```
NNN           NNN   MMM           MMM   LLL
NNN           NNN   MMM           MMM   LLL
NNN           NNN   MMM           MMM   LLL
NNN           NNN   MMMMMM    MMMMMM   LLL
NNN           NNN   MMMMMM    MMMMMM   LLL
NNN           NNN   MMMMMM    MMMMMM   LLL
NNNNNN        NNN   MMM   MMM   MMM   LLL
NNNNNN        NNN   MMM   MMM   MMM   LLL
NNNNNN        NNN   MMM   MMM   MMM   LLL
NNN   NNN     NNN   MMM         MMM   LLL
NNN   NNN     NNN   MMM         MMM   LLL
NNN   NNN     NNN   MMM         MMM   LLL
NNN       NNNNNN   MMM         MMM   LLL
NNN       NNNNNN   MMM         MMM   LLL
NNN       NNNNNN   MMM         MMM   LLL
NNN           NNN   MMM         MMM   LLL
NNN           NNN   MMM         MMM   LLL
NNN           NNN   MMM         MMM   LLL
NNN           NNN   MMM         MMM   LLLLLLLLLLLLLLL
NNN           NNN   MMM         MMM   LLLLLLLLLLLLLLL
NNN           NNN   MMM         MMM   LLLLLLLLLLLLLLL
```

FILEID**NMLPMANIP

```
NN       NN  MM       MM  LL          PPPPPPPP    MM       MM    AAAAAA    NN       NN    IIIIII    PPPPPPPP
NN       NN  MM       MM  LL          PPPPPPPP    MM       MM    AAAAAA    NN       NN    IIIIII    PPPPPPPP
NN       NN  MMMM   MMMM  LL          PP      PP  MMMM   MMMM   AA    AA  NN       NN      II      PP      PP
NN       NN  MMMM   MMMM  LL          PP      PP  MMMM   MMMM   AA    AA  NN       NN      II      PP      PP
NNNN     NN  MM  MM  MM  LL          PP      PP  MM  MM  MM  AA    AA  NNNN     NN      II      PP      PP
NNNN     NN  MM  MM  MM  LL          PP      PP  MM  MM  MM  AA    AA  NNNN     NN      II      PP      PP
NN  NN   NN  MM      MM  LL          PPPPPPPP    MM      MM  AA    AA  NN  NN   NN      II      PPPPPPPP
NN  NN   NN  MM      MM  LL          PPPPPPPP    MM      MM  AA    AA  NN  NN   NN      II      PPPPPPPP
NN    NNNN  MM      MM  LL          PP          MM      MM  AAAAAAAAAA  NN    NNNN      II      PP
NN    NNNN  MM      MM  LL          PP          MM      MM  AAAAAAAAAA  NN    NNNN      II      PP
NN      NN  MM      MM  LL          PP          MM      MM  AA    AA  NN      NN      II      PP              ....
NN      NN  MM      MM  LL          PP          MM      MM  AA    AA  NN      NN      II      PP              ....
NN      NN  MM      MM  LLLLLLLLLL  PP          MM      MM  AA    AA  NN      NN    IIIIII    PP              ....
NN      NN  MM      MM  LLLLLLLLLL  PP          MM      MM  AA    AA  NN      NN    IIIIII    PP              ....

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 'NML internal parameter manipulation module'
;     2          0002  0 MODULE NML$PMANIP (
;     3          0003  0                 LANGUAGE (BLISS32),
;     4          0004  0                 ADDRESSING_MODE (NONEXTERNAL=GENERAL),
;     5          0005  0                 ADDRESSING_MODE (EXTERNAL=GENERAL),
;     6          0006  0                 IDENT = 'V04-000'
;     7          0007  0                 ) =
;     8          0008  1 BEGIN
;     9          0009  1 !
;    10          0010  1 !*****************************************************************************
;    11          0011  1 !*                                                                           *
;    12          0012  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    13          0013  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    14          0014  1 !*  ALL RIGHTS RESERVED.                                                      *
;    15          0015  1 !*                                                                           *
;    16          0016  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;    17          0017  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;    18          0018  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;    19          0019  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;    20          0020  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;    21          0021  1 !*  TRANSFERRED.                                                              *
;    22          0022  1 !*                                                                           *
;    23          0023  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;    24          0024  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;    25          0025  1 !*  CORPORATION.                                                              *
;    26          0026  1 !*                                                                           *
;    27          0027  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;    28          0028  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    29          0029  1 !*                                                                           *
;    30          0030  1 !*                                                                           *
;    31          0031  1 !*****************************************************************************
;    32          0032  1 !
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT:
;    35          0035  1 !
;    36          0036  1 !       This module contains routines to handle internal parameter
;    37          0037  1 !       manipulation functions.
;    38          0038  1 !
;    39          0039  1 ! ENVIRONMENT:  VAX/VMS Operating System
;    40          0040  1 !
;    41          0041  1 ! AUTHOR:  Distributed Systems Software Engineering
;    42          0042  1 !
;    43          0043  1 ! CREATION DATE:  23-JAN-1980
;    44          0044  1 !
;    45          0045  1 ! MODIFIED BY:
;    46          0046  1 !
;    47          0047  1 !       V03-003 MKP0003         Kathy Perko             4-Aug-1983
;    48          0048  1 !               Make permanent database routines transparent to the length
;    49          0049  1 !               of the ISAM keys at the beginning of the records.
;    50          0050  1 !
;    51          0051  1 !       V03-002 MKP0002         Kathy Perko             22-June-1982
;    52          0052  1 !               Add support for specifying "active X25-Protocol network".
;    53          0053  1 !
;    54          0054  1 !       V03-001 MKP0001         Kathy Perko             28-April-1982
;    55          0055  1 !               More modifications for NETACP control QIO.  Add the
;    56          0056  1 !               second search key to NFB.  Also, delete the start key.
;    57          0057  1 !
```

```
;    58          0058  1 !       V02-001 LMK0001         Len Kawell                21-Jul-1981
;    59          0059  1 !               Modifications for new NETACP control QIO.
;    60          0060  1 !--
;    61          0061  1
```

```
;    63          0062  1 %SBTTL 'Declarations'
;    64          0063  1
;    65          0064  1 !
;    66          0065  1 ! TABLE OF CONTENTS:
;    67          0066  1 !
;    68          0067  1
;    69          0068  1 FORWARD ROUTINE
;    70          0069  1     NML$SAVEPARAM,
;    71          0070  1     NML$CHKPRMVAL,
;    72          0071  1     NML$BLDSETQBF       : NOVALUE,
;    73          0072  1     NML$DEL_FIELDS,
;    74          0073  1     NML$ADD_FIELDS,
;    75          0074  1     NML$READPARLIST,
;    76          0075  1     NML$SHOWPARLIST,
;    77          0076  1     NML$BLDALLDES;
;    78          0077  1
;    79          0078  1 !
;    80          0079  1 ! INCLUDE FILES:
;    81          0080  1 !
;    82          0081  1
;    83          0082  1 LIBRARY 'LIB$:NMLLIB.L32';
;    84          0083  1 LIBRARY 'SHRLIB$:NMALIBRY.L32';
;    85          0084  1 LIBRARY 'SHRLIB$:NET.L32';
;    86          0085  1 LIBRARY 'SYS$LIBRARY:STARLET.L32';
;    87          0086  1
;    88          0087  1 !
;    89          0088  1 ! EXTERNAL REFERENCES:
;    90          0089  1 !
;    91          0090  1
;    92          0091  1 $NML_EXTDEF;
;    93          0092  1
;    94          0093  1 EXTERNAL LITERAL
;    95          0094  1     NML$_DSCBFOVF,
;    96          0095  1     NML$_QIOBFOVF,
;    97          0096  1     NML$_RECBFOVF;
;    98          0097  1
;    99          0098  1 EXTERNAL ROUTINE
;   100          0099  1     NMA$SEARCHFLD,
;   101          0100  1     NML$BLD_REPLY,
;   102          0101  1     NML$BLDP2,
;   103          0102  1     NML$ERROR_1,
;   104          0103  1     NML$ERROR_2;
;   105          0104  1
```

```
;  107          0105  1 %SBTTL 'NML$SAVEPARAM  Check parameter value'
;  108          0106  1 GLOBAL ROUTINE NML$SAVEPARAM (CPT_INDEX, LENGTH, POINTER) =
;  109          0107  1
;  110          0108  1 !++
;  111          0109  1 ! FUNCTIONAL DESCRIPTION:
;  112          0110  1 !
;  113          0111  1 !       This routine saves a parameter as a descriptor in the parameter
;  114          0112  1 !       descriptor block.
;  115          0113  1 !
;  116          0114  1 ! FORMAL PARAMETERS:
;  117          0115  1 !
;  118          0116  1 !       CPT_INDEX
;  119          0117  1 !       LENGTH
;  120          0118  1 !       POINTER
;  121          0119  1 !
;  122          0120  1 ! IMPLICIT INPUTS:
;  123          0121  1 !
;  124          0122  1 !       NML$AB_PRMSEM is the parameter semantic table.
;  125          0123  1 !       NML$AW_PRM_DES is the parameter descriptor buffer.
;  126          0124  1 !       NML$GW_PRMDESCNT contains the current number of descriptor entries.
;  127          0125  1 !
;  128          0126  1 ! IMPLICIT OUTPUTS:
;  129          0127  1 !
;  130          0128  1 !       If the parameter is valid then a descriptor entry will be created for
;  131          0129  1 !       it in NML$AW_PRM_DES and NML$GW_PRMDESCNT will be incremented.
;  132          0130  1 !
;  133          0131  1 ! ROUTINE VALUE:
;  134          0132  1 ! COMPLETION CODES:
;  135          0133  1 !
