J 6

```
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**NMLSET

```
NN      NN MM      MM LL         SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT
NN      NN MM      MM LL         SSSSSSSS  EEEEEEEEEE  TTTTTTTTTT
NN      NN MMMM  MMMM LL        SS         EE              TT
NN      NN MMMM  MMMM LL        SS         EE              TT
NNNN    NN MM  MM  MM LL        SS         EE              TT
NNNN    NN MM  MM  MM LL        SS         EE              TT
NN  NN  NN MM      MM LL         SSSSSS    EEEEEEEE        TT
NN  NN  NN MM      MM LL         SSSSSS    EEEEEEEE        TT
NN    NNNN MM      MM LL               SS  EE              TT
NN    NNNN MM      MM LL               SS  EE              TT
NN      NN MM      MM LL               SS  EE              TT       ::::
NN      NN MM      MM LL               SS  EE              TT       ::::
NN      NN MM      MM LLLLLLLLLL SSSSSSSS  EEEEEEEEEE      TT       ::::
NN      NN MM      MM LLLLLLLLLL SSSSSSSS  EEEEEEEEEE      TT       ::::

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

```
%TITLE 'NML SET parameter module'
MODULE NML$SET (
                LANGUAGE (BLISS32),
                ADDRESSING_MODE (NONEXTERNAL=GENERAL),
                ADDRESSING_MODE (EXTERNAL=GENERAL),
                IDENT = 'V04-000'
                ) =
BEGIN
!
!****************************************************************************
!*                                                                          *
!*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
!*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
!*  ALL RIGHTS RESERVED.                                                    *
!*                                                                          *
!*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
!*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
!*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
!*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
!*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
!*  TRANSFERRED.                                                            *
!*                                                                          *
!*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
!*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
!*  CORPORATION.                                                            *
!*                                                                          *
!*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
!*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
!*                                                                          *
!*                                                                          *
!****************************************************************************
!

!++
! FACILITY:  DECnet-VAX Network Management Listener
!
! ABSTRACT:
!
!       These routines provide the operations necessary to perform all
!       NCP SET entity commands.
!
! ENVIRONMENT:  VAX/VMS Operating System
!
! AUTHOR:  Distributed Systems Software Engineering
!
! CREATION DATE:  30-DEC-1979
!
! MODIFIED BY:
!
!       V03-011 MKP0017         Kathy Perko     13-Aug-1984
!               For entities that have at least one required field,
!               don't do a SET ALL if the entity isn't in the permanent
!               database.
!
!       V03-010 MKP0016         Kathy Perko     25-Mar-1984
!               Fix SET LOGGING bug so events are added correctly.
!
```

```
1 !       V03-009 MKP0015         Kathy Perko     3-Feb-1984
1 !               Add X25-Access Module support.
1 !
1 !       V03-008 MKP0014         Kathy Perko     4-Aug-1983
1 !               Make changes to give the node permanent database multiple
1 !               ISAM keys.  This speeds up accessint it.
1 !
1 !       V03-007 MKP0013         Kathy Perko     26-April-1983
1 !               Add SET MODULE CONFIGURATOR ALL commands.
1 !
1 !       V03-006 MKP0012         Kathy Perko     23-Nov_1982
1 !               Add module as a source for events.
1 !
1 !       V03-005 MKP0011         Kathy Perko     27-Oct-1982
1 !               Enlarge NFB buffer used for SETs.
1 !
1 !       V03-004 MKP0010         Kathy Perko     14-Sept-1982
1 !               For logging to the executor node, the sink node address is
1 !               stored in the permanent database as zero.  This allows the
1 !               logging permanent database to be transportable without logging
1 !               events to the old executor.
1 !
1 !       V03-003 MKP0009         Kathy Perko     10-Sept-1982
1 !               Put in fix so SET X25-PROTOCOL KNOWN GROUPS ALL works.
1 !
1 !       V03-002 MKP0008         Kathy Perko     23-June-1982
1 !               Add support to handle X25-Protocol "active network" concept.
1 !               Also, change qualifier handling to use the qualifier's
1 !               Parameter Semantic Table (PST) entry address, instead of
1 !               the qualifier's Network Management parameter code.
1 !               Add X25 and X29 Server and Trace modules.
1 !
1 !       V03-001 MKP0007         Kathy Perko     28-April-1982
1 !               Make changes to add second search key to QIO interface
1 !               to NETACP.
1 !
1 !       V02-005 MKP0006         Kathy Perko     07-Feb-1982
1 !               Repair previous fix to reinstate SET KNOWN LOGGING ALL.
1 !
1 !       V02-004 MKP0005         Kathy Perko     31-Jan-1982
1 !               For SET KNOWN LOGGING commands, change NML so it will do the
1 !               update if the sink has an entry in either the ESI or the
1 !               EFI volatile database.
1 !
1 !       V02-004 MKP0004         Kathy Perko     17-Nov-1981
1 !               Add circuits to logging event sources, and fix buffer
1 !               overflow problem correctly (if not filters defined for
1 !               a sink, error returned was buffer overflow).
1 !
1 !       V02-003 MKP0003         Kathy Perko     16-Nov-1981
1 !               Undo the previous fix.  It's wrong.
1 !
1 !       V02-002 MKP0002         Kathy Perko     17-Sept-1981
1 !               Fix SET KNOWN LOGGING ALL so that buffer overflow
1 !               is not returned if no logging is defined for a sink.
1 !
1 !       V02-001 MKP0001         Kathy Perko     21-July-1981
```

```
1 !                        Add Circuit entity.
1 !--
1
```

```
%SBTTL 'Declarations'
!
! TABLE OF CONTENTS:
!

FORWARD ROUTINE
	NML$SETENTITY		: NOVALUE,
	NML$SETKNOLOG		: NOVALUE,
	NML$SETLOGGING		: NOVALUE,
	NML_SETLOGGING		: NOVALUE,
	NML_SETLOGALL		: NOVALUE,
	NML_ADDALLFIL,
	NML_SETENTITY,
	NML$SETLINE		: NOVALUE,
	NML$SETEXECUTOR		: NOVALUE,
	NML$SETKNOWN		: NOVALUE,
	NML$SETKNONODES		: NOVALUE,
	NML_SETKNOWN		: NOVALUE,
	NML_SETKNOWNALL		: NOVALUE,
	NML_SETEXE		: NOVALUE,
	NML_SETEXEALL		: NOVALUE,
	NML$SET_NI_CONFIG	: NOVALUE;

!
! INCLUDE FILES:
!

LIBRARY 'LIB$:NMLLIB.L32';
LIBRARY 'SHRLIB$:NMALIBRY.L32';
LIBRARY 'SHRLIB$:NET.L32';
LIBRARY 'SYS$LIBRARY:STARLET.L32';

!
! EQUATED SYMBOLS:
!

!
! OWN STORAGE:
!

!
! Buffers and descriptors.
!
OWN
	NML$T_PRMBUFFER	: VECTOR [NML$K_QIOBFLEN, BYTE], ! Parameter (scratch) buffer
	NML$T_NFBBUFFER	: VECTOR [NML$K_NFBBFLEN, BYTE], ! NFB QIO buffer
	NML$T_P2BUFFER	: VECTOR [NML$K_P2BUFLEN,  BYTE], ! P2 QIO buffer
	NML$T_ENTBUFFER	: VECTOR [NML$K_ENTBUFLEN, BYTE]; ! Entity buffer

BIND
	NML$Q_PRMBFDSC	= UPLIT (NML$K_QIOBFLEN, NML$T_PRMBUFFER)
			  : DESCRIPTOR,
	NML$Q_NFBBFDSC	 = UPLIT (%ALLOCATION(NML$T_NFBBUFFER),  NML$T_NFBBUFFER)
			  : DESCRIPTOR,
	NML$Q_P2BFDSC	= UPLIT (%ALLOCATION(NML$T_P2BUFFER),  NML$T_P2BUFFER)
			  : DESCRIPTOR;
```

```
1 OWN
1     NML$Q_ENTBFDSC : DESCRIPTOR
1                      INITIAL (0, NML$T_ENTBUFFER);
1
1 !
1 ! EXTERNAL REFERENCES:
1 !
1
1 $NML_EXTDEF;
1
1 EXTERNAL LITERAL
1     NML$_BADEVTUPD,
1     NML$_NOSNKNOD;
1
1 EXTERNAL ROUTINE
1     NML$MATCHRECORD,
1     NMA$SEARCHFLD,
1     NML$ADDEVENTS,
1     NML$ADDFILTERS,
1     NML$BLDALLDES,
1     NML$BLDP2,
1     NML$BLD_REPLY,
1     NML$BLDSETQBF,
1     NML$CALL_NI_CONFIG,
1     NML$GETEXEID,
1     NML$GETINFTABS,
1     NML$GET_ENTITY_IDS,
1     NML$GETNXTEVT,
1     NML$GETNXTSNK,
1     NML$GETEXEADR,
1     NML$GETRECOWNER,
1     NML$GETTABLE,
1     NML$NETQIO,
1     NML$READ_KNOWN_NODE_REC,
1     NML$READRECORD,
1     NML$SAVEVENTS,
1     NML$SEND;
```

```
%SBTTL 'NML$SETENTITY  Set volatile database parameters'
GLOBAL ROUTINE NML$SETENTITY (ENTITY, ENTITY_LEN, ENTITY_ADR,
                              QUAL_PST, QUAL_LEN, QUAL_ADR) : NOVALUE =

!++
! FUNCTIONAL DESCRIPTION:
!
!       This routine adds the specified parameters to the volatile data base
!       entry fcr the specified component.  The purpose of this routine is
!       to allow the same code (NML_CLEARENTITY, etc.) to be used for both
!       singular and plural entity operations.
!
! FORMAL PARAMETERS:
!
!       ENTITY          Entity type code.
!       ENTITY_LEN      Byte count of entity id string.
!       ENTITY_ADR      Address of entity id string.
!       QUAL_PST        Qualifier's Parameter Semantic Table address
!       QUAL_LEN        Qualifier length
!       QUAL_ADR        Qualifier address
!
! SIDE EFFECTS:
!
!       A NICE response message is built and transmitted.
!
!--

    BEGIN

    LOCAL
        MSGSIZE;

!
! X25 Server and Trace, and X29 Server databases have only one entry.  So
! always do a wildcard zero of these databases.
!
IF .ENTITY EQL NML$C_X25_SERV OR
   .ENTITY EQL NML$C_X29_SERV OR
   .ENTITY EQL NML$C_TRACE THEN
    ENTITY_LEN = -1;

!
! Call the routine to process the entity.
!
    NML_SETENTITY (.ENTITY, .ENTITY_LEN, .ENTITY_ADR,
                   .QUAL_PST, .QUAL_LEN, .QUAL_ADR);
!
! Build and signal the response message.
!
    NML$BLD_REPLY (NML$AB_MSGBLOCK, MSGSIZE);
    NML$SEND (NML$AB_SNDBUFFER, .MSGSIZE);

    END;                                ! End of NML$SETENTITY


                                        .TITLE  NML$SET NML SET parameter module
                                        .IDENT  \V04-000\
```

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

        000004B0  00000 P.AAA:  .LONG   1200
        00000000' 00004         .ADDRESS NML$T_PRMBUFFER
        00000100  00008 P.AAB:  .LONG   256
        00000000' 0000C         .ADDRESS NML$T_NFBBUFFER
        00000068  00010 P.AAC:  .LONG   104
        00000000' 00014         .ADDRESS NML$T_P2BUFFER

                                .PSECT  $OWN$,NOEXE,2

                  00000 NML$T_PRMBUFFER:
                                .BLKB   1200
                  004B0 NML$T_NFBBUFFER:
                                .BLKB   256
                  005B0 NML$T_P2BUFFER:
                                .BLKB   104
                  00618 NML$T_ENTBUFFER:
                                .BLKB   64
        00000000  00658 NML$Q_ENTBFDSC:
                                .LONG   0
        00000000' 0065C         .ADDRESS NML$T_ENTBUFFER

                        NML$Q_PRMBFDSC=         P.AAA
                        NML$Q_NFBBFDSC=         P.AAB
                        NML$Q_P2BFDSC=          P.AAC
                                .EXTRN  NML$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$_BADEVTUPD, NML$_NOSNKNOD
                                                  .EXTRN  NML$MATCHRECORD
                                                  .EXTRN  NMA$SEARCHFLD, NML$ADDEVENTS
                                                  .EXTRN  NML$ADDFILTERS, NML$BLDALLDES
                                                  .EXTRN  NML$BLDP2, NML$BLD_REPLY
                                                  .EXTRN  NML$BLDSETQBF, NML$CALL_NI_CONFIG
                                                  .EXTRN  NML$GETEXEID, NML$GETINFTABS
                                                  .EXTRN  NML$GET_ENTITY_IDS
                                                  .EXTRN  NML$GETNXTEVT, NML$GETNXTSNK
                                                  .EXTRN  NML$GETEXEADR, NML$GETRECOWNER
                                                  .EXTRN  NML$GETTABLE, NML$NETQIO
                                                  .EXTRN  NML$READ_KNOWN_NODE_REC
                                                  .EXTRN  NML$READRECORD, NML$SAVEVENTS
                                                  .EXTRN  NML$SEND

                                                  .PSECT  $CODE$,NOWRT,2

                                 0000 00000       .ENTRY  NML$SETENTITY, Save nothing                        ; 0213
             5E             04   C2 00002         SUBL2   #4, SP                                             ;
             11      04     AC   D1 00005         CMPL    ENTITY, #17                                        ; 0248
                            0C   13 00009         BEQL    1$                                                 ;
             15      04     AC   D1 0000B         CMPL    ENTITY, #21                                        ; 0249
                            06   13 0000F         BEQL    1$                                                 ;
             13      04     AC   D1 00011         CMPL    ENTITY, #19                                        ; 0250
                            04   12 00015         BNEQ    2$                                                 ;
      08     AC             01   CE 00017 1$:     MNEGL   #1, ENTITY_LEN                                     ; 0251
             7E      14     AC   7D 0001B 2$:     MOVQ    QUAL_LEN, -(SP)                                    ; 0257
             7E      0C     AC   7D 0001F         MOVQ    ENTITY_ADR, -(SP)                                  ; 0256
             7E      04     AC   7D 00023         MOVQ    ENTITY, -(SP)                                      ;
00000000V    00             06   FB 00027         CALLS   #6, NML_SETENTITY                                  ;
                            5E   DD 0002E         PUSHL   SP                                                 ; 0261
                 00000000G  00   9F 00030         PUSHAB  NML$AB_MSGBLOCK                                    ;
00000000G    00             02   FB 00036         CALLS   #2, NML$BLD_REPLY                                  ;
                            6E   DD 0003D         PUSHL   MSGSIZE                                            ; 0262
                 00000000G  00   9F 0003F         PUSHAB  NML$AB_SNDBUFFER                                   ;
00000000G    00             02   FB 00045         CALLS   #2, NML$SEND                                       ;
                                 04 0004C         RET                                                        ; 0264

; Routine Size:  77 bytes,    Routine Base:  $CODE$ + 0000
```

```
  1 %SBTTL 'NML$SETKNOLOG  Set parameters for known logging'
  1 GLOBAL ROUTINE NML$SETKNOLOG (ENTITY, DUM1, DUM2) : NOVALUE =
  1
  1 !++
  1 ! FUNCTIONAL DESCRIPTION:
  1 !
  1 !       Add parameters to KNOWN entries in the logging volatile data base.
  1 !
  1 ! FORMAL PARAMETERS:
  1 !
  1 !       ENTITY          Entity type code.
  1 !       DUM1            Not used.
  1 !       DUM2            Not used.
  1 !
  1 !--
  1
  2 BEGIN
  2
  2 !
  2 ! NFBs used to find out if NETACP already has entries in its database for
  2 ! each sink type.
  2 !
P 2     $NFBDSC     (ESINFB,            ! NFB descriptor label.
P 2                 SHOW,               ! Function code
P 2                 ,                   ! No flags
P 2                 ESI,                ! ACP Data base ID
P 2                 SNK,,               ! Search key one ID = Sink type, oper1 = eql
P 2                 ,                   ! Search key two ID = wildcard, oper2 = eql
  2                 );
  2
  2 LOCAL
  2     STATUS,
  2     P3,
  2     DUMDSC:     REF DESCRIPTOR,
  2     NFBDSC:     REF DESCRIPTOR,
  2     P2DSC : DESCRIPTOR,
  2     BLKDSC: DESCRIPTOR,
  2     SRCPTR,
  2     PTR;
  2 !
  2 ! KNOWN LOGGING ALL is defined by whatever is set in the permanent data base.
  2 !
  2 ! KNOWN LOGGING is defined by sink node. In otherwords:
  2 !       SET KNOWN LOG  updates the executor node logging.  This means updating
  2 ! the sink (ESI) and filter (EFI) volatile database entries which already
  2 ! exist for the executor.  Note that the ESI database applies to the
  2 ! executor node only.
  2 !       SET KNOWN LOGGING SINK NODE updates logging for the specified sink
  2 ! node.  This means updating the filter (EFI) volatile database entries which
  2 ! already exist for that node.
  2 !
  2 INCR SNK FROM NMA$C_SNK_CON TO NMA$C_SNK_MON DO
  3     BEGIN
  3     !
  3     ! If processing a SET KNOWN LOGGING ALL command, the logging set in the
  3     ! permanent database determines what gets set in the volatile database.
  3     !
```

```
	IF .NML$GL_PRS_FLGS [NML$V_PRS_ALL]
	THEN
	    NML_SETLOGALL (.SNK)
	ELSE
	    !
	    ! Determine if there are any entries in the volatile database for this
	    ! sink node and sink (console, file, monitor).
	    !
	    BEGIN
	    STATUS = 0;
	    !
	    ! If the sink node is the executor node, check to see if there's
	    ! anything in the sink database.
	    !
	    IF .NML$GL_PRS_FLGS [NML$V_PRS_EXESNK] THEN
		BEGIN
		NML$BLDP2 (0, .SNK, -1, 0, NML$Q_P2BFDSC, P2DSC);
		STATUS = NML$NETQIO (ESINFB, P2DSC, P3, NML$GQ_QIOBFDSC);
		END;
	    !
	    ! If the sink node isn't the executor node, or there isn't anything in
	    ! the ESI database for the exec, check the filter (EFI) database for
	    ! this sink.
	    !
	    IF NOT .STATUS THEN
		BEGIN
		NML$GETINFTABS (NML$C_LOGGING, NML$C_EVENTS, NFBDSC, DUMDSC, 0);
		NML$BLDP2 (0, .NML$GW_EVTSNKADR, -1, 0, NML$Q_P2BFDSC, P2DSC);
		!
		! Get events set for specified sink node.  Then search through the
		! source block of events to see if any of the events are specified
		! for the sink currently being processed.
		!
		IF NML$NETQIO (.NFBDSC, P2DSC, P3, NML$GQ_QIOBFDSC) THEN
		    BEGIN
		    PTR = .NML$GQ_QIOBFDSC [DSC$A_POINTER];
		    BLKDSC [DSC$W_LENGTH] = .(.PTR)<0,16>;
		    BLKDSC [DSC$A_POINTER] = .PTR + 2;
		    SRCPTR = 0;
		    !
		    ! If any of the events are for the sink I am currently
		    ! working on, then do the SET for that sink.
		    !
		    STATUS = NML$GETNXTSNK (BLKDSC, .SNK, SRCPTR);
		    END;
		END;
	    IF .STATUS THEN
		!
		! Either the EFI or the ESI volatile database already has an entry
		! for this sink (File, console, or monitor).  So do the update.
		!
		NML$SETLOGGING (.ENTITY, .SNK, 0);
	    END;
	END;

