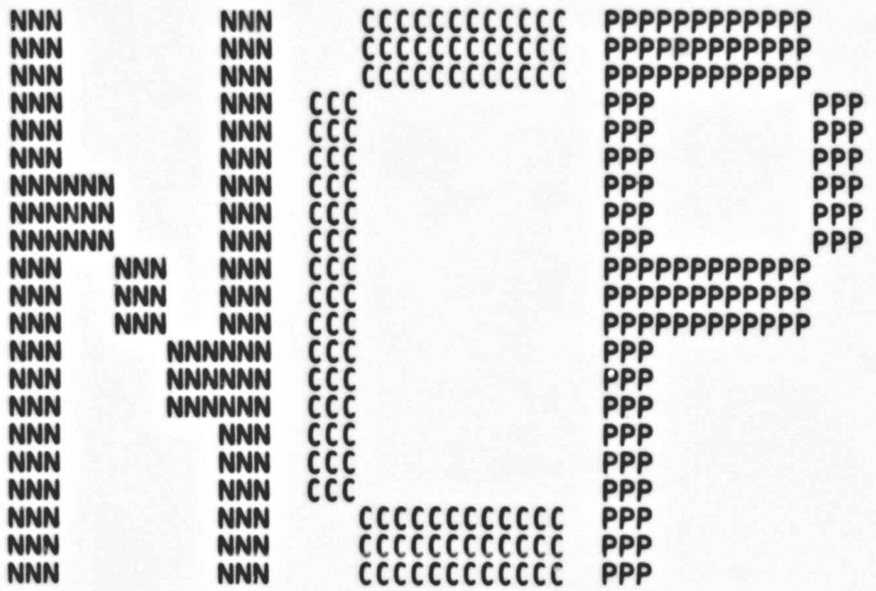
NNN NNN CCCCCCCCCCCC PPPPPPPPPPPP
NNN NNN CCCCCCCCCCCC PPPPPPPPPPPP
NNN NNN CCCCCCCCCCCC PPPPPPPPPPPP
NNN NNN CCC PPP PPP
NNN NNN CCC PPP PPP
NNN NNN CCC PPP PPP
NNNNNN NNN CCC PPP PPP
NNNNNN NNN CCC PPP PPP
NNNNNN NNN CCC PPP PPP
NNN NNN NNN CCC PPPPPPPPPPPP
NNN NNN NNN CCC PPPPPPPPPPPP
NNN NNN NNN CCC PPPPPPPPPPPP
NNN NNNNNN CCC PPP
NNN NNNNNN CCC PPP
NNN NNNNNN CCC PPP
NNN NNN CCC PPP
NNN NNN CCC PPP
NNN NNN CCC PPP
NNN NNN CCCCCCCCCCCC PPP
NNN NNN CCCCCCCCCCCC PPP
NNN NNN CCCCCCCCCCCC PPP

FILEID**NCPSHOLIS

```
NN      NN   CCCCCCCC  PPPPPPPP     SSSSSSSS  HH      HH   OOOOOO    LL          IIIIII     SSSSSSSS
NN      NN   CCCCCCCC  PPPPPPPP     SSSSSSSS  HH      HH   OOOOOO    LL          IIIIII     SSSSSSSS
NN      NN  CC         PP      PP  SS         HH      HH  OO    OO  LL            II      SS
NN      NN  CC         PP      PP  SS         HH      HH  OO    OO  LL            II      SS
NNNN    NN  CC         PP      PP  SS         HH      HH  OO    OO  LL            II      SS
NNNN    NN  CC         PP      PP  SS         HH      HH  OO    OO  LL            II      SS
NN  NN  NN  CC         PPPPPPPP     SSSSSS    HHHHHHHHHH  OO    OO  LL            II        SSSSSS
NN  NN  NN  CC         PPPPPPPP     SSSSSS    HHHHHHHHHH  OO    OO  LL            II        SSSSSS
NN    NNNN  CC         PP                SS   HH      HH  OO    OO  LL            II              SS
NN    NNNN  CC         PP                SS   HH      HH  OO    OO  LL            II              SS
NN      NN  CC         PP                SS   HH      HH  OO    OO  LL            II              SS      ....
NN      NN  CC         PP                SS   HH      HH  OO    OO  LL            II              SS      ....
NN      NN   CCCCCCCC  PP         SSSSSSSS   HH      HH   OOOOOO   LLLLLLLLLL  IIIIII   SSSSSSSS        ....
NN      NN   CCCCCCCC  PP         SSSSSSSS   HH      HH   OOOOOO   LLLLLLLLLL  IIIIII   SSSSSSSS        ....

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

```
;     1          0001  0 %TITLE  'Process Returns from SHOW and LIST'
;     2          0002  0 MODULE NCPSHOLIS (IDENT = 'V04-000',
;     3          0003  0                   ADDRESSING_MODE(EXTERNAL=GENERAL),
;     4          0004  0                   ADDRESSING_MODE(NONEXTERNAL=GENERAL)) =
;     5          0005  1 BEGIN
;     6          0006  1
;     7          0007  1 !
;     8          0008  1 !**********************************************************************************
;     9          0009  1 !*                                                                                *
;    10          0010  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                       *
;    11          0011  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                         *
;    12          0012  1 !*  ALL RIGHTS RESERVED.                                                           *
;    13          0013  1 !*                                                                                *
;    14          0014  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED          *
;    15          0015  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE        *
;    16          0016  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER         *
;    17          0017  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY         *
;    18          0018  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY         *
;    19          0019  1 !*  TRANSFERRED.                                                                   *
;    20          0020  1 !*                                                                                *
;    21          0021  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE         *
;    22          0022  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT         *
;    23          0023  1 !*  CORPORATION.                                                                   *
;    24          0024  1 !*                                                                                *
;    25          0025  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS         *
;    26          0026  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                        *
;    27          0027  1 !*                                                                                *
;    28          0028  1 !*                                                                                *
;    29          0029  1 !**********************************************************************************
;    30          0030  1 !
;    31          0031  1
;    32          0032  1 !++
;    33          0033  1 ! FACILITY:     Network Control Program (NCP)
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:
;    36          0036  1 !
;    37          0037  1 !       This module contains routines to process and print returns from
;    38          0038  1 !       show and list commands.
;    39          0039  1 !
;    40          0040  1 ! ENVIRONMENT:  VAX/VMS Operating System
;    41          0041  1 !
;    42          0042  1 ! AUTHOR:       Darrell Duffy   , CREATION DATE:  13-November-1979
;    43          0043  1 !
;    44          0044  1 ! MODIFIED BY:
;    45          0045  1 !
;    46          0046  1 !       V03-028 PRD0090         Paul R. DeStefano       28-Mar-1984
;    47          0047  1 !               Modified NCP$PARSEPARM to format a node address with
;    48          0048  1 !               area numbers when data type = event.
;    49          0049  1 !
;    50          0050  1 !       V03-027 PRD0076         Paul R. DeStefano       09-Mar-1984
;    51          0051  1 !               Remove PRD0047.  We want to see area 1 after all!
;    52          0052  1 !
;    53          0053  1 !       V03-026 PRD0047         Paul R. DeStefano       05-Feb-1984
;    54          0054  1 !               Don't output area number for area = 1.
;    55          0055  1 !
;    56          0056  1 !       V03-025 RPG0025         Bob Grosso              08-Jun-1983
;    57          0057  1 !               Correct formatting of AREA number.
```

```
;    58          0058  1 !                Columnize SHOW CONFIG STATUS output.
;    59          0059  1 !
;    60          0060  1 !        V03-024 RPG0024         Bob Grosso              03-May-1983
;    61          0061  1 !                Print circuit service data base.
;    62          0062  1 !
;    63          0063  1 !        V03-023 RPG0023         Bob Grosso              10-Mar-1983
;    64          0064  1 !                Format NI addresses with the '-'s between couplets.
;    65          0065  1 !
;    66          0066  1 !        V03-022 RPG0022         Bob Grosso              09-Mar-1983
;    67          0067  1 !                Make more room for file name in Object display.
;    68          0068  1 !
;    69          0069  1 !        V03-021 RPG0021         Bob Grosso              31-Jan-1983
;    70          0070  1 !                Fix "Loop node =" bug.
;    71          0071  1 !                Add SHOW MODULE CONFIGURATOR support.
;    72          0072  1 !
;    73          0073  1 !        V03-020 RPG0020         Bob Grosso              24-Nov-1982
;    74          0074  1 !                Add circuit to SHOW KNOWN AREA SUMMARY column display.
;    75          0075  1 !
;    76          0076  1 !        V03-019 RPG0019         Bob Grosso              24-Nov-1982
;    77          0077  1 !                Add column for circuit name in SHO AREA display.
;    78          0078  1 !
;    79          0079  1 !        V03-018 RPG0018         Bob Grosso              22-Nov-1982
;    80          0080  1 !                Clean up code by accessing areas with Vield definitions.
;    81          0081  1 !
;    82          0082  1 !        V03-017 RPG0017         Bob Grosso              18-Nov-1982
;    83          0083  1 !                Clean up displays to accommodate areas.
;    84          0084  1 !
;    85          0085  1 !        V03-016 RPG0016         Bob Grosso              28-Sep-1982
;    86          0086  1 !                Format Area entity.
;    87          0087  1 !                Widen display for SHO LOG MONITOR since it truncates events.
;    88          0088  1 !
;    89          0089  1 !        V03-015 RPG0015         Bob Grosso              10-Sep-1982
;    90          0090  1 !                Lay groundwork to format RNGL type parameters so
;    91          0091  1 !                that X-P channel lists can be printed on a single line.
;    92          0092  1 !                Restrict 'active' from appearing in header for SHOW
;    93          0093  1 !                X-P KNOWN DTES.
;    94          0094  1 !
;    95          0095  1 !        V03-014 RPG0014         Bob Grosso              09-Aug-1982
;    96          0096  1 !                Correct looping problem.
;    97          0097  1 !                Adjust NODE STATUS table.
;    98          0098  1 !
;    99          0099  1 !        V03-013 RPG0013         Bob Grosso      14-Jul-1982
;   100          0100  1 !                Support MODULE X25-Trace, X29-Server
;   101          0101  1 !
;   102          0102  1 !        V012    RPG0012         Bob Grosso      09-Jun-1982
;   103          0103  1 !                Add support for show module X25-Protocol and
;   104          0104  1 !                X25- Server.
;   105          0105  1 !                Remove numerical ordering of parameters restriction.
;   106          0106  1 !
;   107          0107  1 !        V011    TMH0011         Tim Halvorsen   01-Jun-1982
;   108          0108  1 !                Add support for display of parameters which are entity
;   109          0109  1 !                specifiers or a numeric range.
;   110          0110  1 !
;   111          0111  1 !        V010    TMH0010         Tim Halvorsen   04-Mar-1982
;   112          0112  1 !                Fix display headers to use "circuit" rather than "line".
;   113          0113  1 !
;   114          0114  1 !        V009    TMH0009         Tim Halvorsen   19-Feb-1982
```

```
;  115     0115  1 !               Fix SHOW STATUS LINKS display to show "state" column.
;  116     0116  1 !
;  117     0117  1 !       V008    TMH0008         Tim Halvorsen   11-Jan-1982
;  118     0118  1 !               Add V2.0 NICE mapping to process SHOW returns from
;  119     0119  1 !               older systems.
;  120     0120  1 !
;  121     0121  1 !       V007    TMH0007         Tim Halvorsen    2-Dec-1981
;  122     0122  1 !               Add check to verify that the NICE response to a SHOW
;  123     0123  1 !               request returned the parameters in ascending order,
;  124     0124  1 !               as an additional level of NML validation.  We will
;  125     0125  1 !               continue to output all the parameters, regardless of
;  126     0126  1 !               order, but issue a warning message about the NICE
;  127     0127  1 !               protocol violation to catch problems in NML.
;  128     0128  1 !               Reformat LINKS display to include "remote ID".
;  129     0129  1 !
;  130     0130  1 !       V006    TMH0006         Tim Halvorsen   25-Nov-1981
;  131     0131  1 !               Display source type (Line, Circuit, Node, etc)
;  132     0132  1 !               in the source column, so display distinguishes
;  133     0133  1 !               between source lines and circuits.  Display
;  134     0134  1 !               "(all sources)" for global event masks.
;  135     0135  1 !
;  136     0136  1 !       V005    TMH0005         Tim Halvorsen   11-Nov-1981
;  137     0137  1 !               Show circuit type in source column for show logging.
;  138     0138  1 !               Output "No information available" message if no parameters
;  139     0139  1 !               are returned in a show response message.
;  140     0140  1 !
;  141     0141  1 !       V004    TMH0004         Tim Halvorsen   28-Aug-1981
;  142     0142  1 !               Add compatibility for 2.0 NML show link messages.
;  143     0143  1 !
;  144     0144  1 !       V003    TMH0003         Tim Halvorsen   25-Aug-1981
;  145     0145  1 !               Display links as a single line per link
;  146     0146  1 !               rather than multi-column format.
;  147     0147  1 !
;  148     0148  1 !       V002    TMH0002         Tim Halvorsen   07-Jul-1981
;  149     0149  1 !               Complete code to display circuits
;  150     0150  1 !
;  151     0151  1 !       V001    TMH0001         Tim Halvorsen   22-Jun-1981
;  152     0152  1 !               Add Circuit entity.  Re-do display code to detect
;  153     0153  1 !               system-specific entities vs. NICE entities.
;  154     0154  1 !--
```