;  136          0134  1 !       Always returns NML$_STS_SUC.
;  137          0135  1 !
;  138          0136  1 ! SIDE EFFECTS:
;  139          0137  1 !
;  140          0138  1 !  If the parameter descriptor buffer is full then a software error
;  141          0139  1 ! (NMA$C_STS_MPR) is signalled with optional text to identify the error.
;  142          0140  1 !
;  143          0141  1 !--
;  144          0142  1
;  145          0143  2     BEGIN
;  146          0144  2
;  147          0145  2     BIND
;  148          0146  2         CPT_LIST      = NML$AB_CPTABLE [.CPT_INDEX, 0,0,0,0]
;  149          0147  2                         : BBLOCK [CPT$K_ENTRYLEN],
;  150          0148  2         SEMANTIC_LIST = NML$AB_PRMSEM [.CPT_LIST [CPT$W_PSTINDEX], 0,0,0,0]
;  151          0149  2                         : BBLOCK [PST$K_ENTRYLEN];
;  152          0150  2
;  153          0151  2     LOCAL
;  154          0152  2         VEC_INDEX,
;  155          0153  2         MASK      : BLOCK [1, WORD],
;  156          0154  2         MSGSIZE,                        ! Resultant message size
;  157          0155  2         OFFSET,                         ! Temporary parameter offset
;  158          0156  2         VEC       : REF BLOCKVECTOR [, 2, WORD];
;  159          0157  2
;  160          0158  2
;  161          0159  2 !
;  162          0160  2 ! Check the parameter descriptor buffer to see if there is any room left
;  163          0161  2 !
```

```
;  164     0162  2     IF .NML$GW_PRMDESCNT GEQU PDB$K_NUMBER
;  165     0163  2     THEN
;  166     0164  2         BEGIN
;  167     0165  2 !
;  168     0166  2 ! Signal parameter descriptor buffer overflow.
;  169     0167  2 !
;  170     0168  2         NML$AB_MSGBLOCK [MSB$L_FLAGS] = MSB$M_DET_FLD OR MSB$M_MSG_FLD;
;  171     0169  2         NML$AB_MSGBLOCK [MSB$B_CODE] = NMA$C_STS_MPR; ! Get error code
;  172     0170  2         NML$AB_MSGBLOCK [MSB$W_DETAIL] =
;  173     0171  2             .SEMANTIC_LIST [PST$W_DATAID]; ! Get parameter code detail
;  174     0172  2         NML$AB_MSGBLOCK [MSB$L_TEXT] = NML$_DSCBFOVF;
;  175     0173  2         NML$BLD_REPLY (NML$AB_MSGBLOCK, MSGSIZE); ! Build message
;  176     0174  2         $SIGNAL_MSG (NML$AB_SNDBUFFER, .MSGSIZE); ! Signal error message
;  177     0175  2
;  178     0176  2         END;
;  179     0177  2 !
;  180     0178  2 ! Add descriptor entry for this parameter.
;  181     0179  2 !
;  182     0180  2     NML$AW_PRM_DES [.NML$GW_PRMDESCNT, PDB$W_INDEX] = .CPT_INDEX;
;  183     0181  2     NML$AW_PRM_DES [.NML$GW_PRMDESCNT, PDB$W_COUNT] = .LENGTH;
;  184     0182  2     NML$AW_PRM_DES [.NML$GW_PRMDESCNT, PDB$A_POINTER] = .POINTER;
;  185     0183  2
;  186     0184  2     NML$GW_PRMDESCNT = .NML$GW_PRMDESCNT + 1; ! Increment descriptor count
;  187     0185  2
;  188     0186  2     RETURN NML$_STS_SUC
;  189     0187  2
;  190     0188  1     END;                                    ! End of NML$SAVEPARAM


;                                                           .TITLE  NML$PMANIP NML internal parameter manipulation
                                                                    module
                                                            .IDENT  \V04-000\

                                                            .EXTRN  NML$GB_EVTSRCTYP
                                                            .EXTRN  NML$GQ_EVTSRCDSC
                                                            .EXTRN  NML$GW_EVTCLASS
                                                            .EXTRN  NML$GB_EVTMSKTYP
                                                            .EXTRN  NML$GQ_EVTMSKDSC
                                                            .EXTRN  NML$GW_EVTSNKADR
                                                            .EXTRN  NML$GW_ACP_CHAN
                                                            .EXTRN  NML$GL_LOGMASK, NML$GQ_ENTSTRDSC
                                                            .EXTRN  NML$AB_QIOBUFFER
                                                            .EXTRN  NML$GQ_QIOBFDSC
                                                            .EXTRN  NML$AB_EXEBUFFER
                                                            .EXTRN  NML$GL_EXEDATPTR
                                                            .EXTRN  NML$GQ_EXEDATDSC
                                                            .EXTRN  NML$GQ_EXEBFDSC
                                                            .EXTRN  NML$AB_RCVBUFFER
                                                            .EXTRN  NML$GQ_RCVBFDSC
                                                            .EXTRN  NML$AB_SNDBUFFER
                                                            .EXTRN  NML$GQ_SNDBFDSC
                                                            .EXTRN  NML$GL_RCVDATLEN
                                                            .EXTRN  NML$AB_CPTABLE, NML$AB_MSGBLOCK
                                                            .EXTRN  NML$AB_ENTITY_ID
                                                            .EXTRN  NML$AB_QUALIFIER_ID
                                                            .EXTRN  NML$AB_ENTITYDATA
                                                            .EXTRN  NML$AB_NML_NMV, NML$AB_PRMSEM
```

```
                                                        .EXTRN  NML$AB_RECBUF, NML$AL_ENTINFTAB
                                                        .EXTRN  NML$AL_PERMINFTAB
                                                        .EXTRN  NML$AW_PRM_DES, NML$GB_CMD_VER
                                                        .EXTRN  NML$GB_ENTITY_CODE
                                                        .EXTRN  NML$GB_ENTITY_FORMAT
                                                        .EXTRN  NML$GL_QUALIFIER_PST
                                                        .EXTRN  NML$GB_QUALIFIER_FORMAT
                                                        .EXTRN  NML$GB_FUNCTION
                                                        .EXTRN  NML$GB_INFO, NML$GB_OPTIONS
                                                        .EXTRN  NML$GL_PRMCODE, NML$GL_PRS_FLGS
                                                        .EXTRN  NML$GL_NML_ENTITY
                                                        .EXTRN  NML$GQ_NETNAMDSC
                                                        .EXTRN  NML$GQ_RECBFDSC
                                                        .EXTRN  NML$GW_PRMDESCNT
                                                        .EXTRN  NML$_DSCBFOVF, NML$_QIOBFOVF
                                                        .EXTRN  NML$_RECBFOVF, NMA$SEARCHFLD
                                                        .EXTRN  NML$BLD_REPLY, NML$BLDP2
                                                        .EXTRN  NML$ERROR_1, NML$ERROR_2

                                                        .PSECT  $CODE$,NOWRT,2

                                                001C 00000      .ENTRY  NML$SAVEPARAM, Save R2,R3,R4                    ; 0106
                                          54 00000000G  00   9E 00002  MOVAB   NML$AW_PRM_DES, R4                      ;
                                          53 00000000G  00   9E 00009  MOVAB   NML$GW_PRMDESCNT, R3                   ;
                                          52 00000000G  00   9E 00010  MOVAB   NML$AB_MSGBLOCK, R2                     ;
                                          5E                04   C2 00017  SUBL2   #4, SP                              ;
                                50        04              AC   0A   C5 0001A  MULL3   #10, CPT_INDEX, R0                  ; 0146
                                                00000000G00040   9F 0001F  PUSHAB  NML$AB_CPTABLE[R0]                   ; 0148
                                          50                9E   3C 00026  MOVZWL  @(SP)+, R0                         ;
                                          50                10   C4 00029  MULL2   #16, R0                            ;
                                          20                63   B1 0002C  CMPW    NML$GW_PRMDESCNT, #32                ; 0162
                                                            3A   1F 0002F  BLSSU   1$                                ;
                                          62                06   D0 00031  MOVL    #6, NML$AB_MSGBLOCK                  ; 0168
                                    04    A2                05   8E 00034  MNEGB   #5, NML$AB_MSGBLOCK+4              ; 0169
                                                00000000G00040   9F 00038  PUSHAB  NML$AB_PRMSEM[R0]                   ; 0171
                                    08    A2                9E   B0 0003F  MOVW    @(SP)+, NML$AB_MSGBLOCK+8            ;
                                    0C    A2  00000000G     8F   D0 00043  MOVL    #NML$_DSCBFOVF, NML$AB_MSGBLOCK+12   ; 0172
                                                  4004      8F   BB 0004B  PUSHR   #^M<R2,SP>                       ; 0173
                              00000000G   00                02   FB 0004F  CALLS   #2, NML$BLD_REPLY                  ;
                                                            6E   DD 00056  PUSHL   MSGSIZE                           ; 0174
                                                00000000G   00   9F 00058  PUSHAB  NML$AB_SNDBUFFER                   ;
                                                01F90000    8F   DD 0005E  PUSHL   #33095680                         ;
                              00000000G   00                03   FB 00064  CALLS   #3, LIB$SIGNAL                    ;
                                          50                63   3C 0006B 1$:  MOVZWL  NML$GW_PRMDESCNT, R0             ; 0180
                                                      6440  7F   7F 0006E  PUSHAQ  NML$AW_PRM_DES[R0]                 ;
                                          9E              04   AC   B0 00071  MOVW    CPT_INDEX, @(SP)+                ;
                                                        02 A440  7F 00075  PUSHAQ  NML$AW_PRM_DES+2[R0]              ; 0181
                                          9E              08   AC   B0 00079  MOVW    LENGTH, @(SP)+                   ;
                                                        04 A440  7F 0007D  PUSHAQ  NML$AW_PRM_DES+4[R0]              ; 0182
                                          9E              0C   AC   D0 00081  MOVL    POINTER, @(SP)+                  ;
                                                            63   B6 00085  INCW    NML$GW_PRMDESCNT                 ; 0184
                                          50                01   D0 00087  MOVL    #1, R0                           ; 0186
                                                                04 0008A  RET                                       ; 0188

; Routine Size:  139 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  192          0189  1 %SBTTL 'NML$CHKPRMVAL  Check parameter value'
;  193          0190  1 GLOBAL ROUTINE NML$CHKPRMVAL (CPT_INDEX, LEN, ADR) =
;  194          0191  1
;  195          0192  1 !++
;  196          0193  1 ! FUNCTIONAL DESCRIPTION:
;  197          0194  1 !