END;				! End of NML$SETKNOLOG
```

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

                          00000014  00018 P.AAD:        .LONG   20                                      ;
                          00000000' 0001C               .ADDRESS U.1                                    ;

                                                        .PSECT  $OWN$,NOEXE,2

                                22  00660 ;_NFB
                                          U.1:          .BYTE   34                                      ;
                                00  00661               .BYTE   0                                       ;
                                07  00662               .BYTE   7                                       ;
                                00  00663               .BYTE   0                                       ;
                          07010010  00664               .LONG   117506064                               ;
                          00000001  00668               .LONG   1                                       ;
                                00  0066C               .BYTE   0                                       ;
                                00  0066D               .BYTE   0                                       ;
                              0000  0066E               .WORD   0                                       ;
                          00000000  00670               .LONG   0                                       ;

                                          U.2=                   P.AAD

                                                        .PSECT  $CODE$,NOWRT,2

                              01FC  00000               .ENTRY  NML$SETKNOLOG, Save R2,R3,R4,R5,R6,R7,R8        ; 0266
           58 00000000G  00   9E  00002               MOVAB   NML$NETQIO, R8                          ;
           57 00000000G  00   9E  00009               MOVAB   NML$BLDP2, R7                           ;
           56 00000000G  00   9E  00010               MOVAB   NML$GQ_QIOBFDSC, R6                     ;
           55 00000000'  00   9E  00017               MOVAB   NML$Q_P2BFDSC, R5                       ;
                  5E     20   C2  0001E               SUBL2   #32, SP                                 ;
                  52     01   D0  00021               MOVL    #1, SNK                                 ; 0316
0C 00000000G  00         01   E1  00024 1$:           BBC     #1, NML$GL_PRS_FLGS, 2$                 ; 0322
                         52   DD  0002C               PUSHL   SNK                                     ; 0324
   00000000V  00         01   FB  0002E               CALLS   #1, NML_SETLOGALL                       ;
                       009B   31  00035               BRW     6$                                      ;
                         54   D4  00038 2$:           CLRL    STATUS                                  ; 0331
           22 00000000G  00   E9  0003A               BLBC    NML$GL_PRS_FLGS+1, 3$                   ; 0336
                  18     AE   9F  00041               PUSHAB  P2DSC                                   ; 0338
                         55   DD  00044               PUSHL   R5                                      ;
                         7E   D4  00046               CLRL    -(SP)                                   ;
           7E            01   CE  00048               MNEGL   #1, -(SP)                               ;
                         52   DD  0004B               PUSHL   SNK                                     ;
                         7E   D4  0004D               CLRL    -(SP)                                   ;
           67            06   FB  0004F               CALLS   #6, NML$BLDP2                           ;
                         56   DD  00052               PUSHL   R6                                      ; 0339
                  0C     AE   9F  00054               PUSHAB  P3                                      ;
                  20     AE   9F  00057               PUSHAB  P2DSC                                   ;
                  08     A5   9F  0005A               PUSHAB  ESINFB                                  ;
           68            04   FB  0005D               CALLS   #4, NML$NETQIO                          ;
           54            50   D0  00060               MOVL    R0, STATUS                              ;
           5F            54   E8  00063 3$:           BLBS    STATUS, 5$                              ; 0346
                         7E   D4  00066               CLRL    -(SP)                                   ; 0348
                  04     AE   9F  00068               PUSHAB  DUMDSC                                  ;
                  0C     AE   9F  0006B               PUSHAB  NFBDSC                                  ;
                         04   DD  0006E               PUSHL   #4                                      ;
```

```
                                   01  DD 00070        PUSHL   #1                                   ;
                   00000000G  00   05  FB 00072        CALLS   #5, NML$GETINFTABS                   ;
                           18      AE  9F 00079        PUSHAB  P2DSC                                ; 0349
                                   55  DD 0007C        PUSHL   R5                                   ;
                                   7E  D4 0007E        CLRL    -(SP)                                ;
                              7E   01  CE 00080        MNEGL   #1, -(SP)                            ;
                   7E 00000000G    00  3C 00083        MOVZWL  NML$GW_EVTSNKADR, -(SP)              ;
                                   7E  D4 0008A        CLRL    -(SP)                                ;
                              67   06  FB 0008C        CALLS   #6, NML$BLDP2                        ;
                                   56  DD 0008F        PUSHL   R6                                   ; 0355
                           0C      AE  9F 00091        PUSHAB  P3                                   ;
                           20      AE  9F 00094        PUSHAB  P2DSC                                ;
                           10      AE  DD 00097        PUSHL   NFBDSC                               ;
                              68   04  FB 0009A        CALLS   #4, NML$NETQIO                       ;
                              22   50  E9 0009D        BLBC    R0, 4$                               ;
                              53   04  A6  D0 000A0    MOVL    NML$GQ_QIOBFDSC+4, PTR               ; 0357
                      10      AE   63  B0 000A4        MOVW    (PTR), BLKDSC                        ; 0358
                      14      AE   02  A3  9E 000A8    MOVAB   2(R3), BLKDSC+4                      ; 0359
                           0C      AE  D4 000AD        CLRL    SRCPTR                               ; 0360
                           0C      AE  9F 000B0        PUSHAB  SRCPTR                               ; 0365
                                   52  DD 000B3        PUSHL   SNK                                  ;
                           18      AE  9F 000B5        PUSHAB  BLKDSC                               ;
                   00000000G  00   03  FB 000B8        CALLS   #3, NML$GETNXTSNK                    ;
                              54   50  D0 000BF        MOVL    R0, STATUS                           ;
                              0E   54  E9 000C2  4$:   BLBC    STATUS, 6$                           ; 0368
                                   7E  D4 000C5  5$:   CLRL    -(SP)                                ; 0373
                                   52  DD 000C7        PUSHL   SNK                                  ;
                           04      AC  DD 000C9        PUSHL   ENTITY                               ;
                   00000000V  00   03  FB 000CC        CALLS   #3, NML$SETLOGGING                   ;
         FF4B             52  01   03  F1 000D3  6$:   ACBL    #3, #1, SNK, 1$                      ; 0316
                                       04 000D9        RET                                          ; 0377

; Routine Size:  218 bytes,     Routine Base:  $CODE$ + 004D
```

```
%SBTTL 'NML$SETLOGGING  Set logging parameters'
GLOBAL ROUTINE NML$SETLOGGING (ENTITY, SNK, DUM2) : NOVALUE =

!++
! FUNCTIONAL DESCRIPTION:
!
!       Add parameters to the volatile data base entry for the specified
!       logging entity.
!
! FORMAL PARAMETERS:
!
!       ENTITY          Entity type code.
!       SNK             Logging sink type.
!       DUM2            Not used.
!
! IMPLICIT INPUTS:
!
!       NML$GL_PRS_FLGS Message parsing flags.
!       NML$GW_EVTSNKADR Sink node address.
!
! IMPLICIT OUTPUTS:
!
!       NONE
!
! ROUTINE VALUE:
! COMPLETION CODES:
!
!       NONE
!
! SIDE EFFECTS:
!
!       A NICE response message is built and transmitted.
!
!--

    BEGIN

    LOCAL
        MSG_SIZE;                       ! Message size
!
! Check the parsing flags to see if this is a SET ALL function.
!
    IF .NML$GL_PRS_FLGS [NML$V_PRS_ALL]
    THEN
        NML_SETLOGALL (.SNK)
    ELSE
        BEGIN
!
! Decide if the parameter group is for filters (EFI) or sinks (ESI).
!
        IF .NML$GL_PRS_FLGS [NML$V_PRS_ESIPG]
        THEN
            NML_SETENTITY (NML$C_SINK, 1, SNK, 0, 0, 0)
        ELSE
            NML_SETLOGGING (.SNK, .NML$GW_EVTSNKADR);

        END;
```

```
;  439     0435  2 !
;  440     0436  2 ! Add entity id (sink type code) to entity buffer.
;  441     0437  2 !
;  442     0438  2     NML$Q_ENTBFDSC [DSC$W_LENGTH] = 1;
;  443     0439  2     NML$Q_ENTBFDSC [DSC$A_POINTER] = NML$T_ENTBUFFER;
;  444     0440  2     NML$T_ENTBUFFER<0,8> = .SNK;
;  445     0441  2 !
;  446     0442  2 ! Add entity descriptor to message information.
;  447     0443  2 !
;  448     0444  2     NML$AB_MSGBLOCK [MSB$V_ENTD_FLD] = 1;
;  449     0445  2     NML$AB_MSGBLOCK [MSB$A_ENTITY] = NML$Q_ENTBFDSC;
;  450     0446  2 !
;  451     0447  2 ! Build and send the message.
;  452     0448  2 !
;  453     0449  2     NML$BLD_REPLY (NML$AB_MSGBLOCK, MSG_SIZE);
;  454     0450  2     NML$SEND (NML$AB_SNDBUFFER, .MSG_SIZE); ! Send message
;  455     0451  2
;  456     0452  1     END;                                ! End of NML$SETLOGGING


                                      000C 00000          .ENTRY  NML$SETLOGGING, Save R2,R3                 ; 0379
                     53 00000000G  00   9E 00002          MOVAB   NML$AB_MSGBLOCK, R3                        ;
                     52 00000000'  00   9E 00009          MOVAB   NML$Q_ENTBFDSC, R2                         ;
                     5E            04   C2 00010          SUBL2   #4, SP                                     ;
         0C 00000000G  00          01   E1 00013          BBC     #1, NML$GL_PRS_FLGS, 1$                    ; 0420
                             08    AC   DD 0001B          PUSHL   SNK                                        ; 0422
            00000000V  00          01   FB 0001E          CALLS   #1, NML_SETLOGALL                          ;
                                   2D   11 00025          BRB     3$                                         ;
         14 00000000G  00          04   E1 00027 1$:      BBC     #4, NML$GL_PRS_FLGS+1, 2$                  ; 0428
                                   7E   7C 0002F          CLRQ    -(SP)                                      ; 0430
                                   7E   D4 00031          CLRL    -(SP)                                      ;
                             08    AC   9F 00033          PUSHAB  SNK                                        ;
                                   01   DD 00036          PUSHL   #1                                         ;
                                   02   DD 00038          PUSHL   #2                                         ;
            00000000V  00          06   FB 0003A          CALLS   #6, NML_SETENTITY                          ;
                                   11   11 00041          BRB     3$                                         ;
                     7E 00000000G  00   3C 00043 2$:      MOVZWL  NML$GW_EVTSNKADR, -(SP)                    ; 0432
                             08    AC   DD 0004A          PUSHL   SNK                                        ;
            00000000V  00          02   FB 0004D          CALLS   #2, NML_SETLOGGING                         ;
                       62          01   B0 00054 3$:      MOVW    #1, NML$Q_ENTBFDSC                         ; 0438
                  04   A2    C0    A2   9E 00057          MOVAB   NML$T_ENTBUFFER, NML$Q_ENTBFDSC+4          ; 0439
                  C0   A2    08    AC   90 0005C          MOVB    SNK, NML$T_ENTBUFFER                       ; 0440
                       63          10   88 00061          BISB2   #16, NML$AB_MSGBLOCK                       ; 0444
                  14   A3          62   9E 00064          MOVAB   NML$Q_ENTBFDSC, NML$AB_MSGBLOCK+20         ; 0445
                           4008    8F   BB 00068          PUSHR   #^M<R3,SP>                                 ; 0449
            00000000G  00          02   FB 0006C          CALLS   #2, NML$BLD_REPLY                          ;
                                   6E   DD 00073          PUSHL   MSG_SIZE                                   ; 0450
                        00000000G  00   9F 00075          PUSHAB  NML$AB_SNDBUFFER                           ;
            00000000G  00          02   FB 0007B          CALLS   #2, NML$SEND                               ;
                                        04 00082          RET                                                ; 0452

; Routine Size:  131 bytes,    Routine Base:  $CODE$ + 0127
```

```
1 %SBTTL 'NML_SETLOGGING  Set logging parameters'
1 ROUTINE NML_SETLOGGING (SNK, SNKADR) : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine performs common SET functions for both singular
1 !       and plural logging volatile data base operations.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       SNK             Logging sink type.
1 !       SNKADR          Sink node address.
1 !
1 !--
1
2     BEGIN
2
2     MAP
2         SNKADR : WORD;
2
2     LOCAL
2         DB,                                   ! Database ID
2         SRCHKEY1,                             ! Search key one ID
2         SRCHKEY2,                             ! Search key two ID
2         DUMDSC : REF DESCRIPTOR,              ! Dummy descriptor for table
2         EVTADR,                               ! Address of event parameter
2         EVTLEN,                               ! Length of event parameter
2         NFBDSC : REF DESCRIPTOR,
2         P3,                                   ! Byte count of data returned by NETACP.
2         P2DSC  : DESCRIPTOR,
2         QBFDSC : DESCRIPTOR,
2         RECDSC : DESCRIPTOR,
2         STATUS,
2         TABDES : REF DESCRIPTOR,
2         UPDFLG;                               ! Data base update flag
2
2     RECDSC [DSC$W_LENGTH] = 0;                ! Initial descriptor
2     RECDSC [DSC$A_POINTER] = .NML$GQ_RECBFDSC [DSC$A_POINTER];
2
2     NML$GETINFTABS (NML$C_LOGGING, NML$C_EVENTS, NFBDSC, DUMDSC, 0);
2     NML$BLDP2 (0, .SNKADR, -1, 0, NML$Q_P2BFDSC, P2DSC);
2
2     STATUS = NML$NETQIO (.NFBDSC, P2DSC, P3, NML$GQ_QIOBFDSC);
2
3     IF NOT .STATUS AND (.STATUS NEQ NML$_STS_CMP)
2     THEN
2         RETURN;
2
2     IF .STATUS
2     THEN
3         BEGIN
3
3         EVTLEN = .(.NML$GQ_QIOBFDSC [DSC$A_POINTER])<0,16>;
3         EVTADR = .NML$GQ_QIOBFDSC [DSC$A_POINTER] + 2;
3
3         END
```

```
    ELSE
        !
        ! There is no entry for this sink in NETACPs filter (EFI) database.
        ! Create a null permanent data base entry to which the filter info
        ! from the NICE command will be added.
        !
        BEGIN
        EVTLEN = 0;
        EVTADR = 0;
        END;

    IF NOT NML$SAVEVENTS (.NML$GQ_RECBFDSC [DSC$W_LENGTH],
                          .EVTLEN,
                          .EVTADR,
                          RECDSC)
    THEN
        RETURN;
!
! Add event to record.  If this fails then just return.  The error message
! information will already be set up in the message block.
!
    IF NOT NML$ADDEVENTS (TRUE, RECDSC, .SNK, .SNKADR, UPDFLG)
    THEN
        RETURN;

    IF NOT .UPDFLG
    THEN
        BEGIN

        NML$AB_MSGBLOCK [MSB$L_FLAGS] = MSB$M_MSG_FLD;
        NML$AB_MSGBLOCK [MSB$B_CODE]  = NMA$C_STS_MPR;
        NML$AB_MSGBLOCK [MSB$L_TEXT]  = NML$_BADEVTUPD;
        RETURN;

        END;
!
! If event field was added successfully, then update the volatile data base.
!
    DB = .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$B_DATABASE];
    SRCHKEY1 = .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$L_SRCH_ID1];
    SRCHKEY2 = .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$L_SRCH_ID2];
    TABDES = .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$A_ALLTAB]; ! Table for SET
!
! Build the QIO buffer.
!
    NML$BLDALLDES (RECDSC, .TABDES);
    NML$BLDSETQBF (NFB$C_FC_SET, .DB,
                   .SRCHKEY1, 0, SNKADR,
                   .SRCHKEY2, -1, 0,
                   NML$Q_NFBBFDSC, NFBDSC,
                   NML$Q_P2BFDSC, P2DSC,
                   NML$GQ_QIOBFDSC, QBFDSC);
!
! Set the event information into the volatile data base.
!
    IF NML$NETQIO (NFBDSC, P2DSC, 0, QBFDSC)
    THEN
```

```
;  572          0567 3          BEGIN
;  573          0568 3
;  574          0569 3          NML$AB_MSGBLOCK [MSB$L_FLAGS] = 0;
;  575          0570 3          NML$AB_MSGBLOCK [MSB$B_CODE]  = NMA$C_STS_SUC;
;  576          0571 3
;  577          0572 2          END;
;  578          0573 2
;  579          0574 2      RETURN;
;  580          0575 2
;  581          0576 1      END;                                      ! End of NML_SETLOGGING



                                                  03FC 00000 NML_SETLOGGING:
                                                                   .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9                         ; 0454
                       59 00000000G   00   9E 00002                MOVAB   NML$NETQIO, R9                                       ;
                       58 00000000G   00   9E 00009                MOVAB   NML$GQ_QIOBFDSC, R8                                  ;
                       57 00000000'   00   9E 00010                MOVAB   NML$Q_P2BFDSC, R7                                    ;
                       56 00000000G   00   9E 00017                MOVAB   NML$AB_ENTITYDATA+49, R6                             ;
                       55 00000000G   00   9E 0001E                MOVAB   NML$AB_MSGBLOCK, R5                                  ;
                       5E             28   C2 00025                SUBL2   #40, SP                                              ;
                                10    AE   B4 00028                CLRW    RECDSC                                               ; 0490
              14    AE 00000000G      00   D0 0002B                MOVL    NML$GQ_RECBFDSC+4, RECDSC+4                          ; 0491
                                      7E   D4 00033                CLRL    -(SP)                                                ; 0493
                                04    AE   9F 00035                PUSHAB  DUMDSC                                               ;
                                14    AE   9F 00038                PUSHAB  NFBDSC                                               ;
                                      04   DD 0003B                PUSHL   #4                                                   ;
                                      01   DD 0003D                PUSHL   #1                                                   ;
           00000000G  00              05   FB 0003F                CALLS   #5, NML$GETINFTABS                                   ;
                                20    AE   9F 00046                PUSHAB  P2DSC                                                ; 0494
                                      57   DD 00049                PUSHL   R7                                                   ;
                                      7E   D4 0004B                CLRL    -(SP)                                                ;
                       7E             01   CE 0004D                MNEGL   #1, -(SP)                                            ;
                       7E       08    AC   3C 00050                MOVZWL  SNKADR, -(SP)                                        ;
                                      7E   D4 00054                CLRL    -(SP)                                                ;
           00000000G  00              06   FB 00056                CALLS   #6, NML$BLDP2                                        ;
                                      58   DD 0005D                PUSHL   R8                                                   ; 0496
                                08    AE   9F 0005F                PUSHAB  P3                                                   ;
                                28    AE   9F 00062                PUSHAB  P2DSC                                                ;
                                18    AE   DD 00065                PUSHL   NFBDSC                                               ;
                       69             04   FB 00068                CALLS   #4, NML$NETQIO                                       ;
                       0D             50   E8 0006B                BLBS    STATUS, 2$                                           ; 0498
           FFFFFFF0   8F              50   D1 0006E                CMPL    STATUS, #-16                                         ;
                                      01   13 00075                BEQL    1$                                                   ;
                                      04      00077                RET                                                          ;
                       09             50   E9 00078 1$:            BLBC    STATUS, 3$                                           ; 0502
                       50       04    A8   D0 0007B 2$:            MOVL    NML$GQ_QIOBFDSC+4, R0                                ; 0506
                       51             80   3C 0007F                MOVZWL  (R0)+, EVTLEN                                        ;
                                      02   11 00082                BRB     4$                                                   ; 0502
                                      50   7C 00084 3$:            CLRQ    EVTADR                                               ; 0518
                                10    AE   9F 00086 4$:            PUSHAB  RECDSC                                               ; 0521
                                      50   DD 00089                PUSHL   EVTADR                                               ; 0523
                                      51   DD 0008B                PUSHL   EVTLEN                                               ; 0522