```
;  156          0155  1 %SBTTL  'Definitions'
;  157          0156  1
;  158          0157  1 !
;  159          0158  1 ! TABLE OF CONTENTS:
;  160          0159  1 !
;  161          0160  1
;  162          0161  1 FORWARD ROUTINE
;  163          0162  1         NCP$SHOHEAD      : NOVALUE,      ! Print heading for show/list
;  164          0163  1         NCP$SHOCOLHEAD   : NOVALUE,      ! Setup heading for column output
;  165          0164  1         NCP$SHOLIS       : NOVALUE,      ! Print all params for a return
;  166          0165  1         NCP$SHOCOLFMT    : ,             ! Produce column output
;  167          0166  1         NCP$SHOENTITY    : NOVALUE,      ! Convert entity in return
;  168          0167  1         V2_SHOW_COMPAT,                  ! V2.0 NML SHOW response compatibility
;  169          0168  1         NCP$FORMATPARM   : NOVALUE,      ! Format a parameter or counter
;  170          0169  1         NCP$PARSEPARM    : NOVALUE,      ! Parse a parameter into text
;  171          0170  1         NCP$PARSECOUNTER : NOVALUE,      ! Parse a counter into text
;  172          0171  1         NCP$PARSEVENTS   : NOVALUE,      ! Parse an event mask into text
;  173          0172  1         NCP$PTBSEARCH,                   ! Search a parameter table
;  174          0173  1         NCP$FAOSET       : NOVALUE,      ! Setup fao pointers
;  175          0174  1         NCP$ADDSTR       : NOVALUE,      ! Add one string to another
;  176          0175  1         NCP$ADDFAO       : NOVALUE,      ! Add an entry to the fao list
;  177          0176  1         NCP$FAOWRITE     : NOVALUE,      ! Convert and write fao string
;  178          0177  1         NCP$FAOL         : NOVALUE,      ! Convert fao parameters
;  179          0178  1         NCP$PRIVBITS     : NOVALUE       ! Convert privilege mask to text
;  180          0179  1         ;                                !
;  181          0180  1
;  182          0181  1 !
;  183          0182  1 ! INCLUDE FILES:
;  184          0183  1 !
;  185          0184  1
;  186          0185  1 LIBRARY 'SYS$LIBRARY:STARLET';
;  187          0186  1 LIBRARY 'LIB$:NMALIBRY';
;  188          0187  1 LIBRARY 'LIB$:NCPLIBRY';
```

```
;  190     0188  1
;  191     0189  1 !
;  192     0190  1 ! EQUATED SYMBOLS:
;  193     0191  1 !
;  194     0192  1
;  195     0193  1 LITERAL
;  196     0194  1         FAOLSTSIZ       = 100,          ! Size of the fao arg list
;  197     0195  1         CTRBUFSIZ       = 500,          ! Size of control string buffer
;  198     0196  1         OUTBUFSIZ       = 500           ! Size of fao output buffer
;  199     0197  1         ;
;  200     0198  1
```

```
;  202          0199  1
;  203          0200  1 !
;  204          0201  1 !       Headers for column output
;  205          0202  1 !
;  206          0203  1
;  207          0204  1 BIND
;  208          0205  1
;  209          0206  1     NCP$Q_COLNODSUMHDR =                ! Node Summary header
;  210          0207  1
;  211        P 0208  1         ASCID
;  212        P 0209  1         (
;  213        P 0210  1
;  214        P 0211  1 '    Node              State      Active  Delay   Circuit     Next node',
;  215        P 0212  1 %char(13), %char(10),
;  216        P 0213  1 '                                  Links',
;  217        P 0214  1 %char(13), %char(10), ' '
;  218        P 0215  1
;  219          0216  1         ),
;  220          0217  1
;  221          0218  1     NCP$Q_COLNODSTAHDR =                ! Node Status header
;  222          0219  1
;  223        P 0220  1         ASCID
;  224        P 0221  1         (
;  225        P 0222  1
;  226        P 0223  1 '    Node              State      Active  Delay  Type          Cost  Hops  Circuit',
;  227        P 0224  1 %char(13), %char(10),
;  228        P 0225  1 '                                  Links',
;  229        P 0226  1 %char(13), %char(10), ' '
;  230        P 0227  1
;  231          0228  1         ),
;  232          0229  1
;  233          0230  1     NCP$Q_COLCIRSTAHDR =                ! Circuit Status
;  234          0231  1
;  235        P 0232  1         ASCID
;  236        P 0233  1         (
;  237        P 0234  1
;  238        P 0235  1 '   Circuit           State                  Loopback     Adjacent       Block',
;  239        P 0236  1 %char(13), %char(10),
;  240        P 0237  1 '                                            Name         Node           Size',
;  241        P 0238  1 %char(13), %char(10), ' '
;  242        P 0239  1
;  243          0240  1         ),
;  244          0241  1
;  245          0242  1     NCP$Q_COLCIRSUMHDR =              ! Show Circuit Summary
;  246          0243  1
;  247        P 0244  1         ASCID
;  248        P 0245  1         (
;  249        P 0246  1
;  250        P 0247  1 '   Circuit           State                  Loopback     Adjacent',
;  251        P 0248  1 %char(13), %char(10),
;  252        P 0249  1 '                                            Name         Node',
;  253        P 0250  1 %char(13), %char(10), ' '
;  254        P 0251  1
;  255          0252  1         ),
;  256          0253  1
;  257          0254  1     NCP$Q_COLLISCIRHDR =              ! List Circuit
;  258          0255  1
```

```
;  259          P 0256  1          ASCID
;  260          P 0257  1          (
;  261          P 0258  1
;  262          P 0259  1  '   Circuit              State',
;  263          P 0260  1  %char(13), %char(10), ' '
;  264          P 0261  1
;  265             0262  1          ),
;  266             0263  1
;  267             0264  1
;  268             0265  1      NCP$Q_COLLINHDR =                  ! Line header
;  269             0266  1
;  270          P 0267  1          ASCID
;  271          P 0268  1          (
;  272          P 0269  1
;  273          P 0270  1  '   Line                State',
;  274          P 0271  1  %char(13), %char(10), ' '
;  275          P 0272  1
;  276             0273  1          ),
;  277             0274  1
;  278             0275  1      NCP$Q_COLLNKHDR =                  ! Header for links
;  279             0276  1
;  280          P 0277  1          ASCID
;  281          P 0278  1          (
;  282          P 0279  1
;  283          P 0280  1  '   Link       Node              PID     Process     Remote link  Remote user',
;  284          P 0281  1  %char(13), %char(10), ' '
;  285          P 0282  1
;  286             0283  1          ),
;  287             0284  1
;  288             0285  1      NCP$Q_COLLNKSTAHDR =               ! Header for link status
;  289             0286  1
;  290          P 0287  1          ASCID
;  291          P 0288  1          (
;  292          P 0289  1
;  293          P 0290  1  '   Link       Node              PID     Process     Remote link  State',
;  294          P 0291  1  %char(13), %char(10), ' '
;  295          P 0292  1
;  296             0293  1          ),
;  297             0294  1
;  298             0295  1      NCP$Q_COLOBJHDR =                  ! Header for Objects
;  299             0296  1
;  300          P 0297  1          ASCID
;  301          P 0298  1          (
;  302          P 0299  1
;  303          P 0300  1  '   Object   Number  File/PID                    User Id          Password',
;  304          P 0301  1  %char(13), %char(10), ' ',
;  305          P 0302  1  %char(13), %char(10)
;  306          P 0303  1
;  307             0304  1          ),
;  308             0305  1
;  309             0306  1      NCP$Q_COLLOGSUMHDR =               ! Header for logging summary
;  310             0307  1
;  311          P 0308  1          ASCID
;  312          P 0309  1          (
;  313          P 0310  1
;  314          P 0311  1  '    Sink Node       Source              Events                    State Name',
;  315          P 0312  1  %char(13), %char(10), ' ',
```

```
;   316        P 0313  1 %char(13), %char(10)
;   317        P 0314  1
;   318          0315  1         ),
;   319          0316  1
;   320          0317  1     NCP$Q_COLCNFSUMHDR =                ! Header for Show Module Configurator Summary
;   321          0318  1
;   322        P 0319  1         ASCID
;   323        P 0320  1         (
;   324        P 0321  1
;   325        P 0322  1 '  Circuit           Surveillance     Elapsed Time',
;   326        P 0323  1 %char(13), %char(10), ' '
;   327        P 0324  1
;   328          0325  1         ),
;   329          0326  1
;   330          0327  1     NCP$Q_COLCNFSTAHDR =                ! Header for Show Module Configurater Status
;   331          0328  1
;   332        P 0329  1         ASCID
;   333        P 0330  1         (
;   334        P 0331  1
;   335        P 0332  1 '  Circuit   Physical address   Last report      Hardware address',
;   336        P 0333  1 %char(13), %char(10), ' '
;   337        P 0334  1
;   338          0335  1         ),
;   339          0336  1
;   340          0337  1     NCP$Q_COLMPRGRPHDR =                ! Header for logging Chacteristics
;   341          0338  1                                         ! of Module X25-Protocol Groups
;   342        P 0339  1         ASCID
;   343        P 0340  1         (
;   344        P 0341  1
;   345        P 0342  1 ' Group               DTE                 Number    Type',
;   346        P 0343  1 %char(13), %char(10), ' ',
;   347        P 0344  1 %char(13), %char(10)
;   348        P 0345  1
;   349          0346  1         ),
;   350          0347  1
;   351          0348  1     NCP$Q_COLMPRDTEHDR =                ! Header for logging status and summary
;   352          0349  1                                         ! of LIST Module X25-Protocol DTE
;   353        P 0350  1         ASCID
;   354        P 0351  1         (
;   355        P 0352  1
;   356        P 0353  1 '  DTE                 State                   Active Channels  Active Switched',
;   357        P 0354  1 %char(13), %char(10), ' ',
;   358        P 0355  1 %char(13), %char(10)
;   359        P 0356  1
;   360          0357  1         ),
;   361          0358  1
;   362          0359  1     NCP$Q_COLMPRDTEPHDR =               ! Header for logging status and summary
;   363          0360  1                                         ! of SHOW Module X25-Protocol DTE
;   364        P 0361  1         ASCID
;   365        P 0362  1         (
;   366        P 0363  1
;   367        P 0364  1 '  DTE                 State',
;   368        P 0365  1 %char(13), %char(10), ' ',
;   369        P 0366  1 %char(13), %char(10)
;   370        P 0367  1
;   371          0368  1         ),
;   372          0369  1
```

```
;  373          0370  1     NCP$Q_COLARESTAHDR =
;  374          0371  1
;  375        P 0372  1         ASCID
;  376        P 0373  1         (
;  377        P 0374  1
;  378        P 0375  1 '  Area     State       Cost   Hops    Circuit          Next node to area',
;  379        P 0376  1 %char(13), %char(10), ' '
;  380        P 0377  1
;  381          0378  1         ),
;  382          0379  1
;  383          0380  1     NCP$Q_COLARESUMHDR =
;  384          0381  1
;  385        P 0382  1         ASCID
;  386        P 0383  1         (
;  387        P 0384  1
;  388        P 0385  1 '  Area     State       ', ' Circuit          Next node to area',
;  389        P 0386  1 %char(13), %char(10), ' '
;  390        P 0387  1
;  391          0388  2         )
;  392          0389  1     ;
```

```
;  394     0390  1
;  395     0391  1 !
;  396     0392  1 !       Parameter tables for column output
;  397     0393  1 !
;  398     0394  1
;  399     0395  1 !
;  400     0396  1 !       Tables have the following format
;  401     0397  1 !       Word (parameter id code),
;  402     0398  1 !       Byte (offset into line, size of field in line),
;  403     0399  1 !       ...
;  404     0400  1 !       Word (-1)
;  405     0401  1 !
;  406     0402  1
;  407     0403  1 BIND
;  408     0404  1
;  409     0405  1     NCP$T_COLNODSUMTBL =                ! Node summary
;  410     0406  1
;  411     0407  1         UPLIT
;  412     0408  1         (
;  413     0409  1         WORD (NMA$C_PCNO_STA), BYTE (17, 11),
;  414     0410  1         WORD (NMA$C_PCNO_ACL), BYTE (29, 5),
;  415     0411  1         WORD (NMA$C_PCNO_DEL), BYTE (37, 5),
;  416     0412  1         WORD (NMA$C_PCNO_DLI), BYTE (46, 12),
;  417     0413  1         WORD (NMA$C_PCNO_NLI), BYTE (46, 12),
;  418     0414  1         WORD (NMA$C_PCNO_NND), BYTE (59, 16),
;  419     0415  1         WORD (-1)
;  420     0416  1         )
;  421     0417  1         ,
;  422     0418  1
;  423     0419  1     NCP$T_COLNODSTATBL =                ! Node status
;  424     0420  1
;  425     0421  1         UPLIT
;  426     0422  1         (
;  427     0423  1         WORD (NMA$C_PCNO_STA), BYTE (17, 11),
;  428     0424  1         WORD (NMA$C_PCNO_ACL), BYTE (29, 5),
;  429     0425  1         WORD (NMA$C_PCNO_DEL), BYTE (37, 5),
;  430     0426  1         WORD (NMA$C_PCNO_DTY), BYTE (44, 14),
;  431     0427  1         WORD (NMA$C_PCNO_DCO), BYTE (59, 5),
;  432     0428  1         WORD (NMA$C_PCNO_DHO), BYTE (65, 5),
;  433     0429  1         WORD (NMA$C_PCNO_DLI), BYTE (72, 16),
;  434     0430  1         WORD (NMA$C_PCNO_NLI), BYTE (72, 16),
;  435     0431  1         WORD (-1)
;  436     0432  1         )
;  437     0433  1         ,
;  438     0434  1
;  439     0435  1     NCP$T_COLCIRTBL =           ! Circuit status and summary
;  440     0436  1
;  441     0437  1         UPLIT
;  442     0438  1         (
;  443     0439  1         WORD (NMA$C_PCCI_STA), BYTE (20, 7),
;  444     0440  1         WORD (NMA$C_PCCI_SUB), BYTE (29, 15),
;  445     0441  1         WORD (NMA$C_PCCI_SSB), BYTE (29, 15),   ! Substate from circuit service database
;  446     0442  1         WORD (NMA$C_PCCI_LOO), BYTE (45, 6),
;  447     0443  1         WORD (NMA$C_PCCI_ADJ), BYTE (53, 16),
;  448     0444  1         WORD (NMA$C_PCCI_SPY), BYTE (53, 17),   ! NI addr from circuit service database
;  449     0445  1         WORD (NMA$C_PCCI_BLO), BYTE (71, 5),
;  450     0446  1         WORD (-1)
```