;  198          0195  1 !       This routine verifies that parameter values from the NICE message
;  199          0196  1 !       fall within valid boundaries.
;  200          0197  1 !
;  201          0198  1 ! FORMAL PARAMETERS:
;  202          0199  1 !
;  203          0200  1 !       CPT_INDEX       Index into change parameter table.
;  204          0201  1 !       LEN             Byte count of parameter.
;  205          0202  1 !       ADR             Address of parameter.
;  206          0203  1 !
;  207          0204  1 ! IMPLICIT INPUTS:
;  208          0205  1 !
;  209          0206  1 !       NONE
;  210          0207  1 !
;  211          0208  1 ! IMPLICIT OUTPUTS:
;  212          0209  1 !
;  213          0210  1 !       NONE
;  214          0211  1 !
;  215          0212  1 ! ROUTINE VALUE:
;  216          0213  1 ! COMPLETION CODES:
;  217          0214  1 !
;  218          0215  1 !       Returns success (NML$_STS_SUC) if the paramter value is within range.
;  219          0216  1 !
;  220          0217  1 ! SIDE EFFECTS:
;  221          0218  1 !
;  222          0219  1 !       An error message (NMA$C_STS_PVA) is signalled if the value is bad.
;  223          0220  1 !
;  224          0221  1 !--
;  225          0222  1
;  226          0223  2     BEGIN
;  227          0224  2
;  228          0225  2     LOCAL
;  229          0226  2         MAX,                        ! Maximum parameter value (0 if no limit)
;  230          0227  2         MIN,                        ! Minimum parameter value
;  231          0228  2         VAL,                        ! Parameter value to compare
;  232          0229  2         STATUS;                     ! Status of the range checking operations
;  233          0230  2 !
;  234          0231  2 ! The parameter semantic table index is determined by looking in the change
;  235          0232  2 ! parameter table.
;  236          0233  2 !
;  237          0234  2     BIND
;  238          0235  2         CPT_LIST      = NML$AB_CPTABLE [.CPT_INDEX, 0,0,0,0]
;  239          0236  2                       : BBLOCK [CPT$K_ENTRYLEN],
;  240          0237  2         SEMANTIC_LIST = NML$AB_PRMSEM [.CPT_LIST [CPT$W_PSTINDEX], 0,0,0,0]
;  241          0238  2                       : BBLOCK [PST$K_ENTRYLEN];
;  242          0239  2 !
;  243          0240  2 ! Pick up the values for comparison.
;  244          0241  2 !
;  245          0242  2     MIN = .SEMANTIC_LIST [PST$L_MINVALUE];
;  246          0243  2     MAX = .SEMANTIC_LIST [PST$L_MAXVALUE];
;  247          0244  2     STATUS = NML$_STS_SUC;
;  248          0245  2 !
```

```
;  249          0246  2 ! If the parameter is a string then get the byte count (a byte).  If the
;  250          0247  2 ! parameter is not a string then get the value of the appropriate width
;  251          0248  2 ! (byte, word, longword).
;  252          0249  2 !
;  253          0250  2     IF .SEMANTIC_LIST [PST$B_FORMAT] EQLU NML$K_STRING
;  254          0251  2     THEN
;  255          0252  2         VAL = .LEN
;  256          0253  2     ELSE
;  257          0254  2         VAL = .(.ADR)<0,.LEN*8>;
;  258          0255  2 !
;  259          0256  2 ! Check the minimum parameter value.
;  260          0257  2 !
;  261          0258  2     IF .VAL LSSU .MIN
;  262          0259  2     THEN
;  263          0260  2         STATUS = NML$_STS_PVA;
;  264          0261  2 !
;  265          0262  2 ! If the maximum value has a zero in it then don't bother to check it.
;  266          0263  2 !
;  267          0264  2     IF  .MAX NEQU 0
;  268          0265  2     AND .VAL GTRU .MAX
;  269          0266  2     THEN
;  270          0267  2         STATUS = NML$_STS_PVA;
;  271          0268  2 !
;  272          0269  2 ! If the parameter is not within range then signal a parameter value error.
;  273          0270  2 !
;  274          0271  2     IF NOT .STATUS
;  275          0272  2     THEN
;  276          0273  2         NML$ERROR_2 (NMA$C_STS_PVA, .SEMANTIC_LIST [PST$W_DATAID]);
;  277          0274  2
;  278          0275  2     RETURN NML$_STS_SUC
;  279          0276  2
;  280          0277  1     END;                                    ! End of NML$CHKPRMVAL


                                                003C 00000          .ENTRY  NML$CHKPRMVAL, Save R2,R3,R4,R5                 ; 0190
                    50      04      AC              0A   C5 00002   MULL3   #10, CPT_INDEX, R0                               ; 0235
                                    0000000000040   9F   00007   PUSHAB  NML$AB_CPTABLE[R0]                              ; 0237
                                50                  9E   3C 0000E   MOVZWL  @(SP)+, R0                                      ;
                                50                  10   C4 00011   MULL2   #16, R0                                        ;
                                51  0000000000040  9E   00014   MOVAB   NML$AB_PRMSEM[R0], R1                           ;
                                55          04      A1   D0 0001C   MOVL    4(R1), MIN                                      ; 0242
                                54          08      A1   D0 00020   MOVL    8(R1), MAX                                      ; 0243
                                53                  01   D0 00024   MOVL    #1, STATUS                                      ; 0244
                                03          02      A1   91 00027   CMPB    2(R1), #3                                       ; 0250
                                                    06   12 0002B   BNEQ    1$                                              ;
                                52          08      AC   D0 0002D   MOVL    LEN, VAL                                        ; 0252
                                                    0B   11 00031   BRB     2$                                              ;
                    50      08      AC              03   78 00033 1$: ASHL    #3, LEN, R0                                     ; 0254
          52        0C      BC      50              00   EF 00038   EXTZV   #0, R0, @ADR, VAL                               ;
                                55                  52   D1 0003E 2$: CMPL    VAL, MIN                                        ; 0258
                                                    03   1E 00041   BGEQU   3$                                              ;
                                53                  20   CE 00043   MNEGL   #32, STATUS                                     ; 0260
                                                    54   D5 00046 3$: TSTL    MAX                                             ; 0264
                                                    08   13 00048   BEQL    4$                                              ;
```

```
                                54              52  D1 0004A           CMPL    VAL, MAX                                ; 0265
                                                03  1B 0004D           BLEQU   4$                                      ;
                                53              20  CE 0004F           MNEGL   #32, STATUS                             ; 0267
                                0D              53  E8 00052 4$:       BLBS    STATUS, 5$                              ; 0271
                                7E              61  3C 00055           MOVZWL  (R1), -(SP)                             ; 0273
                                7E              10  CE 00058           MNEGL   #16, -(SP)                              ;
                     00000000G  00              02  FB 0005B           CALLS   #2, NML$ERROR_2                         ;
                                50              01  D0 00062 5$:       MOVL    #1, R0                                  ; 0275
                                                    04 00065           RET                                             ; 0277

; Routine Size:  102 bytes,    Routine Base:  $CODE$ + 008B
```

```
;  282          0278  1 %SBTTL 'NML$BLDSETQBF  Build SET/CLEAR QIO buffers'
;  283          0279  1 GLOBAL ROUTINE NML$BLDSETQBF (FUNC, DB,
;  284          0280  1                                KEYID1, KEYLEN1, KEYADR1,
;  285          0281  1                                KEYID2, KEYLEN2, KEYADR2,
;  286          0282  1                                NFBBFDSC, NFBDSC,
;  287          0283  1                                P2BFDSC,  P2DSC,
;  288          0284  1                                VALBFDSC, VALDSC)
;  289          0285  1                                    : NOVALUE =
;  290          0286  1
;  291          0287  1 !++
;  292          0288  1 ! FUNCTIONAL DESCRIPTION:
;  293          0289  1 !
;  294          0290  1 !       This routine builds a QIO buffer for parameter modifications.
;  295          0291  1 !
;  296          0292  1 ! FORMAL PARAMETERS:
;  297          0293  1 !
;  298          0294  1 !       FUNC            Control function - NFB$C_FC_SET or NFB$C_FC_CLEAR
;  299          0295  1 !       DB              Database ID
;  300          0296  1 !       KEYID1          Search key one ID
;  301          0297  1 !       KEYLEN1         Search key one length
;  302          0298  1 !       KEYADR1         Search key one address
;  303          0299  1 !       KEYID2          Search key two ID
;  304          0300  1 !       KEYLEN2         Search key two length
;  305          0301  1 !       KEYADR2         Search key two address
;  306          0302  1 !       NFBBFDSC        Descriptor of control function buffer (P1)
;  307          0303  1 !       NFBDSC          Descriptor of resulting control function buffer (P1)
;  308          0304  1 !       P2BFDSC         Descriptor of P2 buffer
;  309          0305  1 !       P2DSC           Descriptor of resulting P2 buffer
;  310          0306  1 !       VALBFDSC        Descriptor of parameter value buffer (P4)
;  311          0307  1 !       VALDSC          Descriptor of resulting parameter value data (P4)
;  312          0308  1 !