                       7E 00000000G   00   3C 0008D                MOVZWL  NML$GQ_RECBFDSC, -(SP)                               ; 0521
           00000000G  00              04   FB 00094                CALLS   #4, NML$SAVEVENTS                                    ;
```

```
                7F              50  E9 0009B          BLBC    R0, 6$                                   ;
                        08      AE  9F 0009E          PUSHAB  UPDFLG                                   ; 0531
                7E      08      AC  3C 000A1          MOVZWL  SNKADR, -(SP)                            ;
                        04      AC  DD 000A5          PUSHL   SNK                                      ;
                        1C      AE  9F 000A8          PUSHAB  RECDSC                                   ;
                                01  DD 000AB          PUSHL   #1                                       ;
       00000000G  00            05  FB 000AD          CALLS   #5, NML$ADDEVENTS                        ;
                6F              50  E9 000B4          BLBC    R0, 7$                                   ;
                10      08      AE  E8 000B7          BLBS    UPDFLG, 5$                               ; 0535
                65              04  D0 000BB          MOVL    #4, NML$AB_MSGBLOCK                      ; 0539
        04      A5              05  8E 000BE          MNEGB   #5, NML$AB_MSGBLOCK+4                    ; 0540
        0C      A5 00000000G    8F  D0 000C2          MOVL    #NML$_BADEVTUPD, NML$AB_MSGBLOCK+12      ; 0541
                                    04 000CA          RET                                              ; 0537
                54              66  9A 000CB 5$:      MOVZBL  NML$AB_ENTITYDATA+49, DB                 ; 0548
                53      01      A6  D0 000CE          MOVL    NML$AB_ENTITYDATA+50, SRCHKEY1           ; 0549
                52      05      A6  D0 000D2          MOVL    NML$AB_ENTITYDATA+54, SRCHKEY2           ; 0550
                50      23      A6  D0 000D6          MOVL    NML$AB_ENTITYDATA+84, TABDES             ; 0551
                                50  DD 000DA          PUSHL   TABDES                                   ; 0555
                        14      AE  9F 000DC          PUSHAB  RECDSC                                   ;
       00000000G  00            02  FB 000DF          CALLS   #2, NML$BLDALLDES                        ;
                        18      AE  9F 000E6          PUSHAB  QBFDSC                                   ; 0556
                                58  DD 000E9          PUSHL   R8                                       ;
                        28      AE  9F 000EB          PUSHAB  P2DSC                                    ;
                                57  DD 000EE          PUSHL   R7                                       ;
                        1C      AE  9F 000F0          PUSHAB  NFBDSC                                   ;
                        F8      A7  9F 000F3          PUSHAB  NML$Q_NFBBFDSC                           ;
                                7E  D4 000F6          CLRL    -(SP)                                    ;
                7E              01  CE 000F8          MNEGL   #1, -(SP)                                ; 0558
                                52  DD 000FB          PUSHL   SRCHKEY2                                 ;
                        08      AC  9F 000FD          PUSHAB  SNKADR                                   ; 0556
                                7E  D4 00100          CLRL    -(SP)                                    ;
                                53  DD 00102          PUSHL   SRCHKEY1                                 ; 0557
                                54  DD 00104          PUSHL   DB                                       ; 0556
                                23  DD 00106          PUSHL   #35                                      ;
       00000000G  00            0E  FB 00108          CALLS   #14, NML$BLDSETQBF                       ;
                        18      AE  9F 0010F          PUSHAB  QBFDSC                                   ; 0565
                                7E  D4 00112          CLRL    -(SP)                                    ;
                        28      AE  9F 00114          PUSHAB  P2DSC                                    ;
                        18      AE  9F 00117          PUSHAB  NFBDSC                                   ;
                69              04  FB 0011A          CALLS   #4, NML$NETQIO                           ;
                06              50  E9 0011D 6$:      BLBC    R0, 7$                                   ;
                                65  D4 00120          CLRL    NML$AB_MSGBLOCK                          ; 0569
        04      A5              01  90 00122          MOVB    #1, NML$AB_MSGBLOCK+4                    ; 0570
                                    04 00126 7$:      RET                                              ; 0576

; Routine Size:  295 bytes,    Routine Base:  $CODE$ + 01AA
```

```
1 %SBTTL 'NML_SETLOGALL  Set all logging parameters'
1 ROUTINE NML_SETLOGALL (SNK) : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine sets all permanent parameters for the specified
1 !       logging sink type into the volatile data base.
1 !
1 !--
1
2 BEGIN
2
2 LOCAL
2     DB,                               ! Database ID
2     SRCHKEY1,                         ! Search key one ID
2     SRCHKEY2,                         ! Search key two ID
2     SINK_NODE_PTR,                    ! Pointer to sink node address in NICE message.
2     SINK_NODE_ADD,                    ! Sink node address from NICE message.
2     SINK_NODE_LEN,                    ! Length of sink node address (2).
2     FID,                              ! File id code
2     KEY,
2     OWNER,                            ! Search key
2     NFBDSC : DESCRIPTOR,              ! NFB buffer descriptor
2     P2DSC  : DESCRIPTOR,              ! QIO P2 buffer descriptor
2     QBFDSC : DESCRIPTOR,              ! QIO P4 buffer descriptor
2     RECDSC : DESCRIPTOR,              ! Record descriptor
2     SETDSC : DESCRIPTOR,              ! Event parameter descriptor
2     STATUS,
2     TABDES : REF DESCRIPTOR,          ! Table for SET ALL
2     EXEC_ADDRESS;                     ! Address of executor node.
2 !
2 ! Get entity information.
2 !
2 DB      = .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$B_DATABASE];
2 SRCHKEY1= .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$L_SRCH_ID1];
2 SRCHKEY2= .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$L_SRCH_ID2];
2 FID     = .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$B_FILEID];
2 OWNER   = .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$W_KEY];
2 TABDES = .NML$AB_ENTITYDATA [NML$C_LOGGING, EIT$A_ALLTAB];
2 !
2 ! The logging database is broken up into two databases -
2 !       - the Filter database (EFI) uses event filters to determine whether
2 !         or not a locally generated event is being logged and what sink
2 !         node to send it to for logging.
2 !       - the Sink database (ESI) is used by the sink node receiving the event
2 !         to determine if the sink type (file, console, monitor) is logging
2 !         events (sink state = on).
2 !
2 ! First, set the Sink database
2 !
2 IF NOT NML_SETENTITY (NML$C_SINK, 1, SNK, 0, 0, 0) THEN
2     RETURN;
2 !
2 ! Now set the Filter database parameters.
2 ! At this point, it is assumed that the logging permanent data base file
2 ! is open.
```

```
!
STATUS = NML$_STS_SUC;                          ! Initialize status
KEY = 0;                                        ! Start at beginning of file
NML$GETEXEADR (EXEC_ADDRESS);
!
! Search through the logging file for entries in the Filter database.
! There is one entry for each sink node.  From each sink node's entry,
! extract all the event filters for the sink type currently being set.
! Then issue a QIO to NETACP to set the events for the sink node and
! sink in the volatile database.
!
WHILE NML$MATCHRECORD (.FID,
                       NML$GQ_RECBFDSC,
                       KEY,
                       .OWNER,
                       0, 0,    ! No entity length or id.
                       0, 0, 0,         ! No qualifier
                       RECDSC) DO
    BEGIN
    !
    ! Get the record owner.  In the case of the logging filter database,
    ! this is the sink node address.
    !
    SINK_NODE_PTR = 0;
    STATUS = NMA$SEARCHFLD (RECDSC, .OWNER, SINK_NODE_LEN, SINK_NODE_PTR);
    !
    ! If no sink node address was found in the record then something is
    ! wrong with this record in the file.
    !
    IF NOT .STATUS THEN
        BEGIN
        NML$AB_MSGBLOCK [MSB$L_FLAGS]  = MSB$M_DET_FLD OR MSB$M_MSG_FLD;
        NML$AB_MSGBLOCK [MSB$B_CODE]   = NMA$C_STS_FCO;
        NML$AB_MSGBLOCK [MSB$W_DETAIL] = NMA$C_FOPDTL_PDB;
        NML$AB_MSGBLOCK [MSB$L_TEXT]   = NML$_NOSNKNOD;
        RETURN;
        END;
    !
    ! Find event parameter in record.  If it is not found then skip this
    ! record.
    !
    SETDSC [DSC$A_POINTER] = 0;
    STATUS = NMA$SEARCHFLD (RECDSC,
                            NMA$C_PCLO_EVE,
                            SETDSC [0,0,32,0],
                            SETDSC [DSC$A_POINTER]);
    IF .STATUS THEN
        !
        ! Build QIO buffer to tell NETACP what filters to put into the logging
        ! database for this sink node.
        !
        BEGIN
        !
        ! If the sink node address in the permanent database record is zero,
        ! then the sink node is the executor.  Find out the executor's real
        ! address, and use that to specify the sink node to NETACP.
        !
```

```
4                 SINK_NODE_ADD = .(.SINK_NODE_PTR)<0,16>;
4                 IF .SINK_NODE_ADD EQL 0 THEN
4                     SINK_NODE_ADD = .EXEC_ADDRESS;
4                 IF NML_ADDALLFIL (RECDSC, SETDSC, .SNK, .SINK_NODE_ADD) THEN
5                     BEGIN
5                     NML$BLDALLDES (RECDSC, .TABDES); ! Build parameter descriptors
5                     NML$BLDSETQBF (NFB$C_FC_SET, .DB,
5                                    .SRCHKEY1, 0, SINK_NODE_ADD,
5                                    .SRCHKEY2, -1, 0,
5                                    NML$Q_NFBBFDSC, NFBDSC,
5                                    NML$Q_P2BFDSC, P2DSC,
5                                    NML$GQ_QIOBFDSC, QBFDSC);
5                 !
5                 ! Add the parameters to volatile data base entry.
5                 !
5                     STATUS = NML$NETQIO (NFBDSC, P2DSC, 0, QBFDSC);
5                     IF NOT .STATUS THEN
5                         EXITLOOP;
4                     END;
4                 END
3             ELSE
3                 STATUS = NML$_STS_SUC;            ! Reset return status
3
3             KEY = .KEY + 1;                       ! Increment record key
2             END;
2         !
2         ! If the operation was successful then set up the successful response
2         ! message.
2         !
2         IF .STATUS THEN
3             BEGIN
3             NML$AB_MSGBLOCK [MSB$L_FLAGS] = 0;
3             NML$AB_MSGBLOCK [MSB$B_CODE] = NMA$C_STS_SUC;
3             END;
2
1         END;                                      ! End of NML_SETLOGALL
```

```
                              0FFC 00000 NML_SETLOGALL:
                                                 .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11      ; 0578
5B 00000000G      00   9E 00002                  MOVAB   NMA$SEARCHFLD, R11                        ;
5A 00000000G      00   9E 00009                  MOVAB   NML$AB_ENTITYDATA+49, R10                 ;
59 00000000G      00   9E 00010                  MOVAB   NML$AB_MSGBLOCK, R9                       ;
5E                3C   C2 00017                  SUBL2   #60, SP                                   ;
58                6A   9A 0001A                  MOVZBL  NML$AB_ENTITYDATA+49, DB                  ; 0611
57          01    AA   D0 0001D                  MOVL    NML$AB_ENTITYDATA+50, SRCHKEY1            ; 0612
56          05    AA   D0 00021                  MOVL    NML$AB_ENTITYDATA+54, SRCHKEY2            ; 0613
55          FB    AA   9A 00025                  MOVZBL  NML$AB_ENTITYDATA+44, FID                 ; 0614
53          FE    AA   3C 00029                  MOVZWL  NML$AB_ENTITYDATA+47, OWNER               ; 0615
54          23    AA   D0 0002D                  MOVL    NML$AB_ENTITYDATA+84, TABDES              ; 0616
                  7E   7C 00031                  CLRQ    -(SP)                                     ; 0628
                  7E   D4 00033                  CLRL    -(SP)                                     ;
            04    AC   9F 00035                  PUSHAB  SNK                                       ;
                  01   DD 00038                  PUSHL   #1                                        ;
```

```
                                  02  DD 0003A          PUSHL   #2                                       :
             00000000V  00        06  FB 0003C          CALLS   #6, NML_SETENTITY                        :
                        01        50  E8 00043          BLBS    R0, 1$                                   :
                                      04 00046          RET                                              :
                        52        01  D0 00047 1$:      MOVL    #1, STATUS                               : 0635
                           04     AE  D4 0004A          CLRL    KEY                                      : 0636
                                  5E  DD 0004D          PUSHL   SP                                       : 0637
             00000000G  00        01  FB 0004F          CALLS   #1, NML$GETEXEADR                        :
                           1C     AE  9F 00056 2$:      PUSHAB  RECDSC                                   : 0645
                                  7E  7C 00059          CLRQ    -(SP)                                    :
                                  7E  7C 0005B          CLRQ    -(SP)                                    :
                                  7E  D4 0005D          CLRL    -(SP)                                    :
                                  53  DD 0005F          PUSHL   OWNER                                    : 0648
                           20     AE  9F 00061          PUSHAB  KEY                                      : 0645
                   00000000G      00  9F 00064          PUSHAB  NML$GQ_RECBFDSC                          :
                                  55  DD 0006A          PUSHL   FID                                      :
             00000000G  00        0A  FB 0006C          CALLS   #10, NML$MATCHRECORD                     :
                        03        50  E8 00073          BLBS    R0, 3$                                   :
                            00C7  31  00076             BRW     8$                                       :
                           08     AE  D4 00079 3$:      CLRL    SINK_NODE_PTR                            : 0657
                           08     AE  9F 0007C          PUSHAB  SINK_NODE_PTR                            : 0658
                           10     AE  9F 0007F          PUSHAB  SINK_NODE_LEN                            :
                                  53  DD 00082          PUSHL   OWNER                                    :
                           28     AE  9F 00084          PUSHAB  RECDSC                                   :
                        6B        04  FB 00087          CALLS   #4, NMA$SEARCHFLD                        :
                        52        50  D0 0008A          MOVL    R0, STATUS                               :
                        13        52  E8 0008D          BLBS    STATUS, 4$                               : 0663
                        69        06  D0 00090          MOVL    #6, NML$AB_MSGBLOCK                      : 0665
                  04    A9        0E  8E 00093          MNEGB   #14, NML$AB_MSGBLOCK+4                   : 0666
                           08     A9  B4 00097          CLRW    NML$AB_MSGBLOCK+8                        : 0667
          0C    A9 00000000G      8F  D0 0009A          MOVL    #NML$_NOSNKNOD, NML$AB_MSGBLOCK+12       : 0668
                                      04 000A2          RET                                              : 0664
                           18     AE  D4 000A3 4$:      CLRL    SETDSC+4                                 : 0675
                           18     AE  9F 000A6          PUSHAB  SETDSC+4                                 : 0679
                           18     AE  9F 000A9          PUSHAB  SETDSC                                   : 0678
                        7E        C9  8F  9A 000AC      MOVZBL  #201, -(SP)                              : 0676
                           28     AE  9F 000B0          PUSHAB  RECDSC                                   :
                        6B        04  FB 000B3          CALLS   #4, NMA$SEARCHFLD                        :
                        52        50  D0 000B6          MOVL    R0, STATUS                               :
                        7B        52  E9 000B9          BLBC    STATUS, 6$                               : 0680
                  10    AE    08  BE  3C 000BC          MOVZWL  @SINK_NODE_PTR, SINK_NODE_ADD            : 0691
                                  04  12 000C1          BNEQ    5$                                       : 0692
                  10    AE        6E  D0 000C3          MOVL    EXEC_ADDRESS, SINK_NODE_ADD              : 0693
                           10     AE  DD 000C7 5$:      PUSHL   SINK_NODE_ADD                            : 0694
                           04     AC  DD 000CA          PUSHL   SNK                                      :
                           1C     AE  9F 000CD          PUSHAB  SETDSC                                   :
                           28     AE  9F 000D0          PUSHAB  RECDSC                                   :
             00000000V  00        04  FB 000D3          CALLS   #4, NML_ADDALLFIL                        :
                        5D        50  E9 000DA          BLBC    R0, 7$                                   :
                                  54  DD 000DD          PUSHL   1ABDES                                   : 0696
                           20     AE  9F 000DF          PUSHAB  RECDSC                                   :
             00000000G  00        02  FB 000E2          CALLS   #2, NML$BLDALLDES                        :
                           24     AE  9F 000E9          PUSHAB  QBFDSC                                   : 0697
                   00000000G      00  9F 000EC          PUSHAB  NML$GQ_QIOBFDSC                          :
                           34     AE  9F 000F2          PUSHAB  P2DSC                                    :
                   00000000'      00  9F 000F5          PUSHAB  NML$Q_P2BFDSC                            :
                           44     AE  9F 000FB          PUSHAB  NFBDSC                                   :
```

```
          00000000'  00  9F 000FE          PUSHAB  NML$Q_NFBBFDSC               ;
                         7E  D4 00104          CLRL    -(SP)                        ;
               7E        01  CE 00106          MNEGL   #1, -(SP)                    ; 0699
                         56  DD 00109          PUSHL   SRCHKEY2                     ;
                    34   AE  9F 0010B          PUSHAB  SINK_NODE_ADD                ; 0697
                         7E  D4 0010E          CLRL    -(SP)                        ;
                         57  DD 00110          PUSHL   SRCHKEY1                     ; 0698
                         58  DD 00112          PUSHL   DB                           ; 0697
                         23  DD 00114          PUSHL   #35                          ;
      00000000G  00          0E  FB 00116          CALLS   #14, NML$BLDSETQBF           ;
                    24   AE  9F 0011D          PUSHAB  QBFDSC                       ; 0706
                         7E  D4 00120          CLRL    -(SP)                        ;
                    34   AE  9F 00122          PUSHAB  P2DSC                        ;
                    40   AE  9F 00125          PUSHAB  NFBDSC                       ;
      00000000G  00          04  FB 00128          CALLS   #4, NML$NETQIO               ;
                 52          50  D0 0012F          MOVL    R0, STATUS                   ;
                 05          52  E8 00132          BLBS    STATUS, 7$                   ; 0707
                             09  11 00135          BRB     8$                           ; 0708
                 52          01  D0 00137 6$:      MOVL    #1, STATUS                   ; 0712
                    04   AE  D6 0013A 7$:      INCL    KEY                          ; 0714
                      FF16   31 0013D          BRW     2$                           ; 0645
                 06          52  E9 00140 8$:      BLBC    STATUS, 9$                   ; 0720
                             69  D4 00143          CLRL    NML$AB_MSGBLOCK              ; 0722
            04   A9          01  90 00145          MOVB    #1, NML$AB_MSGBLOCK+4        ; 0723
                             04 00149 9$:      RET                                  ; 0726

; Routine Size:  330 bytes,    Routine Base:  $CODE$ + 02D1
```

```
%SBTTL 'NML_ADDALLFIL  Add all logging filters'
ROUTINE NML_ADDALLFIL (RECDSC, SETDSC, SNK, SNKADR) =

!++
! FUNCTIONAL DESCRIPTION:
!
!       This routine adds all permanent filters to the volatile filters
!       for the specified sink type and sink node.
!
! FORMAL PARAMETERS:
!
!       RECDSC          Descriptor of permanent data base record.
!       SETDSC          Descriptor of existing event parameter.
!       SNK             Logging sink type.
!       SNKADR          Sink node address.
!
! IMPLICIT INPUTS:
!
!       NML$GB_EVTMSKTYP .
!
! IMPLICIT OUTPUTS:
!
!       NONE
!
! ROUTINE VALUE:
! COMPLETION CODES:
!
!       NONE
!
! SIDE EFFECTS:
!
!       NONE
!
!--

    BEGIN

    MAP
        RECDSC : REF DESCRIPTOR,
        SETDSC : REF DESCRIPTOR,
        SNKADR : WORD;

    LOCAL
        DUMDSC : REF DESCRIPTOR,
        ENTDSC : DESCRIPTOR,
        EVEDSC : DESCRIPTOR,
        EVTPTR : REF BBLOCK,
        NFBDSC : REF DESCRIPTOR,
        P2DSC  : DESCRIPTOR,
        SRCPTR : REF BBLOCK,
        STATUS;

!
! If there is an entry in the volatile data base then update it with the
! permanent data base information.  If no entry exists then create one.
!
    !
```

```
	! Get the sink's entry in the volatile data base, if there is one.
	!
	NML$GETINFTABS (NML$C_LOGGING, NML$C_EVENTS, NFBDSC, DUMDSC, 0);
	NML$BLDP2 (0, .SNKADR, -1, 0, NML$Q_P2BFDSC, P2DSC);
	STATUS = NML$NETQIO (.NFBDSC, P2DSC, 0, NML$GQ_QIOBFDSC);

	IF .STATUS
	THEN
	    BEGIN
	    !
	    ! Convert the volatile data base information into permanent data
	    ! base format.  The filters from the permanent data base will be
	    ! added to this.
	    !
	    EVEDSC [DSC$W_LENGTH] = .(.NML$GQ_QIOBFDSC [DSC$A_POINTER])<0,16>;