```
;   451     0447  1          ),
;   452     0448  1
;   453     0449  1      NCP$T_COLLINTBL =           ! Line status
;   454     0450  1
;   455     0451  1          UPLIT
;   456     0452  1          (
;   457     0453  1          WORD (NMA$C_PCLI_STA), BYTE (20, 7),
;   458     0454  1          WORD (NMA$C_PCLI_SUB), BYTE (29, 15),
;   459     0455  1          WORD (NMA$C_PCLI_LOO), BYTE (45, 6),
;   460     0456  1          WORD (NMA$C_PCLI_ADJ), BYTE (53, 14),
;   461     0457  1          WORD (NMA$C_PCLI_BLO), BYTE (69, 5),
;   462     0458  1          WORD (-1)
;   463     0459  1          ),
;   464     0460  1
;   465     0461  1      NCP$T_COLLNKTBL =                   ! Link display
;   466     0462  1
;   467     0463  1          UPLIT
;   468     0464  1          (
;   469     0465  1          WORD (NMA$C_PCLK_NID), BYTE (9,16),     ! Node ID
;   470     0466  1          WORD (NMA$C_PCLK_PID), BYTE (27,8),     ! PID of local process
;   471     0467  1          WORD (NMA$C_PCLK_PRC), BYTE (37,16),    ! Local process name
;   472     0468  1          WORD (NMA$C_PCLK_RLN), BYTE (55,5),     ! Remote link number
;   473     0469  1          WORD (NMA$C_PCLK_RID), BYTE (62,16),    ! Remote ID (username or PID)
;   474     0470  1          WORD (-1)
;   475     0471  1          ),
;   476     0472  1
;   477     0473  1      NCP$T_COLLNKSTATBL =                ! Link status display
;   478     0474  1
;   479     0475  1          UPLIT
;   480     0476  1          (
;   481     0477  1          WORD (NMA$C_PCLK_NID), BYTE (9,16),     ! Node ID
;   482     0478  1          WORD (NMA$C_PCLK_PID), BYTE (27,8),     ! PID of local process
;   483     0479  1          WORD (NMA$C_PCLK_PRC), BYTE (37,16),    ! Local process name
;   484     0480  1          WORD (NMA$C_PCLK_RLN), BYTE (55,5),     ! Remote link number
;   485     0481  1          WORD (NMA$C_PCLK_STA), BYTE (62,10),    ! State
;   486     0482  1          WORD (-1)
;   487     0483  1          ),
;   488     0484  1
;   489     0485  1      NCP$T_COLOBJTBL =                   ! Object display
;   490     0486  1
;   491     0487  1          UPLIT
;   492     0488  1          (
;   493     0489  1          WORD (NMA$C_PCOB_NUM), BYTE (13,5),     ! Object number
;   494     0490  1          WORD (NMA$C_PCOB_FID), BYTE (20,26),    ! File id
;   495     0491  1          WORD (NMA$C_PCOB_PID), BYTE (20,16),    ! Process id
;   496     0492  1          WORD (NMA$C_PCOB_USR), BYTE (47,16),    ! User id
;   497     0493  1          WORD (NMA$C_PCOB_PSW), BYTE (64,16),    ! Password
;   498     0494  1          WORD (-1)
;   499     0495  1          ),
;   500     0496  1
;   501     0497  1      NCP$T_COLLOGTBL =                   ! Logging display
;   502     0498  1
;   503     0499  1          UPLIT
;   504     0500  1          (
;   505     0501  1
;   506     0502  1          WORD (NMA$C_PCLO_STA), BYTE (67,4),     ! Logging state
;   507     0503  1          WORD (NMA$C_PCLO_LNA), BYTE (62,32),    ! Logging name
```

```
;  508     0504  1          WORD (NMA$C_PCLO_EVE), BYTE (20,36),     ! Events
;  509     0505  1          WORD (NMA$C_PCLO_SIN), BYTE (3,16),      ! Sink node
;  510     0506  1          WORD (-1)
;  511     0507  1          ),
;  512     0508  1
;  513     0509  1
;  514     0510  1      NCP$T_COLCNFSUMTBL =                         ! Module Configurator summary
;  515     0511  1
;  516     0512  1          UPLIT
;  517     0513  1          (
;  518     0514  1
;  519     0515  1          WORD (NMA$C_PCCN_CIR), BYTE (2, 16),
;  520     0516  1          WORD (NMA$C_PCCN_SUR), BYTE (20, 8),
;  521     0517  1          WORD (NMA$C_PCCN_ELT), BYTE (37, 25),
;  522     0518  1          WORD (-1)
;  523     0519  1          ),
;  524     0520  1
;  525     0521  1      NCP$T_COLCNFSTATBL =                         ! Module Configurator status
;  526     0522  1
;  527     0523  1          UPLIT
;  528     0524  1          (
;  529     0525  1
;  530     0526  1          WORD (NMA$C_PCCN_CIR), BYTE (2, 8),
;  531     0527  1          WORD (NMA$C_PCCN_PHA), BYTE (12, 17),
;  532     0528  1          WORD (NMA$C_PCCN_LRP), BYTE (31, 16),
;  533     0529  1          WORD (NMA$C_PCCN_HWA), BYTE (48, 17),
;  534     0530  1          WORD (-1)
;  535     0531  1          ),
;  536     0532  1
;  537     0533  1      NCP$T_COLMPRGRPTBL =                             ! Module X25-Protocol Group
;  538     0534  1
;  539     0535  1          UPLIT
;  540     0536  1          (
;  541     0537  1
;  542     0538  1          WORD (NMA$C_PCXP_GRP), BYTE (2,16),      ! Group
;  543     0539  1          WORD (NMA$C_PCXP_GDT), BYTE (19,16),     ! DTE
;  544     0540  1          WORD (NMA$C_PCXP_GNM), BYTE (37,5),      ! Number
;  545     0541  1          WORD (NMA$C_PCXP_GTY), BYTE (45,9),      ! Type
;  546     0542  1          WORD (-1)
;  547     0543  1          ),
;  548     0544  1
;  549     0545  1      NCP$T_COLMPRDTETBL =                             ! Module X25-Protocol DTE
;  550     0546  1
;  551     0547  1          UPLIT
;  552     0548  1          (
;  553     0549  1
;  554     0550  1          WORD (NMA$C_PCXP_DTE), BYTE (2,16),      ! DTE
;  555     0551  1          WORD (NMA$C_PCXP_STA), BYTE (19,4),      ! State
;  556     0552  1          WORD (NMA$C_PCXP_SBS), BYTE (24,16),     ! Substate
;  557     0553  1          WORD (NMA$C_PCXP_ACH), BYTE (44,5),      ! Active channels
;  558     0554  1          WORD (NMA$C_PCXP_ASW), BYTE (56,5),      ! Active switched
;  559     0555  1          WORD (-1)
;  560     0556  1          ),
;  561     0557  1
;  562     0558  1      NCP$T_COLARESTATBL =                     ! Area status
;  563     0559  1
;  564     0560  1          UPLIT
```

```
;  565     0561  1          (
;  566     0562  1          WORD (NMA$C_PCAR_STA), BYTE (10, 11),
;  567     0563  1          WORD (NMA$C_PCAR_COS), BYTE (20, 5),
;  568     0564  1          WORD (NMA$C_PCAR_HOP), BYTE (28, 5),
;  569     0565  1          WORD (NMA$C_PCAR_CIR), BYTE (35, 16),
;  570     0566  1          WORD (NMA$C_PCAR_NND), BYTE (53, 16),
;  571     0567  1          WORD (-1)
;  572     0568  1          )
;  573     0569  1          ,
;  574     0570  1
;  575     0571  1      NCP$T_COLARESUMTBL =                ! Area summary
;  576     0572  1
;  577     0573  1          UPLIT
;  578     0574  1          (
;  579     0575  1          WORD (NMA$C_PCAR_STA), BYTE (10, 11),
;  580     0576  1          WORD (NMA$C_PCAR_CIR), BYTE (26, 16),
;  581     0577  1          WORD (NMA$C_PCAR_NND), BYTE (44, 16),
;  582     0578  1          WORD (-1)
;  583     0579  1          )
;  584     0580  1
;  585     0581  1          ;
```

```
;  587          0582  1
;  588          0583  1 !
;  589          0584  1 ! OWN STORAGE:
;  590          0585  1 !
;  591          0586  1
;  592          0587  1 OWN
;  593          0588  1         FAOPTR,                         ! Pointer to fao list
;  594          0589  1         FAOLST : VECTOR [FAOLSTSIZ],    ! List of args for FAO
;  595          0590  1         CTRBUF : VECTOR [CTRBUFSIZ, BYTE], ! Control string for fao
;  596          0591  1         OUTDSC : VECTOR [2],            ! Descriptor of output buffer
;  597          0592  1         OUTBUF : VECTOR [OUTBUFSIZ,BYTE], ! Fao output buffer
;  598          0593  1         NCP$A_COLHEAD,                  ! Address of descriptor for header
;  599          0594  1         NCP$B_COLHEAD   : BYTE,         ! True for head printed
;  600          0595  1         NCP$B_LOGTYPE   : BYTE,         ! Type code for logging
;  601          0596  1         NCP$A_COLTBL;                   ! Table for column output
;  602          0597  1
;  603          0598  1 GLOBAL
;  604          0599  1         NCP$GQ_CTRDSC : VECTOR [3]      ! Descriptor of control string
;  605          0600  1         ;
;  606          0601  1
;  607          0602  1 BIND
;  608          0603  1         CTRDSC = NCP$GQ_CTRDSC : VECTOR [3] ! Local alias
;  609          0604  1         ;
;  610          0605  1
;  611          0606  1 !
;  612          0607  1 ! EXTERNAL REFERENCES:
;  613          0608  1 !
;  614          0609  1
;  615          0610  1 EXTERNAL LITERAL
;  616          0611  1         NCP$_BADMSG,                    ! Invalid management message
;  617          0612  1         NCP$_LOGIC;                     ! NCP logic error
;  618          0613  1
;  619          0614  1 EXTERNAL
;  620          0615  1         NCP$GA_TBL_EVESOUR,             ! Event source table
;  621          0616  1         NCP$GL_OPTION : BBLOCK,         ! Option byte
;  622          0617  1         NCP$GL_ENTITY : SIGNED,         ! Entity type code.  If negative,
;  623          0618  1                                         ! system-specific entity (NMA$C_SENT_)
;  624          0619  1         NCP$GL_MODTYP,                  ! Type of MODULE entity
;  625          0620  1         NCP$GW_PRMTYP,                  ! Type of parameter
;  626          0621  1         PDB$G_VRB_ENT : BBLOCK;         ! Plural entity type code
;  627          0622  1                                         ! (Known, Active, etc.)
;  628          0623  1
;  629          0624  1 EXTERNAL
;  630          0625  1         NCP$GA_PRM_NOD,                 ! Parameter tables
;  631          0626  1         NCP$GA_PRM_CIR,
;  632          0627  1         NCP$GA_PRM_LIN,
;  633          0628  1         NCP$GA_PRM_LOG,
;  634          0629  1         NCP$GA_PRM_MODCNF,              ! module Configurator
;  635          0630  1 !!      NCP$GA_PRM_MODCNS,              ! module Console
;  636          0631  1 !!!     NCP$GA_PRM_MODLOA,              ! module Loader
;  637          0632  1 !!      NCP$GA_PRM_MODLOO,              ! module Looper
;  638          0633  1         NCP$GA_PRM_MODACC,              ! module X25-Access
;  639          0634  1         NCP$GA_PRM_MODPRO,              ! module X25-Protocol
;  640          0635  1         NCP$GA_PRM_MODSER,              ! module X25-Server
;  641          0636  1         NCP$GA_PRM_MODTRC,              ! module X25-Trace
;  642          0637  1         NCP$GA_PRM_ARE,                 ! Area parameter table
;  643          0638  1         NCP$GA_PRM_OBJ,
```

```
;  644     0639  1          NCP$GA_PRM_LNK,
;  645     0640  1          NCP$GA_PRM_CIRCNTR,          ! Circuit counter table
;  646     0641  1          NCP$GA_PRM_LINCNTR,          ! Line counter table
;  647     0642  1          NCP$GA_PRM_NODCNTR,          ! Node counter table
;  648     0643  1          NCP$GA_PRM_MPRCNTR,          ! Module X25-Protocol counter table
;  649     0644  1          NCP$GA_PRM_MSECNTR;          ! Module X25-Server counter table
;  650     0645  1
;  651     0646  1  EXTERNAL ROUTINE
;  652     0647  1          NCP$TABLESEARCH,             ! Search for table entry
;  653     0648  1          NCP$WRITESHO : NOVALUE       ! Write a message to show/list output
;  654     0649  1          ;
```

```
;  656     0650  1 %SBTTL  'NCP$SHOHEAD  Display Heading for SHOW and LIST'
;  657     0651  1 GLOBAL ROUTINE NCP$SHOHEAD :NOVALUE =   !
;  658     0652  1
;  659     0653  1 !++
;  660     0654  1 ! FUNCTIONAL DESCRIPTION:
;  661     0655  1 !
;  662     0656  1 !       Create and display the standard heading for show and list data.
;  663     0657  1 !       The parts come from the option byte and the entity which were
;  664     0658  1 !       used to build the message originally.
;  665     0659  1 !
;  666     0660  1 ! FORMAL PARAMETERS:
;  667     0661  1 !
;  668     0662  1 !       NONE
;  669     0663  1 !
;  670     0664  1 ! IMPLICIT INPUTS:
;  671     0665  1 !
;  672     0666  1 !       NCP$GL_OPTION           Option byte
;  673     0667  1 !       NCP$GL_ENTITY           Entity type code (negative if system-specific)
;  674     0668  1 !       PDB$G_VRB_ENT           Plural entity code (known, active, etc.)
;  675     0669  1 !
;  676     0670  1 ! IMPLICIT OUTPUTS:
;  677     0671  1 !
;  678     0672  1 !       NONE
;  679     0673  1 !
;  680     0674  1 ! ROUTINE VALUE:
;  681     0675  1 ! COMPLETION CODES:
;  682     0676  1 !
;  683     0677  1 !       NONE
;  684     0678  1 !
;  685     0679  1 ! SIDE EFFECTS:
;  686     0680  1 !
;  687     0681  1 !       NONE
;  688     0682  1 !
;  689     0683  1 !--
;  690     0684  1
;  691     0685  2     BEGIN
;  692     0686  2
;  693     0687  2     NCP$FAOSET ();                      ! Set the fao pointers
;  694     0688  2
;  695     0689  2     ADDSTR (' !/ !/');                  ! Setup some formatting
;  696     0690  2
;  697     0691  2     SELECTONE .(PDB$G_VRB_ENT [PDB$T_DATA]) <0, 8, 1>
;  698     0692  2     OF                                  ! Plural entity formats
;  699     0693  2         SET
;  700     0694  2         [NMA$C_ENT_ACT] : ADDSTR ('Active ');
;  701     0695  2         [NMA$C_ENT_LOO] : ADDSTR ('Loop ');
;  702     0696  2         [NMA$C_ENT_KNO] : ADDSTR ('Known ');
;  703     0697  2         TES;
;  704     0698  2
;  705     0699  2     SELECTONE .NCP$GL_ENTITY            ! The type of entity
;  706     0700  2     OF
;  707     0701  2         SET
;  708     0702  2         [NMA$C_ENT_NOD] : ADDSTR ('Node');
;  709     0703  2         [NMA$C_ENT_LIN] : ADDSTR ('Line');
;  710     0704  2         [NMA$C_ENT_LOG] : ADDSTR ('Logging');
;  711     0705  2         [NMA$C_ENT_CIR] : ADDSTR ('Circuit');
;  712     0706  2         [NMA$C_ENT_MOD] :
```

