EVL$GW_NETSHOCHAN,
;   241   P 0238  2                 FUNC = IO$_ACPCONTROL,
;   242   P 0239  2                 IOSB = IOSB,
;   243   P 0240  2                 P1 = NFBDSC,
;   244   P 0241  2                 P2 = KEYDSC,
;   245   P 0242  2                 P3 = RTNLENGTH,
;   246     0243  2                 P4 = .RTNBFR);
;   247     0244  2
;   248     0245  2 IF .STATUS                                          ! Obtain the worst status
;   249     0246  2 THEN
;   250     0247  2     STATUS = .IOSB [IOSB$W_STS];
;   251     0248  2
;   252     0249  2 IF NOT .STATUS                                      ! Check it out
;   253     0250  2     AND .STATUS NEQ SS$_ENDOFFILE
;   254     0251  2 THEN
;   255     0252  2     SIGNAL_STOP (EVL$_ACPSHO, 0, .STATUS);          ! Report the error
;   256     0253  2
```

```
;  257     0254  2 IF .CONTEXT NEQ 0                                   ! If current position updated,
;  258     0255  2 THEN
;  259     0256  2     CH$MOVE(.(KEY+4) <0,16> + 2, KEY+4, .CONTEXT); ! Copy it back to context area
;  260     0257  2
;  261     0258  2 IF NOT NULLPARAMETER(9)                               ! If caller wants buffer size,
;  262     0259  2 THEN
;  263     0260  2     .RTNLEN = .RTNLENGTH;                             ! Return the length
;  264     0261  2
;  265     0262  2 RETURN .STATUS;                                       ! and the status
;  266     0263  2
;  267     0264  1 END;


                                                        .EXTRN  SYS$QIOW

                                                 01FC 00000     .ENTRY  EVL$NETSHOW, Save R2,R3,R4,R5,R6,R7,R8      ; 0141
                                          5E  FF04   CE  9E 00002     MOVAB   -252(SP), SP
                10                00      6E         00  2C 00007     MOVC5   #0, (SP), #0, #16, NFB                  ;  0196
                                              A0     AD  9E 0000C
                                    A0    AD         22  90 0000E     MOVB    #34, NFB                               ;  0197
                                    A2    AD   04    AC  90 00012     MOVB    DATABASE, NFB+2                        ;  0198
                                          53   18    AE  9E 00017     MOVAB   KEY+4, PTR                             ;  0200
                                    A4    AD   08    AC  D0 0001B     MOVL    SEARCHID, NFB+4                        ;  0201
                                          51   0C    AC  D0 00020     MOVL    SEARCHVAL, R1                          ;  0202
                                                     17  13 00024     BEQL    3$
                02        0A      AC         02      00  ED 00026     CMPZV   #0, #2, SEARCHID+2, #2                 ;  0206
                                                     08  12 0002C     BNEQ    1$
                                          50         61  3C 0002E     MOVZWL  (R1), R0                               ;  0207
                                          50         02  C0 00031     ADDL2   #2, R0
                                                     03  11 00034     BRB     2$
                                          50         04  D0 00036 1$:  MOVL    #4, R0                                 ;  0206
                                    63    61         50  28 00039 2$:  MOVC3   R0, (R1), (PTR)                       ;  0209
                                          57   10    AC  D0 0003D 3$:  MOVL    CONTEXT, R7                            ;  0212
                                                     58  D4 00041     CLRL    R8
                                                     57  D5 00043     TSTL    R7
                                                     0E  13 00045     BEQL    4$
                                                     58  D6 00047     INCL    R8
                                          50         67  3C 00049     MOVZWL  (R7), R0                               ;  0215
                                          50         02  C0 0004C     ADDL2   #2, R0
                                    63    67         50  28 0004F     MOVC3   R0, (R7), (PTR)                        ;  0216
                                                     04  11 00053     BRB     5$                                     ;  0214
                                    A1    AD         04  88 00055 4$:  BISB2   #4, NFB+1                            ;  0218
                                          50   14    AE  9E 00059 5$:  MOVAB   KEY, R0                              ;  0220
                0C        AE            53         50  C3 0005D     SUBL3   R0, PTR, KEYDSC
                                    10    AE   14    AE  9E 00062     MOVAB   KEY, KEYDSC+4                          ;  0221
                                          15         58  E9 00067     BLBC    R8, 7$                                 ;  0222
                                          50   0C    AE  D0 0006A     MOVL    KEYDSC, R0                             ;  0224
                          00000044        8F         50  D1 0006E     CMPL    R0, #68
                                                     04  1E 00075     BGEQU   6$
                                          50   44    8F  9A 00077     MOVZBL  #68, R0
                                    0C    AE         50  D0 0007B 6$:  MOVL    R0, KEYDSC
                                          50   14    AC  D0 0007F 7$:  MOVL    FIELDS, R0                             ;  0226
                                          14         50  D1 00083     CMPL    R0, #20
                                                     03  15 00086     BLEQ    8$
                                          50         14  D0 00088     MOVL    #20, R0
                                          53   B0    AD  9E 0008B 8$:  MOVAB   NFB+16, PTR                          ;  0227
```