	    EVEDSC [DSC$A_POINTER] = NML$T_PRMBUFFER;

	    CH$MOVE (.EVEDSC [DSC$W_LENGTH],
		     .NML$GQ_QIOBFDSC [DSC$A_POINTER] + 2,
		     NML$T_PRMBUFFER);
	    END
	ELSE
	    BEGIN

	    IF .STATUS EQL NML$_STS_CMP
	    THEN
		BEGIN
		!
		! Since there was no entry for the sink in the volatile data base,
		! create a null permanent data base entry to which the filters
		! from the permanent data base will be added.
		!
		EVEDSC [DSC$W_LENGTH] = 0;
		EVEDSC [DSC$A_POINTER] = NML$T_PRMBUFFER;

		END
	    ELSE
		RETURN .STATUS;

	    END;
!
! Now go through the permenent data base record for this sink node and
! add the sink's filters to the volatile data base filters (which are now
! in permanent data base format).
!
	STATUS = NML$_STS_PMS;
	SRCPTR = 0;
	WHILE NML$GETNXTSNK (.SETDSC, .SNK, SRCPTR) DO
	    BEGIN
	    STATUS = NML$_STS_SUC;		! At least one new filter was found for
						! sink.
!
! Each event source type is handled differently.
!
	    SELECTONEU .SRCPTR [SRC$B_SRCTYPE] OF
		SET
```

```
3              [NMA$C_ENT_CIR, NMA$C_ENT_LIN,         ! Circuit, Line or
3               NMA$C_ENT_MOD]:                       !          Module source
4                  BEGIN
4                  ENTDSC [DSC$W_LENGTH] = .SRCPTR [SRC$B_IDLENGTH];
4                  ENTDSC [DSC$A_POINTER] = SRCPTR [SRC$T_ID];
3                  END;
3
3              [NMA$C_ENT_NOD]:                       ! Node source
4                  BEGIN
4                  ENTDSC [DSC$W_LENGTH] = 0;
4                  ENTDSC [DSC$A_POINTER] = .SRCPTR [SRC$W_NODADR];
3                  END;
3
3              [OTHERWISE]:                           ! Null source
4                  BEGIN
4                  ENTDSC [DSC$W_LENGTH] = 0;
4                  ENTDSC [DSC$A_POINTER] = 0;
3                  END;
3              TES;
3
3          NML$GB_EVTMSKTYP = 0;                      ! Set to add individual filters
3          EVTPTR = 0;
3          WHILE NML$GETNXTEVT (.SRCPTR, EVTPTR) DO
4              BEGIN
4  !
4  ! Add all the SET filters.
4  !
4              STATUS = NML$ADDFILTERS (1,
4                                       NML$Q_PRMBFDSC,
4                                       .SNK,
4                                       .SRCPTR [SRC$B_SRCTYPE],
4                                       ENTDSC,
4                                       .EVTPTR [EVT$W_CLASS],
4                                       8,
4                                       EVTPTR [EVT$Q_LOGMSK],
4                                       EVEDSC);
4  !
4  ! If that was successful then add all the CLEAR filters.
4  !
4              IF .STATUS
4              THEN
4                  STATUS = NML$ADDFILTERS (0,
4                                           NML$Q_PRMBFDSC,
4                                           .SNK,
4                                           .SRCPTR [SRC$B_SRCTYPE],
4                                           ENTDSC,
4                                           .EVTPTR [EVT$W_CLASS],
4                                           8,
4                                           EVTPTR [EVT$Q_FILTERMSK],
4                                           EVEDSC);
4              IF NOT .STATUS
4              THEN
5                  BEGIN
5
5                  NML$AB_MSGBLOCK [MSB$L_FLAGS] = MSB$M_MSG_FLD;
5                  NML$AB_MSGBLOCK [MSB$B_CODE]  = NMA$C_STS_MPR;
5                  NML$AB_MSGBLOCK [MSB$L_TEXT]  = NML$_BADEVTUPD;
```

```
5                   RETURN .STATUS;
5
4                   END;
3               END;
2           END;
2   !
2   ! If the permanent data base filter record contained at least one event
2   ! filter for the sink,
2   ! store the modified filters in the temporary data base record.  If there
2   ! were no filters for the sink, no update will be done to the volatile
2   ! data base.
2   !
2       IF .STATUS THEN
2           STATUS = NML$SAVEVENTS (.NML$GQ_RECBFDSC [DSC$W_LENGTH],
2                                   .EVEDSC [DSC$W_LENGTH],
2                                   .EVEDSC [DSC$A_POINTER],
2                                   .RECDSC);
2       RETURN .STATUS
1       END;                                    ! End of NML_ADDALLFIL


                                  07FC 00000 NML_ADDALLFIL:
                                                   .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10          ; 0728
            5A 00000000G   00  9E 00002            MOVAB   NML$ADDFILTERS, R10                       ;
            59 00000000G   00  9E 00009            MOVAB   NML$AB_MSGBLOCK, R9                       ;
            58 00000000'   00  9E 00010            MOVAB   NML$T_PRMBUFFER, R8                       ;
            57 00000000'   00  9E 00017            MOVAB   NML$Q_PRMBFDSC, R7                        ;
            5E             28  C2 0001E            SUBL2   #40, SP                                   ;
                           7E  D4 00021            CLRL    -(SP)                                     ; 0786
                      04   AE  9F 00023            PUSHAB  DUMDSC                                    ;
                      0C   AE  9F 00026            PUSHAB  NFBDSC                                    ;
                           04  DD 00029            PUSHL   #4                                        ;
                           01  DD 0002B            PUSHL   #1                                        ;
      00000000G   00       05  FB 0002D            CALLS   #5, NML$GETINFTABS                        ;
                      10   AE  9F 00034            PUSHAB  P2DSC                                     ; 0787
                      10   A7  9F 00037            PUSHAB  NML$Q_P2BFDSC                             ;
                           7E  D4 0003A            CLRL    -(SP)                                     ;
            7E             01  CE 0003C            MNEGL   #1, -(SP)                                 ;
            7E        10   AC  3C 0003F            MOVZWL  SNKADR, -(SP)                             ;
                           7E  D4 00043            CLRL    -(SP)                                     ;
      00000000G   00       06  FB 00045            CALLS   #6, NML$BLDP2                             ;
                  00000000G 00 9F 0004C            PUSHAB  NML$GQ_QIOBFDSC                           ; 0788
                           7E  D4 00052            CLRL    -(SP)                                     ;
                      18   AE  9F 00054            PUSHAB  P2DSC                                     ;
                      10   AE  DD 00057            PUSHL   NFBDSC                                    ;
      00000000G   00       04  FB 0005A            CALLS   #4, NML$NETQIO                            ;
            56             50  D0 00061            MOVL    R0, STATUS                                ;
            17             56  E9 00064            BLBC    STATUS, 1$                                ; 0790
            50 00000000G   00  D0 00067            MOVL    NML$GQ_QIOBFDSC+4, R0                     ; 0798
         18 AE             60  B0 0006E            MOVW    (R0), EVEDSC                              ;
         1C AE             68  9E 00072            MOVAB   NML$T_PRMBUFFER, EVEDSC+4                 ; 0799
   68    02 A0        18   AE  28 00076            MOVC3   EVEDSC, 2(R0), NML$T_PRMBUFFER            ; 0801
            13             11     0007C            BRB     3$                                        ; 0790
      FFFFFFF0    8F       56  D1 0007E 1$:        CMPL    STATUS, #-16                              ; 0808
```

```
                                          03  13 00085       BEQL    2$                              ;
                                        00EC  31 00087       BRW     13$                             ;
                               18         AE  B4 0008A 2$:   CLRW    EVEDSC                          ; 0816
               1C      AE                 68  9E 0008D       MOVAB   NML$T_PRMBUFFER, EVEDSC+4       ; 0817
                       56                 3A  CE 00091 3$:   MNEGL   #58, STATUS                     ; 0829
                       08                 AE  D4 00094       CLRL    SRCPTR                          ; 0830
                       08                 AE  9F 00097 4$:   PUSHAB  SRCPTR                          ; 0831
                       7E          08     AC  7D 0009A       MOVQ    SETDSC, -(SP)                   ;
          00000000G    00                 03  FB 0009E       CALLS   #3, NML$GETNXTSNK               ;
                       03                 50  E8 000A5       BLBS    R0, 5$                          ;
                                        00AD  31 000A8       BRW     12$                             ;
                       56                 01  D0 000AB 5$:   MOVL    #1, STATUS                      ; 0833
                       52          08     AE  D0 000AE       MOVL    SRCPTR, R2                      ; 0838
                       50          03     A2  9A 000B2       MOVZBL  3(R2), R0                       ;
                       01                 50  91 000B6       CMPB    R0, #1                          ; 0841
                                          0A  13 000B9       BEQL    6$                              ;
                       03                 50  91 000BB       CMPB    R0, #3                          ;
                                          11  1F 000BE       BLSSU   7$                              ;
                       04                 50  91 000C0       CMPB    R0, #4                          ;
                                          0C  1A 000C3       BGTRU   7$                              ;
               20      AE          04     A2  9B 000C5 6$:   MOVZBW  4(R2), ENTDSC                   ; 0844
               24      AE          05     A2  9E 000CA       MOVAB   5(R2), ENTDSC+4                 ; 0845
                                          14  11 000CF       BRB     9$                              ; 0838
                                          50  D5 000D1 7$:   TSTL    R0                              ; 0848
                                          0A  12 000D3       BNEQ    8$                              ;
                                   20     AE  B4 000D5       CLRW    ENTDSC                          ; 0850
               24      AE          04     A2  3C 000D8       MOVZWL  4(R2), ENTDSC+4                 ; 0851
                                          06  11 000DD       BRB     9$                              ; 0838
                                   20     AE  B4 000DF 8$:   CLRW    ENTDSC                          ; 0856
                                   24     AE  D4 000E2       CLRL    ENTDSC+4                        ; 0857
                            00000000G     00  94 000E5 9$:   CLRB    NML$GB_EVTMSKTYP                ; 0861
                                   0C     AE  D4 000EB       CLRL    EVTPTR                          ; 0862
                                   0C     AE  9F 000EE 10$:  PUSHAB  EVTPTR                          ; 0863
                                          52  DD 000F1       PUSHL   R2                              ;
          00000000G    00                 02  FB 000F3       CALLS   #2, NML$GETNXTEVT               ;
                       9A                 50  E9 000FA       BLBC    R0, 4$                          ;
                                   18     AE  9F 000FD       PUSHAB  EVEDSC                          ; 0875
       7E      10      AE                 04  C1 00100       ADDL3   #4, EVTPTR, -(SP)               ;
                                          08  DD 00105       PUSHL   #8                              ;
                       7E          18     BE  3C 00107       MOVZWL  @EVTPTR, -(SP)                  ;
                                   30     AE  9F 0010B       PUSHAB  ENTDSC                          ; 0868
                       7E          03     A2  9A 0010E       MOVZBL  3(R2), -(SP)                    ; 0875
                                   0C     AC  DD 00112       PUSHL   SNK                             ;
                                          57  DD 00115       PUSHL   R7                              ; 0868
                                          01  DD 00117       PUSHL   #1                              ; 0875
                       6A                 09  FB 00119       CALLS   #9, NML$ADDFILTERS              ;
                       56                 50  D0 0011C       MOVL    R0, STATUS                      ;
                       25                 56  E9 0011F       BLBC    STATUS, 11$                     ; 0880
                                   18     AE  9F 00122       PUSHAB  EVEDSC                          ; 0889
       7E      10      AE                 0C  C1 00125       ADDL3   #12, EVTPTR, -(SP)              ;
                                          08  DD 0012A       PUSHL   #8                              ;
                       7E          18     BE  3C 0012C       MOVZWL  @EVTPTR, -(SP)                  ;
                                   30     AE  9F 00130       PUSHAB  ENTDSC                          ; 0882
                       7E          03     A2  9A 00133       MOVZBL  3(R2), -(SP)                    ; 0889
                                   0C     AC  DD 00137       PUSHL   SNK                             ;
                                          57  DD 0013A       PUSHL   R7                              ; 0882
                                          7E  D4 0013C       CLRL    -(SP)                           ; 0889
```

```
                6A             09 FB 0013E       CALLS   #9, NML$ADDFILTERS                        ;
                56             50 D0 00141       MOVL    R0, STATUS                                ;
                A7             56 E8 00144       BLBS    STATUS, 10$                               ; 0891
                69             04 D0 00147 11$:  MOVL    #4, NML$AB_MSGBLOCK                       ; 0895
          04    A9             05 8E 0014A       MNEGB   #5, NML$AB_MSGBLOCK+4                     ; 0896
          0C    A9 00000000G   8F D0 0014E       MOVL    #NML$_BADEVTUPD, NML$AB_MSGBLOCK+12       ; 0897
                               1E 11 00156       BRB     13$                                       ; 0898
                1B             56 E9 00158 12$:  BLBC    STATUS, 13$                               ; 0910
                         04    AC DD 0015B       PUSHL   RECDSC                                    ; 0914
                         20    AE DD 0015E       PUSHL   EVEDSC+4                                  ; 0913
                7E       20    AE 3C 00161       MOVZWL  EVEDSC, -(SP)                             ; 0912
                7E 00000000G   00 3C 00165       MOVZWL  NML$GQ_RECBFDSC, -(SP)                    ; 0911
       00000000G  00           04 FB 0016C       CALLS   #4, NML$SAVEVENTS                         ;
                56             50 D0 00173       MOVL    R0, STATUS                                ;
                50             56 D0 00176 13$:  MOVL    STATUS, R0                                ; 0915
                               04 00179          RET                                               ; 0916

; Routine Size:  378 bytes,    Routine Base:  $CODE$ + 041B
```

```
1 %SBTTL 'NML_SETENTITY  Set volatile database parameters'
1 ROUTINE NML_SETENTITY (ENTITY, ENTITY_LEN, ENTITY_ADR,
1                        QUAL_PST, QUAL_LEN, QUAL_ADR) =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine adds the specified parameters to the volatile data base
1 !       entry for the specified component.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       ENTITY          Entity type code.
1 !       ENTITY_LEN      Byte count of entity id string.
1 !       ENTITY_ADR      Address of entity id string.
1 !       QUAL_PST        Qualifier's Parameter Semantic Table entry address.
1 !       QUAL_LEN        Byte count of qualifier id string
1 !       QUAL_ADR        Address of qualifier id string
1 !
1 !
1 ! ROUTINE VALUE:
1 ! COMPLETION CODES:
1 !
1 !       The translated status of the SET QIO is returned.
1 !--
1
2 BEGIN
2
2 LOCAL
2     db,                                       ! Database ID
2     srchkey1,                                 ! Search key one ID
2     srchkey2,                                 ! Search key two ID
2     detail : WORD,                            ! NICE error detail code
2     fid,                                      ! File id code
2     key,
2     owner,                                    ! Search key
2     nfbdsc : DESCRIPTOR,                      ! NFB buffer descriptor
2     p2dsc  : DESCRIPTOR,                      ! QIO P2 buffer descriptor
2     qbfdsc : DESCRIPTOR,                      ! QIO P4 buffer descriptor
2     recdsc : DESCRIPTOR,                      ! Record descriptor
2     status,
2     dummy,
2     tabdes : REF DESCRIPTOR;
2
2 status = nml$_sts_suc;
2 !
2 ! Get entity information.
2 !
2 db       = .nml$ab_entitydata [.entity, eit$b_database]; ! Database ID
2 srchkey1 = .nml$ab_entitydata [.entity, eit$l_srch_id1]; ! Search key one
2 srchkey2 = .nml$ab_entitydata [.entity, eit$l_srch_id2]; ! Search key two
2 detail   = .nml$ab_entitydata [.entity, eit$w_detail]; ! NICE error detail
2 fid      = .nml$ab_entitydata [.entity, eit$b_fileid]; ! File id code
2 owner    = .nml$ab_entitydata [.entity, eit$w_key];    ! Search key
2 tabdes   = .nml$ab_entitydata [.entity, eit$a_alltab]; ! Table for SET ALL
2 !
2 ! If no parameters were specified in the NICE message then all parameters
```

```
2 ! from the permanent data base should be used.
2 !
2 IF .nml$gl_prs_flgs [nml$v_prs_all] THEN
3     BEGIN
3     !
3     ! Find a permanent data base entry for this entity.  If one is not
3     ! found then create a descriptor for a null record.
3     !
3     IF .fid NEQ nma$c_opn_node THEN
4         BEGIN
4         key = 0;
4         IF NOT nml$matchrecord (.fid,
4                                 nml$gq_recbfdsc,
4                                 key,
4                                 .owner, .entity_len, .entity_adr,
4                                 .qual_pst, .qual_len, .qual_adr,
4                                 recdsc) THEN
5             BEGIN
5             !
5             ! If no data base record was found for a logging sink then
5             ! just return.  This check avoids extraneous error messages
5             ! as the result of the NCP SET KNOWN LOGGING ALL command.
5             !
5             IF .entity EQLU nml$c_sink THEN
5                 RETURN nml$_sts_suc;
5             !
5             ! NICE does not differentiate between
5             !               SET entity abc
5             !                   and
5             !               SET entity abc ALL
5             ! However, some entities require at least one other parameter
5             ! (objects must have a number).  So, for entities with are not in
5             ! the perm database, assume that it's simply a SET command unless
5             ! the entity has a required parameter.
5             !
5             IF .entity EQLU nml$c_object THEN
5                 status = nml$_sts_cmp;
5             recdsc [dsc$w_length]  = 0;
5             recdsc [dsc$a_pointer] = .nml$gq_recbfdsc [dsc$a_pointer] +
5                                                 nml$k_perm_keys_len;
4             END;
4         !
4         ! If entity is X25-Protocol network, no entity ID was supplied with the
4         ! NICE command.  So, get the network name from the permanent data
4         ! base record, and set it up as the entity ID for the QIO.
4         !
4         IF .entity EQLU nml$c_prot_net THEN
4             status = nma$searchfld (recdsc, .owner, entity_len, entity_adr);
4         END
3     ELSE
3         !
3         ! The ISAM key structure for the node permanent database is different
3         ! so it will be faster (it can be much bigger than the others).
3         !
3         BEGIN
4         IF NOT nml$readrecord (.fid, owner, entity_len,
4                                nml$gq_recbfdsc, recdsc, dummy) THEN
```

```
4             status = nml$_sts_cmp;
3         END;
3     !
3     ! Using the SET ALL parameter list (in NMLDAT) for this entity, build
3     ! the a parameter descriptor list in the form generated by the NPARSE
3     ! tables.
3     !
3     IF .status THEN
3         nml$bldalldes (recdsc, .tabdes); ! Build descriptors
2     END;
2
2 IF .status EQL nml$_sts_cmp THEN
3     BEGIN
3     nml$ab_msgblock [msb$l_flags]  = msb$m_det_fld;
3     nml$ab_msgblock [msb$w_detail] = .detail;
3     nml$ab_msgblock [msb$b_code] = nma$c_sts_cmp;
3     RETURN nml$_sts_cmp;
2     END;
2 IF .status THEN
3     BEGIN
3     !
3     ! Add the parameters to volatile data base entry.
3     !
3     nml$bldsetqbf (nfb$c_fc_set, .db,
3                   .srchkey1, .entity_len, .entity_adr,
3                   .srchkey2, .qual_len, .qual_adr,
3                   nml$q_nfbbfdsc, nfbdsc,
3                   nml$q_p2bfdsc, p2dsc,
3                   nml$gq_qiobfdsc, qbfdsc);
3     status = nml$netqio (nfbdsc, p2dsc, 0, qbfdsc);
3     IF .status THEN
4         BEGIN
4         nml$ab_msgblock [msb$l_flags] = 0;
4         nml$ab_msgblock [msb$b_code] = nma$c_sts_suc;
3         END;
2     END;
2 RETURN .status
1 END;                                  ! End of NML_SETENTITY
```

```
                                  07FC 00000 NML_SETENTITY:
                                             .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10          ; 0918
             5A 00000000G   00  9E 00002     MOVAB   NML$GQ_RECBFDSC, R10                      ;
             59 00000000G   00  9E 00009     MOVAB   NML$AB_MSGBLOCK, R9                       ;
             58 00000000G   00  9E 00010     MOVAB   NML$AB_ENTITYDATA+5, R8                   ;
             5E             2C  C2 00017     SUBL2   #44, SP                                   ;
             52             01  D0 0001A     MOVL    #1, STATUS                                ; 0961
  50     04  AC             2C  C5 0001D     MULL3   #44, ENTITY, R0                           ; 0965
             57           6840  9A 00022     MOVZBL  NML$AB_ENTITYDATA+5[R0], DB               ;
                       01 A840  9F 00026     PUSHAB  NML$AB_ENTITYDATA+6[R0]                   ; 0966
             56             9E  D0 0002A     MOVL    @(SP)+, SRCHKEY1                          ;
                       05 A840  9F 0002D     PUSHAB  NML$AB_ENTITYDATA+10[R0]                  ; 0967
             55             9E  D0 00031     MOVL    @(SP)+, SRCHKEY2                          ;
                       FC A840  9F 00034     PUSHAB  NML$AB_ENTITYDATA+1[R0]                   ; 0968
```

```
                   54          9E  B0 00038      MOVW    @(SP)+, DETAIL                          ;
                   51    FB A840  9A 0003B      MOVZBL  NML$AB_ENTITYDATA[R0], FID              ; 0969
                         FE A840  9F 00040      PUSHAB  NML$AB_ENTITYDATA+3[R0]                 ; 0970