;  313          0309  1 !       NML$AW_PRM_DES  List of parameter descriptors
;  314          0310  1 !       NML$GW_PRMDESCNT Count of parameter descriptors
;  315          0311  1 !
;  316          0312  1 ! OUTPUTS:
;  317          0313  1 !
;  318          0314  1 !       Output buffers and descriptors built.
;  319          0315  1 !--
;  320          0316  1
;  321          0317  2 BEGIN
;  322          0318  2
;  323          0319  2 MAP
;  324          0320  2     NFBBFDSC : REF DESCRIPTOR,
;  325          0321  2     NFBDSC   : REF DESCRIPTOR,
;  326          0322  2     P2BFDSC  : REF DESCRIPTOR,
;  327          0323  2     P2DSC    : REF DESCRIPTOR,
;  328          0324  2     VALBFDSC : REF DESCRIPTOR,
;  329          0325  2     VALDSC   : REF DESCRIPTOR;
;  330          0326  2
;  331          0327  2 LOCAL
;  332          0328  2     MSGSIZE,
;  333          0329  2     CPT_INDEX,
;  334          0330  2     NFB : REF BBLOCK [NFB$C_LENGTH],
;  335          0331  2     VALADR,
;  336          0332  2     VALPTR,
;  337          0333  2     VALLEN,
;  338          0334  2     VALTYP,
```

```
;  339     0335  2      CPT:        REF BBLOCK [CPT$K_ENTRYLEN],
;  340     0336  2      PST:        REF BBLOCK [PST$K_ENTRYLEN];
;  341     0337  2
;  342     0338  2  BIND
;  343     0339  2      VALBUF = VALBFDSC [DSC$A_POINTER] : REF BBLOCK;
;  344     0340  2
;  345     0341  2
;  346     0342  2  !
;  347     0343  2  ! Setup NFB descriptor and header
;  348     0344  2  !
;  349     0345  2
;  350     0346  2  NFBDSC [DSC$A_POINTER] = NFB = .NFBBFDSC [DSC$A_POINTER];
;  351     0347  2  CH$FILL(0, $BYTEOFFSET(NFB$L_FLDID), .NFB);      ! Zero NFB header
;  352     0348  2  NFB [NFB$B_FCT] = .FUNC;
;  353     0349  2  NFB [NFB$B_DATABASE] = .DB;
;  354     0350  2  NFB [NFB$L_SRCH_KEY] = .KEYID1;
;  355     0351  2  NFB [NFB$L_SRCH2_KEY] = .KEYID2;
;  356     0352  2  NFB = NFB[NFB$L_FLDID];
;  357     0353  2
;  358     0354  2  !
;  359     0355  2  ! Build the P2 buffer.  The P2 buffer contains the valurs of the search
;  360     0356  2  ! keys.  The search keys are used by NETACP to identify the database entry
;  361     0357  2  ! to be updated.
;  362     0358  2  !
;  363     0359  2  SELECTONEU .KEYID1 OF
;  364     0360  2      SET
;  365     0361  2      [NFB$C_EFI_SIN]:            ! Logging filters (sink node)
;  366     0362  2          NML$BLDP2 (0, .(.KEYADR1)<0,16>, -1, 0, .P2BFDSC, .P2DSC);
;  367     0363  2
;  368     0364  2      [NFB$C_ESI_SNK]:            ! Logging sink
;  369     0365  2          NML$BLDP2 (0, .(.KEYADR1)<0,8>, -1, 0, .P2BFDSC, .P2DSC);
;  370     0366  2
;  371     0367  2      [NFB$C_NDI_ADD]:            ! Node (by address)
;  372     0368  2          NML$BLDP2 (0, .(.KEYADR1)<0,16>, -1, 0, .P2BFDSC, .P2DSC);
;  373     0369  2
;  374     0370  2      [NFB$C_XGI_GRP]:            ! Protocol Groups.
;  375     0371  2          !
;  376     0372  2          ! Protocol Group is always qualified with a DTE ID.  The DTE ID
;  377     0373  2          ! is the second search key.
;  378     0374  2          !
;  379     0375  2          NML$BLDP2 (.KEYLEN1, .KEYADR1, .KEYLEN2, .KEYADR2,
;  380     0376  2                                          .P2BFDSC, .P2DSC);
;  381     0377  2
;  382     0378  2      [NFB$C_XNI_NET]:
;  383     0379  3          BEGIN
;  384     0380  3          !
;  385     0381  3          ! If search key 1 length is zero, then the QIO is for the "active
;  386     0382  3          ! X-25 Protocol network".  Pass NML$BLDP2 the code to indicate that
;  387     0383  3          ! the key value is a word.
;  388     0384  3          !
;  389     0385  3          IF .KEYLEN1 EQL 0 THEN
;  390     0386  3              KEYLEN1 = -2;
;  391     0387  3          NML$BLDP2 (.KEYLEN1, .KEYADR1, -1, 0, .P2BFDSC, .P2DSC);
;  392     0388  2          END;
;  393     0389  2
;  394     0390  2      [OTHERWISE]:
;  395     0391  2          NML$BLDP2 (.KEYLEN1, .KEYADR1, -1, 0, .P2BFDSC, .P2DSC);
```

```
;  396          0392  2
;  397          0393  2      TES;
;  398          0394  2
;  399          0395  2  !
;  400          0396  2  ! Setup parameter value buffer descriptor
;  401          0397  2  !
;  402          0398  2
;  403          0399  2  VALDSC [DSC$A_POINTER] = VALPTR = .VALBFDSC [DSC$A_POINTER];
;  404          0400  2
;  405          0401  2  !
;  406          0402  2  ! For each entry in the parameter descriptor list, add its ACP identifier
;  407          0403  2  ! to the NFB and its value to the value buffer.
;  408          0404  2  !
;  409          0405  2  INCR I FROM 0 TO .NML$GW_PRMDESCNT - 1 DO
;  410          0406  3      BEGIN
;  411          0407  3
;  412          0408  3      CPT_INDEX = .NML$AW_PRM_DES [.I, PDB$W_INDEX];
;  413          0409  3      CPT = NML$AB_CPTABLE [.CPT_INDEX, 0,0,0,0];
;  414          0410  3      PST = NML$AB_PRMSEM [.CPT [CPT$W_PSTINDEX], 0,0,0,0];
;  415          0411  3      VALLEN =  .NML$AW_PRM_DES [.I, PDB$W_COUNT];
;  416          0412  3      VALADR = .NML$AW_PRM_DES [.I, PDB$A_POINTER];
;  417          0413  3
;  418          0414  4      IF (.VALPTR + .VALLEN + 2 LSSU
;  419          0415  3          .VALBFDSC [DSC$A_POINTER] + .VALBFDSC [DSC$W_LENGTH]) AND
;  420          0416  4          (.NFB + 4 LSSU
;  421          0417  4          .NFBBFDSC [DSC$A_POINTER] + .NFBBFDSC [DSC$W_LENGTH])
;  422          0418  4
;  423          0419  3      THEN
;  424          0420  4          BEGIN
;  425          0421  4          NFB[0,0,32,0] = .PST [PST$L_NFBID];
;  426          0422  4          NFB = .NFB + 4;
;  427          0423  4
;  428          0424  4          IF .VALLEN  GTRU 0
;  429          0425  4          THEN
;  430          0426  5              BEGIN
;  431          0427  6              VALTYP = .(PST [PST$L_NFBID])
;  432          0428  5                           <$BITPOSITION (NFB$V_TYP),
;  433          0429  5                            $FIELDWIDTH (NFB$V_TYP)>;
;  434          0430  5              IF .VALTYP FQLU NFB$C_TYP_STR
;  435          0431  5              THEN
;  436          0432  6                  BEGIN
;  437          0433  6
;  438          0434  6                  (.VALPTR)<0,16> = .VALLEN;       ! Set count
;  439          0435  6                  VALPTR = .VALPTR + 2;
;  440          0436  6                  VALPTR = CH$MOVE (.VALLEN, .VALADR, .VALPTR);
;  441          0437  6
;  442          0438  6                  END
;  443          0439  5              ELSE
;  444          0440  6                  BEGIN
;  445          0441  6
;  446          0442  6                  (.VALPTR)<0,32> = .(.VALADR)<0,.VALLEN*8>;
;  447          0443  6                  VALPTR = .VALPTR + 4;    ! Increment data pointer
;  448          0444  6
;  449          0445  5                  END;
;  450          0446  4              END;
;  451          0447  4          END
;  452          0448  3      ELSE
```

```
;  453          0449  4          BEGIN
;  454          0450  4
;  455          0451  4          NML$AB_MSGBLOCK [MSB$L_FLAGS] = MSB$M_MSG_FLD; ! Set message text flag
;  456          0452  4          NML$AB_MSGBLOCK [MSB$B_CODE] = NMA$C_STS_MPR;
;  457          0453  4          NML$AB_MSGBLOCK [MSB$L_TEXT] = NML$_QIOBFOVF;
;  458          0454  4          NML$BLD_REPLY (NML$AB_MSGBLOCK, MSGSIZE); ! Build message
;  459          0455  4          $SIGNAL_MSG (NML$AB_SNDBUFFER, .MSGSIZE); ! Signal it
;  460          0456  4
;  461          0457  3          END;
;  462          0458  3
;  463          0459  2      END;
;  464          0460  2
;  465          0461  2 NFB [0,0,32,0] = NFB$C_ENDOFLIST;
;  466          0462  2 NFBDSC [DSC$W_LENGTH] = .NFB - .NFBDSC[DSC$A_POINTER] + 4;
;  467          0463  2
;  468          0464  2 VALDSC [DSC$W_LENGTH] = .VALPTR - .VALDSC[DSC$A_POINTER];
;  469          0465  2
;  470          0466  1 END;                                ! End of NML$BLDSETQBF


                                                  0FFC 00000          .ENTRY  NML$BLDSETQBF, Save R2,R3,R4,R5,R6,R7,R8,-      ; 0279
                                                                               R9,R10,R11
                                        5E              14  C2 00002          SUBL2   #20, SP                                 ;
                                                  34    AC  DD 00005          PUSHL   VALBFDSC                                ; 0339
                                  50    24        AC      04  C1 00008          ADDL3   #4, NFBBFDSC, R0                        ; 0346
                                        56              60  D0 0000D          MOVL    (R0), NFB                               ;
                                  51    28        AC      04  C1 00010          ADDL3   #4, NFBDSC, R1                          ;
                                        61              56  D0 00015          MOVL    NFB, (R1)                               ;
                            10    00                6E    00  2C 00018          MOVC5   #0, (SP), #0, #16, (NFB)                ; 0347
                                                          66      0001D
                                        86        04    AC  90 0001E          MOVB    FUNC, (NFB)+                            ; 0348
                                  01    A6        08    AC  90 00022          MOVB    DB, 1(NFB)                              ; 0349
                                        50        0C    AC  D0 00027          MOVL    KEYID1, R0                              ; 0350
                                  03    A6              50  D0 0002B          MOVL    R0, 3(NFB)                              ;
                                  07    A6        18    AC  D0 0002F          MOVL    KEYID2, 7(NFB)                          ; 0351
                                        56              0F  C0 00034          ADDL2   #15, NFB                                ; 0352
                                06010010  8F              50  D1 00037          CMPL    R0, #100728848                          ; 0361
                                                          21  13 0003E          BEQL    2$                                      ;
                                07010010  8F              50  D1 00040          CMPL    R0, #117506064                          ; 0364
                                                          0F  12 00047          BNEQ    1$                                      ;
                                        7E        2C    AC  7D 00049          MOVQ    P2BFDSC, -(SP)                          ; 0365
                                                          7E  D4 0004D          CLRL    -(SP)                                   ;
                                        7E                01  CE 0004F          MNEGL   #1, -(SP)                               ;