```
                                52       FF    A0  9E 0008F          MOVAB   -1(R0), R2                                  ; 0228
                                                 51  D4 00093          CLRL    I                                           ; 0230
                                                 08  11 00095          BRB     10$
                             6341       18 BC41  D0 00097 9$:       MOVL    @FIELDSADR[I], (PTR)[I]
                                                 51  D6 0009D          INCL    I
                                52               51  D1 0009F 10$:      CMPL    I, R2
                                                 F3  1B 000A2          BLEQU   9$
                98   AD            50           02  78 000A4          ASHL    #2, NUMFIELDS, NFBDSC                       ; 0232
                          98   AD    50           10  C0 000A9          ADDL2   #16, NFBDSC
                          9C   AD          A0    AD  9E 000AD          MOVAB   NFB, NFBDSC+4                               ; 0233
                                           7E    7C 000B2          CLRQ    -(SP)                                       ; 0243
                                           1C    AC  DD 000B4          PUSHL   RTNBFR
                                           0C    AE  9F 000B7          PUSHAB  RTNLENGTH
                                           1C    AE  9F 000BA          PUSHAB  KEYDSC
                                           98    AD  9F 000BD          PUSHAB  NFBDSC
                                           7E    7C 000C0          CLRQ    -(SP)
                                           24    AE  9F 000C2          PUSHAB  IOSB
                                                 38  DD 000C5          PUSHL   #56
                                7E       0000'    CF  3C 000C7          MOVZWL  EVL$GW_NETSHOCHAN, -(SP)
                                                 01  DD 000CC          PUSHL   #1
                     00000000G  00               0C  FB 000CE          CALLS   #12, SYS$QIOW
                                56               50  D0 000D5          MOVL    R0, STATUS
                                07               56  E9 000D8          BLBC    STATUS, 11$                                 ; 0245
                                56       04      AE  3C 000DB          MOVZWL  IOSB, STATUS                                ; 0247
                                1A               56  E8 000DF          BLBS    STATUS, 12$                                 ; 0249
                     00000870   8F               56  D1 000E2 11$:      CMPL    STATUS, #2160                               ; 0250
                                                 11  13 000E9          BEQL    12$
                                                 56  DD 000EB          PUSHL   STATUS                                      ; 0252
                                                 7E  D4 000ED          CLRL    -(SP)
                                     00000000G   8F  DD 000EF          PUSHL   #EVL$_ACPSHO
                          00000000G  00               03  FB 000F5          CALLS   #3, LIB$STOP
                                     0C               58  E9 000FC 12$:      BLBC    R8, 13$                                 ; 0254
                                     50       18      AE  3C 000FF          MOVZWL  KEY+4, R0                               ; 0256
                                     50               02  C0 00103          ADDL2   #2, R0
                          67         18      AE           50  28 00106          MOVC3   R0, KEY+4, (R7)
                                     09               6C  91 0010B 13$:      CMPB    (AP), #9                                ; 0258
                                                 09  1F 0010E          BLSSU   14$
                                           24    AC  D5 00110          TSTL    36(AP)
                                                 04  13 00113          BEQL    14$
                                20   BC           6E  3C 00115          MOVZWL  RTNLENGTH, @RTNLEN                          ; 0260
                                     50           56  D0 00119 14$:      MOVL    STATUS, R0                                  ; 0262
                                                 04  0011C          RET                                                 ; 0264