              08   AE          9E  3C 00044      MOVZWL  @(SP)+, OWNER                           ;
                         23 A840  9F 00048      PUSHAB  NML$AB_ENTITYDATA+40[R0]                ; 0971
                   53          9E  D0 0004C      MOVL    @(SP)+, TABDES                          ;
    03 00000000G   00          01  E0 0004F      BBS     #1, NML$GL_PRS_FLGS, 1$                 ; 0976
                             008E  31 00057      BRW     7$                                      ;
                               51  D5 0005A 1$:  TSTL    FID                                     ; 0982
                               5E  13 0005C      BEQL    5$                                      ;
                               6E  D4 0005E      CLRL    KEY                                     ; 0984
                         0C    AE  9F 00060      PUSHAB  RECDSC                                  ; 0985
                   7E    14    AC  7D 00063      MOVQ    QUAL_LEN, -(SP)                         ; 0989
                   7E    0C    AC  7D 00067      MOVQ    ENTITY_ADR, -(SP)                       ; 0988
                         08    AC  DD 0006B      PUSHL   ENTITY_LEN                              ;
                         20    AE  DD 0006E      PUSHL   OWNER                                   ;
                         1C    AE  9F 00071      PUSHAB  KEY                                     ; 0985
                             0402  8F  BB 00074  PUSHR   #^M<R1,R10>                             ;
       00000000G   00          0A  FB 00078      CALLS   #10, NML$MATCHRECORD                    ;
                   1C          50  E8 0007F      BLBS    R0, 4$                                  ;
                   02    04    AC  D1 00082      CMPL    ENTITY, #2                              ; 0997
                               04  12 00086      BNEQ    2$                                      ;
                   50          01  D0 00088      MOVL    #1, R0                                  ; 0998
                                   04 0008B      RET                                             ;
                   08    04    AC  D1 0008C 2$:  CMPL    ENTITY, #8                              ; 1009
                               03  12 00090      BNEQ    3$                                      ;
                   52          10  CE 00092      MNEGL   #16, STATUS                             ; 1010
                         0C    AE  B4 00095 3$:  CLRW    RECDSC                                  ; 1011
    10   AE        04   AA    02  C1 00098      ADDL3   #2, NML$GQ_RECBFDSC+4, RECDSC+4          ; 1012
                   0E    04    AC  D1 0009E 4$:  CMPL    ENTITY, #14                             ; 1020
                               35  12 000A2      BNEQ    6$                                      ;
                         0C    AC  9F 000A4      PUSHAB  ENTITY_ADR                              ; 1021
                         08    AC  9F 000A7      PUSHAB  ENTITY_LEN                              ;
                         10    AE  DD 000AA      PUSHL   OWNER                                   ;
                         18    AE  9F 000AD      PUSHAB  RECDSC                                  ;
       00000000G   00          04  FB 000B0      CALLS   #4, NMA$SEARCHFLD                       ;
                   52          50  D0 000B7      MOVL    R0, STATUS                              ;
                               1D  11 000BA      BRB     6$                                      ; 0982
                         04    AE  9F 000BC 5$:  PUSHAB  DUMMY                                   ; 1029
                         10    AE  9F 000BF      PUSHAB  RECDSC                                  ;
                               5A  DD 000C2      PUSHL   R10                                     ;
                         08    AC  9F 000C4      PUSHAB  ENTITY_LEN                              ;
                         18    AE  9F 000C7      PUSHAB  OWNER                                   ;
                               51  DD 000CA      PUSHL   FID                                     ;
       00000000G   00          06  FB 000CC      CALLS   #6, NML$READRECORD                      ;
                   03          50  E8 000D3      BLBS    R0, 6$                                  ;
                   52          10  CE 000D6      MNEGL   #16, STATUS                             ; 1031
                   0C          52  E9 000D9 6$:  BLBC    STATUS, 7$                              ; 1038
                               53  DD 000DC      PUSHL   TABDES                                  ; 1039
                         10    AE  9F 000DE      PUSHAB  RECDSC                                  ;
       00000000G   00          02  FB 000E1      CALLS   #2, NML$BLDALLDES                       ;
       FFFFFFF0    8F          52  D1 000E8 7$:  CMPL    STATUS, #-16                            ; 1042
                               0F  12 000EF      BNEQ    8$                                      ;
                   69          02  D0 000F1      MOVL    #2, NML$AB_MSGBLOCK                     ; 1044
              08   A9          54  B0 000F4      MOVW    DETAIL, NML$AB_MSGBLOCK+8               ; 1045
              04   A9          08  8E 000F8      MNEGB   #8, NML$AB_MSGBLOCK+4                   ; 1046
                   50          10  CE 000FC      MNEGL   #16, R0                                 ; 1047
```

```
                                       04 000FF          RET                                                   ;
               50                  52  E9 00100  8$:     BLBC     STATUS, 9$                                   ; 1049
                          14       AE  9F 00103          PUSHAB   QBFDSC                                       ; 1054
                   00000000G       00  9F 00106          PUSHAB   NML$GQ_QIOBFDSC                              ;
                          24       AE  9F 0010C          PUSHAB   P2DSC                                        ;
                   00000000'       00  9F 0010F          PUSHAB   NML$Q_P2BFDSC                                ;
                          34       AE  9F 00115          PUSHAB   NFBDSC                                       ;
                   00000000'       00  9F 00118          PUSHAB   NML$Q_NFBBFDSC                               ;
               7E         14       AC  7D 0011E          MOVQ     QUAL_LEN, -(SP)                              ; 1056
                                   55  DD 00122          PUSHL    SRCHKEY2                                     ;
               7E         08       AC  7D 00124          MOVQ     ENTITY_LEN, -(SP)                            ; 1055
                                   56  DD 00128          PUSHL    SRCHKEY1                                     ;
                                   57  DD 0012A          PUSHL    DB                                           ; 1054
                                   23  DD 0012C          PUSHL    #35                                          ;
       00000000G   00              0E  FB 0012E          CALLS    #14, NML$BLDSETQBF                           ;
                          14       AE  9F 00135          PUSHAB   QBFDSC                                       ; 1060
                                   7E  D4 00138          CLRL     -(SP)                                        ;
                          24       AE  9F 0013A          PUSHAB   P2DSC                                        ;
                          30       AE  9F 0013D          PUSHAB   NFBDSC                                       ;
       00000000G   00              04  FB 00140          CALLS    #4, NML$NETQIO                               ;
                   52              50  D0 00147          MOVL     R0, STATUS                                   ;
                   06              52  E9 0014A          BLBC     STATUS, 9$                                   ; 1061
                                   69  D4 0014D          CLRL     NML$AB_MSGBLOCK                              ; 1063
               04  A9              01  90 0014F          MOVB     #1, NML$AB_MSGBLOCK+4                        ; 1064
                   50              52  D0 00153  9$:     MOVL     STATUS, R0                                   ; 1067
                                       04 00156          RET                                                   ; 1068

; Routine Size:  343 bytes,     Routine Base:  $CODE$ + 0595
```

```
1 %SBTTL 'NMLSSETLINE  Set volatile line parameters'
1 GLOBAL ROUTINE NMLSSETLINE (ENTITY, LEN, ADR) : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine adds the specified parameters to the volatile data base
1 !       entry for the specified line.  This routine will eventually provide
1 !       support for wildcards in the line id.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       ENTITY          Entity type code.
1 !       LEN             Byte of line id string.
1 !       ADR             Address of line id string.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 ! COMPLETION CODES:
1 !
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !       NONE
1 !
1 !--
1
2    BEGIN
2 !
2 ! Call normal entity routine to handle line parameters.
2 !
2    NMLSSETENTITY (.ENTITY, .LEN, .ADR);
2
1    END;                              ! End of NMLSSETLINE


                          0000 00000     .ENTRY  NMLSSETLINE, Save nothing          ; 1070
          7E    08   AC   7D 00002       MOVQ    LEN, -(SP)                         ; 1108
                04   AC   DD 00006       PUSHL   ENTITY                             ;
     F906   CF        03  FB 00009       CALLS   #3, NMLSSETENTITY                  ;
                          04 0000E       RET                                        ; 1110

; Routine Size:  15 bytes,    Routine Base:  $CODE$ + 06EC
```

```
; 1121     1111  1 %SBTTL 'NML$SETEXECUTOR  Set executor parameters'
; 1122     1112  1 GLOBAL ROUTINE NML$SETEXECUTOR (ENTITY, DUM1, DUM2) : NOVALUE =
; 1123     1113  1
; 1124     1114  1 !++
; 1125     1115  1 ! FUNCTIONAL DESCRIPTION:
; 1126     1116  1 !
; 1127     1117  1 !       This routine adds the specified parameters to the volatile data base
; 1128     1118  1 !       entry for the executor node.
; 1129     1119  1 !
; 1130     1120  1 ! FORMAL PARAMETERS:
; 1131     1121  1 !
; 1132     1122  1 !       ENTITY          Entity code.
; 1133     1123  1 !       DUM1            Not used.
; 1134     1124  1 !       DUM2            Not used.
; 1135     1125  1 !
; 1136     1126  1 ! IMPLICIT INPUTS:
; 1137     1127  1 !
; 1138     1128  1 !       NONE
; 1139     1129  1 !
; 1140     1130  1 ! IMPLICIT OUTPUTS:
; 1141     1131  1 !
; 1142     1132  1 !       NONE
; 1143     1133  1 !
; 1144     1134  1 ! ROUTINE VALUE:
; 1145     1135  1 ! COMPLETION CODES:
; 1146     1136  1 !
; 1147     1137  1 !       NONE
; 1148     1138  1 !
; 1149     1139  1 ! SIDE EFFECTS:
; 1150     1140  1 !
; 1151     1141  1 !       NONE
; 1152     1142  1 !
; 1153     1143  1 !--
; 1154     1144  1
; 1155     1145  2   BEGIN
; 1156     1146  2 !
; 1157     1147  2 ! If no parameters were specified in the NICE message then all parameters
; 1158     1148  2 ! from the permanent data base should be used.
; 1159     1149  2 !
; 1160     1150  2   IF .NML$GL_PRS_FLGS [NML$V_PRS_ALL]
; 1161     1151  2   THEN
; 1162     1152  2       NML_SETEXEALL ()
; 1163     1153  2   ELSE
; 1164     1154  2       NML_SETEXE ();
; 1165     1155  2
; 1166     1156  1   END;                                    ! End of NML$SETEXECUTOR


                                      0000 00000           .ENTRY  NML$SETEXECUTOR, Save nothing        ; 1112
          08 0000000G   00         01  E1 00002           BBC     #1, NML$GL_PRS_FLGS, 1$               ; 1150
             0000000CV  00         00  FB 0000A           CALLS   #0, NML_SETEXEALL                     ; 1152
                                       04 00011           RET                                           ;
             00000000V  00         00  FB 00012 1$:       CALLS   #0, NML_SETEXE                        ; 1154
                                       04 00019           RET                                           ; 1156
```

```
; Routine Size:  26 bytes,    Routine Base:  $CODE$ + 06FB
```

```
1 %SBTTL 'NML$SETKNOWN  Set volatile entity parameters'
1 GLOBAL ROUTINE NML$SETKNOWN (ENTITY, ENTITY_LEN, ENTITY_ADR,
1                               DUM3, DUM4, DUM5) : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine sets the specified parameters for each of the components
1 !       of the given entity type.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       ENTITY          Entity type code.
1 !       ENTITY_LEN      Byte count of entity id string.
1 !       ENTITY_ADR      Address of entity id string.
1 !       DUM3 - DUM5     Not used.
1 !
1 !--
1
2 BEGIN
2 !
2 ! If this no parameters specified then add all permanent data base information
2 ! to volatile data base.
2 !
2 IF .NML$GL_PRS_FLGS [NML$V_PRS_ALL]
2 THEN
2       NML_SETKNOWNALL (.ENTITY, .ENTITY_LEN, .ENTITY_ADR)
2 ELSE
2       NML_SETKNOWN (.ENTITY);
2
1 END;                                   ! End of NML$SETKNOWN


                                0000 00000          .ENTRY  NML$SETKNOWN, Save nothing            ; 1158
      0F 00000000G  00            01  E1 00002          BBC     #1, NML$GL_PRS_FLGS, 1$               ; 1181
                    7E      08    AC  7D 0000A          MOVQ    ENTITY_LEN, -(SP)                     ; 1183
                            04    AC  DD 0000E          PUSHL   ENTITY                                ;
         00000000V  00            03  FB 00011          CALLS   #3, NML_SETKNOWNALL                   ;
                                      04 00018          RET                                           ;
                            04    AC  DD 00019 1$:      PUSHL   ENTITY                                ; 1185
         00000000V  00            01  FB 0001C          CALLS   #1, NML_SETKNOWN                      ;
                                      04 00023          RET                                           ; 1187

; Routine Size:  36 bytes,    Routine Base:  $CODE$ + 0715


1
```

```
1 %SBTTL 'NML$SETKNONODES  Set volatile node parameters'
1 GLOBAL ROUTINE NML$SETKNONODES (ENTITY, DUM1, DUM2) : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine sets the specified parameters for each of the nodes.
1 !
1 ! FORMAL PARAMETERS:
1 !
1 !       ENTITY          Entity type code.
1 !       DUM1            Not used.
1 !       DUM2            Not used.
1 !
1 ! IMPLICIT INPUTS:
1 !
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 ! COMPLETION CODES:
1 !
1 !       NONE
1 !
1 ! SIDE EFFECTS:
1 !
1 !       NONE
1 !
1 !--
1
2     BEGIN
2 !
2 ! If this no parameters specified then add all permanent data base information
2 ! to volatile data base.
2 !
2     IF .NML$GL_PRS_FLGS [NML$V_PRS_ALL]
2     THEN
3         BEGIN
3         NML_SETEXEALL ();
3         NML_SETKNOWNALL (NML$C_NODE);
3         NML_SETKNOWNALL (NML$C_LOOPNODE);
3
3         END
2     ELSE
3         BEGIN
3         NML_SETEXE ();
3         NML_SETKNOWN (NML$C_NODE);
3
2         END;
2
1     END;                                    ! End of NML$SETKNONODES
```

```
                                  0004 00000       .ENTRY  NML$SETKNONODES, Save R2            ; 1190
              52 00000000V  00      9E 00002       MOVAB   NML_SETKNOWNALL, R2                 ;
     12 00000000G  00             01 E1 00009       BBC     #1, NML$GL_PRS_FLGS, 1$             ; 1227
        00000000V  00             00 FB 00011       CALLS   #0, NML_SETEXEALL                   ; 1230
                                  03 DD 00018       PUSHL   #3                                  ; 1231
                   62             01 FB 0001A       CALLS   #1, NML_SETKNOWNALL                 ;
                                  05 DD 0001D       PUSHL   #5                                  ; 1232
                   62             01 FB 0001F       CALLS   #1, NML_SETKNOWNALL                 ;
                                     04 00022       RET                                         ; 1227
        00000000V  00             00 FB 00023 1$:   CALLS   #0, NML_SETEXE                      ; 1237
                                  03 DD 0002A       PUSHL   #3                                  ; 1238
        00000000V  00             01 FB 0002C       CALLS   #1, NML_SETKNOWN                    ;
                                     04 00033       RET                                         ; 1242

; Routine Size:  52 bytes,    Routine Base:  $CODE$ + 0739

; 1255          1243  1
```

```
%SBTTL 'NML_SETKNOWN  Set volatile entity parameters'
ROUTINE NML_SETKNOWN (ENTITY) : NOVALUE =

!++
! FUNCTIONAL DESCRIPTION:
!
!       This routine sets the specified parameters for each of the components
!       of the given entity type.
!
! INPUTS:
!
!       ENTITY          Entity type code.
!
!--

BEGIN

LOCAL
    BUFEND,
    DB,                             ! Database ID
    SRCHKEY1,                       ! Search key one ID
    SRCHKEY2,                       ! Search key two ID
    ENTITYADD,
    ENTITYLEN,
    LISDSC    : DESCRIPTOR,
    MSGSIZE,
    ENTIDPTR,
    PTR,
    NFBDSC    : DESCRIPTOR,                 ! NFB buffer descriptor
    P2DSC     : DESCRIPTOR,                 ! QIO P2 buffer descriptor
    QBFDSC    : DESCRIPTOR,                 ! QIO P2 buffer descriptor
    STATUS,
    STRTFLG;
!
! Build QIO buffer.
!
DB       = .NML$AB_ENTITYDATA [.ENTITY, EIT$B_DATABASE]; ! Database ID
SRCHKEY1 = .NML$AB_ENTITYDATA [.ENTITY, EIT$L_SRCH_ID1];  ! Search key one
SRCHKEY2 = .NML$AB_ENTITYDATA [.ENTITY, EIT$L_SRCH_ID2];  ! Search key two
!
! Process every entry in the data base.
!
! The NICE protocol specifies that, for multiple entity changes, one NICE
! response message must be returned to NCP for each entity changed.  Each
! message must contain the ID of the entity changed.  Therefore, the following
! loop issues one QIO to the ACP to get a bufferfull of entity IDs, and then
! issues one QIO for each entity in the buffer to perform the set.  This
! process continues until the ACP returns end-of-file to the request for more
! entity IDs.
!
STRTFLG = FALSE;
WHILE NML$GET_ENTITY_IDS (.ENTITY, NMA$C_ENT_KNO, 0, .STRTFLG, LISDSC) DO
    BEGIN

    STRTFLG = TRUE;

    BUFEND = .LISDSC [DSC$A_POINTER] + .LISDSC [DSC$W_LENGTH];
```

```
        PTR = .LISDSC [DSC$A_POINTER];

        WHILE .PTR LSSA .BUFEND DO
            BEGIN

            ENTIDPTR = NML$T_ENTBUFFER;
            NML$Q_ENTBFDSC [DSC$W_LENGTH] = NML$K_ENTBUFLEN;
            !
            ! Entity IDs for the ACP are different from those used by NICE.  The
            ! most common cause of this is that the ACP uses a word for the entity
            ! string length, and NICE uses a byte.  The following code sets up the
            ! two ID forms: the NICE entity ID for the response to NCP
            !               the ACP entity ID to be used in the P2 buffer of the
            !                         clear QIO.
            !
            SELECTONEU .ENTITY OF
                SET
                [NML$C_NODE]:
                    BEGIN

                    PTR = .PTR + 4; ! Skip loopnode flag.
                    ENTITYADD = .PTR;           ! Point to node address
                    ENTIDPTR = CH$MOVE (2, .PTR, .ENTIDPTR);
                    PTR = .PTR + 4;

                    ENTITYLEN = .(.PTR)<0,8>; ! Get name length
                    PTR = .PTR + 2;
                    CH$WCHAR_A (.ENTITYLEN, ENTIDPTR);
                    ENTIDPTR = CH$MOVE (.ENTITYLEN, .PTR, .ENTIDPTR);
                    PTR = .PTR + .ENTITYLEN;
                    ENTITYLEN = 2;  ! Get length of node address
                    END;

                !
                ! The entity ID for the following modules is the string
                ! identifying the module (e.g. 'X25-PROTOCOL') followed by the
                ! parameter id, parameter type, and string for the qualifier
                ! being set.
                !
                [NML$C_X25_ACCESS,
                NML$C_PROT_NET,
                NML$C_PROT_DTE,
                NML$C_PROT_GRP,
                NML$C_X25_SERV_DEST,
                NML$C_TRACEPNT,
                NML$C_X29_SERV_DEST]:
                    BEGIN
                    SELECTONEU .ENTITY OF
                        SET
                        [NML$C_X25_ACCESS]:
                            $MOVE_ASCIC ('X25-ACCESS', ENTIDPTR);
                        [NML$C_PROT_NET, NML$C_PROT_DTE, NML$C_PROT_GRP]:
                            $MOVE_ASCIC ('X25-PROTOCOL', ENTIDPTR);
                        [NML$C_X25_SERV_DEST]:
                            $MOVE_ASCIC ('X25-SERVER', ENTIDPTR);
                        [NML$C_TRACEPNT]:
                            $MOVE_ASCIC ('X25-TRACE', ENTIDPTR);
```

```
5                       [NML$C_X29_SERV_DEST]:
5                           $MOVE_ASCIC ('X29-SERVER', ENTIDPTR);
5                       TES;
5                   ENTIDPTR = CH$MOVE (2,
5                                   NML$AB_ENTITYDATA [.ENTITY, EIT$W_DETAIL],
5                                   .ENTIDPTR);
5                   CH$WCHAR_A (NMA$M_PTY_ASC, ENTIDPTR);
4                   END;
4
4               TES;
4
4           IF .ENTITY NEQ NML$C_NODE THEN
5                   BEGIN
5                   ENTITYLEN = .(.PTR)<0,16>;
5                   PTR = .PTR + 2;
5                   ENTITYADD = .PTR;
5                   CH$WCHAR_A (.ENTITYLEN, ENTIDPTR);
5                   ENTIDPTR = CH$MOVE (.ENTITYLEN,
5                                   .ENTITYADD,
5                                   .ENTIDPTR);
5                   PTR = .PTR + .ENTITYLEN;
4                   END;
4
4           NML$Q_ENTBFDSC [DSC$W_LENGTH] = .ENTIDPTR - NML$T_ENTBUFFER;
4           !
4           ! Add the parameters to volatile data base entry.
4           !
4           NML$BLDSETQBF (NFB$C_FC_SET, .DB,
4                           .SRCHKEY1, .ENTITYLEN, .ENTITYADD,
4                           .SRCHKEY2, -1, 0,
4                           NML$Q_NFBBFDSC, NFBDSC,
4                           NML$Q_P2BFDSC, P2DSC,
4                           NML$GQ_QIOBFDSC, QBFDSC);
4
4           IF NML$NETQIO (NFBDSC, P2DSC, 0, QBFDSC)
4           THEN
5               BEGIN
5
5               NML$AB_MSGBLOCK [MSB$L_FLAGS] = 0;
5               NML$AB_MSGBLOCK [MSB$B_CODE] = NMA$C_STS_SUC;
5
4               END;
4           !
4           ! Add the entity identification to the message.
4           !
4           NML$AB_MSGBLOCK [MSB$V_ENTD_FLD] = 1;
4           NML$AB_MSGBLOCK [MSB$A_ENTITY] = NML$Q_ENTBFDSC;
4           !
4           ! Build and send the response message.
4           !
4           NML$BLD_REPLY (NML$AB_MSGBLOCK, MSGSIZE);
4           NML$SEND (NML$AB_SNDBUFFER, .MSGSIZE);
3           END;
2       END;
2
1   END;                                    ! End of NML_SETKNOWN
```