                                        7E        14    BC  9A 00052          MOVZBL  @KEYADR1, -(SP)                         ;
                                                          16  11 00056          BRB     3$                                      ;
                                02010012  8F              50  D1 00058  1$:     CMPL    R0, #33619986                           ; 0367
                                                          11  12 0005F          BNEQ    4$                                      ;
                                        7E        2C    AC  7D 00061  2$:     MOVQ    P2BFDSC, -(SP)                          ; 0368
                                                          7E  D4 00065          CLRL    -(SP)                                   ;
                                        7E                01  CE 00067          MNEGL   #1, -(SP)                               ;
                                        7E        14    BC  3C 0006A          MOVZWL  @KEYADR1, -(SP)                         ;
                                                          7E  D4 0006E  3$:     CLRL    -(SP)                                   ;
                                                          32  11 00070          BRB     8$                                      ;
                                0A020041  8F              50  D1 00072  4$:     CMPL    R0, #167903297                          ; 0370
```

```
                                                  0A  12 00079        BNEQ    5$                                   ;
                                   7E        2C   AC  7D 0007B        MOVQ    P2BFDSC, -(SP)                       ; 0376
                                   7E        1C   AC  7D 0007F        MOVQ    KEYLEN2, -(SP)                       ; 0375
                                                  1B  11 00083        BRB     7$                                   ;
                        09020041   8F             50  D1 00085 5$:    CMPL    R0, #151126081                       ; 0378
                                                  09  12 0008C        BNEQ    6$                                   ;
                                             10   AC  D5 0008E        TSTL    KEYLEN1                              ; 0385
                                                  04  12 00091        BNEQ    6$                                   ;
                              10   AC             02  CE 00093        MNEGL   #2, KEYLEN1                          ; 0386
                                   7E        2C   AC  7D 00097 6$:    MOVQ    P2BFDSC, -(SP)                       ; 0391
                                                  7E  D4 0009B        CLRL    -(SP)                                ;
                                   7E             01  CE 0009D        MNEGL   #1, -(SP)                            ;
                                   7E        10   AC  7D 000A0 7$:    MOVQ    KEYLEN1, -(SP)                       ;
                        00000000G  00             06  FB 000A4 8$:    CALLS   #6, NML$BLDP2                        ;
                                   5B        38   AC  D0 000AB        MOVL    VALDSC, R11                          ; 0399
                        50         6E             04  C1 000AF        ADDL3   #4, (SP), R0                         ;
                                   53             60  D0 000B3        MOVL    (R0), VALPTR                         ;
                              04   AB             53  D0 000B6        MOVL    VALPTR, 4(R11)                       ;
                              0C   AE  00000000G  00  3C 000BA        MOVZWL  NML$GW_PRMDESCNT, 12(SP)             ; 0405
                                   5A             01  CE 000C2        MNEGL   #1, I                                ;
                                                0008  31 000C5        BRW     12$                                  ;
                                       00000000G004A  7F 000C8 9$:    PUSHAQ  NML$AW_PRM_DES[I]                    ; 0408
                                   59             9E  3C 000CF        MOVZWL  @(SP)+, CPT_INDEX                    ;
                        50         59             0A  C5 000D2        MULL3   #10, CPT_INDEX, R0                   ; 0409
                                   58  00000000G0040  9E 000D6        MOVAB   NML$AB_CPTABLE[R0], CPT              ;
                                   50             68  3C 000DE        MOVZWL  (CPT), R0                            ; 0410
                                   50             10  C4 000E1        MULL2   #16, R0                              ;
                              04   AE  00000000G0040  9E 000E4        MOVAB   NML$AB_PRMSEM[R0], PST               ;
                                       00000000G004A  7F 000ED        PUSHAQ  NML$AW_PRM_DES+2[I]                  ; 0411
                                   57             9E  3C 000F4        MOVZWL  @(SP)+, VALLEN                       ;
                                       00000000G004A  7F 000F7        PUSHAQ  NML$AW_PRM_DES+4[I]                  ; 0412
                              08   AE             9E  D0 000FE        MOVL    @(SP)+, VALADR                       ;
                                   51        02 A743  9E 00102        MOVAB   2(VALLEN)[VALPTR], R1                ; 0414
                                   52        34   BC  3C 00107        MOVZWL  @VALBFDSC, R2                        ; 0415
                        54         6E             04  C1 0010B        ADDL3   #4, (SP), R4                         ;
                        50         64             52  C1 0010F        ADDL3   R2, (R4), R0                         ;
                                   50             51  D1 00113        CMPL    R1, R0                               ;
                                                  49  1E 00116        BGEQU   11$                                  ;
                                   51        04   A6  9E 00118        MOVAB   4(R6), R1                            ; 0416
                                   52        24   BC  3C 0011C        MOVZWL  @NFBBFDSC, R2                        ; 0417
                        54    24   AC             04  C1 00120        ADDL3   #4, NFBBFDSC, R4                     ;
                        50         64             52  C1 00125        ADDL3   R2, (R4), R0                         ;
                                   50             51  D1 00129        CMPL    R1, R0                               ;
                                                  33  1E 0012C        BGEQU   11$                                  ;
                        50    04   AE             0C  C1 0012E        ADDL3   #12, PST, R0                         ; 0421
                                   86             60  D0 00133        MOVL    (R0), (NFB)+                         ;
                                                  57  D5 00136        TSTL    VALLEN                               ; 0424
                                                  66  13 00138        BEQL    12$                                  ;
                        50    04   AE             0E  C1 0013A        ADDL3   #14, PST, R0                         ; 0427
    10   AE             60         02             00  EF 0013F        EXTZV   #0, #2, (R0), VALTYP                 ;
                                   02        10   AE  D1 00145        CMPL    VALTYP, #2                           ; 0430
                                                  0A  12 00149        BNEQ    10$                                  ;
                                   83             57  B0 0014B        MOVW    VALLEN, (VALPTR)+                    ; 0434
                        63    08   BE             57  28 0014E        MOVC3   VALLEN, @VALADR, (VALPTR)            ; 0436
                                                  4B  11 00153        BRB     12$                                  ; 0430
                        50         57             03  78 00155 10$:   ASHL    #3, VALLEN. R0                       ; 0442
         83        08   BE         50             00  EF 00159        EXTZV   #0, R0, @VALADR, (VALPTR)+           ;
```

```
                                         3F  11 0015F         BRB      12$                                          ; 0424
                000000000G  00           04  D0 00161 11$:    MOVL     #4, NML$AB_MSGBLOCK                          ; 0451
                000000000G  00           05  8E 00168         MNEGB    #5, NML$AB_MSGBLOCK+4                        ; 0452
                000000000G  00 000000000G 8F  D0 0016F        MOVL     #NML$_QIOBFOVF, NML$AB_MSGBLOCK+12           ; 0453
                                    14   AE  9F 0017A         PUSHAB   MSGSIZE                                      ; 0454
                               000000000G 00  9F 0017D        PUSHAB   NML$AB_MSGBLOCK                              ;
                000000000G  00           02  FB 00183         CALLS    #2, NML$BLD_REPLY                            ;
                                    14   AE  DD 0018A         PUSHL    MSGSIZE                                      ; 0455
                               000000000G 00  9F 0018D        PUSHAB   NML$AB_SNDBUFFER                             ;
                                01F90000  8F  DD 00193        PUSHL    #33095680                                    ;
                000000000G  00           03  FB 00199         CALLS    #3, LIB$SIGNAL                               ;
          02              5A        0C   AE  F2 001A0 12$:    AOBLSS   12(SP), I, 13$                               ; 0405
                                         03  11 001A5         BRB      14$                                          ;
                                       FF1E  31 001A7 13$:    BRW      9$                                           ;
                                         66  D4 001AA 14$:    CLRL     (NFB)                                        ; 0461
          50          28  AC             04  C1 001AC         ADDL3    #4, NFBDSC, R0                               ; 0462
                              56         60  C2 001B1         SUBL2    (R0), R6                                     ;
      28  BC                  56         04  A1 001B4         ADDW3    #4, R6, @NFBDSC                              ;
          6B                  53    04   AB  A3 001B9         SUBW3    4(R11), VALPTR, (R11)                        ; 0464
                                             04 001BE         RET                                                   ; 0466

; Routine Size:  447 bytes,    Routine Base:  $CODE$ + 00F1

;  471          0467  1
```

```
;  473     0468  1 %SBTTL 'NML$ADD_FIELDS  Add parameter fields to record'
;  474     0469  1 GLOBAL ROUTINE NML$ADD_FIELDS (BUFSIZE, RTNDSC) =
;  475     0470  1
;  476     0471  1 !++
;  477     0472  1 ! FUNCTIONAL DESCRIPTION:
;  478     0473  1 !