; Routine Size:  285 bytes,    Routine Base:  $CODE$ + 0027
```

```
;   269          0265  1 %SBTTL  'EVL$INITLOCALNODE  Obtain Local Node Address and Name'
;   270          0266  1 GLOBAL ROUTINE EVL$INITLOCALNODE :NOVALUE =
;   271          0267  1
;   272          0268  1 !++
;   273          0269  1 ! FUNCTIONAL DESCRIPTION:
;   274          0270  1 !
;   275          0271  1 !       Obtain the local nodes address and name and format in DNA form.
;   276          0272  1 !
;   277          0273  1 ! FORMAL PARAMETERS:
;   278          0274  1 !
;   279          0275  1 !       NONE
;   280          0276  1 !
;   281          0277  1 ! IMPLICIT OUTPUTS:
;   282          0278  1 !
;   283          0279  1 !       EVL$GT_LOCALNODE
;   284          0280  1 !
;   285          0281  1 ! ROUTINE VALUE:
;   286          0282  1 !
;   287          0283  1 !       NONE
;   288          0284  1 !--
;   289          0285  1
;   290          0286  2     BEGIN
;   291          0287  2
;   292          0288  2     LOCAL
;   293          0289  2         DPTR,
;   294          0290  2         RETDSC : VECTOR [2],            ! Return buffer descriptor
;   295          0291  2         RETBFR : VECTOR [16, BYTE];     ! Return buffer
;   296          0292  2
;   297          0293  2     RETDSC [0] = 16;                    ! Setup return descriptor
;   298          0294  2     RETDSC [1] = RETBFR;
;   299          0295  2
;   300          0296  2     EVL$NETSHOW(                        ! Obtain the data
;   301          0297  2         NFB$C_DB_LNI,
;   302          0298  2         NFB$C_WILDCARD, 0,
;   303          0299  2         0,
;   304          0300  2         2, UPLIT (NFB$C_LNI_ADD, NFB$C_LNI_NAM),
;   305          0301  2         RETDSC);
;   306          0302  2
;   307          0303  2     DPTR = EVL$GT_LOCALNODE;            ! Set pointers
;   308          0304  2     DPTR = CH$MOVE (2, RETBFR, .DPTR);  ! Address for two bytes
;   309          0305  2     CH$WCHAR_A(CH$RCHAR(RETBFR+4),DPTR); ! Copy the count
;   310          0306  2     DPTR = CH$MOVE (CH$RCHAR(RETBFR+4), RETBFR+6, .DPTR); ! Copy name
;   311          0307  2     EVL$GB_LOCALNODE = .DPTR - EVL$GT_LOCALNODE; ! Set count
;   312          0308  2
;   313          0309  1     END;


                                                .PSECT  $PLIT$,NOWRT,NOEXE,2
                        01020041  01010010  00010 P.AAC:  .LONG   16842768, 16908353                    ;

                                                .PSECT  $CODE$,NOWRT,2
                                    003C 00000          .ENTRY  EVL$INITLOCALNODE, Save R2,R3,R4,R5     ; 0266
```

```
                        5E              18 C2 00002     SUBL2   #24, SP                          ;
                   10   AE              10 D0 00005     MOVL    #16, RETDSC                      ; 0293
                   14   AE              6E 9E 00009     MOVAB   RETBFR, RETDSC+4                 ; 0294
                                  10    AE 9F 0000D     PUSHAB  RETDSC                           ; 0296
                               0000'    CF 9F 00010     PUSHAB  P.AAC                            ; 0300
                                        02 DD 00014     PUSHL   #2                               ; 0296
                                        7E 7C 00016     CLRQ    -(SP)                            ;
                                        01 DD 00018     PUSHL   #1                               ;
                                        01 DD 0001A     PUSHL   #1                               ;
                 FEC2   CF              07 FB 0001C     CALLS   #7, EVL$NETSHOW                  ;
                        53     0000G    CF 9E 00021     MOVAB   EVL$GT_LOCALNODE, DPTR           ; 0303
                        83              6E B0 00026     MOVW    RETBFR, (DPTR)+                  ; 0304
                        83        04    AE 90 00029     MOVB    RETBFR+4, (DPTR)+                ; 0305
                        50        04    AE 9A 0002D     MOVZBL  RETBFR+4, R0                     ; 0306
             63    06   AE              50 28 00031     MOVC3   R0, RETBFR+6, (DPTR)             ;
                        50     0000G    CF 9E 00036     MOVAB   EVL$GT_LOCALNODE, R0             ; 0307
        0000G  CF       53              50 83 0003B     SUBB3   R0, DPTR, EVL$GB_LOCALNODE       ;
                                           04 00041     RET                                      ; 0309

; Routine Size:  66 bytes,    Routine Base:  $CODE$ + 0144
```

```
;  315          0310  1 END                          !End of module
;  316          0311  0 ELUDOM


                                                     .EXTRN  LIB$STOP

;                              PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $GLOBAL$                         2  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                          24  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                         390  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)


;                            Library Statistics
;                                        --------  Symbols  --------      Pages       Processing
;       File                             Total     Loaded    Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1      9776          7          0         581        00:01.0
;  _$255$DUA28:[EVL.OBJ]EVLIBRARY.L32;1    191          5          2          14        00:00.2
;  _$255$DUA28:[SHRLIB]NET.L32;1          1279         14          1          63        00:00.9



;                                    COMMAND QUALIFIERS

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

; Size:         390 code + 26 data bytes
; Run Time:         00:09.6
; Elapsed Time:     00:20.5
; Lines/CPU Min:     1947
; Lexemes/CPU-Min: 13058
; Memory Used:  105 pages
; Compilation Complete
```

EVLJULIAN
LIS

EVCDEF
LIS

ERRMSG
LIS

EVLIBRARY
LIS

EVLTRANS
LIS

EVLSHOW
LIS

EVLMAIN
LIS

RECEIVER
LIS