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

            53 53 45 43 43 41 2D 35 32 58 0A 00020 P.AAE:           .ASCII  <10>\X25-ACCESS\                          ;
      4C 4F 43 4F 54 4F 52 50 2D 35 32 58 0C 0002B P.AAF:           .ASCII  <12>\X25-PROTOCOL\                        ;
            52 45 56 52 45 53 2D 35 32 58 0A 00038 P.AAG:           .ASCII  <10>\X25-SERVER\                          ;
               45 43 41 52 54 2D 35 32 58 09 00043 P.AAH:           .ASCII  <9>\X25-TRACE\                            ;
            52 45 56 52 45 53 2D 39 32 58 0A 0004D P.AAI:           .ASCII  <10>\X29-SERVER\                          ;


                                                                            .PSECT  $CODE$,NOWRT,2

                                          0FFC 00000 NML_SETKNOWN:
                                                                            .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11     ; 1245
                           5E            34  C2 00002                       SUBL2   #52, SP                                   ;
                           57     04     AC  D0 00005                       MOVL    ENTITY, R7                                ; 1280
                 56        57            2C  C5 00009                       MULL3   #44, R7, R6                               ;
                      0C   AE 00000000G00046 9A 0000D                       MOVZBL  NML$AB_ENTITYDATA+5[R6], DB               ;
                              00000000G00046 9F 00016                       PUSHAB  NML$AB_ENTITYDATA+6[R6]                   ; 1281
                      08   AE            9E  D0 0001D                       MOVL    @(SP)+, SRCHKEY1                          ;
                              00000000G00046 9F 00021                       PUSHAB  NML$AB_ENTITYDATA+10[R6]                  ; 1282
                      04   AE            9E  D0 00028                       MOVL    @(SP)+, SRCHKEY2                          ;
                                         5B  D4 0002C                       CLRL    STRTFLG                                   ; 1294
                                  2C     AE  9F 0002E 1$:                   PUSHAB  LISDSC                                    ; 1295
                                         5B  DD 00031                       PUSHL   STRTFLG                                   ;
                                         7E  D4 00033                       CLRL    -(SP)                                     ;
                           7E            01  CE 00035                       MNEGL   #1, -(SP)                                 ;
                                         57  DD 00038                       PUSHL   R7                                        ;
            00000000G      00            05  FB 0003A                       CALLS   #5, NML$GET_ENTITY_IDS                    ;
                           01            50  E8 00041                       BLBS    R0, 2$                                    ;
                                             04 00044                       RET                                               ;
                           5B            01  D0 00045 2$:                   MOVL    #1, STRTFLG                               ; 1298
                           50     2C     AE  3C 00048                       MOVZWL  LISDSC, R0                                ; 1300
                           6E     30   BE40  9E 0004C                       MOVAB   @LISDSC+4[R0], BUFEND                     ;
                           58     30     AE  D0 00051                       MOVL    LISDSC+4, PTR                             ; 1301
                           6E            58  D1 00055 3$:                   CMPL    PTR, BUFEND                               ; 1303
                           D4            1E  00058                          BGEQU   1$                                        ;
                           53 00000000'  00  9E 0005A                       MOVAB   NML$T_ENTBUFFER, ENTIDPTR                 ; 1306
            00000000'      00     40     8F  9B 00061                       MOVZBW  #64, NML$Q_ENTBFDSC                       ; 1307
                           03            57  D1 00069                       CMPL    R7, #3                                    ; 1318
                           20            12  0006C                          BNEQ    4$                                        ;
                           58            04  C0 0006E                       ADDL2   #4, PTR                                   ; 1321
                           5A            58  D0 00071                       MOVL    PTR, ENTITYADD                            ; 1322
                           83            88  B0 00074                       MOVW    (PTR)+, (ENTIDPTR)+                       ; 1323
                           58            02  C0 00077                       ADDL2   #2, PTR                                   ; 1324
                           59            88  9A 0007A                       MOVZBL  (PTR)+, ENTITYLEN                         ; 1326
                                         58  D6 0007D                       INCL    PTR                                       ; 1327
                           83            59  90 0007F                       MOVB    ENTITYLEN, (ENTIDPTR)+                    ; 1328
                 63        68            59  28 00082                       MOVC3   ENTITYLEN, (PTR), (ENTIDPTR)              ; 1329
                           58            59  C0 00086                       ADDL2   ENTITYLEN, PTR                            ; 1330
                           59            02  D0 00089                       MOVL    #2, ENTITYLEN                             ; 1331
                                         75  11 0008C                       BRB     12$                                       ; 1316
                           0D            57  D1 0008E 4$:                   CMPL    R7, #13                                   ; 1340
                                         05  1F 00091                       BLSSU   5$                                        ;
                           10            57  D1 00093                       CMPL    R7, #16                                   ;
```

```
                                    0F    1B 00096          BLEQU   6$                                           ;
                       12           57    D1 00098 5$:      CMPL    R7, #18                                      ;
                                    0A    13 0009B          BEQL    6$                                           ;
                       14           57    D1 0009D          CMPL    R7, #20                                      ;
                                    05    13 000A0          BEQL    6$                                           ;
                       16           57    D1 000A2          CMPL    R7, #22                                      ;
                                    5C    12 000A5          BNEQ    12$                                          ;
                       0D           57    D1 000A7 6$:      CMPL    R7, #13                                      ; 1350
                                    0A    12 000AA          BNEQ    7$                                           ;
          63 00000000'  00          0B    28 000AC          MOVC3   #11, P.AAE, (ENTIDPTR)                       ; 1351
                                    3F    11 000B4          BRB     11$                                          ;
                       0E           57    D1 000B6 7$:      CMPL    R7, #14                                      ; 1352
                                    0F    1F 000B9          BLSSU   8$                                           ;
                       10           57    D1 000BB          CMPL    R7, #16                                      ;
                                    0A    1A 000BE          BGTRU   8$                                           ;
          63 00000000'  00          0D    28 000C0          MOVC3   #13, P.AAF, (ENTIDPTR)                       ; 1353
                                    2B    11 000C8          BRB     11$                                          ;
                       12           57    D1 000CA 8$:      CMPL    R7, #18                                      ; 1354
                                    0A    12 000CD          BNEQ    9$                                           ;
          63 00000000'  00          0B    28 000CF          MOVC3   #11, P.AAG, (ENTIDPTR)                       ; 1355
                                    1C    11 000D7          BRB     11$                                          ;
                       14           57    D1 000D9 9$:      CMPL    R7, #20                                      ; 1356
                                    0A    12 000DC          BNEQ    10$                                          ;
          63 00000000'  00          0A    28 000DE          MOVC3   #10, P.AAH, (ENTIDPTR)                       ; 1357
                                    0D    11 000E6          BRB     11$                                          ;
                       16           57    D1 000E8 10$:     CMPL    R7, #22                                      ; 1358
                                    08    12 000EB          BNEQ    11$                                          ;
          63 00000000'  00          0B    28 000ED          MOVC3   #11, P.AAI, (ENTIDPTR)                       ; 1359
                         00000000G0046    9F 000F5 11$:     PUSHAB  NML$AB_ENTITYDATA+1[R6]                      ; 1363
                       83           9E    B0 000FC          MOVW    @(SP)+, (ENTIDPTR)+                          ;
                       83       40  8F    90 000FF          MOVB    #64, (ENTIDPTR)+                             ; 1364
                       03           57    D1 00103 12$:     CMPL    R7, #3                                       ; 1369
                                    10    13 00106          BEQL    13$                                          ;
                       59           88    3C 00108          MOVZWL  (PTR)+, ENTITYLEN                            ; 1371
                       5A           58    D0 0010B          MOVL    PTR, ENTITYADD                               ; 1373
                       83           59    90 0010E          MOVB    ENTITYLEN, (ENTIDPTR)+                       ; 1374
          63           6A           59    28 00111          MOVC3   ENTITYLEN, (ENTITYADD), (ENTIDPTR)           ; 1377
                       58           59    C0 00115          ADDL2   ENTITYLEN, PTR                               ; 1378
                       50  00000000'  00  9E 00118 13$:     MOVAB   NML$T_ENTBUFFER, R0                          ; 1381
00000000'  00          53           50    A3 0011F          SUBW3   R0, ENTIDPTR, NML$Q_ENTBFDSC                 ;
                                14  AE    9F 00127          PUSHAB  QBFDSC                                       ; 1385
                         00000000G  00    9F 0012A          PUSHAB  NML$GQ_QIOBFDSC                              ;
                                24  AE    9F 00130          PUSHAB  P2DSC                                        ;
                         00000000'  00    9F 00133          PUSHAB  NML$Q_P2BFDSC                                ;
                                34  AE    9F 00139          PUSHAB  NFBDSC                                       ;
                         00000000'  00    9F 0013C          PUSHAB  NML$Q_NFBBFDSC                               ;
                                    7E    D4 00142          CLRL    -(SP)                                        ;
                       7E           01    CE 00144          MNEGL   #1, -(SP)                                    ; 1387
                                24  AE    DD 00147          PUSHL   SRCHKEY2                                     ;
                       7E           59    7D 0014A          MOVQ    ENTITYLEN, -(SP)                             ; 1386
                                34  AE    DD 0014D          PUSHL   SRCHKEY1                                     ;
                                3C  AE    DD 00150          PUSHL   DB                                           ; 1385
                                    23    DD 00153          PUSHL   #35                                          ;
          00000000G  00             0E    FB 00155          CALLS   #14, NML$BLDSETQBF                           ;
                                14  AE    9F 0015C          PUSHAB  QBFDSC                                       ; 1392
                                    7E    D4 0015F          CLRL    -(SP)                                        ;
                                24  AE    9F 00161          PUSHAB  P2DSC                                        ;
```

```
                                30    AE   9F 00164          PUSHAB  NFBDSC                                      ;
          00000000G  00               04   FB 00167          CALLS   #4, NML$NETQIO                              ;
                     0D               50   E9 0016E          BLBC    R0, 14$                                     ;
                        00000000G     00   D4 00171          CLRL    NML$AB_MSGBLOCK                             ; 1396
          00000000G  00               01   90 00177          MOVB    #1, NML$AB_MSGBLOCK+4                       ; 1397
          00000000G  00               10   88 0017E  14$:    BISB2   #16, NML$AB_MSGBLOCK                        ; 1403
          00000000G  00  00000000'    00   9E 00185          MOVAB   NML$Q_ENTBFDSC, NML$AB_MSGBLOCK+20          ; 1404
                                10    AE   9F 00190          PUSHAB  MSGSIZE                                     ; 1408
                        00000000G     00   9F 00193          PUSHAB  NML$AB_MSGBLOCK                             ;
          00000000G  00               02   FB 00199          CALLS   #2, NML$BLD_REPLY                           ;
                                10    AE   DD 001A0          PUSHL   MSGSIZE                                     ; 1409
                        00000000G     00   9F 001A3          PUSHAB  NML$AB_SNDBUFFER                            ;
          00000000G  00               02   FB 001A9          CALLS   #2, NML$SEND                                ;
                               FEA2   31 001B0          BRW     3$                                          ; 1303
                                      04 001B3          RET                                                 ; 1413

; Routine Size:  436 bytes,    Routine Base:  $CODE$ + 076D


; 1427          1414  1
```

```
%SBTTL 'NML_SETKNOWNALL  Set volatile database parameters'
ROUTINE NML_SETKNOWNALL (ENTITY, ENTITY_LEN, ENTITY_ADR) : NOVALUE =

!++
! FUNCTIONAL DESCRIPTION:
!
!       This routine sets all permanent parameters for the specified
!       entity type into the volatile data base.
!
! INPUTS:
!
!       ENTITY          Entity type code.
!       ENTITY_LEN      Byte count of entity id string.
!       ENTITY_ADR      Address of entity id string.
!
!--

BEGIN

LOCAL
    bufend,
    db,                                 ! Database ID
    srchkey1,                           ! Search key one ID
    srchkey2,                           ! Search key two ID
    entityadd,
    entitylen,
    fid,                                ! File id code
    fldadr,
    fldsize,
    key,
    qual_len,
    qual_adr,
    msgsize,
    owner,                              ! Search key
    nfbdsc   : DESCRIPTOR,              ! NFB buffer descriptor
    p2dsc    : DESCRIPTOR,              ! QIO P2 buffer descriptor
    qbfdsc   : DESCRIPTOR,              ! QIO P2 buffer descriptor
    recdsc : DESCRIPTOR,                ! Record descriptor
    status,
    rewind_flag,                        ! Used for node database only.  Set at
                                        !       at the beginning of an operation.
    tabdes : REF DESCRIPTOR;            ! Table for SET ALL

!
! Get entity information.
!
db       = .nml$ab_entitydata [.entity, eit$b_database]; ! Database ID
srchkey1 = .nml$ab_entitydata [.entity, eit$l_srch_id1];! Search key one
srchkey2 = .nml$ab_entitydata [.entity, eit$l_srch_id2];! Search key two
fid      = .nml$ab_entitydata [.entity, eit$b_fileid];   ! File id code
owner    = .nml$ab_entitydata [.entity, eit$w_key];      ! Search key
tabdes   = .nml$ab_entitydata [.entity, eit$a_alltab];   ! Table for SET ALL

key = 0;
status = nml$_sts_suc;
rewind_flag = true;
!
```

```
! Keep looking for records in the permanent database which match the entity
! being SET.  For each one found, build QIO buffers containing all the
! parameters in the permanent data base record, and issue the SET to the ACP.
! Note that the entity information passed to NML$MATCHRECORD is normally
! zeros.  It is non zero for commands such as
!       SET X25-PROTOCOL GROUP XYZ ALL
! which, although it is a single entity, has one record for each DTE assigned
! to the group.  Therefore, it is a multiple QIO operation.
!
WHILE .status DO
    BEGIN
    !
    ! The structure of the node permanent database is different from the
    ! others because it can be much bigger and therefore needs to be faster.
    !
    IF .fid NEQ nma$c_opn_node THEN
        status = nml$matchrecord (.fid, nml$gq_recbfdsc, key,
                                  .owner, .entity_len, .entity_adr,
                                  0, 0, 0,
                                  recdsc)
    ELSE
        BEGIN
        status = nml$read_known_node_rec (.entity, nml$gq_recbfdsc,
                                          recdsc, .rewind_flag);
        rewind_flag = false;
        END;
    IF .status THEN
        BEGIN
        nml$ab_msgblock [msb$b_code] = 0; ! Reset status code
        !
        ! Get entity id.
        !
        entityadd = 0;
        status = nma$searchfld (recdsc, .owner, entitylen, entityadd);
        !
        ! Get entity id for SET QIO and id string for response message.
        !
        SELECTONEU .entity OF
            SET
            [nml$c_loopnode]:
                BEGIN
                !
                ! Id for loop node is node name.
                !
                entityadd = 0;
                status = nma$searchfld (recdsc, nma$c_pcno_nna,
                                        entitylen, entityadd);
                END;

            [nml$c_prot_grp]:
                !
                ! Get the DTE ID from the Group record just read.  The DTE
                ! is the second search key ID for the SET QIO.
                !
                BEGIN
                qual_adr = 0;           ! Search from beginning of record.
                status = nma$searchfld (recdsc, nma$c_pcxp_gdt,
```

```
; 1543      1529 5                                     qual_len, qual_adr);
; 1544      1530 4                     END;
; 1545      1531 4              TES;
; 1546      1532 3            END;
; 1547      1533 3        IF .status THEN
; 1548      1534 4            BEGIN
; 1549      1535 4            !
; 1550      1536 4            ! Build QIO buffer.
; 1551      1537 4            !
; 1552      1538 4            nml$bldalldes (recdsc, .tabdes); ! Build parameter descriptors
; 1553      1539 4            nml$bldsetqbf (nfb$c_fc_set, .db,
; 1554      1540 4                            .srchkey1, .entitylen, .entityadd,
; 1555      1541 4                            .srchkey2, .qual_len, .qual_adr,
; 1556      1542 4                            nml$q_nfbbfdsc, nfbdsc,
; 1557      1543 4                            nml$q_p2bfdsc, p2dsc,
; 1558      1544 4                            nml$gq_qiobfdsc, qbfdsc);
; 1559      1545 4            !
; 1560      1546 4            ! Add the parameters to volatile data base entry.
; 1561      1547 4            !
; 1562      1548 4            IF nml$netqio (nfbdsc, p2dsc, 0, qbfdsc) THEN
; 1563      1549 5                BEGIN
; 1564      1550 5                nml$ab_msgblock [msb$l_flags] = 0;
; 1565      1551 5                nml$ab_msgblock [msb$b_code] = nma$c_sts_suc;
; 1566      1552 4                END;
; 1567      1553 4            !
; 1568      1554 4            ! Add the entity identification to the response message.
; 1569      1555 4            !
; 1570      1556 4            nml$q_entbfdsc [dsc$w_length] = nml$k_entbuflen;
; 1571      1557 4            nml$getrecowner (recdsc,
; 1572      1558 4                            .entity,
; 1573      1559 4                            nml$q_entbfdsc,
; 1574      1560 4                            nml$q_entbfdsc [dsc$w_length]);
; 1575      1561 4
; 1576      1562 4            nml$ab_msgblock [msb$v_entd_fld] = 1;
; 1577      1563 4            nml$ab_msgblock [msb$a_entity] = nml$q_entbfdsc;
; 1578      1564 4            !
; 1579      1565 4            ! Build and send the response message.
; 1580      1566 4            !
; 1581      1567 4            nml$bld_reply (nml$ab_msgblock, msgsize);
; 1582      1568 4            nml$send (nml$ab_sndbuffer, .msgsize);
; 1583      1569 3            END;
; 1584      1570 3        key = .key + 1;                          ! Increment record key
; 1585      1571 2        END;
; 1586      1572 2
; 1587      1573 1 END;                                  ! End of NML_SETKNOWNALL