```
;   713     0707  3                 BEGIN
;   714     0708  3                 ADDSTR ('Module');
;   715     0709  3                 SELECTONE .NCP$GL_MODTYP OF
;   716     0710  3                     SET
;   717     0711  3                     [NCP$C_ENT_MODCNF] : ADDSTR (' Configurator');
;   718     0712  3  !                  [NCP$C_ENT_MODCNS] : ADDSTR (' Console');
;   719     0713  3  !                  [NCP$C_ENT_MODLOA] : ADDSTR (' Loader');
;   720     0714  3  !                  [NCP$C_ENT_MODLOO] : ADDSTR (' Looper');
;   721     0715  3                     [NCP$C_ENT_MODACC] : ADDSTR (' X25-Access');
;   722     0716  3                     [NCP$C_ENT_MODPRO] : ADDSTR (' X25-Protocol');
;   723     0717  3                     [NCP$C_ENT_MODSER] : ADDSTR (' X25-Server');
;   724     0718  3                     [NCP$C_ENT_MODTRC] : ADDSTR (' X25-Trace');
;   725     0719  3                     [NCP$C_ENT_MOD29S] : ADDSTR (' X29-Server');
;   726     0720  3                     TES;
;   727     0721  3                 END;
;   728     0722  2             [NMA$C_ENT_ARE] : ADDSTR ('Area');
;   729     0723  2             [-NMA$C_SENT_OBJ] : ADDSTR ('Object');
;   730     0724  2             [-NMA$C_SENT_LNK] : ADDSTR ('Link');
;   731     0725  2             TES;
;   732     0726  2
;   733     0727  2         IF .NCP$GL_OPTION [NMA$V_OPT_INF] NEQ NMA$C_OPINF_COU
;   734     0728  2         THEN
;   735     0729  3             BEGIN
;   736     0730  3             IF .NCP$GL_OPTION [NMA$V_OPT_PER] ! Indicate this is SHOW or LIST
;   737     0731  3             THEN
;   738     0732  3                 ADDSTR (' Permanent')       ! List
;   739     0733  3             ELSE
;   740     0734  3                 ADDSTR (' Volatile')        ! Show
;   741     0735  3             END;
;   742     0736  2
;   743     0737  2         SELECTONE .NCP$GL_OPTION [NMA$V_OPT_INF]    ! Information type
;   744     0738  2         OF
;   745     0739  2             SET
;   746     0740  2             [NMA$C_OPINF_SUM] : ADDSTR (' Summary');
;   747     0741  2             [NMA$C_OPINF_STA] : ADDSTR (' Status');
;   748     0742  2             [NMA$C_OPINF_CHA] : ADDSTR (' Characteristics');
;   749     0743  2             [NMA$C_OPINF_COU] : ADDSTR (' Counters');
;   750     0744  2             [NMA$C_OPINF_EVE] : ADDSTR (' Events');
;   751     0745  2             TES;
;   752     0746  2
;   753     0747  2         ADDSTR (' as of !20%D!/ ');         ! And the exact date and time
;   754     0748  2         ADDFAO (0);                         ! Quadword code for system datetime
;   755     0749  2         ADDFAO (0);
;   756     0750  2
;   757     0751  2         NCP$FAOWRITE ();                    ! Convert and write it to file
;   758     0752  2
;   759     0753  2         NCP$SHOCOLHEAD ();                  ! Set up for column header output
;   760     0754  2
;   761     0755  2         RETURN
;   762     0756  2
;   763     0757  1         END;


                                                    .TITLE  NCPSHOLIS Process Returns from SHOW and LIST
                                                    .IDENT  \V04-000\