;  479     0474  1 !       This routine adds fields to a permanent data base record.
;  480     0475  1 !
;  481     0476  1 ! FORMAL PARAMETERS:
;  482     0477  1 !
;  483     0478  1 !       BUFSIZE Maximum size of the record buffer.
;  484     0479  1 !       RTNDSC  Address of the current record descriptor.
;  485     0480  1 !
;  486     0481  1 ! IMPLICIT INPUTS:
;  487     0482  1 !
;  488     0483  1 !       NONE
;  489     0484  1 !
;  490     0485  1 ! IMPLICIT OUTPUTS:
;  491     0486  1 !
;  492     0487  1 !       The record descriptor pointed to by RTNDSC is updated to include
;  493     0488  1 !       any fields added to the record.
;  494     0489  1 !
;  495     0490  1 ! ROUTINE VALUE:
;  496     0491  1 ! COMPLETION CODES:
;  497     0492  1 !
;  498     0493  1 !       NONE
;  499     0494  1 !
;  500     0495  1 ! SIDE EFFECTS:
;  501     0496  1 !
;  502     0497  1 !       NONE
;  503     0498  1 !
;  504     0499  1 !--
;  505     0500  1
;  506     0501  2     BEGIN
;  507     0502  2
;  508     0503  2     LOCAL
;  509     0504  2         CPT_INDEX,                      ! Change parameter table index
;  510     0505  2         SEM_INDEX,                      ! Semantic table index
;  511     0506  2         FLDLEN,                         ! Field length
;  512     0507  2         FLDADR,                         ! Field address
;  513     0508  2         MSGSIZE,                        ! Message size
;  514     0509  2         ROUTINE_ADR,                    ! Temporary routine address
;  515     0510  2         STATUS;
;  516     0511  2
;  517     0512  2     INCR I FROM 0 TO .NML$GW_PRMDESCNT - 1 DO
;  518     0513  3         BEGIN
;  519     0514  3
;  520     0515  3         FLDLEN = .NML$AW_PRM_DES [.I, PDB$W_COUNT];
;  521     0516  3         FLDADR = .NML$AW_PRM_DES [.I, PDB$A_POINTER];
;  522     0517  3
;  523     0518  3         CPT_INDEX = .NML$AW_PRM_DES [.I, PDB$W_INDEX];
;  524     0519  3         ROUTINE_ADR = .NML$AB_CPTABLE [.CPT_INDEX, CPT$A_DEFINE_RTN];
;  525     0520  3         SEM_INDEX = .NML$AB_CPTABLE [.CPT_INDEX, CPT$W_PSTINDEX];
;  526     0521  3
;  527     0522  4         IF NOT (STATUS =
;  528     0523  4                 (.ROUTINE_ADR) (NML$AB_PRMSEM [.SEM_INDEX, 0,0,0,0],
;  529     0524  4                                 .BUFSIZE,
```

```
;  530          0525  4                                   .FLDLEN,
;  531          0526  4                                   .FLDADR,
;  532          0527  4                                   .RTNDSC))
;  533          0528  3          THEN
;  534          0529  3              RETURN .STATUS
;  535          0530  3
;  536          0531  2          END;
;  537          0532  2
;  538          0533  2      RETURN NML$_STS_SUC
;  539          0534  2
;  540          0535  1      END;                                ! End of NML$ADD_FIELDS


                                        03FC 00000          .ENTRY  NML$ADD_FIELDS, Save R2,R3,R4,R5,R6,R7,R8,- ; 0469
                                                                    R9
                                59 00000000G   00    9E 00002       MOVAB   NML$AW_PRM_DES+2, R9                  ;
                                55 00000000G   00    3C 00009       MOVZWL  NML$GW_PRMDESCNT, R5                  ; 0512
                                54                01 CE 00010       MNEGL   #1, I                                 ; 0524
                                                  47 11 00013       BRB     2$                                    ;
                                                6944 7F 00015 1$:   PUSHAQ  NML$AW_PRM_DES+2[I]                   ; 0515
                                58                9E 3C 00018       MOVZWL  @(SP)+, FLDLEN                        ;
                                             02 A944 7F 0001B       PUSHAQ  NML$AW_PRM_DES+4[I]                   ; 0516
                                57                9E D0 0001F       MOVL    @(SP)+, FLDADR                        ;
                                             FE A944 7F 00022       PUSHAQ  NML$AW_PRM_DES[I]                     ; 0518
                                53                9E 3C 00026       MOVZWL  @(SP)+, CPT_INDEX                     ;
                          51    53                0A C5 00029       MULL3   #10, CPT_INDEX, R1                    ; 0519
                                   00000000G0041  9F 0002D       PUSHAB  NML$AB_CPTABLE+2[R1]                  ;
                                56                9E D0 00034       MOVL    @(SP)+, ROUTINE_ADR                   ;
                                   00000000G0041  9F 00037       PUSHAB  NML$AB_CPTABLE[R1]                    ; 0520
                                52                9E 3C 0003E       MOVZWL  @(SP)+, SEM_INDEX                     ;
                                             08   AC DD 00041       PUSHL   RTNDSC                                ; 0527
                                                  57 DD 00044       PUSHL   FLDADR                                ; 0526
                                                  58 DD 00046       PUSHL   FLDLEN                                ; 0525
                                             04   AC DD 00048       PUSHL   BUFSIZE                               ; 0524
                          51    52                04 78 0004B       ASHL    #4, SEM_INDEX, R1                     ; 0523
                                   00000000G0041  9F 0004F       PUSHAB  NML$AB_PRMSEM[R1]                     ;
                                66                05 FB 00056       CALLS   #5, (ROUTINE_ADR)                     ;
                                07                50 E9 00059       BLBC    STATUS, 3$                            ;
                          B5    54                55 F2 0005C 2$:   AOBLSS  R5, I, 1$                            ; 0522
                                50                01 D0 00060       MOVL    #1, R0                                ; 0533
                                                     04 00063 3$:   RET                                           ; 0535

; Routine Size:  100 bytes,    Routine Base:  $CODE$ + 02B0
```

```
;  542     0536  1 %SBTTL 'NML$DEL_FIELDS  Delete parameter fields from record'
;  543     0537  1 GLOBAL ROUTINE NML$DEL_FIELDS (RTNDSC) =
;  544     0538  1
;  545     0539  1 !++
;  546     0540  1 ! FUNCTIONAL DESCRIPTION:
;  547     0541  1 !
;  548     0542  1 !       This routine deletes the entire list of parameters in the parameter
;  549     0543  1 !       descriptor buffer from the specified record buffer.
;  550     0544  1 !
;  551     0545  1 ! FORMAL PARAMETERS:
;  552     0546  1 !
;  553     0547  1 !       RTNDSC contains the address of the current record descriptor.
;  554     0548  1 !
;  555     0549  1 ! IMPLICIT INPUTS:
;  556     0550  1 !
;  557     0551  1 !       NML$GW_PRMDESCNT contains the number of parameter descriptors.
;  558     0552  1 !       NML$AW_PRM_DES is a list of parameter descriptors.
;  559     0553  1 !       NML$AB_PRMSEM is the parameter semantic table.
;  560     0554  1 !
;  561     0555  1 ! IMPLICIT OUTPUTS:
;  562     0556  1 !
;  563     0557  1 !       The record descriptor pointed to by RTNDSC is updated to reflect
;  564     0558  1 !       any fields deleted from the record.
;  565     0559  1 !
;  566     0560  1 ! ROUTINE VALUE:
;  567     0561  1 ! COMPLETION CODES:
;  568     0562  1 !
;  569     0563  1 !       Always returns success (NML$_STS_SUC).
;  570     0564  1 !
;  571     0565  1 ! SIDE EFFECTS:
;  572     0566  1 !
;  573     0567  1 !       NONE
;  574     0568  1 !