                                        0FFC 00000 NML_SETKNOWNALL:
                                                        .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11       ; 1416
                     5B 00000000G   00  9E 00002        MOVAB   NML$AB_MSGBLOCK, R11                       ;
                     5E             38  C2 00009        SUBL2   #56, SP                                    ;
                     53       04    AC  D0 0000C        MOVL    ENTITY, R3                                 ; 1461
        50           53             2C  C5 00010        MULL3   #44, R3, R0                                ;
                     5A 00000000G0040   9A 00014        MOVZBL  NML$AB_ENTITYDATA+5[R0], DB                ;
```

```
                    00000000G0040  9F  0001C         PUSHAB  NML$AB_ENTITYDATA+6[R0]                    ; 1462
                  59           9E  D0  00023         MOVL    @(SP)+, SRCHKEY1                           ;
                    00000000G0040  9F  00026         PUSHAB  NML$AB_ENTITYDATA+10[R0]                   ; 1463
                  58           9E  D0  0002D         MOVL    @(SP)+, SRCHKEY2                           ;
                  56 00000000G0040 9A  00030         MOVZBL  NML$AB_ENTITYDATA[R0], FID                 ; 1464
                    00000000G0040  9F  00038         PUSHAB  NML$AB_ENTITYDATA+3[R0]                    ; 1465
                  55           9E  3C  0003F         MOVZWL  @(SP)+, OWNER                              ;
                    00000000G0040  9F  00042         PUSHAB  NML$AB_ENTITYDATA+40[R0]                   ; 1466
                  57           9E  D0  00049         MOVL    @(SP)+, TABDES                             ;
                               6E  D4  0004C         CLRL    KEY                                        ; 1468
                  52           01  D0  0004E         MOVL    #1, STATUS                                 ; 1469
                  54           01  D0  00051         MOVL    #1, REWIND_FLAG                            ; 1470
                  01           52  E8  00054 1$:     BLBS    STATUS, 2$                                 ; 1481
                                   04  00057         RET                                                ;
                               56  D5  00058 2$:     TSTL    FID                                        ; 1487
                               24  13  0005A         BEQL    3$                                         ;
                         18    AE  9F  0005C         PUSHAB  RECDSC                                     ; 1488
                               7E  7C  0005F         CLRQ    -(SP)                                      ;
                               7E  D4  00061         CLRL    -(SP)                                      ;
                  7E     08    AC  7D  00063         MOVQ    ENTITY_LEN, -(SP)                          ; 1489
                               55  DD  00067         PUSHL   OWNER                                      ;
                         1C    AE  9F  00069         PUSHAB  KEY                                        ; 1488
                    00000000G  00  9F  0006C         PUSHAB  NML$GQ_RECBFDSC                            ;
                               56  DD  00072         PUSHL   FID                                        ;
         00000000G  00         0A  FB  00074         CALLS   #10, NML$MATCHRECORD                       ;
                    52         50  D0  0007B         MOVL    R0, STATUS                                 ;
                               19  11  0007E         BRB     4$                                         ;
                               54  DD  00080 3$:     PUSHL   REWIND_FLAG                                ; 1495
                         1C    AE  9F  00082         PUSHAB  RECDSC                                     ; 1494
                    00000000G  00  9F  00085         PUSHAB  NML$GQ_RECBFDSC                            ;
                               53  DD  0008B         PUSHL   R3                                         ;
         00000000G  00         04  FB  0008D         CALLS   #4, NML$READ_KNOWN_NODE_REC                ;
                    52         50  D0  00094         MOVL    R0, STATUS                                 ;
                               54  D4  00097         CLRL    REWIND_FLAG                                ; 1496
                  50           52  E9  00099 4$:     BLBC    STATUS, 7$                                 ; 1498
                         04    AB  94  0009C         CLRB    NML$AB_MSGBLOCK+4                          ; 1500
                         04    AE  D4  0009F         CLRL    ENTITYADD                                  ; 1504
                         04    AE  9F  000A2         PUSHAB  ENTITYADD                                  ; 1505
                         0C    AE  9F  000A5         PUSHAB  ENTITYLEN                                  ;
                               55  DD  000A8         PUSHL   OWNER                                      ;
                         24    AE  9F  000AA         PUSHAB  RECDSC                                     ;
         00000000G  00         04  FB  000AD         CALLS   #4, NMA$SEARCHFLD                          ;
                    52         50  D0  000B4         MOVL    R0, STATUS                                 ;
                    05         53  D1  000B7         CMPL    R3, #5                                     ; 1511
                               10  12  000BA         BNEQ    5$                                         ;
                         04    AE  D4  000BC         CLRL    ENTITYADD                                  ; 1516
                         04    AE  9F  000BF         PUSHAB  ENTITYADD                                  ; 1517
                         0C    AE  9F  000C2         PUSHAB  ENTITYLEN                                  ;
                  7E   01F4    8F  3C  000C5         MOVZWL  #500, -(SP)                                ;
                               13  11  000CA         BRB     6$                                         ;
                  10           53  D1  000CC 5$:     CMPL    R3, #16                                    ; 1521
                               1B  12  000CF         BNEQ    7$                                         ;
                         0C    AE  D4  000D1         CLRL    QUAL_ADR                                   ; 1527
                         0C    AE  9F  000D4         PUSHAB  QUAL_ADR                                   ; 1528
                         14    AE  9F  000D7         PUSHAB  QUAL_LEN                                   ;
                  7E   0492    8F  3C  000DA         MOVZWL  #1170, -(SP)                               ;
                         24    AE  9F  000DF 6$:     PUSHAB  RECDSC                                     ;
```

```
     00000000G  00                04  FB 000E2         CALLS   #4, NMA$SEARCHFLD                          ;
                52                50  D0 000E9         MOVL    R0, STATUS                                 ;
                03                52  E8 000EC 7$:     BLBS    STATUS, 8$                                 ; 1533
                            00A4  31  000EF            BRW     10$                                        ;
                                  57  DD 000F2 8$:     PUSHL   TABDES                                     ; 1538
                          1C      AE  9F 000F4         PUSHAB  RECDSC                                     ;
     00000000G  00                02  FB 000F7         CALLS   #2, NML$BLDALLDES                          ;
                          20      AE  9F 000FE         PUSHAB  QBFDSC                                     ; 1539
                     00000000G    00  9F 00101         PUSHAB  NML$GQ_QIOBFDSC                            ;
                          30      AE  9F 00107         PUSHAB  P2DSC                                      ;
                     00000000'    00  9F 0010A         PUSHAB  NML$Q_P2BFDSC                              ;
                          40      AE  9F 00110         PUSHAB  NFBDSC                                     ;
                     00000000'    00  9F 00113         PUSHAB  NML$Q_NFBBFDSC                             ;
                          24      AE  DD 00119         PUSHL   QUAL_ADR                                   ; 1541
                          2C      AE  DD 0011C         PUSHL   QUAL_LEN                                   ;
                                  58  DD 0011F         PUSHL   SRCHKEY2                                   ;
                          28      AE  DD 00121         PUSHL   ENTITYADD                                  ; 1540
                          30      AE  DD 00124         PUSHL   ENTITYLEN                                  ;
                                  59  DD 00127         PUSHL   SRCHKEY1                                   ;
                                  5A  DD 00129         PUSHL   DB                                         ; 1539
                                  23  DD 0012B         PUSHL   #35                                        ;
     00000000G  00                0E  FB 0012D         CALLS   #14, NML$BLDSETQBF                         ;
                          20      AE  9F 00134         PUSHAB  QBFDSC                                     ; 1548
                                  7E  D4 00137         CLRL    -(SP)                                      ;
                          30      AE  9F 00139         PUSHAB  P2DSC                                      ;
                          3C      AE  9F 0013C         PUSHAB  NFBDSC                                     ;
     00000000G  00                04  FB 0013F         CALLS   #4, NML$NETQIO                             ;
                06                50  E9 00146         BLBC    R0, 9$                                     ;
                                  6B  D4 00149         CLRL    NML$AB_MSGBLOCK                            ; 1550
            04  AB                01  90 0014B         MOVB    #1, NML$AB_MSGBLOCK+4                      ; 1551
     00000000'  00        40      8F  9B 0014F 9$:     MOVZBW  #64, NML$Q_ENTBFDSC                        ; 1556
                     00000000'    00  9F 00157         PUSHAB  NML$Q_ENTBFDSC                             ; 1560
                     00000000'    00  9F 0015D         PUSHAB  NML$Q_ENTBFDSC                             ; 1557
                                  53  DD 00163         PUSHL   R3                                         ; 1560
                          24      AE  9F 00165         PUSHAB  RECDSC                                     ; 1557
     00000000G  00                04  FB 00168         CALLS   #4, NML$GETRECOWNER                        ; 1560
                6B                10  88 0016F         BISB2   #16, NML$AB_MSGBLOCK                       ; 1562
            14  AB   00000000'    00  9E 00172         MOVAB   NML$Q_ENTBFDSC, NML$AB_MSGBLOCK+20         ; 1563
                          14      AE  9F 0017A         PUSHAB  MSGSIZE                                    ; 1567
                                  5B  DD 0017D         PUSHL   R11                                        ;
     00000000G  00                02  FB 0017F         CALLS   #2, NML$BLD_REPLY                          ;
                          14      AE  DD 00186         PUSHL   MSGSIZE                                    ; 1568
                     00000000G    00  9F 00189         PUSHAB  NML$AB_SNDBUFFER                           ;
     00000000G  00                02  FB 0018F         CALLS   #2, NML$SEND                               ;
                                  6E  D6 00196 10$:    INCL    KEY                                        ; 1570
                            FEB9  31  00198            BRW     1$                                         ; 1481
                                  04  0019B            RET                                                ; 1573

; Routine Size:  412 bytes,    Routine Base:  $CODE$ + 0921
```

```
%SBTTL 'NML_SETEXE  Set executor parameters'
ROUTINE NML_SETEXE : NOVALUE =

!++
! FUNCTIONAL DESCRIPTION:
!
!       This routine adds the specified parameters to the volatile data base
!       entry for the executor node.
!
! FORMAL PARAMETERS:
!
!       NONE
!
! IMPLICIT INPUTS:
!
!       NONE
!
! IMPLICIT OUTPUTS:
!
!       NONE
!
! ROUTINE VALUE:
! COMPLETION CODES:
!
!       NONE
!
! SIDE EFFECTS:
!
!       NONE
!
!--

    BEGIN

    LOCAL
        DB,                                     ! Database ID
        SRCHKEY1,                               ! Search key one ID
        SRCHKEY2,                               ! Search key two ID
        ENTITYADD,
        ENTITYLEN,
        MSGSIZE,
        NODADDR,
        NFBDSC    : DESCRIPTOR,                 ! NFB buffer descriptor
        P2DSC     : DESCRIPTOR,                 ! QIO P2 buffer descriptor
        QBFDSC    : DESCRIPTOR;                 ! QIO P2 buffer descriptor

!
! Add the entity identification to the message.
!
NML$GETEXEID (NML$Q_ENTBFDSC, NML$Q_ENTBFDSC [DSC$W_LENGTH]);
NML$AB_MSGBLOCK [MSB$V_ENTD_FLD] = T;
NML$AB_MSGBLOCK [MSB$A_ENTITY] = NML$Q_ENTBFDSC;

!
! If executor specific parameters, use the local node database,
! otherwise, use the node database.
!
```

```
2 SRCHKEY2 = NFB$C_WILDCARD;
2 IF .NML$GL_PRS_FLGS [NML$V_PRS_EXEPG]
2     THEN
3         BEGIN
3
3         DB = NFB$C_DB_LNI;
3         SRCHKEY1 = NFB$C_WILDCARD;
3         ENTITYLEN = -1;                                ! No id necessary for
3         ENTITYADD = 0;                                 !  executor parameters
3
3         END
2     ELSE
3         BEGIN
3
3         DB = NFB$C_DB_NDI;
3         SRCHKEY1 = NFB$C_NDI_ADD;
3         NODADDR = 0;                                   ! Zero node address
3         ENTITYLEN = 2;
3         ENTITYADD = NODADDR;
3
2         END;
2
2     NML$BLDSETQBF (NFB$C_FC_SET, .DB,
2                    .SRCHKEY1, .ENTITYLEN, .ENTITYADD,
2                    .SRCHKEY2, -1, 0,
2                    NML$Q_NFBBFDSC, NFBDSC,
2                    NML$Q_P2BFDSC,  P2DSC,
2                    NML$GQ_QIOBFDSC, QBFDSC);
2 !
2 ! Add the parameters to volatile data base entry.
2 !
2 IF NML$NETQIO (NFBDSC, P2DSC, 0, QBFDSC) THEN
3     BEGIN
3     NML$AB_MSGBLOCK [MSB$L_FLAGS] = 0;
3     NML$AB_MSGBLOCK [MSB$B_CODE] = NMA$C_STS_SUC;
2     END;
2 !
2 ! Build and signal the response message.
2 !
2 NML$BLD_REPLY (NML$AB_MSGBLOCK, MSGSIZE);
2 NML$SEND (NML$AB_SNDBUFFER, .MSGSIZE);
2
1 END;                                   ! End of NML_SETEXE


                                  007C 00000 NML_SETEXE:
                                             .WORD   Save R2,R3,R4,R5,R6                     ; 1575
          56 00000000'    00 9E 00002        MOVAB   NML$Q_ENTBFDSC, R6                      ;
          55 00000000G    00 9E 00009        MOVAB   NML$AB_MSGBLOCK, R5                     ;
          5E              20 C2 00010        SUBL2   #32, SP                                 ;
                          56 DD 00013        PUSHL   R6                                      ; 1623
                          56 DD 00015        PUSHL   R6                                      ;
     00000000G  00        02 FB 00017        CALLS   #2, NML$GETEXEID                        ;
                65        10 88 0001E        BISB2   #16, NML$AB_MSGBLOCK                    ; 1624
```

```
              14   A5               66  9E 00021        MOVAB   NML$Q_ENTBFDSC, NML$AB_MSGBLOCK+20         ; 1625
                   54               01  D0 00025        MOVL    #1, SRCHKEY2                               ; 1631
                   0D 00000000G     00  E9 00028        BLBC    NML$GL_PRS_FLGS+1, 1$                      ; 1632
                   53               01  D0 0002F        MOVL    #1, DB                                     ; 1636
                   52               01  D0 00032        MOVL    #1, SRCHKEY1                               ; 1637
                   51               01  CE 00035        MNEGL   #1, ENTITYLEN                              ; 1638
                                    50  D4 00038        CLRL    ENTITYADD                                  ; 1639
                                    12  11 0003A        BRB     2$                                         ; 1632
                   53               02  D0 0003C 1$:    MOVL    #2, DB                                     ; 1645
                   52 02010012      8F  D0 0003F        MOVL    #33619986, SRCHKEY1                        ; 1646
                                    6E  D4 00046        CLRL    NODADDR                                    ; 1647
                   51               02  D0 00048        MOVL    #2, ENTITYLEN                              ; 1648
                   50               6E  9E 0004B        MOVAB   NODADDR, ENTITYADD                         ; 1649
                           08       AE  9F 0004E 2$:    PUSHAB  QBFDSC                                     ; 1653
                      00000000G     00  9F 00051        PUSHAB  NML$GQ_QIOBFDSC                            ;
                           18       AE  9F 00057        PUSHAB  P2DSC                                      ;
                      00000000'     00  9F 0005A        PUSHAB  NML$Q_P2BFDSC                              ;
                           28       AE  9F 00060        PUSHAB  NFBDSC                                     ;
                      00000000'     00  9F 00063        PUSHAB  NML$Q_NFBBFDSC                             ;
                                    7E  D4 00069        CLRL    -(SP)                                      ;
                   7E               01  CE 0006B        MNEGL   #1, -(SP)                                  ; 1655
                                    11  BB 0006E        PUSHR   #^M<R0,R4>                                 ; 1654
                                    51  DD 00070        PUSHL   ENTITYLEN                                  ;
                                    52  DD 00072        PUSHL   SRCHKEY1                                   ;
                                    53  DD 00074        PUSHL   DB                                         ; 1653
                                    23  DD 00076        PUSHL   #35                                        ;
         00000000G  00              0E  FB 00078        CALLS   #14, NML$BLDSETQBF                         ;
                           08       AE  9F 0007F        PUSHAB  QBFDSC                                     ; 1662
                                    7E  D4 00082        CLRL    -(SP)                                      ;
                           18       AE  9F 00084        PUSHAB  P2DSC                                      ;
                           24       AE  9F 00087        PUSHAB  NFBDSC                                     ;
         00000000G  00              04  FB 0008A        CALLS   #4, NML$NETQIO                             ;
                    06              50  E9 00091        BLBC    R0, 3$                                     ;
                                    65  D4 00094        CLRL    NML$AB_MSGBLOCK                            ; 1664
              04   A5               01  90 00096        MOVB    #1, NM[$AB_MSGBLOCK+4                      ; 1665
                           04       AE  9F 0009A 3$:    PUSHAB  MSGSIZE                                    ; 1670
                                    55  DD 0009D        PUSHL   R5                                         ;
         00000000G  00              02  FB 0009F        CALLS   #2, NML$BLD_REPLY                          ;
                           04       AE  DD 000A6        PUSHL   MSGSIZE                                    ; 1671
                      00000000G     00  9F 000A9        PUSHAB  NML$AB_SNDBUFFER                           ;
         00000000G  00              02  FB 000AF        CALLS   #2, NM[$SEND                               ;
                                        04 000B6        RET                                                ; 1673

; Routine Size:  183 bytes,    Routine Base:  $CODE$ + 0ABD
```

```
%SBTTL 'NML_SETEXEALL  Set all executor parameters'
ROUTINE NML_SETEXEALL : NOVALUE =

!++
! FUNCTIONAL DESCRIPTION:
!
!       This routine sets all executor parameters from the permanent data base
!       into the volatile data base.
!
! FORMAL PARAMETERS:
!
!       NONE
!
! IMPLICIT INPUTS:
!
!       This routine assumes the node data base file is open.
!
! IMPLICIT OUTPUTS:
!
!       NONE
!
! ROUTINE VALUE:
! COMPLETION CODES:
!
!       NONE
!
! SIDE EFFECTS:
!
!       NONE
!
!--

BEGIN

LOCAL
    exeadr,
    fid,
    key,
    msgsize,
    nfbdsc   : DESCRIPTOR,              ! NFB buffer descriptor
    p2dsc    : DESCRIPTOR,              ! QIO P2 buffer descriptor
    qbfdsc   : DESCRIPTOR,              ! QIO P2 buffer descriptor
    recdsc   : DESCRIPTOR,
    dummy,
    status;
!
! Read executor node record.  If one does not exist then create a
! descriptor for a null record.
!
key = .nml$ab_entitydata [nml$c_executor, eit$w_key];
IF NOT nml$readrecord (nma$c_opn_node,
                        key,
                        UPLIT (nmn$c_typ_key_len,
                               UPLIT (nml$c_executor)),
                        nml$gq_recbfdsc,
                        recdsc,
                        dummy) THEN
```

```
3       BEGIN
3       recdsc [dsc$w_length]  = 0;
3       recdsc [dsc$a_pointer] = .nml$gg_recbfdsc [dsc$a_pointer] +
3                                       nmn$c_node_keys_len;
2       END;
2    !
2    ! Build a parameter descriptor for each parameter that is set in the record
2    ! and set it in the volatile data base.
2    !
2    nml$bldalldes (recdsc,
2               .nml$ab_entitydata [nml$c_executor, eit$a_alltab]);
2    nml$bldsetqbf (nfb$c_fc_set, nfb$c_db_lni,
2                   nfb$c_wildcard, -1, 0,
2                   nfb$c_wildcard, -1, 0,
2                   nml$q_nfbbfdsc, nfbdsc,
2                   nml$q_p2bfdsc, p2dsc,
2                   nml$gq_qiobfdsc, qbfdsc);
2    status = nml$netqio (nfbdsc, p2dsc, 0, qbfdsc);
2    !
2    ! If the first operation was successful then repeat the operation for the
2    ! parameters that the executor has in common with remote nodes.  Otherwise
2    ! send an error response message.
2    !
2    IF .status THEN
3       BEGIN
3       nml$bldalldes (recdsc, .nml$ab_entitydata [nml$c_node, eit$a_alltab]);
3       exeadr = 0;
3       nml$bldsetqbf (nfb$c_fc_set, nfb$c_db_ndi,
3                          nfb$c_ndi_add, 0, exeadr,
3                          nfb$c_wildcard, -1, 0,
3                          nml$q_nfbbfdsc, nfbdsc,
3                          nml$q_p2bfdsc, p2dsc,
3                          nml$gq_qiobfdsc, qbfdsc);
3
3       IF nml$netqio (nfbdsc, p2dsc, 0, qbfdsc) THEN
4           BEGIN
4           nml$ab_msgblock [msb$l_flags] = 0;
4           nml$ab_msgblock [msb$b_code] = nma$c_sts_suc; ! Successful
3           END;
2       END;
2    !
2    ! Add the executor entity descriptor to the message.
2    !
2    nml$q_entbfdsc [dsc$w_length] = nml$k_entbuflen;
2    nml$getrecowner (recdsc,
2                     nml$c_executor,
2                     nml$q_entbfdsc,
2                     nml$q_entbfdsc [dsc$w_length]);
2
2    nml$ab_msgblock [msb$v_entd_fld] = 1;
2    nml$ab_msgblock [msb$a_entity] = nml$q_entbfdsc;
2    !
2    ! Build and send the response message.
2    !
2    nml$bld_reply (nml$ab_msgblock, msgsize);
2    nml$send (nml$ab_sndbuffer, .msgsize);
2
```