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

```
20 20 20 20 20 20 20 65 64 6F 4E 20 20 20 20 00000 P.AAB:  .ASCII  \    Node            State      Active  De\
20 20 20 20 20 20 65 74 61 74 53 20 20 20 20 0000F
                  65 44 20 20 65 76 69 74 63 41 0001E
20 20 74 69 75 63 72 69 43 20 20 20 79 61 6C 00028        .ASCII  \lay   Circuit     Next node\<13><10>\ \
20 0A 0D 65 64 6F 6E 20 74 78 65 4E 20 20 20 00037
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 00046        .ASCII  \                                Links\<13>
4C 20 20 20 20 20 20 20 20 20 20 20 20 20 20 00055
                              0D 73 6B 6E 69 00064
                                 00 20 0A 00069        .ASCII  <10>\ \<0>
                                    0000006B 0006C P.AAA: .LONG   107
                                    00000000' 00070       .ADDRESS P.AAB
20 20 20 20 20 20 20 65 64 6F 4E 20 20 20 20 00074 P.AAD:  .ASCII  \    Node            State      Active  De\
20 20 20 20 20 20 65 74 61 74 53 20 20 20 20 00083
                  65 44 20 20 65 76 69 74 63 41 00092
20 20 20 20 20 20 65 70 79 54 20 20 79 61 6C 0009C        .ASCII  \lay  Type           Cost  Hops  Circuit-
73 70 6F 48 20 20 74 73 6F 43 20 20 20 20 20 000AB                \<13>
                  0D 74 69 75 63 72 69 43 20 20 000BA
20 20 20 20 20 20 20 20 20 20 20 20 20 20 0A 000C4        .ASCII  <10>\                              Link\
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 000D3
                              6B 6E 69 4C 20 000E2
                              00 20 0A 0D 73 000E7        .ASCII  \s\<13><10>\ \<0>
                                    00000077 000EC P.AAC: .LONG   119
                                    00000000' 000F0       .ADDRESS P.AAD
20 20 20 20 20 74 69 75 63 72 69 43 20 20 20 000F4 P.AAF:  .ASCII  \   Circuit          State                \
20 20 20 20 20 65 74 61 74 53 20 20 20 20 20 00103
                  20 20 20 20 20 20 20 20 20 20 00112
20 20 20 6B 63 61 62 70 6F 6F 4C 20 20 20 20 0011C        .ASCII  \    Loopback     Adjacent       Block-
20 20 20 20 20 74 6E 65 63 61 6A 64 41 20 20 0012B                \<13>
                  0D 6B 63 6F 6C 42 20 20 0013A
20 20 20 20 20 20 20 20 20 20 20 20 20 20 0A 00142        .ASCII  <10>\                                  \
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 00151
                              20 20 20 20 20 00160
6D 61 4E 20 20 20 20 20 20 20 20 20 20 20 20 00165        .ASCII  \              Name          Node          Si\
20 65 64 6F 4E 20 20 20 20 20 20 20 20 20 65 00174
                  69 53 20 20 20 20 20 20 20 20 00183
                        00 00 20 0A 0D 65 7A 0018D        .ASCII  \ze\<13><10>\ \<0><0>
                                    0000009E 00194 P.AAE: .LONG   158
                                    00000000' 00198       .ADDRESS P.AAF
20 20 20 20 20 74 69 75 63 72 69 43 20 20 20 0019C P.AAH:  .ASCII  \   Circuit          State                \
20 20 20 20 20 65 74 61 74 53 20 20 20 20 20 001AB
                  20 20 20 20 20 20 20 20 20 20 001BA
20 20 20 6B 63 61 62 70 6F 6F 4C 20 20 20 20 001C4        .ASCII  \    Loopback     Adjacent\<13><10>\  \
   20 20 0A 0D 74 6E 65 63 61 6A 64 41 20 20 20 001D3
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 001E1        .ASCII  \                                      \
20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 001F0
                  20 20 20 20 20 20 20 20 20 20 001FF
20 20 20 20 20 20 20 65 6D 61 4E 20 20 20 20 00209        .ASCII  \    Name          Node\<13><10>\ \<0>
                  00 20 0A 0D 65 64 6F 4E 20 20 00218
                              00 00 00222       .ASCII  <0><0>
                                    00000085 00224 P.AAG: .LONG   133
                                    00000000' 00228       .ADDRESS P.AAH
20 20 20 20 20 74 69 75 63 72 69 43 20 20 20 0022C P.AAJ:  .ASCII  \   Circuit          State\<13><10>\ \
   20 0A 0D 65 74 61 74 53 20 20 20 20 20 20 0023B
                                    0000001C 00248 P.AAI: .LONG   28
                                    00000000' 0024C       .ADDRESS P.AAJ
20 20 20 20 20 20 20 20 65 6E 69 4C 20 20 20 00250 P.AAL:  .ASCII  \   Line            State\<13><10>\ \
```

```
                20  0A  0D  65  74  61  74  53  20  20  20  20  20  0025F
                                                0000001C    0026C P.AAK:  .LONG   28
                                                00000000'   00270         .ADDRESS P.AAL
4E  20  20  20  20  20  20  20  6B  6E  69  4C  20  20  20   00274 P.AAN:  .ASCII  \   Link       Node            PID     Pro\
50  20  20  20  20  20  20  20  20  20  20  65  64  6F  00283
                    6F  72  50  20  20  20  20  20  44  49  00292
65  74  6F  6D  65  52  20  20  20  20  20  73  73  65  63   0029C         .ASCII  \cess     Remote link  Remote user\<13>-
75  20  65  74  6F  6D  65  52  20  20  6B  6E  69  6C  20   002AB                 <10>
                                    0A  0D  72  65  73  00  002BA
                                                20  002BF         .ASCII  \ \
                                                0000004C    002C0 P.AAM:  .LONG   76
                                                00000000'   002C4         .ADDRESS P.AAN
4E  20  20  20  20  20  20  20  6B  6E  69  4C  20  20  20   002C8 P.AAP:  .ASCII  \   Link       Node            PID     Pro\
50  20  20  20  20  20  20  20  20  20  20  65  64  6F  002D7
                    6F  72  50  20  20  20  20  20  44  49  002E6
65  74  6F  6D  65  52  20  20  20  20  20  73  73  65  63   002F0         .ASCII  \cess     Remote link  State\<13><10>\ \
20  0A  0D  65  74  61  74  53  20  20  6B  6E  69  6C  20   002FF
                                    00  00  0030E         .ASCII  <0><0>
                                                00000046    00310 P.AAO:  .LONG   70
                                                00000000'   00314         .ADDRESS P.AAP
6D  75  4E  20  20  20  74  63  65  6A  62  4F  20  20  20   00318 P.AAR:  .ASCII  \   Object   Number  File/PID          \
20  20  44  49  50  2F  65  6C  69  46  20  20  72  65  62   00327
                    20  20  20  20  20  20  20  20  20  20  00336
20  64  49  20  72  65  73  55  20  20  20  20  20  20  20   00340         .ASCII  \       User Id          Password\<13><10>
6F  77  73  73  61  50  20  20  20  20  20  20  20  20  20   0034F
                                    0A  0D  64  72  0035E
                            00  00  00  0A  0D  20  00362         .ASCII  \ \<13><10><0><0><0>
                                                0000004D    00368 P.AAQ:  .LONG   77
                                                00000000'   0036C         .ADDRESS P.AAR
20  20  65  64  6F  4E  20  6B  6E  69  53  20  20  20  20   00370 P.AAT:  .ASCII  \    Sink Node       Source            \
20  20  20  20  65  63  72  75  6F  53  20  20  20  20  20   0037F
                    20  20  20  20  20  20  20  20  20  20  0038E
20  20  20  20  20  20  20  20  73  74  6E  65  76  45  20   00398         .ASCII  \ Events                    State Name\<13>
74  61  74  53  20  20  20  20  20  20  20  20  20  20  20   003A7
                                0D  65  6D  61  4E  20  65   003B6
                                00  00  00  0A  0D  20  0A   003BD         .ASCII  <10>\ \<13><10><0><0><0>
                                                00000051    003C4 P.AAS:  .LONG   81
                                                00000000'   003C8         .ADDRESS P.AAT
20  20  20  20  20  20  74  69  75  63  72  69  43  20  20   003CC P.AAV:  .ASCII  \  Circuit           Surveillance     Ela\
6E  61  6C  6C  69  65  76  72  75  53  20  20  20  20  20   003DB
                    61  6C  45  20  20  20  20  20  65  63  003EA
                20  0A  0D  65  6D  69  54  20  64  65  73  70  003F4         .ASCII  \psed Time\<13><10>\ \
                                                00000034    00400 P.AAU:  .LONG   52
                                                00000000'   00404         .ADDRESS P.AAV
79  68  50  20  20  20  74  69  75  63  72  69  43  20  20   00408 P.AAX:  .ASCII  \  Circuit   Physical address   Last repo\
20  20  73  73  65  72  64  64  61  20  6C  61  63  69  73   00417
                    6F  70  65  72  20  74  73  61  4C  20  00426
72  61  77  64  72  61  48  20  20  20  20  20  20  74  72   00430         .ASCII  \rt      Hardware address\<13><10>\ \<0>
                00  20  0A  0D  73  73  65  72  64  64  61  20  65  0043F
                                                00000043    0044C P.AAW:  .LONG   67
                                                00000000'   00450         .ADDRESS P.AAX
20  20  20  20  20  20  20  20  20  70  75  6F  72  47  20   00454 P.AAZ:  .ASCII  \ Group              DTE               Numb\
20  20  20  20  20  20  20  20  45  54  44  20  20  20  20   00463
                    62  6D  75  4E  20  20  20  20  20  20  00472
0A  0D  20  0A  0D  65  70  79  54  20  20  20  20  72  65   0047C         .ASCII  \er    Type\<13><10>\ \<13><10><0>
                                                00  0048B
                                                00000037    0048C P.AAY:  .LONG   55
```

```
                                                  00000000'  00490                .ADDRESS P.AAZ
20 20 20 20 20 20 20 20 20 20 45 54 44 20 20  00494 P.ABB:   .ASCII  \  DTE                State                \
20 20 20 20 20 20 65 74 61 74 53 20 20 20 20  004A3
                  20 20 20 20 20 20 20 20 20  004B2
6E 6E 61 68 43 20 65 76 69 74 63 41 20 20 20  004BC          .ASCII  \   Active Channels  Active Switched\<13>
69 77 53 20 65 76 69 74 63 41 20 20 73 6C 65  004CB
                              0D 64 65 68 63 74  004DA
                                 0A 0D 20 0A     004E0          .ASCII  <10>\ \<13><10>
                                       00000050  004E4 P.ABA:   .LONG   80
                                       00000000' 004E8          .ADDRESS P.ABB
20 20 20 20 20 20 20 20 20 20 45 54 44 20 20  004EC P.ABD:   .ASCII  \  DTE                State\<13><10>\ \<13>
   0D 20 0A 0D 65 74 61 74 53 20 20 20 20     004FB
                                 00 00 00 0A     00508          .ASCII  <10><0><0><0>
                                       0000001D  0050C P.ABC:   .LONG   29
                                       00000000' 00510          .ADDRESS P.ABD
65 74 61 74 53 20 20 20 20 61 65 72 41 20 20  00514 P.ABF:   .ASCII  \  Area    State      Cost   Hops    Circ\
6F 48 20 20 20 74 73 6F 43 20 20 20 20 20 20  00523
                  63 72 69 43 20 20 20 20 73  00532
4E 20 20 20 20 20 20 20 20 20 20 20 74 69 75  0053C          .ASCII  \uit           Next node to area\<13><10>
65 72 61 20 6F 74 20 65 64 6F 6E 20 74 78 65  0054B
                                 0A 0D 61     0055A
                                 00 00 20     0055D          .ASCII  \ \<0><0>
                                       0000004A  00560 P.ABE:   .LONG   74
                                       00000000' 00564          .ADDRESS P.ABF
65 74 61 74 53 20 20 20 20 61 65 72 41 20 20  00568 P.ABH:   .ASCII  \  Area    State          Circuit          \
75 63 72 69 43 20 20 20 20 20 20 20 20 20 20  00577
                  20 20 20 20 20 20 20 20 74  00586
6F 74 20 65 64 6F 6E 20 74 78 65 4E 20 20 20  00590          .ASCII  \   Next node to area\<13><10>\ \<0>
                        00 20 0A 0D 61 65 72  0059F
                                       0000003F  005A8 P.ABG:   .LONG   63
                                       00000000' 005AC          .ADDRESS P.ABH
                                          0000   005B0 P.ABI:   .WORD   0
                                       0B  11    005B2          .BYTE   17, 11
                                         0258    005B4          .WORD   600
                                       05  1D    005B6          .BYTE   29, 5
                                         0259    005B8          .WORD   601
                                       05  25    005BA          .BYTE   37, 5
                                         0336    005BC          .WORD   822
                                       0C  2E    005BE          .BYTE   46, 12
                                         01F5    005C0          .WORD   501
                                       0C  2E    005C2          .BYTE   46, 12
                                         033E    005C4          .WORD   830
                                       10  3B    005C6          .BYTE   59, 16
                                         FFFF    005C8          .WORD   -1
                                                 005CA          .BLKB   2
                                          0000   005CC P.ABJ:   .WORD   0
                                       0B  11    005CE          .BYTE   17, 11
                                         0258    005D0          .WORD   600
                                       05  1D    005D2          .BYTE   29, 5
                                         0259    005D4          .WORD   601
                                       05  25    005D6          .BYTE   37, 5
                                         032A    005D8          .WORD   810
                                       0E  2C    005DA          .BYTE   44, 14
                                         0334    005DC          .WORD   820
                                       05  3B    005DE          .BYTE   59, 5
                                         0335    005E0          .WORD   821
                                       05  41    005E2          .BYTE   65, 5
```

```
                0336  005E4          .WORD   822
              10  48  005E6          .BYTE   72, 16
                01F5  005E8          .WORD   501
              10  48  005EA          .BYTE   72, 16
                FFFF  005EC          .WORD   -1
                      005EE          .BLKB   2
                0000  005F0 P.ABK:   .WORD   0
              07  14  005F2          .BYTE   20, 7
                0001  005F4          .WORD   1
              0F  1D  005F6          .BYTE   29, 15
                0079  005F8          .WORD   121
              0F  1D  005FA          .BYTE   29, 15
                0190  005FC          .WORD   400
              06  2D  005FE          .BYTE   45, 6
                0320  00600          .WORD   800
              10  35  00602          .BYTE   53, 16
                0078  00604          .WORD   120
              11  35  00606          .BYTE   53, 17
                032A  00608          .WORD   810
              05  47  0060A          .BYTE   71, 5
                FFFF  0060C          .WORD   -1
                      0060E          .BLKB   2
                0000  00610 P.ABL:   .WORD   0
              07  14  00612          .BYTE   20, 7
                0001  00614          .WORD   1
              0F  1D  00616          .BYTE   29, 15
                0190  00618          .WORD   400
              06  2D  0061A          .BYTE   45, 6
                0320  0061C          .WORD   800
              0E  35  0061E          .BYTE   53, 14
                032A  00620          .WORD   810
              05  45  00622          .BYTE   69, 5
                FFFF  00624          .WORD   -1
                      00626          .BLKB   2
                0066  00628 P.ABM:   .WORD   102
              10  09  0062A          .BYTE   9, 16
                0065  0062C          .WORD   101
              08  1B  0062E          .BYTE   27, 8
                0083  00630          .WORD   131
              10  25  00632          .BYTE   37, 16
                0078  00634          .WORD   120
              05  37  00636          .BYTE   55, 5
                0079  00638          .WORD   121
              10  3E  0063A          .BYTE   62, 16
                FFFF  0063C          .WORD   -1
                      0063E          .BLKB   2
                0066  00640 P.ABN:   .WORD   102
              10  09  00642          .BYTE   9, 16
                0065  00644          .WORD   101
              08  1B  00646          .BYTE   27, 8
                0083  00648          .WORD   131
              10  25  0064A          .BYTE   37, 16
                0078  0064C          .WORD   120
              05  37  0064E          .BYTE   55, 5
                0000  00650          .WORD   0
              0A  3E  00652          .BYTE   62, 10
                FFFF  00654          .WORD   -1
```

```
                     00656                .BLKB   2
   0201              00658 P.ABO:         .WORD   513
05  0D               0065A                .BYTE   13, 5
   0212              0065C                .WORD   530
1A  14               0065E                .BYTE   20, 26
   0217              00660                .WORD   535
10  14               00662                .BYTE   20, 16
   0226              00664                .WORD   550
10  2F               00666                .BYTE   47, 16
   0228              00668                .WORD   552
10  40               0066A                .BYTE   64, 16
   FFFF              0066C                .WORD   -1
                     0066E                .BLKB   2
   0000              00670 P.ABP:         .WORD   0
04  43               00672                .BYTE   67, 4
   0064              00674                .WORD   100
20  3E               00676                .BYTE   62, 32
   00C9              00678                .WORD   201
24  14               0067A                .BYTE   20, 36
   00C8              0067C                .WORD   200
10  03               0067E                .BYTE   3, 16
   FFFF              00680                .WORD   -1
                     00682                .BLKB   2
   0064              00684 P.ABQ:         .WORD   100
10  02               00686                .BYTE   2, 16
   006E              00688                .WORD   110
08  14               0068A                .BYTE   20, 8
   006F              0068C                .WORD   111
19  25               0068E                .BYTE   37, 25
   FFFF              00690                .WORD   -1
                     00692                .BLKB   2
   0064              00694 P.ABR:         .WORD   100
08  02               00696                .BYTE   2, 8
   0078              00698                .WORD   120
11  0C               0069A                .BYTE   12, 17
   0082              0069C                .WORD   130
10  1F               0069E                .BYTE   31, 16
   4E27              006A0                .WORD   20007
11  30               006A2                .BYTE   48, 17
   FFFF              006A4                .WORD   -1
                     006A6                .BLKB   2
   044D              006A8 P.ABS:         .WORD   1101
10  02               006AA                .BYTE   2, 16
   0492              006AC                .WORD   1170
10  13               006AE                .BYTE   19, 16
   0493              006B0                .WORD   1171
05  25               006B2                .BYTE   37, 5
   0494              006B4                .WORD   1172
09  2D               006B6                .BYTE   45, 9
   FFFF              006B8                .WORD   -1
                     006BA                .BLKB   2
   044C              006BC P.ABT:         .WORD   1100
10  02               006BE                .BYTE   2, 16
   0000              006C0                .WORD   0
04  13               006C2                .BYTE   19, 4
   0AA0              006C4                .WORD   2720
10  18               006C6                .BYTE   24, 16
```

```
                                                        03E8 006C8          .WORD   1000
                                                      05  2C 006CA          .BYTE   44, 5
                                                        03F2 006CC          .WORD   1010
                                                      05  38 006CE          .BYTE   56, 5
                                                        FFFF 006D0          .WORD   -1
                                                             006D2          .BLKB   2
                                                        0000 006D4 P.ABU:   .WORD   0