;  575     0569  1 !--
;  576     0570  1
;  577     0571  2     BEGIN
;  578     0572  2
;  579     0573  2     LOCAL
;  580     0574  2         CPT_INDEX,
;  581     0575  2         SEM_INDEX,
;  582     0576  2         ROUTINE_ADR;
;  583     0577  2
;  584     0578  2     INCR I FROM 0 TO .NML$GW_PRMDESCNT - 1 DO
;  585     0579  3         BEGIN
;  586     0580  3
;  587     0581  3         CPT_INDEX = .NML$AW_PRM_DES [.I, PDB$W_INDEX];
;  588     0582  3         ROUTINE_ADR = .NML$AB_CPTABLE [.CPT_INDEX, CPT$A_PURGE_RTN];
;  589     0583  3         SEM_INDEX = .NML$AB_CPTABLE [.CPT_INDEX, CPT$W_PSTINDEX];
;  590     0584  3
;  591     0585  3         (.ROUTINE_ADR) (.RTNDSC,
;  592     0586  3                         NML$AB_PRMSEM [.SEM_INDEX, 0,0,0,0]);
;  593     0587  3
;  594     0588  2         END;
;  595     0589  2
;  596     0590  2     RETURN NML$_STS_SUC
;  597     0591  2
;  598     0592  1     END;                                    ! End of NML$DEL_FIELDS
```

```
                                                007C 00000          .ENTRY  NML$DEL_FIELDS, Save R2,R3,R4,R5,R6          ; 0537
                                     55 00000000G  00   3C 00002          MOVZWL  NML$GW_PRMDESCNT, R5                         ; 0578
                                     54            01   CE 00009          MNEGL   #1, I                                        ; 0585
                                                   33   11 0000C          BRB     2$                                           ;
                                        00000000G0044   7F 0000E 1$:       PUSHAQ  NML$AW_PRM_DES[I]                            ; 0581
                                     53            9E   3C 00015          MOVZWL  @(SP)+, CPT_INDEX                            ;
                                50   53            0A   C5 00018          MULL3   #10, CPT_INDEX, R0                           ; 0582
                                        00000000G0040   9F 0001C          PUSHAB  NML$AB_CPTABLE+6[R0]                         ;
                                     56            9E   D0 00023          MOVL    @(SP)+, ROUTINE_ADR                          ;
                                        00000000G0040   9F 00026          PUSHAB  NML$AB_CPTABLE[R0]                           ; 0583
                                     52            9E   3C 0002D          MOVZWL  @(SP)+, SEM_INDEX                            ;
                                50   52            04   78 00030          ASHL    #4, SEM_INDEX, R0                            ; 0586
                                        00000000G0040   9F 00034          PUSHAB  NML$AB_PRMSEM[R0]                            ;
                                              04    AC   DD 0003B          PUSHL   RTNDSC                                       ;
                                     66            02   FB 0003E          CALLS   #2, (ROUTINE_ADR)                            ;
                                C9   54            55   F2 00041 2$:       AOBLSS  R5, I, 1$                                    ; 0578
                                     50            01   D0 00045          MOVL    #1, R0                                       ; 0590
                                                        04 00048          RET                                                  ; 0592

; Routine Size:  73 bytes,    Routine Base:  $CODE$ + 0314
```

```
;  600     0593  1 %SBTTL 'NML$READPARLIST  Show parameters from buffer'
;  601     0594  1 GLOBAL ROUTINE NML$READPARLIST (BUFDSC, MSGSIZE, TABDSC, DATDSC) =
;  602     0595  1
;  603     0596  1 !++
;  604     0597  1 ! FUNCTIONAL DESCRIPTION:
;  605     0598  1 !
;  606     0599  1 !       This routine builds a message from the list of parameters specified.
;  607     0600  1 !
;  608     0601  1 ! FORMAL PARAMETERS:
;  609     0602  1 !
;  610     0603  1 !       BUFDSC          Address of message buffer descriptor.
;  611     0604  1 !       MSGSIZE         Address of longword to contain resulting message size.
;  612     0605  1 !       TABDSC          Address of parameter table descriptor.
;  613     0606  1 !       DATDSC          Address of data buffer descriptor.
;  614     0607  1 !
;  615     0608  1 ! IMPLICIT INPUTS:
;  616     0609  1 !
;  617     0610  1 !       NONE
;  618     0611  1 !
;  619     0612  1 ! IMPLICIT OUTPUTS:
;  620     0613  1 !
;  621     0614  1 !       NONE
;  622     0615  1 !
;  623     0616  1 ! ROUTINE VALUE:
;  624     0617  1 ! COMPLETION CODES:
;  625     0618  1 !
;  626     0619  1 !       Always returns success (NML$_STS_SUC).
;  627     0620  1 !
;  628     0621  1 ! SIDE EFFECTS:
;  629     0622  1 !
;  630     0623  1 !       NONE
;  631     0624  1 !
;  632     0625  1 !--
;  633     0626  1
;  634     0627  2     BEGIN
;  635     0628  2
;  636     0629  2     MAP
;  637     0630  2         TABDSC : REF DESCRIPTOR;
;  638     0631  2
;  639     0632  2     LOCAL
;  640     0633  2         INDEX;
;  641     0634  2
;  642     0635  2     BIND
;  643     0636  2         TABLE = TABDSC [DSC$A_POINTER] : REF BBLOCKVECTOR [, 6];
;  644     0637  2
;  645     0638  2 !
;  646     0639  2 ! If table address is null then the specified information type is not
;  647     0640  2 ! applicable to this entity.
;  648     0641  2 !
;  649     0642  2     IF .TABLE EQLA 0
;  650     0643  2     THEN
;  651     0644  2         NML$ERROR_1 (NMA$C_STS_FOP);
;  652     0645  2
;  653     0646  2     INCR I FROM 0 TO .TABDSC [DSC$W_LENGTH] - 1 DO
;  654     0647  3         BEGIN
;  655     0648  3
;  656     0649  3         INDEX = .TABLE [.I, 0,0,16,0];  ! Get table index
```

```
;  657          0650  3
;  658          0651  3          (.TABLE [.I, 2,0,32,0]) (NML$AB_PRMSEM [.INDEX, 0,0,0,0],
;  659          0652  3                                                  .BUFDSC,
;  660          0653  3                                                  .MSGSIZE,
;  661          0654  3                                                  .DATDSC);
;  662          0655  3
;  663          0656  2          END;
;  664          0657  2
;  665          0658  2      RETURN NML$_STS_SUC
;  666          0659  2
;  667          0660  1      END;                                    ! End of NML$READPARLIST


                                                 003C 00000          .ENTRY  NML$READPARLIST, Save R2,R3,R4,R5        ; 0594
                    54        0C    AC               04   C1 00002          ADDL3   #4, TABDSC, R4                   ; 0636
                                                     64   D5 00007          TSTL    (R4)                             ; 0642
                                                     0A   12 00009          BNEQ    1$                               ;
                          7E                         0D   CE 0000B          MNEGL   #13, -(SP)                       ; 0644
                 00000000G  00                       01   FB 0000E          CALLS   #1, NML$ERROR_1                  ;
                          55        0C               BC   3C 00015 1$:      MOVZWL  @TABDSC, R5                      ; 0646
                          52                         01   CE 00019          MNEGL   #1, I                            ; 0652
                                                     20   11 0001C          BRB     3$                               ;
                    51    52                         06   C5 0001E 2$:      MULL3   #6, I, R1                        ; 0649
                          51                         64   C0 00022          ADDL2   (R4), R1                         ;
                          53                         61   3C 00025          MOVZWL  (R1), INDEX                      ;
                                    10    AC         DD   00028          PUSHL   DATDSC                           ; 0654
                          7E        04    AC         7D   0002B          MOVQ    BUFDSC, -(SP)                    ; 0652
                    50    53                         04   78 0002F          ASHL    #4, INDEX, R0                    ; 0651
                                 00000000G0040         9F   00033          PUSHAB  NML$AB_PRMSEM[R0]                ;
                       02   B1                        04   FB 0003A          CALLS   #4, @2(R1)                       ;
                    DC    52                         55   F2 0003E 3$:      AOBLSS  R5, I, 2$                        ; 0646
                          50                         01   D0 00042          MOVL    #1, R0                           ; 0658
                                                          04 00045          RET                                      ; 0660

; Routine Size:  70 bytes,    Routine Base:  $CODE$ + 035D
```

```
;  669          0661  1 %SBTTL 'NML$SHOWPARLIST  Show parameters from QIO buffer'
;  670          0662  1 GLOBAL ROUTINE NML$SHOWPARLIST (BUFDSC, MSGSIZE, TABDSC, DATDSC, DATPTR) =
;  671          0663  1
;  672          0664  1 !++
;  673          0665  1 ! FUNCTIONAL DESCRIPTION:
;  674          0666  1 !
;  675          0667  1 !       This routine builds a message from the list of parameters specified.
;  676          0668  1 !
;  677          0669  1 ! FORMAL PARAMETERS:
;  678          0670  1 !
;  679          0671  1 !       BUFDSC          Address of message buffer descriptor.
;  680          0672  1 !       MSGSIZE         Address of longword to contain resulting message size.
;  681          0673  1 !       TABDSC          Address of parameter table descriptor.
;  682          0674  1 !       DATDSC          Address of data buffer descriptor.
;  683          0675  1 !       DATPTR          Address of data buffer pointer.
;  684          0676  1 !
;  685          0677  1 ! IMPLICIT INPUTS:
;  686          0678  1 !
;  687          0679  1 !       NONE
;  688          0680  1 !
;  689          0681  1 ! IMPLICIT OUTPUTS:
;  690          0682  1 !
;  691          0683  1 !       NONE
;  692          0684  1 !
;  693          0685  1 ! ROUTINE VALUE:
;  694          0686  1 ! COMPLETION CODES:
;  695          0687  1 !
;  696          0688  1 !       Always returns success (NML$_STS_SUC).
;  697          0689  1 !
;  698          0690  1 ! SIDE EFFECTS:
;  699          0691  1 !
;  700          0692  1 !       NONE
;  701          0693  1 !