```
; 1804           1788  1 END;                         ! End of NML_SETEXEALL


                                                                     .PSECT  $PLIT$,NOWRT,NOEXE,2

                                 00000007  00058 P.AAK:  .LONG   7                                   ;
                                 00000002  0005C P.AAJ:  .LONG   2                                   ;
                                 00000000' 00060         .ADDRESS P.AAK                              ;


                                                         .PSECT  $CODE$,NOWRT,2

                                      03FC 00000 NML_SETEXEALL:
                                                         .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9        ; 1675
                  59 00000000G  00   9E 00002            MOVAB   NML$NETQIO, R9                      ;
                  58 00000000G  00   9E 00009            MOVAB   NML$BLDSETQBF, R8                   ;
                  57 00000000G  00   9E 00010            MOVAB   NML$GQ_QIOBFDSC, R7                 ;
                  56 00000000G  00   9E 00017            MOVAB   NML$BLDALLDES, R6                   ;
                  55 00000000G  00   9E 0001E            MOVAB   NML$AB_ENTITYDATA+311, R5           ;
                  54 00000000'  00   9E 00025            MOVAB   NML$Q_ENTBFDSC, R4                  ;
                  53 00000000G  00   9E 0002C            MOVAB   NML$AB_MSGBLOCK, R3                 ;
                  52 00000000'  00   9E 00033            MOVAB   NML$Q_P2BFDSC, R2                   ;
                  5E            30   C2 0003A            SUBL2   #48, SP                             ;
           04     AE            65   3C 0003D            MOVZWL  NML$AB_ENTITYDATA+311, KEY          ; 1723
                                5E   DD 00041            PUSHL   SP                                  ; 1724
                         14     AE   9F 00043            PUSHAB  RECDSC                              ;
                   00000000G    00   9F 00046            PUSHAB  NML$GQ_RECBFDSC                     ;
                         4C     A2   9F 0004C            PUSHAB  P.AAJ                               ; 1726
                         14     AE   9F 0004F            PUSHAB  KEY                                 ; 1724
                                7E   D4 00052            CLRL    -(SP)                               ;
        00000000G  00           06   FB 00054            CALLS   #6, NML$READRECORD                  ;
                   0C           50   E8 0005B            BLBS    R0, 1$                              ;
                         10     AE   B4 0005E            CLRW    RECDSC                              ; 1732
   14     AE 00000000G  00      0A   C1 00061            ADDL3   #10, NML$GQ_RECBFDSC+4, RECDSC+4    ; 1733
                         25     A5   DD 0006A 1$:        PUSHL   NML$AB_ENTITYDATA+348               ; 1741
                         14     AE   9F 0006D            PUSHAB  RECDSC                              ; 1740
                   66           02   FB 00070            CALLS   #2, NML$BLDALLDES                   ;
                         18     AE   9F 00073            PUSHAB  QBFDSC                              ; 1742
                                57   DD 00076            PUSHL   R7                                  ;
                         28     AE   9F 00078            PUSHAB  P2DSC                               ;
                                52   DD 0007B            PUSHL   R2                                  ;
                         38     AE   9F 0007D            PUSHAB  NFBDSC                              ;
                         F8     A2   9F 00080            PUSHAB  NML$Q_NFBBFDSC                      ;
                                7E   D4 00083            CLRL    -(SP)                               ;
                   7E           01   CE 00085            MNEGL   #1, -(SP)                           ; 1744
                                01   DD 00088            PUSHL   #1                                  ; 1742
                                7E   D4 0008A            CLRL    -(SP)                               ;
                   7E           01   CE 0008C            MNEGL   #1, -(SP)                           ; 1743
                                01   DD 0008F            PUSHL   #1                                  ; 1742
                                01   DD 00091            PUSHL   #1                                  ;
                                23   DD 00093            PUSHL   #35                                 ;
                   68           0E   FB 00095            CALLS   #14, NML$BLDSETQBF                  ;
                         18     AE   9F 00098            PUSHAB  QBFDSC                              ; 1748
                                7E   D4 0009B            CLRL    -(SP)                               ;
                         28     AE   9F 0009D            PUSHAB  P2DSC                               ;
```

```
                              34   AE   9F  000A0           PUSHAB  NFBDSC                                   ;
                  69               04   FB  000A3           CALLS   #4, NML$NETQIO                           ;
                  4D               50   E9  000A6           BLBC    STATUS, 2$                               ; 1754
                            FF75   C5   DD  000A9           PUSHL   NML$AB_ENTITYDATA+172                    ; 1756
                              14   AE   9F  000AD           PUSHAB  RECDSC                                   ;
                  66               02   FB  000B0           CALLS   #2, NML$BLDALLDES                        ;
                              08   AE   D4  000B3           CLRL    EXEADR                                   ; 1757
                              18   AE   9F  000B6           PUSHAB  QBFDSC                                   ; 1758
                                   57   DD  000B9           PUSHL   R7                                       ;
                              28   AE   9F  000BB           PUSHAB  P2DSC                                    ;
                                   52   DD  000BE           PUSHL   R2                                       ;
                              38   AE   9F  000C0           PUSHAB  NFBDSC                                   ;
                              F8   A2   9F  000C3           PUSHAB  NML$Q_NFBBFDSC                           ;
                                   7E   D4  000C6           CLRL    -(SP)                                    ;
                  7E               01   CE  000C8           MNEGL   #1, -(SP)                                ; 1760
                                   01   DD  000CB           PUSHL   #1                                       ; 1758
                              2C   AE   9F  000CD           PUSHAB  EXEADR                                   ;
                                   7E   D4  000D0           CLRL    -(SP)                                    ;
                        02010012   8F   DD  000D2           PUSHL   #33619986                                ;
                                   02   DD  000D8           PUSHL   #2                                       ;
                                   23   DD  000DA           PUSHL   #35                                      ;
                  68               0E   FB  000DC           CALLS   #14, NML$BLDSETQBF                       ;
                              18   AE   9F  000DF           PUSHAB  QBFDSC                                   ; 1765
                                   7E   D4  000E2           CLRL    -(SP)                                    ;
                              28   AE   9F  000E4           PUSHAB  P2DSC                                    ;
                              34   AE   9F  000E7           PUSHAB  NFBDSC                                   ;
                  69               04   FB  000EA           CALLS   #4, NML$NETQIO                           ;
                  06               50   E9  000ED           BLBC    R0, 2$                                   ;
                                   63   D4  000F0           CLRL    NML$AB_MSGBLOCK                          ; 1767
              04  A3               01   90  000F2           MOVB    #1, NML$AB_MSGBLOCK+4                    ; 1768
                  64          40   8F   9B  000F6  2$:      MOVZBW  #64, NML$Q_ENTBFDSC                      ; 1774
                                   54   DD  000FA           PUSHL   R4                                       ; 1778
                                   54   DD  000FC           PUSHL   R4                                       ; 1775
                                   07   DD  000FE           PUSHL   #7                                       ; 1778
                              1C   AE   9F  00100           PUSHAB  RECDSC                                   ; 1775
       00000000G  00               04   FB  00103           CALLS   #4, NML$GETRECOWNER                      ; 1778
                  63               10   88  0010A           BISB2   #16, NML$AB_MSGBLOCK                     ; 1780
              14  A3               64   9E  0010D           MOVAB   NML$Q_ENTBFDSC, NML$AB_MSGBLOCK+20       ; 1781
                              0C   AE   9F  00111           PUSHAB  MSGSIZE                                  ; 1785
                                   53   DD  00114           PUSHL   R3                                       ;
       00000000G  00               02   FB  00116           CALLS   #2, NML$BLD_REPLY                        ;
                              0C   AE   DD  0011D           PUSHL   MSGSIZE                                  ; 1786
                       00000000G   00   9F  00120           PUSHAB  NML$AB_SNDBUFFER                         ;
       00000000G  00               02   FB  00126           CALLS   #2, NML$SEND                             ;
                                   04   0012D               RET                                              ; 1788

; Routine Size:  302 bytes,    Routine Base:  $CODE$ + 0B74
```

```
1 %SBTTL 'NML$SET_NI_CONFIG  Set Configurator Module volatile database parameters'
1 GLOBAL ROUTINE NML$SET_NI_CONFIG (ENTITY, ENTITY_LEN, ENTITY_ADR) : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine reads the permanent database for the Configurator
1 !       Module and builds NICE SET messages to set up the config volatile
1 !       database.  It forwards these messages to the Configurator
1 !       Module for processing.
1 !
1 ! INPUTS:
1 !
1 !       ENTITY          Entity type code.
1 !       ENTITY_LEN      Byte count of entity id string.
1 !       ENTITY_ADR      Address of entity id string.
1 !
1 !--
1
2 BEGIN
2
2 LOCAL
2       FID,                                    ! File id code
2       KEY,
2       OWNER,                                  ! Search key
2       FLDSIZE,
2       FLDADR,
2       CPT_INDEX,
2       PST_INDEX,
2       RECDSC : DESCRIPTOR,                    ! Record descriptor
2       STATUS,
2       TABDES : REF DESCRIPTOR,                ! Table for SET ALL
2       PTR;
2 !
2 ! If it's not a SET ALL command, simply forward the set command to the
2 ! Configurator module for processing.
2 !
2 IF NOT .NML$GL_PRS_FLGS [NML$V_PRS_ALL] THEN
3     BEGIN
3     NML$CALL_NI_CONFIG ();
3     RETURN;
3     END;
2 !
2 ! It's a SET ALL command.  Prepare to access the permanent database.
2 ! Get entity information from Entity Table.
2 !
2 FID     = .NML$AB_ENTITYDATA [.ENTITY, EIT$B_FILEID]; ! File id code
2 OWNER   = .NML$AB_ENTITYDATA [.ENTITY, EIT$W_KEY];    ! Search key
2 TABDES  = .NML$AB_ENTITYDATA [.ENTITY, EIT$A_ALLTAB];
2
2 KEY = 0;
2 !
2 ! For each circuit record in the permanent database, forward one SET command
2 ! to the Configurator Module.
2 !
2 WHILE NML$MATCHRECORD (.FID, NML$GQ_RECBFDSC, KEY,
2                         .OWNER, .ENTITY_LEN, .ENTITY_ADR,
```

```
                            0, 0, 0,
                            RECDSC) DO
        !
        ! Build NICE message to forward to the Configurator Module.  Since
        ! I am processing a SET MODULE CONFIGURATOR ... ALL command, the
        ! function code, option byte, and entity ID (CONFIGURATOR) are already
        ! in the buffer.
        !
        BEGIN
        PTR = NML$AB_RCVBUFFER + 15;
        INCR I FROM 0 TO .TABDES [DSC$W_LENGTH] - 1 DO
            BEGIN

            BIND
                TABLE = TABDES [DSC$A_POINTER] : REF BLOCK;

            FLDADR = 0;
            !
            ! If the parameter in the SET ALL table is in the permanent database
            ! record, add it to the NICE SET command.
            !
            IF NMA$SEARCHFLD (RECDSC,
                              .TABLE [.I, 0, 16, 0],
                              FLDSIZE,
                              FLDADR) THEN
                BEGIN
                !
                ! Add the NICE parameter ID
                !
                PTR = CH$MOVE (2, TABLE [.I, 0, 16, 0], .PTR);
                !
                ! If it's a string, add the string length.  The PST table indicates
                ! if it's a string.
                !
                CPT_INDEX = .TABLE [.I, 16, 16, 0];
                PST_INDEX = .NML$AB_CPTABLE [.CPT_INDEX, CPT$W_PSTINDEX];
                IF .NML$AB_PRMSEM [.PST_INDEX, PST$B_FORMAT] EQL NML$K_STRING THEN
                    CH$WCHAR_A (.FLDSIZE, PTR);
                !
                ! Add the parameter value to the SET command.
                !
                PTR = CH$MOVE (.FLDSIZE, .FLDADR, .PTR);
                END;
            END;
        !
        ! Send the SET command to the Configurator Module.  It will put the
        ! circuit into it's database, and return a response, which
        ! NML$CALL_NI_CONFIG will then forward to NCP.
        !
        NML$GL_RCVDATLEN = .PTR - NML$AB_RCVBUFFER;
        NML$CALL_NI_CONFIG ();
        !
        ! If it's a SET ALL for one circuit, quit here.
        !
        IF .NML$GB_ENTITY_FORMAT NEQ (NMA$C_ENT_KNO AND %X'FF') THEN
            EXITLOOP;
        KEY = .KEY + 1;                     ! Increment record key
```

```
; 1920       1903  2      END;
; 1921       1904  2
; 1922       1905  1 END;                                ! End of NML$SET_NI_CONFIG


                                             0FFC 00000         .ENTRY  NML$SET_NI_CONFIG, Save R2,R3,R4,R5,R6,R7,-  ; 1790
                                                                        R8,R9,R10,R11                                ;
                           5E            1C C2 00002            SUBL2   #28, SP                                      ;
           08 00000000G    00            01 E0 00005            BBS     #1, NML$GL_PRS_FLGS, 1$                      ; 1826
              00000000G    00            00 FB 0000D            CALLS   #0, NML$CALL_NI_CONFIG                       ; 1828
                                         04 00014               RET                                                  ; 1827
           50         04   AC            2C C5 00015 1$:        MULL3   #44, ENTITY, R0                              ; 1835
                      04   AE 00000000G0040 9A 0001A            MOVZBL  NML$AB_ENTITYDATA[R0], FID                   ;
                              00000000G0040 9F 00023            PUSHAB  NML$AB_ENTITYDATA+3[R0]                      ; 1836
                           6E            9E 3C 0002A            MOVZWL  @(SP)+, OWNER                                ;
                              00000000G0040 9F 0002D            PUSHAB  NML$AB_ENTITYDATA+40[R0]                     ; 1837
                           56            9E D0 00034            MOVL    @(SP)+, TABDES                               ;
                                    08   AE D4 00037            CLRL    KEY                                          ; 1839
                           5A       04   A6 9E 0003A            MOVAB   4(R6), R10                                   ; 1860
                                    14   AE 9F 0003E 2$:        PUSHAB  RECDSC                                       ; 1844
                                         7E 7C 00041            CLRQ    -(SP)                                        ;
                                         7E D4 00043            CLRL    -(SP)                                        ;
                           7E       08   AC 7D 00045            MOVQ    ENTITY_LEN, -(SP)                            ; 1845
                                    18   AE DD 00049            PUSHL   OWNER                                        ;
                                    24   AE 9F 0004C            PUSHAB  KEY                                          ; 1844
                              00000000G  00 9F 0004F            PUSHAB  NML$GQ_RECBFDSC                              ;
                                    28   AE DD 00055            PUSHL   FID                                          ;
              00000000G    00            0A FB 00058            CALLS   #10, NML$MATCHRECORD                         ;
                           01            50 E8 0005F            BLBS    R0, 3$                                       ;
                                         04 00062               RET                                                  ;
                           53 00000000G  00 9E 00063 3$:        MOVAB   NML$AB_RCVBUFFER+15, PTR                     ; 1855
                           5B            66 3C 0006A            MOVZWL  (TABDES), R11                                ; 1856
                           59            01 CE 0006D            MNEGL   #1, I                                        ;
                                         51 11 00070            BRB     6$                                           ;
                                    0C   AE D4 00072 4$:        CLRL    FLDADR                                       ; 1862
                                    0C   AE 9F 00075            PUSHAB  FLDADR                                       ; 1867
                                    14   AE 9F 00078            PUSHAB  FLDSIZE                                      ;
                                    00 BA49 DF 0007B            PUSHAL  @0(R10)[I]                                   ; 1868
                           7E            9E 3C 0007F            MOVZWL  @(SP)+, -(SP)                                ;
                                    20   AE 9F 00082            PUSHAB  RECDSC                                       ; 1867
              00000000G    00            04 FB 00085            CALLS   #4, NMA$SEARCHFLD                            ;
                           34            50 E9 0008C            BLBC    R0, 6$                                       ;
                           83       00 BA49 F7 0008F            CVTLW   @0(R10)[I], (PTR)+                           ; 1875
                                    00 BA49 DF 00094            PUSHAL  @0(R10)[I]                                   ; 1880
     58            9E      10            10 EF 00098            EXTZV   #16, #16, @(SP)+, CPT_INDEX                  ;
                   50      58            0A C5 0009D            MULL3   #10, CPT_INDEX, R0                           ; 1881
                              00000000G0040 9F 000A1            PUSHAB  NML$AB_CPTABLE[R0]                           ;
                           57            9E 3C 000A8            MOVZWL  @(SP)+, PST_INDEX                            ;
                   50      57            04 78 000AB            ASHL    #4, PST_INDEX, R0                            ; 1882
                           03 00000000G0040 91 000AF            CMPB    NML$AB_PRMSEM+2[R0], #3                      ;
                                         04 12 000B7            BNEQ    5$                                           ;
                           83       10   AE 90 000B9            MOVB    FLDSIZE, (PTR)+                              ; 1883
                   63      0C BE    10   AE 28 000BD 5$:        MOVC3   FLDSIZE, @FLDADR, (PTR)                      ; 1887
                   AB      59            5B F2 000C3 6$:        AOBLSS  R11, I, 4$                                   ; 1856
```

```
                                  50 00000000G  00  9E 000C7          MOVAB   NML$AB_RCVBUFFER, R0                          ; 1895
          00000000G  00           53            50  C3 000CE          SUBL3   R0, PTR, NML$GL_RCVDATLEN                     ;
                         00000000G 00            00  FB 000D6          CALLS   #0, NML$CALL_NI_CONFIG                        ; 1896
                                FF 8F 00000000G 00  91 000DD          CMPB    NML$GB_ENTITY_FORMAT, #255                    ; 1900
                                                06  12 000E5          BNEQ    7$                                            ;
                                     08         AE  D6 000E7          INCL    KEY                                           ; 1902
                                              FF51  31 000EA          BRW     2$                                            ; 1844
                                                    04 000ED 7$:      RET                                                   ; 1905

; Routine Size:  238 bytes,    Routine Base:  $CODE$ + 0CA2


; 1923          1906  1
; 1924          1907  1 END                                       ! End of module
; 1925          1908  1
; 1926          1909  0 ELUDOM




;                                   PSECT SUMMARY
;
;       Name                     Bytes                       Attributes
;
;   $OWN$                          1652  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;   $PLIT$                          100  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;   $CODE$                         3472  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         80         23         27         00:00.1
;   _$255$DUA28:[SHRLIB]NMALIBRY.L32;1        887         17          1         47         00:00.2
;   _$255$DUA28:[SHRLIB]NET.L32;1            1279          9          0         63         00:00.3
;   _$255$DUA28:[SYSLIB]STARLET.L32;1        9776          2          0        581         00:03.3




;                                  COMMAND QUALIFIERS

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

; Size:          3472 code + 1752 data bytes
; Run Time:         00:55.0
; Elapsed Time:     02:08.4
; Lines/CPU Min:    2082
```

```
; Lexemes/CPU-Min: 12416
; Memory Used:  212 pages
; Compilation Complete
```

0286 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

NMLREALOG
LIS

NMLSHOPRM
LIS

NMLSET
LIS

NMLSEDEST
LIS