                                                      0B  0A 006D6          .BYTE   10, 11
                                                        0334 006D8          .WORD   820
                                                      05  14 006DA          .BYTE   20, 5
                                                        0335 006DC          .WORD   821
                                                      05  1C 006DE          .BYTE   28, 5
                                                        0336 006E0          .WORD   822
                                                      10  23 006E2          .BYTE   35, 16
                                                        033E 006E4          .WORD   830
                                                      10  35 006E6          .BYTE   53, 16
                                                        FFFF 006E8          .WORD   -1
                                                             006EA          .BLKB   2
                                                        0000 006EC P.ABV:   .WORD   0
                                                      0B  0A 006EE          .BYTE   10, 11
                                                        0336 006F0          .WORD   822
                                                      10  1A 006F2          .BYTE   26, 16
                                                        033E 006F4          .WORD   830
                                                      10  2C 006F6          .BYTE   44, 16
                                                        FFFF 006F8          .WORD   -1
                                  2F  21  20  2F  21  20  06 006FA P.ABW:   .ASCII  <6>\ !/ !/\
                              20  65  76  69  74  63  41  07 00701 P.ABX:   .ASCII  <7>\Active \
                                      20  70  6F  6F  4C  05 00709 P.ABY:   .ASCII  <5>\Loop \
                                  20  6E  77  6F  6E  4B  06 0070F P.ABZ:   .ASCII  <6>\Known \
                                          65  64  6F  4E  04 00716 P.ACA:   .ASCII  <4>\Node\
                                          65  6E  69  4C  04 0071B P.ACB:   .ASCII  <4>\Line\
                              67  6E  69  67  67  6F  4C  07 00720 P.ACC:   .ASCII  <7>\Logging\
                              74  69  75  63  72  69  43  07 00728 P.ACD:   .ASCII  <7>\Circuit\
                                  65  6C  75  64  6F  4D  06 00730 P.ACE:   .ASCII  <6>\Module\
      72  6F  74  61  72  75  67  69  66  6E  6F  43  20  0D 00737 P.ACF:   .ASCII  <13>\ Configurator\
              73  73  65  63  63  41  2D  35  32  58  20  0B 00745 P.ACG:   .ASCII  <11>\ X25-Access\
      6C  6F  63  6F  74  6F  72  50  2D  35  32  58  20  0D 00751 P.ACH:   .ASCII  <13>\ X25-Protocol\
              72  65  76  72  65  53  2D  35  32  58  20  0B 0075F P.ACI:   .ASCII  <11>\ X25-Server\
                  65  63  61  72  54  2D  35  32  58  20  0A 0076B P.ACJ:   .ASCII  <10>\ X25-Trace\
              72  65  76  72  65  53  2D  39  32  58  20  0B 00776 P.ACK:   .ASCII  <11>\ X29-Server\
                                          61  65  72  41  04 00782 P.ACL:   .ASCII  <4>\Area\
                                  74  63  65  6A  62  4F  06 00787 P.ACM:   .ASCII  <6>\Object\
                                          6B  6E  69  4C  04 0078E P.ACN:   .ASCII  <4>\Link\
                  74  6E  65  6E  61  6D  72  65  50  20  0A 00793 P.ACO:   .ASCII  <10>\ Permanent\
                      65  6C  69  74  61  6C  6F  56  20  09 0079E P.ACP:   .ASCII  <9>\ Volatile\
                          79  72  61  6D  6D  75  53  20  08 007A8 P.ACQ:   .ASCII  <8>\ Summary\
                              73  75  74  61  74  53  20  07 007B1 P.ACR:   .ASCII  <7>\ Status\
  69  74  73  69  72  65  74  63  61  72  61  68  43  20  10 007B9 P.ACS:   .ASCII  <16>\ Characteristics\
                                                      73  63 007C8
                      73  72  65  74  6E  75  6F  43  20  09 007CA P.ACT:   .ASCII  <9>\ Counters\
                              73  74  6E  65  76  45  20  07 007D4 P.ACU:   .ASCII  <7>\ Events\
  2F  21  44  25  30  32  21  20  66  6F  20  73  61  20  0F 007DC P.ACV:   .ASCII  <15>\ as of !20%D!/ \
                                                          20 007EB

                                                                            .PSECT  $OWN$,NOEXE,2

                                                             00000 FAOPTR:  .BLKB   4
```

```
                                00004 FAOLST: .BLKB   400
                                00194 CTRBUF: .BLKB   500
                                00388 OUTDSC: .BLKB   8
                                00390 OUTBUF: .BLKB   500
                                00584 NCP$A_COLHEAD:
                                              .BLKB   4
                                00588 NCP$B_COLHEAD:
                                              .BLKB   1
                                00589 NCP$B_LOGTYPE:
                                              .BLKB   1
                                0058A         .BLKB   2
                                0058C NCP$A_COLTBL:
                                              .BLKB   4

                                              .PSECT  $GLOBAL$,NOEXE,2

                                00000 NCP$GQ_CTRDSC::
                                              .BLKB   12

                                      NCP$Q_COLNODSUMHDR= P.AAA
                                      NCP$Q_COLNODSTAHDR= P.AAC
                                      NCP$Q_COLCIRSTAHDR= P.AAE
                                      NCP$Q_COLCIRSUMHDR= P.AAG
                                      NCP$Q_COLLISCIRHDR= P.AAI
                                      NCP$Q_COLLINHDR=    P.AAK
                                      NCP$Q_COLLNKHDR=    P.AAM
                                      NCP$Q_COLLNKSTAHDR= P.AAO
                                      NCP$Q_COLOBJHDR=    P.AAQ
                                      NCP$Q_COLLOGSUMHDR= P.AAS
                                      NCP$Q_COLCNFSUMHDR= P.AAU
                                      NCP$Q_COLCNFSTAHDR= P.AAW
                                      NCP$Q_COLMPRGRPHDR= P.AAY
                                      NCP$Q_COLMPRDTEHDR= P.ABA
                                      NCP$Q_COLMPRDTEPHDR=P.ABC
                                      NCP$Q_COLARESTAHDR= P.ABE
                                      NCP$Q_COLARESUMHDR= P.ABG
                                      NCP$T_COLNODSUMTBL= P.ABI
                                      NCP$T_COLNODSTATBL= P.ABJ
                                      NCP$T_COLCIRTBL=    P.ABK
                                      NCP$T_COLLINTBL=    P.ABL
                                      NCP$T_COLLNKTBL=    P.ABM
                                      NCP$T_COLLNKSTATBL= P.ABN
                                      NCP$T_COLOBJTBL=    P.ABO
                                      NCP$T_COLLOGTBL=    P.ABP
                                      NCP$T_COLCNFSUMTBL= P.ABQ
                                      NCP$T_COLCNFSTATBL= P.ABR
                                      NCP$T_COLMPRGRPTBL= P.ABS
                                      NCP$T_COLMPRDTETBL= P.ABT
                                      NCP$T_COLARESTATBL= P.ABU
                                      NCP$T_COLARESUMTBL= P.ABV
                                CTRDSC=               NCP$GQ_CTRDSC
                                              .EXTRN  NCP$_BADMSG, NCP$_LOGIC
                                              .EXTRN  NCP$GA_TBL_EVESOUR
                                              .EXTRN  NCP$GL_OPTION, NCP$GL_ENTITY
                                              .EXTRN  NCP$GL_MODTYP, NCP$GW_PRMTYP
                                              .EXTRN  PDB$G_VRB_ENT, NCP$GA_PRM_NOD
                                              .EXTRN  NCP$GA_PRM_CIR, NCP$GA_PRM_LIN
```

```
                                                        .EXTRN  NCP$GA_PRM_LOG, NCP$GA_PRM_MODCNF
                                                        .EXTRN  NCP$GA_PRM_MODACC
                                                        .EXTRN  NCP$GA_PRM_MODPRO
                                                        .EXTRN  NCP$GA_PRM_MODSER
                                                        .EXTRN  NCP$GA_PRM_MODTRC
                                                        .EXTRN  NCP$GA_PRM_ARE, NCP$GA_PRM_OBJ
                                                        .EXTRN  NCP$GA_PRM_LNK, NCP$GA_PRM_CIRCNTR
                                                        .EXTRN  NCP$GA_PRM_LINCNTR
                                                        .EXTRN  NCP$GA_PRM_NODCNTR
                                                        .EXTRN  NCP$GA_PRM_MPRCNTR
                                                        .EXTRN  NCP$GA_PRM_MSECNTR
                                                        .EXTRN  NCP$TABLESEARCH
                                                        .EXTRN  NCP$WRITESHO

                                                        .PSECT  $CODE$,NOWRT,2

                                    00FC 00000          .ENTRY  NCP$SHOHEAD, Save R2,R3,R4,R5,R6,R7             ; 0651
                 57 00000000V   00   9E 00002          MOVAB   NCP$ADDFAO, R7                                  ;
                 56 00000000G   00   9E 00009          MOVAB   NCP$GL_OPTION, R6                               ;
                 55 00000000V   00   9E 00010          MOVAB   NCP$ADDSTR, R5                                  ;
                 54 00000000'   00   9E 00017          MOVAB   P.ABW, R4                                       ;
                 53 00000000'   00   9E 0001E          MOVAB   NCP$GQ_CTRDSC, R3                               ;
00000000V        00                 00   FB 00025      CALLS   #0, NCP$FAOSET                                  ; 0687
                                    53   DD 0002C      PUSHL   R3                                              ; 0689
                                    54   DD 0002E      PUSHL   R4                                              ;
                 65                 02   FB 00030      CALLS   #2, NCP$ADDSTR                                  ;
                 50 00000000G       00   98 00033      CVTBL   PDB$G_VRB_ENT+1, R0                             ; 0691
           FE    8F                 50   91 0003A      CMPB    R0, #-2                                         ; 0694
                                    07   12 0003E      BNEQ    1$                                              ;
                                    53   DD 00040      PUSHL   R3                                              ;
                       07           A4   9F 00042      PUSHAB  P.ABX                                           ;
                                    18   11 00045      BRB     3$                                              ;
           FD    8F                 50   91 00047 1$:  CMPB    R0, #-3                                         ; 0695
                                    07   12 0004B      BNEQ    2$                                              ;
                                    53   DD 0004D      PUSHL   R3                                              ;
                       0F           A4   9F 0004F      PUSHAB  P.ABY                                           ;
                                    0B   11 00052      BRB     3$                                              ;
           FF    8F                 50   91 00054 2$:  CMPB    R0, #-1                                         ; 0696
                                    08   12 00058      BNEQ    4$                                              ;
                                    53   DD 0005A      PUSHL   R3                                              ;
                       15           A4   9F 0005C      PUSHAB  P.ABZ                                           ;
                 65                 02   FB 0005F 3$:  CALLS   #2, NCP$ADDSTR                                  ;
                 52 00000000G       00   D0 00062 4$:  MOVL    NCP$GL_ENTITY, R2                               ; 0699
                                    07   12 00069      BNEQ    5$                                              ; 0702
                                    53   DD 0006B      PUSHL   R3                                              ;
                       1C           A4   9F 0006D      PUSHAB  P.ACA                                           ;
                                    7E   11 00070      BRB     14$                                             ;
                 01                 52   D1 00072 5$:  CMPL    R2, #1                                          ; 0703
                                    07   12 00075      BNEQ    6$                                              ;
                                    53   DD 00077      PUSHL   R3                                              ;
                       21           A4   9F 00079      PUSHAB  P.ACB                                           ;
                                    7F   11 0007C      BRB     16$                                             ;
                 02                 52   D1 0007E 6$:  CMPL    R2, #2                                          ; 0704
                                    07   12 00081      BNEQ    7$                                              ;
                                    53   DD 00083      PUSHL   R3                                              ;
                       26           A4   9F 00085      PUSHAB  P.ACC                                           ;
                                    73   11 00088      BRB     16$                                             ;
```

```
                           03          52 D1 0008A 7$:    CMPL    R2, #3                     ; 0705
                                       07 12 0008D        BNEQ    8$                         ;
                                       53 DD 0008F        PUSHL   R3                         ;
                                 2E    A4 9F 00091        PUSHAB  P.ACD                      ;
                                       78 11 00094        BRB     18$                        ;
                           04          52 D1 00096 8$:    CMPL    R2, #4                     ; 0706
                                       57 12 00099        BNEQ    15$                        ;
                                       53 DD 0009B        PUSHL   R3                         ; 0708
                                 36    A4 9F 0009D        PUSHAB  P.ACE                      ;
                           65          02 FB 000A0        CALLS   #2, NCP$ADDSTR             ;
                           52 00000000G 00 D0 000A3       MOVL    NCP$GL_MODTYP, R2          ; 0709
                           01          52 D1 000AA        CMPL    R2, #1                     ; 0711
                                       07 12 000AD        BNEQ    9$                         ;
                                       53 DD 000AF        PUSHL   R3                         ;
                                 3D    A4 9F 000B1        PUSHAB  P.ACF                      ;
                                       69 11 000B4        BRB     20$                        ;
                           05          52 D1 000B6 9$:    CMPL    R2, #5                     ; 0715
                                       07 12 000B9        BNEQ    10$                        ;
                                       53 DD 000BB        PUSHL   R3                         ;
                                 4B    A4 9F 000BD        PUSHAB  P.ACG                      ;
                                       5D 11 000C0        BRB     20$                        ;
                           06          52 D1 000C2 10$:   CMPL    R2, #6                     ; 0716
                                       07 12 000C5        BNEQ    11$                        ;
                                       53 DD 000C7        PUSHL   R3                         ;
                                 57    A4 9F 000C9        PUSHAB  P.ACH                      ;
                                       51 11 000CC        BRB     20$                        ;
                           07          52 D1 000CE 11$:   CMPL    R2, #7                     ; 0717
                                       07 12 000D1        BNEQ    12$                        ;
                                       53 DD 000D3        PUSHL   R3                         ;
                                 65    A4 9F 000D5        PUSHAB  P.ACI                      ;
                                       45 11 000D8        BRB     20$                        ;
                           08          52 D1 000DA 12$:   CMPL    R2, #8                     ; 0718
                                       07 12 000DD        BNEQ    13$                        ;
                                       53 DD 000DF        PUSHL   R3                         ;
                                 71    A4 9F 000E1        PUSHAB  P.ACJ                      ;
                                       39 11 000E4        BRB     20$                        ;
                           09          52 D1 000E6 13$:   CMPL    R2, #9                     ; 0719
                                       37 12 000E9        BNEQ    21$                        ;
                                       53 DD 000EB        PUSHL   R3                         ;
                                 7C    A4 9F 000ED        PUSHAB  P.ACK                      ;
                                       2D 11 000F0 14$:   BRB     20$                        ;
                           05          52 D1 000F2 15$:   CMPL    R2, #5                     ; 0722
                                       08 12 000F5        BNEQ    17$                        ;
                                       53 DD 000F7        PUSHL   R3                         ;
                               0088    C4 9F 000F9        PUSHAB  P.ACL                      ;
                                       20 11 000FD 16$:   BRB     20$                        ;
                 FFFFFFFC  8F          52 D1 000FF 17$:   CMPL    R2, #-4                    ; 0723
                                       08 12 00106        BNEQ    19$                        ;
                                       53 DD 00108        PUSHL   R3                         ;
                               008D    C4 9F 0010A        PUSHAB  P.ACM                      ;
                                       0F 11 0010E 18$:   BRB     20$                        ;
                 FFFFFFF9  8F          52 D1 00110 19$:   CMPL    R2, #-7                    ; 0724
                                       09 12 00117        BNEQ    21$                        ;
                                       53 DD 00119        PUSHL   R3                         ;
                               0094    C4 9F 0011B        PUSHAB  P.ACN                      ;
                           65          02 FB 0011F 20$:   CALLS   #2, NCP$ADDSTR             ;
      03         66        03          04 ED 00122 21$:   CMPZV   #4, #3, NCP$GL_OPTION, #3  ; 0727
```

```
                                                     15  13 00127          BEQL    24$                                                     ;
                                                     66  95 00129          TSTB    NCP$GL_OPTION                                           ; 0730
                                                     08  18 0012B          BGEQ    22$                                                     ;
                                                     53  DD 0012D          PUSHL   R3                                                      ; 0732
                                          0099       C4  9F 0012F          PUSHAB  P.ACO                                                   ;
                                                     06  11 00133          BRB     23$                                                     ;
                                                     53  DD 00135 22$:      PUSHL   R3                                                      ; 0734