;  702          0694  1 !--
;  703          0695  1
;  704          0696  2     BEGIN
;  705          0697  2
;  706          0698  2     MAP
;  707          0699  2         TABDSC : REF DESCRIPTOR;
;  708          0700  2
;  709          0701  2     LOCAL
;  710          0702  2         INDEX;
;  711          0703  2
;  712          0704  2     BIND
;  713          0705  2         TABLE = TABDSC [DSC$A_POINTER] : REF BBLOCKVECTOR [, 6];
;  714          0706  2
;  715          0707  2     INCR I FROM 0 TO .TABDSC [DSC$W_LENGTH] - 1 DO
;  716          0708  3         BEGIN
;  717          0709  3
;  718          0710  3         INDEX = .TABLE [.I, 0,0,16,0];  ! Get table index
;  719          0711  3
;  720          0712  3         (.TABLE [.I, 2,0,32,0]) (NML$AB_PRMSEM [.INDEX, 0,0,0,0],
;  721          0713  3                                  .BUFDSC,
;  722          0714  3                                  .MSGSIZE,
;  723          0715  3                                  .DATDSC,
;  724          0716  3                                  .DATPTR);
;  725          0717  3
```

```
;  726          0718  2          END;
;  727          0719  2
;  728          0720  2      RETURN NML$_STS_SUC
;  729          0721  2
;  730          0722  1      END;                                    ! End of NML$SHOWPARLIST


                                                003C 00000          .ENTRY  NML$SHOWPARLIST, Save R2,R3,R4,R5       ; 0662
                    55      0C      AC              04   C1 00002          ADDL3   #4, TABDSC, R5                  ; 0705
                                    54      0C      BC   3C 00007          MOVZWL  @TABDSC, R4                     ; 0707
                                    52              01   CE 0000B          MNEGL   #1, I                           ; 0713
                                                    21   11 0000E          BRB     2$                              ;
                    51              52              06   C5 00010 1$:      MULL3   #6, I, R1                       ; 0710
                                    51              65   C0 00014          ADDL2   (R5), R1                        ;
                                    53              61   3C 00017          MOVZWL  (R1), INDEX                     ;
                                    7E      10      AC   7D 0001A          MOVQ    DATDSC, -(SP)                   ; 0715
                                    7E      04      AC   7D 0001E          MOVQ    BUFDSC, -(SP)                   ; 0713
                    50              53              04   78 00022          ASHL    #4, INDEX, R0                   ; 0712
                                            00000000G00040  9F 00026          PUSHAB  NML$AB_PRMSEM[R0]               ;
                                02  B1                  05   FB 0002D          CALLS   #5, @2(R1)                      ;
                    DB              52                  54   F2 00031 2$:      AOBLSS  R4, I, 1$                       ; 0707
                                    50                  01   D0 00035          MOVL    #1, R0                          ; 0720
                                                        04   00038          RET                                     ; 0722

; Routine Size:  57 bytes,    Routine Base:  $CODE$ + 03A3
```

```
;  732     0723  1 %SBTTL 'NML$BLDALLDES  Build parameter descriptors from record'
;  733     0724  1 GLOBAL ROUTINE NML$BLDALLDES (RECDSC, TABDSC) =
;  734     0725  1
;  735     0726  1 !++
;  736     0727  1 ! FUNCTIONAL DESCRIPTION:
;  737     0728  1 !
;  738     0729  1 !       This routine is used by SET ALL functions to build parameter
;  739     0730  1 !       descriptors from a permanent data base record.
;  740     0731  1 !
;  741     0732  1 ! FORMAL PARAMETERS:
;  742     0733  1 !
;  743     0734  1 !       RECDSC          Address of the current record descriptor.
;  744     0735  1 !       TABDSC          Address of parameter table descriptor.
;  745     0736  1 !
;  746     0737  1 ! IMPLICIT INPUTS:
;  747     0738  1 !
;  748     0739  1 !       NML$AB_PRMSEM is the parameter semantic table.
;  749     0740  1 !
;  750     0741  1 ! IMPLICIT OUTPUTS:
;  751     0742  1 !
;  752     0743  1 !       NONE
;  753     0744  1 !
;  754     0745  1 ! ROUTINE VALUE:
;  755     0746  1 ! COMPLETION CODES:
;  756     0747  1 !
;  757     0748  1 !       Always returns success (NML$_STS_SUC).
;  758     0749  1 !
;  759     0750  1 ! SIDE EFFECTS:
;  760     0751  1 !
;  761     0752  1 !       NONE
;  762     0753  1 !
;  763     0754  1 !--
;  764     0755  1
;  765     0756  2     BEGIN
;  766     0757  2
;  767     0758  2     MAP
;  768     0759  2         RECDSC : REF DESCRIPTOR,
;  769     0760  2         TABDSC : REF DESCRIPTOR;
;  770     0761  2
;  771     0762  2     LOCAL
;  772     0763  2         FLDADR,
;  773     0764  2         FLDSIZE,
;  774     0765  2         INDEX;
;  775     0766  2
;  776     0767  2     BIND
;  777     0768  2         TABLE = TABDSC [DSC$A_POINTER] : REF BLOCK;
;  778     0769  2
;  779     0770  2     NML$GW_PRMDESCNT = 0;                   ! Reset parameter descriptor count
;  780     0771  2
;  781     0772  2     INCR I FROM 0 TO .TABDSC [DSC$W_LENGTH] - 1 DO
;  782     0773  3         BEGIN
;  783     0774  3
;  784     0775  3         FLDADR = 0;
;  785     0776  3
;  786     0777  3         IF NMA$SEARCHFLD (.RECDSC,
;  787     0778  3                           .TABLE [.I,0,16,0],
;  788     0779  3                          FLDSIZE,
```

```
;  789          0780  3                                FLDADR)
;  790          0781  3          THEN
;  791          0782  4              BEGIN
;  792          0783  4
;  793          0784  4              INDEX = .TABLE [.I,16,16,0];
;  794          0785  4
;  795          0786  4              NML$SAVEPARAM (.INDEX,
;  796          0787  4                             .FLDSIZE,
;  797          0788  4                             .FLDADR);
;  798          0789  3              END;
;  799          0790  2          END;
;  800          0791  2
;  801          0792  2      RETURN NML$_STS_SUC
;  802          0793  2
;  803          0794  1      END;                                ! End of NML$BLDALLDES


                                        003C 00000          .ENTRY  NML$BLDALLDES, Save R2,R3,R4,R5                 ; 0724
                                5E        08  C2 00002          SUBL2   #8, SP                                          ;
                          53    08  AC      04  C1 00005          ADDL3   #4, TABDSC, R3                                  ; 0768
                                  00000000G 00  B4 0000A          CLRW    NML$GW_PRMDESCNT                                ; 0770
                                54        08  BC  3C 00010          MOVZWL  @TABDSC, R4                                     ; 0772
                                52            01  CE 00014          MNEGL   #1, I                                           ; 0777
                                              30  11 00017          BRB     2$                                              ;
                                              6E  D4 00019 1$:       CLRL    FLDADR                                          ; 0775
                                              5E  DD 0001B           PUSHL   SP                                              ; 0777
                                          08  AE  9F 0001D           PUSHAB  FLDSIZE                                         ;
                                          00 B342  DF 00020          PUSHAL  @0(R3)[I]                                       ; 0778
                                    7E            9E  3C 00024          MOVZWL  @(SP)+, -(SP)                                   ;
                                          04  AC  DD 00027          PUSHL   RECDSC                                          ; 0777
                              00000000G   00        04  FB 0002A          CALLS   #4, NMA$SEARCHFLD                               ;
                                    15            50  E9 00031          BLBC    R0, 2$                                          ;
                                          00 B342  DF 00034          PUSHAL  @0(R3)[I]                                       ; 0784
                55                9E        10      10  EF 00038          EXTZV   #16, #16, @(SP)+, INDEX                         ;
                                              6E  DD 0003D           PUSHL   FLDADR                                          ; 0788
                                          08  AE  DD 0003F          PUSHL   FLDSIZE                                         ; 0787
                                              55  DD 00042          PUSHL   INDEX                                           ; 0786
                                    FBDB      CF      03  FB 00044          CALLS   #3, NML$SAVEPARAM                               ;
                          CC                52      54  F2 00049 2$:       AOBLSS  R4, I, 1$                               ; 0772
                                            50      01  D0 0004D          MOVL    #1, R0                                  ; 0792
                                                        04 00050          RET                                             ; 0794

; Routine Size:  81 bytes,    Routine Base:  $CODE$ + 03DC
```

```
;  805          0795  1 END
;  806          0796  1
;  807          0797  0 ELUDOM



                                                        .EXTRN  LIB$SIGNAL

;                                  PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $CODE$                          1069  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                                Library Statistics
;
;                                          -------- Symbols --------      Pages       Processing
;       File                               Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[NML.OBJ]NMLLIB.L32;1          341        38        11          27        00:00.1
;  _$255$DUA28:[SHRLIB]NMALIBRY.L32;1         887         3         0          47        00:00.2
;  _$255$DUA28:[SHRLIB]NET.L32;1             1279        14         1          63        00:00.3
;  _$255$DUA28:[SYSLIB]STARLET.L32;1         9776         5         0         581        00:03.2



;                                     COMMAND QUALIFIERS

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

;  Size:          1069 code + 0 data bytes
;  Run Time:         00:24.0
;  Elapsed Time:     01:02.4
;  Lines/CPU Min:     1991
;  Lexemes/CPU-Min:  12432
;  Memory Used:   169 pages
;  Compilation Complete
```

NMLPURGE
LIS

NMLPARINI
LIS

NMLNODFIL
LIS

NMLREAD
LIS

NMLPARPRM
LIS

NMLPMANIP
LIS