                                          00A4       C4  9F 00137          PUSHAB  P.ACP                                                   ;
                                    65               02  FB 0013B 23$:      CALLS   #2, NCP$ADDSTR                                          ;
                52                  66   03          04  EF 0013E 24$:      EXTZV   #4, #3, NCP$GL_OPTION, R2                               ; 0737
                                                     08  12 00143          BNEQ    25$                                                     ; 0740
                                                     53  DD 00145          PUSHL   R3                                                      ;
                                          00AE       C4  9F 00147          PUSHAB  P.ACQ                                                   ;
                                                     32  11 0014B          BRB     29$                                                     ;
                                    01               52  D1 0014D 25$:      CMPL    R2, #1                                                  ; 0741
                                                     08  12 00150          BNEQ    26$                                                     ;
                                                     53  DD 00152          PUSHL   R3                                                      ;
                                          00B7       C4  9F 00154          PUSHAB  P.ACR                                                   ;
                                                     25  11 00158          BRB     29$                                                     ;
                                    02               52  D1 0015A 26$:      CMPL    R2, #2                                                  ; 0742
                                                     08  12 0015D          BNEQ    27$                                                     ;
                                                     53  DD 0015F          PUSHL   R3                                                      ;
                                          00BF       C4  9F 00161          PUSHAB  P.ACS                                                   ;
                                                     18  11 00165          BRB     29$                                                     ;
                                    03               52  D1 00167 27$:      CMPL    R2, #3                                                  ; 0743
                                                     08  12 0016A          BNEQ    28$                                                     ;
                                                     53  DD 0016C          PUSHL   R3                                                      ;
                                          00D0       C4  9F 0016E          PUSHAB  P.ACT                                                   ;
                                                     0B  11 00172          BRB     29$                                                     ;
                                    04               52  D1 00174 28$:      CMPL    R2, #4                                                  ; 0744
                                                     09  12 00177          BNEQ    30$                                                     ;
                                                     53  DD 00179          PUSHL   R3                                                      ;
                                          00DA       C4  9F 0017B          PUSHAB  P.ACU                                                   ;
                                    65               02  FB 0017F 29$:      CALLS   #2, NCP$ADDSTR                                          ;
                                                     53  DD 00182 30$:      PUSHL   R3                                                      ; 0747
                                          00E2       C4  9F 00184          PUSHAB  P.ACV                                                   ;
                                    65               02  FB 00188          CALLS   #2, NCP$ADDSTR                                          ;
                                                     7E  D4 0018B          CLRL    -(SP)                                                   ; 0748
                                    67               01  FB 0018D          CALLS   #1, NCP$ADDFAO                                          ;
                                                     7E  D4 00190          CLRL    -(SP)                                                   ; 0749
                                    67               01  FB 00192          CALLS   #1, NCP$ADDFAO                                          ;
                      00000000V     00               00  FB 00195          CALLS   #0, NCP$FAOWRITE                                        ; 0751
                      00000000V     00               00  FB 0019C          CALLS   #0, NCP$SHOCOLHEAD                                      ; 0753
                                                         04  001A3          RET                                                             ; 0757

; Routine Size:  420 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  765          0758  1 %SBTTL  'NCP$SHOCOLHEAD  Setup Header for Column Output'
;  766          0759  1 ROUTINE NCP$SHOCOLHEAD :NOVALUE =       !
;  767          0760  1
;  768          0761  1 !++
;  769          0762  1 ! FUNCTIONAL DESCRIPTION:
;  770          0763  1 !
;  771          0764  1 !       Set the addresses of the column header text string and parameter
;  772          0765  1 !       decoding table if column output should be used for this return.
;  773          0766  1 !
;  774          0767  1 ! FORMAL PARAMETERS:
;  775          0768  1 !
;  776          0769  1 !       NONE
;  777          0770  1 !
;  778          0771  1 ! IMPLICIT INPUTS:
;  779          0772  1 !
;  780          0773  1 !       NCP$GL_OPTION           Option byte
;  781          0774  1 !       NCP$GL_ENTITY           Entity type code (negative if system-specific)
;  782          0775  1 !
;  783          0776  1 ! IMPLICIT OUTPUTS:
;  784          0777  1 !
;  785          0778  1 !       NCP$A_COLHEAD
;  786          0779  1 !       NCP$A_COLTBL
;  787          0780  1 !       NCP$B_COLHEAD
;  788          0781  1 !
;  789          0782  1 ! ROUTINE VALUE:
;  790          0783  1 ! COMPLETION CODES:
;  791          0784  1 !
;  792          0785  1 !       NONE
;  793          0786  1 !
;  794          0787  1 ! SIDE EFFECTS:
;  795          0788  1 !
;  796          0789  1 !       NONE
;  797          0790  1 !
;  798          0791  1 !--
;  799          0792  1
;  800          0793  2     BEGIN
;  801          0794  2
;  802          0795  2     LOCAL
;  803          0796  2         INFO;                                       ! Type of info requested
;  804          0797  2
;  805          0798  2     NCP$A_COLHEAD = 0;                              ! Assume not a column display
;  806          0799  2     NCP$A_COLTBL = 0;
;  807          0800  2
;  808          0801  2     INFO = .NCP$GL_OPTION [NMA$V_OPT_INF];          ! Get type of information
;  809          0802  2
;  810          0803  2     SELECTONE .NCP$GL_ENTITY                        ! Dispatch on entity type
;  811          0804  2     OF
;  812          0805  2         SET
;  813          0806  2         [NMA$C_ENT_NOD]:
;  814          0807  2             IF NOT .NCP$GL_OPTION [NMA$V_OPT_PER]   ! Volatile only:
;  815          0808  2             THEN
;  816          0809  2                 IF .INFO EQL NMA$C_OPINF_SUM        ! Node summary
;  817          0810  2                 THEN
;  818          0811  3                     BEGIN
;  819          0812  3                     NCP$A_COLHEAD = NCP$Q_COLNODSUMHDR;
;  820          0813  3                     NCP$A_COLTBL = NCP$T_COLNODSUMTBL;
;  821          0814  3                     END
```

```
;   822     0815  2                 ELSE IF .INFO EQL NMA$C_OPINF_STA       ! Node status
;   823     0816  2                 THEN
;   824     0817  2                     BEGIN
;   825     0818  2                     NCP$A_COLHEAD = NCP$Q_COLNODSTAHDR;
;   826     0819  2                     NCP$A_COLTBL = NCP$T_COLNODSTATBL;
;   827     0820  2                     END;
;   828     0821  2
;   829     0822  2             [NMA$C_ENT_CIR]:
;   830     0823  2                 IF .NCP$GL_OPTION [NMA$V_OPT_PER]           ! Permanent
;   831     0824  2                 AND
;   832     0825  2                     (.INFO EQL NMA$C_OPINF_STA OR           ! Circuit status
;   833     0826  2                     .INFO EQL NMA$C_OPINF_SUM)              ! Circuit summary
;   834     0827  2                 THEN
;   835     0828  2                     BEGIN
;   836     0829  2                     NCP$A_COLHEAD = NCP$Q_COLLISCIRHDR;
;   837     0830  2                     NCP$A_COLTBL = NCP$T_COLCIRTBL;
;   838     0831  2                     END
;   839     0832  2
;   840     0833  2                 ELSE                                        ! Volatile
;   841     0834  2                     BEGIN
;   842     0835  3                     IF .INFO EQL NMA$C_OPINF_STA            ! Circuit status
;   843     0836  3                     THEN
;   844     0837  4                         BEGIN
;   845     0838  4                         NCP$A_COLHEAD = NCP$Q_COLCIRSTAHDR;
;   846     0839  4                         NCP$A_COLTBL = NCP$T_COLCIRTBL;
;   847     0840  4                         END
;   848     0841  3                     ELSE IF .INFO EQL NMA$C_OPINF_SUM       ! Circuit summary
;   849     0842  3                     THEN
;   850     0843  4                         BEGIN
;   851     0844  4                         NCP$A_COLHEAD = NCP$Q_COLCIRSUMHDR;
;   852     0845  4                         NCP$A_COLTBL = NCP$T_COLCIRTBL;
;   853     0846  3                         END;
;   854     0847  2                     END;
;   855     0848  2
;   856     0849  2             [NMA$C_ENT_LIN]:                                ! for Show and List
;   857     0850  2                 IF .INFO EQL NMA$C_OPINF_STA OR             ! Line status
;   858     0851  2                     .INFO EQL NMA$C_OPINF_SUM               ! Line summary
;   859     0852  2                 THEN
;   860     0853  2                     BEGIN
;   861     0854  2                     NCP$A_COLHEAD = NCP$Q_COLLINHDR;
;   862     0855  2                     NCP$A_COLTBL = NCP$T_COLLINTBL;
;   863     0856  2                     END;
;   864     0857  2
;   865     0858  2             [-NMA$C_ENT_LNK]:
;   866     0859  2                 IF NOT .NCP$GL_OPTION [NMA$V_OPT_PER]       ! Volatile only:
;   867     0860  2                 THEN
;   868     0861  2                     IF .INFO EQL NMA$C_OPINF_SUM            ! Link summary
;   869     0862  2                     THEN
;   870     0863  2                         BEGIN
;   871     0864  2                         NCP$A_COLHEAD = NCP$Q_COLLNKHDR;
;   872     0865  2                         NCP$A_COLTBL = NCP$T_COLLNKTBL;
;   873     0866  2                         END
;   874     0867  2                     ELSE IF .INFO EQL NMA$C_OPINF_STA       ! Link status
;   875     0868  2                     THEN
;   876     0869  2                         BEGIN
;   877     0870  3                         NCP$A_COLHEAD = NCP$Q_COLLNKSTAHDR;
;   878     0871  3                         NCP$A_COLTBL = NCP$T_COLLNKSTATBL;
```

```
;  879          0872  2                    END;
;  880          0873  2
;  881          0874  2            [-NMA$C_SENT_OBJ]:
;  882          0875  2                IF NOT .NCP$GL_OPTION [NMA$V_OPT_PER]       ! Volatile only:
;  883          0876  2                THEN
;  884          0877  2                    IF .INFO EQL NMA$C_OPINF_STA            ! Object status
;  885          0878  2                        OR .INFO EQL NMA$C_OPINF_SUM        ! Object summary
;  886          0879  2                    THEN
;  887          0880  3                        BEGIN
;  888          0881  3                        NCP$A_COLHEAD = NCP$Q_COLOBJHDR;
;  889          0882  3                        NCP$A_COLTBL = NCP$T_COLOBJTBL
;  890          0883  3                        END;
;  891          0884  2
;  892          0885  2            [NMA$C_ENT_LOG]:
;  893          0886  2                IF .INFO EQL NMA$C_OPINF_SUM                ! Logging summary
;  894          0887  2                THEN
;  895          0888  3                    BEGIN
;  896          0889  3                    NCP$A_COLHEAD = NCP$Q_COLLOGSUMHDR;
;  897          0890  3                    NCP$A_COLTBL = NCP$T_COLLOGTBL
;  898          0891  3                    END;
;  899          0892  2
;  900          0893  2            [NMA$C_ENT_MOD]:
;  901          0894  2                IF .NCP$GL_MODTYP EQL NCP$C_ENT_MODPRO      ! Module X25-Protocol
;  902          0895  2                THEN
;  903          0896  3                    BEGIN
;  904          0897  3                    SELECTONE .NCP$GW_PRMTYP OF
;  905          0898  3                        SET
;  906          0899  3                        [NMA$C_PCXP_DTE] :                  ! DTE
;  907          0900  4                            IF .INFO EQL NMA$C_OPINF_STA    ! status or summary
;  908          0901  4                               OR .INFO EQL NMA$C_OPINF_SUM
;  909          0902  4                            THEN
;  910          0903  4                                BEGIN
;  911          0904  4                                NCP$A_COLTBL = NCP$T_COLMPRDTETBL;
;  912          0905  4                                IF NOT .NCP$GL_OPTION [NMA$V_OPT_PER] ! Volatile only:
;  913          0906  4                                THEN
;  914          0907  4                                    NCP$A_COLHEAD = NCP$Q_COLMPRDTEHDR
;  915          0908  4                                ELSE
;  916          0909  4                                    NCP$A_COLHEAD = NCP$Q_COLMPRDTEPHDR;
;  917          0910  4                                END;
;  918          0911  3
;  919          0912  3                        [NMA$C_PCXP_GRP] :                  ! Group
;  920          0913  4                            IF .INFO EQL NMA$C_OPINF_STA    ! status
;  921          0914  4                               OR .INFO EQL NMA$C_OPINF_SUM ! summary
;  922          0915  4                               OR .INFO EQL NMA$C_OPINF_CHA ! or characteristics
;  923          0916  4                            THEN
;  924          0917  4                                BEGIN
;  925          0918  4                                NCP$A_COLHEAD = NCP$Q_COLMPRGRPHDR;
;  926          0919  4                                NCP$A_COLTBL = NCP$T_COLMPRGRPTBL
;  927          0920  4                                END;
;  928          0921  3                        TES;
;  929          0922  2                    END
;  930          0923  2                ELSE
;  931          0924  3                    BEGIN
;  932          0925  3                    IF .NCP$GL_MODTYP EQL NCP$C_ENT_MODCNF  ! Module Configurator
;  933          0926  3                    THEN
;  934          0927  4                        BEGIN
;  935          0928  4                        IF .INFO EQL NMA$C_OPINF_SUM        ! summary
```

```
;  936          0929  4                    THEN
;  937          0930  4                        IF NOT .NCP$GL_OPTION [NMA$V_OPT_PER]   ! Volatile only:
;  938          0931  4                        THEN
;  939          0932  5                            BEGIN
;  940          0933  5                            NCP$A_COLTBL = NCP$T_COLCNFSUMTBL;
;  941          0934  5                            NCP$A_COLHEAD = NCP$Q_COLCNFSUMHDR;
;  942          0935  4                            END;
;  943          0936  4                    IF .INFO EQL NMA$C_OPINF_STA        ! status
;  944          0937  4                    THEN
;  945          0938  4                        IF NOT .NCP$GL_OPTION [NMA$V_OPT_PER]   ! Volatile only:
;  946          0939  4                        THEN
;  947          0940  5                            BEGIN
;  948          0941  5                            NCP$A_COLTBL = NCP$T_COLCNFSTATBL;
;  949          0942  5                            NCP$A_COLHEAD = NCP$Q_COLCNFSTAHDR;
;  950          0943  4                            END;
;  951          0944  3                    END;
;  952          0945  2                END;
;  953          0946  2
;  954          0947  2            [NMA$C_ENT_ARE]:
;  955          0948  2                IF NOT .NCP$GL_OPTION [NMA$V_OPT_PER]       ! Volatile only:
;  956          0949  2                THEN
;  957          0950  2                    IF .INFO EQL NMA$C_OPINF_SUM            ! Area summary
;  958          0951  2                    THEN
;  959          0952  3                        BEGIN
;  960          0953  3                        NCP$A_COLHEAD = NCP$Q_COLARESUMHDR;
;  961          0954  3                        NCP$A_COLTBL = NCP$T_COLARESUMTBL;
;  962          0955  3                        END
;  963          0956  2                    ELSE IF .INFO EQL NMA$C_OPINF_STA       ! Area status
;  964          0957  2                    THEN
;  965          0958  3                        BEGIN
;  966          0959  3                        NCP$A_COLHEAD = NCP$Q_COLARESTAHDR;
;  967          0960  3                        NCP$A_COLTBL = NCP$T_COLARESTATBL;
;  968          0961  2                        END;
;  969          0962  2
;  970          0963  2            TES;
;  971          0964  2
;  972          0965  2        NCP$B_COLHEAD = FALSE;                          ! Head not printed yet
;  973          0966  2        NCP$B_LOGTYPE = -1                              ! Invalid logging type
;  974          0967  2
;  975          0968  1        END;


                                                003C 00000 NCP$SHOCOLHEAD:
                                                           .WORD   Save R2,R3,R4,R5                                    ; 0759
                                    55 00000000G   00   9E 00002          MOVAB   NCP$GL_OPTION, R5                     ;
                                    54 00000000'   00   9E 00009          MOVAB   NCP$Q_COLNODSUMHDR, R4                ;
                                    53 00000000'   00   9E 00010          MOVAB   NCP$A_COLHEAD, R3                     ;
                                                   63   D4 00017          CLRL    NCP$A_COLHEAD                         ; 0798
                                                   08   A3   D4 00019     CLRL    NCP$A_COLTBL                          ; 0799
                50                  65             03   04   EF 0001C     EXTZV   #4, #3, NCP$GL_OPTION, INFO           ; 0801
                                    51 00000000G   00   D0 00021          MOVL    NCP$GL_ENTITY, R1                     ; 0803
                                                   25   12 00028          BNEQ    2$                                    ; 0806
                                                   65   95 0002A          TSTB    NCP$GL_OPTION                         ; 0807
                                                   7D   19 0002C          BLSS    10$                                   ;
```

```
                                                50  D5  0002E          TSTL    INFO                                    ; 0809
                                                0B  12  00030          BNEQ    1$
                                          63    64  9E  00032          MOVAB   NCP$Q_COLNODSUMHDR, NCP$A_COLHEAD        ; 0812
                                    08    A3    C4  9E  00035          MOVAB   NCP$T_COLNODSUMTBL, NCP$A_COLTBL         ; 0813
                                                7F  11  0003B          BRB     11$                                     ; 0809
                                          01    50  D1  0003D  1$:     CMPL    INFO, #1                                ; 0815
                                                7F  12  00040          BNEQ    13$
                                          63    C4  9E  00042          MOVAB   NCP$Q_COLNODSTAHDR, NCP$A_COLHEAD       ; 0818
                                    08    A3    C4  9E  00047          MOVAB   NCP$T_COLNODSTATBL, NCP$A_COLTBL         ; 0819
                                                7F  11  0004D          BRB     14$                                     ; 0809
                                          03    51  D1  0004F  2$:     CMPL    R1, #3                                  ; 0822
                                                31  12  00052          BNEQ    7$
                                                65  95  00054          TSTB    NCP$GL_OPTION                           ; 0823
                                                10  18  00056          BGEQ    4$
                                          01    50  D1  00058          CMPL    INFO, #1                                ; 0825
                                                04  13  0005B          BEQL    3$
                                                50  D5  0005D          TSTL    INFO                                    ; 0826
                                                07  12  0005F          BNEQ    4$
                                          63    C4  9E  00061  3$:     MOVAB   NCP$Q_COLLISCIRHDR, NCP$A_COLHEAD       ; 0829
                                                15  11  00066          BRB     6$                                      ; 0830
                                          01    50  D1  00068  4$:     CMPL    INFO, #1                                ; 0835
                                                07  12  0006B          BNEQ    5$
                                          63    C4  9E  0006D          MOVAB   NCP$Q_COLCIRSTAHDR, NCP$A_COLHEAD       ; 0838
                                                09  11  00072          BRB     6$                                      ; 0839
                                                50  D5  00074  5$:     TSTL    INFO                                    ; 0841
                                                79  12  00076          BNEQ    17$
                                          63    C4  9E  00078          MOVAB   NCP$Q_COLCIRSUMHDR, NCP$A_COLHEAD       ; 0844
                                    08    A3    C4  9E  0007D  6$:     MOVAB   NCP$T_COLCIRTBL, NCP$A_COLTBL           ; 0845
                                                6C  11  00083          BRB     17$                                     ; 0823
                                          01    51  D1  00085  7$:     CMPL    R1, #1                                  ; 0849
                                                16  12  00088          BNEQ    9$
                                          01    50  D1  0008A          CMPL    INFO, #1                                ; 0850
                                                04  13  0008D          BEQL    8$
                                                50  D5  0008F          TSTL    INFO                                    ; 0851
                                                74  12  00091          BNEQ    19$
                                          63    C4  9E  00093  8$:     MOVAB   NCP$Q_COLLINHDR, NCP$A_COLHEAD          ; 0854
                                    08    A3    C4  9E  00098          MOVAB   NCP$T_COLLINTBL, NCP$A_COLTBL           ; 0855
                                                67  11  0009E          BRB     19$                                     ; 0850
                              FFFFFFF9    8F    51  D1  000A0  9$:     CMPL    R1, #-7                                 ; 0858
                                                27  12  000A7          BNEQ    15$
                                                65  95  000A9          TSTB    NCP$GL_OPTION                           ; 0859
                                                5A  19  000AB  10$:     BLSS    19$
                                                50  D5  000AD          TSTL    INFO                                    ; 0861
                                                0D  12  000AF          BNEQ    12$
                                          63    C4  9E  000B1          MOVAB   NCP$Q_COLLNKHDR, NCP$A_COLHEAD          ; 0864
                                    08    A3    C4  9E  000B6          MOVAB   NCP$T_COLLNKTBL, NCP$A_COLTBL           ; 0865
                                                49  11  000BC  11$:     BRB     19$                                     ; 0861
                                          01    50  D1  000BE  12$:     CMPL    INFO, #1                                ; 0867
                                                71  12  000C1  13$:     BNEQ    22$
                                          63    C4  9E  000C3          MOVAB   NCP$Q_COLLNKSTAHDR, NCP$A_COLHEAD       ; 0870
                                    08    A3    C4  9E  000C8          MOVAB   NCP$T_COLLNKSTATBL, NCP$A_COLTBL        ; 0871
                                                7C  11  000CE  14$:     BRB     25$                                     ; 0861
                              FFFFFFFC    8F    51  D1  000D0  15$:     CMPL    R1, #-4                                 ; 0874
                                                1A  12  000D7          BNEQ    18$
                                                65  95  000D9          TSTB    NCP$GL_OPTION                           ; 0875
                                                6F  19  000DB          BLSS    25$
                                          01    50  D1  000DD          CMPL    INFO, #1                                ; 0877
```

```
                                                04  13  000E0          BEQL    16$                                             ;
                                                50  D5  000E2          TSTL    INFO                                            ; 0878
                                                7D  12  000E4          BNEQ    27$                                             ;
                                    63  02FC  C4  9E  000E6 16$:      MOVAB   NCP$Q_COLOBJHDR, NCP$A_COLHEAD                  ; 0881
                                08  A3  05EC  C4  9E  000EB          MOVAB   NCP$T_COLOBJTBL, NCP$A_COLTBL                   ; 0882
                                                7D  11  000F1 17$:     BRB     29$                                             ; 0875
                                    02          51  D1  000F3 18$:     CMPL    R1, #2                                          ; 0885
                                                11  12  000F6          BNEQ    20$                                             ;
                                                50  D5  000F8          TSTL    INFO                                            ; 0886
                                                79  12  000FA          BNEQ    31$                                             ;
                                    63  0358  C4  9E  000FC          MOVAB   NCP$Q_COLLOGSUMHDR, NCP$A_COLHEAD               ; 0889
                                08  A3  0604  C4  9E  00101          MOVAB   NCP$T_COLLOGTBL, NCP$A_COLTBL                   ; 0890
                                                67  11  00107 19$:     BRB     29$                                             ; 0886
                                    04          51  D1  00109 20$:     CMPL    R1, #4                                          ; 0893
                                                03  13  0010C          BEQL    21$                                             ;
                                              008F  31  0010E          BRW     33$                                             ;
                                    52  00000000G  00  D0  00111 21$:     MOVL    NCP$GL_MODTYP, R2                           ; 0894
                                    06          52  D1  00118          CMPL    R2, #6                                          ;
                                                55  12  0011B          BNEQ    30$                                             ;
                                    51  00000000G  00  D0  0011D          MOVL    NCP$GW_PRMTYP, R1                           ; 0897
                            0000044C    8F          51  D1  00124          CMPL    R1, #1100                                   ; 0899
                                                21  12  0012B          BNEQ    26$                                             ;
                                    01          50  D1  0012D          CMPL    INFO, #1                                        ; 0900
                                                04  13  00130          BEQL    23$                                             ;
                                                50  D5  00132          TSTL    INFO                                            ; 0901
                                                6D  12  00134 22$:     BNEQ    34$                                             ;
                                08  A3  0650  C4  9E  00136 23$:     MOVAB   NCP$T_COLMPRDTETBL, NCP$A_COLTBL                ; 0904
                                                65  95  0013C          TSTB    NCP$GL_OPTION                                   ; 0905
                                                07  19  0013E          BLSS    24$                                             ;
                                    63  0478  C4  9E  00140          MOVAB   NCP$Q_COLMPRDTEHDR, NCP$A_COLHEAD               ; 0907
                                                71  11  00145          BRB     35$                                             ;
                                    63  04A0  C4  9E  00147 24$:     MOVAB   NCP$Q_COLMPRDTEPHDR, NCP$A_COLHEAD              ; 0909
                                                7C  11  0014C 25$:     BRB     37$                                             ; 0900
                            0000044D    8F          51  D1  0014E 26$:     CMPL    R1, #1101                                   ; 0912
                                                73  12  00155          BNEQ    37$                                             ;
                                    01          50  D1  00157          CMPL    INFO, #1                                        ; 0913
                                                09  13  0015A          BEQL    28$                                             ;
                                                50  D5  0015C          TSTL    INFO                                            ; 0914
                                                05  13  0015E          BEQL    28$                                             ;
                                    02          50  D1  00160          CMPL    INFO, #2                                        ; 0915
                                                65  12  00163 27$:     BNEQ    37$                                             ;
                                    63  0420  C4  9E  00165 28$:     MOVAB   NCP$Q_COLMPRGRPHDR, NCP$A_COLHEAD               ; 0918
                                08  A3  063C  C4  9E  0016A          MOVAB   NCP$T_COLMPRGRPTBL, NCP$A_COLTBL                ; 0919
                                                58  11  00170 29$:     BRB     37$                                             ; 0894
                                    01          52  D1  00172 30$:     CMPL    R2, #1                                          ; 0925
                                                53  12  00175 31$:     BNEQ    37$                                             ;
                                                50  D5  00177          TSTL    INFO                                            ; 0928
                                                0F  12  00179          BNEQ    32$                                             ;
                                                65  95  0017B          TSTB    NCP$GL_OPTION                                   ; 0930
                                                0B  19  0017D          BLSS    32$                                             ;
                                08  A3  0618  C4  9E  0017F          MOVAB   NCP$T_COLCNFSUMTBL, NCP$A_COLTBL                ; 0933
                                    63  0394  C4  9E  00185          MOVAB   NCP$Q_COLCNFSUMHDR, NCP$A_COLHEAD               ; 0934
                                    01          50  D1  0018A 32$:     CMPL    INFO, #1                                        ; 0936
                                                3B  12  0018D          BNEQ    37$                                             ;
                                                65  95  0018F          TSTB    NCP$GL_OPTION                                   ; 0938
                                                37  19  00191          BLSS    37$                                             ;
                                08  A3  0628  C4  9E  00193          MOVAB   NCP$T_COLCNFSTATBL, NCP$A_COLTBL                ; 0941
```

```
                         63     03E0    C4  9E 00199          MOVAB   NCP$Q_COLCNFSTAHDR, NCP$A_COLHEAD       ; 0942
                                        2A  11 0019E          BRB     37$                                     ; 0894
                         05             51  D1 001A0 33$:     CMPL    R1, #5                                  ; 0947
                                        25  12 001A3 34$:     BNEQ    37$                                     ;
                                        65  95 001A5          TSTB    NCP$GL_OPTION                           ; 0948
                                        21  19 001A7          BLSS    37$                                     ;
                                        50  D5 001A9          TSTL    INFO                                    ; 0950
                                        0D  12 001AB          BNEQ    36$                                     ;
                         63     053C    C4  9E 001AD          MOVAB   NCP$Q_COLARESUMHDR, NCP$A_COLHEAD       ; 0953
                   08    A3     0680    C4  9E 001B2          MOVAB   NCP$T_COLARESUMTBL, NCP$A_COLTBL        ; 0954
                                        10  11 001B8 35$:     BRB     37$                                     ; 0950
                         01             50  D1 001BA 36$:     CMPL    INFO, #1                                ; 0956
                                        0B  12 001BD          BNEQ    37$                                     ;
                         63     04F4    C4  9E 001BF          MOVAB   NCP$Q_COLARESTAHDR, NCP$A_COLHEAD       ; 0959
                   08    A3     0668    C4  9E 001C4          MOVAB   NCP$T_COLARESTATBL, NCP$A_COLTBL        ; 0960
                   04    A3     FF00    8F  B0 001CA 37$:     MOVW    #65280, NCP$B_COLHEAD                   ; 0965
                                        04  001D0             RET                                             ; 0968

; Routine Size:  465 bytes,    Routine Base:  $CODE$ + 01A4
```

```
;  977          0969  1 %SBTTL  'NCP$SHOLIS  Format a Single Return Message'
;  978          0970  1 GLOBAL ROUTINE NCP$SHOLIS (LEN, BFR) :NOVALUE = !
;  979          0971  1
;  980          0972  1 !++
;  981          0973  1 ! FUNCTIONAL DESCRIPTION:
;  982          0974  1 !
;  983          0975  1 !       This routine formats a single return message and produces
;  984          0976  1 !       the title and the list of parameters or counters.
;  985          0977  1 !
;  986          0978  1 ! FORMAL PARAMETERS:
;  987          0979  1 !
;  988          0980  1 !       LEN             Value of the length of the data
;  989          0981  1 !       BFR             Address of the data
;  990          0982  1 !
;  991          0983  1 ! IMPLICIT INPUTS:
;  992          0984  1 !
;  993          0985  1 !       NCP$GL_ENTITY   Entity code
;  994          0986  1 !
;  995          0987  1 ! IMPLICIT OUTPUTS:
;  996          0988  1 !
;  997          0989  1 !       NONE
;  998          0990  1 !
;  999          0991  1 ! ROUTINE VALUE:
; 1000          0992  1 ! COMPLETION CODES:
; 1001          0993  1 !
; 1002          0994  1 !       NONE
; 1003          0995  1 !
; 1004          0996  1 ! SIDE EFFECTS:
; 1005          0997  1 !
; 1006          0998  1 !       NONE
; 1007          0999  1 !
; 1008          1000  1 !--
; 1009          1001  1
; 1010          1002  2     BEGIN
; 1011          1003  2
; 1012          1004  2     LOCAL
; 1013          1005  2         STATUS,                         ! Return of shocolfmt
; 1014          1006  2         PTR:    REF BBLOCK,             ! General pointer
; 1015          1007  2         PEND,                           ! Pointer to end of message
; 1016          1008  2         CTR                             ! General counter
; 1017          1009  2         ;               !
; 1018          1010  2
; 1019          1011  2     PTR = .BFR;                         ! Pointer to buffer
; 1020          1012  2     PEND = .BFR + .LEN;                 ! End of data pointer
; 1021          1013  2
; 1022          1014  2 !
; 1023          1015  2 !       If this is a link response message from a 2.0 NML,
; 1024          1016  2 !       then re-format it into 2.2 link response messages
; 1025          1017  2 !       recursively call ourself to format each of the links.
; 1026          1018  2 !       A 2.0 message consists of a node name followed by a
; 1027          1019  2 !       series of link/pid coded multiples.  A 2.2 message
; 1028          1020  2 !       describes only a single link and its attributes.  So,
; 1029          1021  2 !       we must reformat the 2.0 message into many 2.2 "messages".
; 1030          1022  2 !
; 1031          1023  2
; 1032          1024  2     IF .ncp$gl_entity EQL -nma$c_sent_lnk  ! If its a link message
; 1033          1025  2         AND CH$RCHAR(.ptr) NEQ 0           ! If not a format byte of zero
```

```
;  1034          1026  2     THEN
;  1035          1027  2         BEGIN                                 ! then assume its a node name
;  1036          1028  2         LOCAL
;  1037          1029  2             new_buffer: BBLOCKVECTOR [32,1]; ! Allocate space for new message
;  1038          1030  2
;  1039          1031  2         CH$FILL(0, 32, new_buffer);           ! Preset entire buffer to zero
;  1040          1032  2         ! new_buffer [0,0,0,8,0] = 0;         ! Format byte of zero
;  1041          1033  2                                               ! Leave space for word link address
;  1042          1034  2         new_buffer [3,0,0,16,0] = nma$c_pclk_pid; ! Parameter code for PID
;  1043          1035  2         new_buffer [5,nma$v_pty_nty] = 2; ! Hex image
;  1044          1036  2         new_buffer [5,nma$v_pty_nle] = 4; ! longword
;  1045          1037  2                                               ! Leave space for longword PID
;  1046          1038  2         new_buffer [10,0,0,16,0] = nma$c_pclk_nid; ! Parameter code for NID
;  1047          1039  2         new_buffer [12,nma$v_pty_cod] = true;     ! Coded
;  1048          1040  2         new_buffer [12,nma$v_pty_mul] = true;     ! Multiple
;  1049          1041  2         new_buffer [12,nma$v_pty_cle] = 2;        ! of 2 fields
;  1050          1042  2         new_buffer [13,nma$v_pty_nle] = 2;        ! Field 1 is a decimal word
;  1051          1043  2         new_buffer [14,0,0,16,0] = .(.ptr) <0,16>; ! Copy node address
;  1052          1044  2         new_buffer [16,nma$v_pty_asc] = true;     ! Field 2 is an ASCIC string
;  1053          1045  2         new_buffer [17,0,0,8,0] = CH$RCHAR(.ptr+2); ! Copy node name
;  1054          1046  2         CH$MOVE(CH$RCHAR(.ptr+2), .ptr+3, new_buffer+18);
;  1055          1047  2
;  1056          1048  2         ptr = .ptr + 3 + CH$RCHAR(.ptr+2); ! Point to first 2.0 parameter
;  1057          1049  2
;  1058          1050  2         WHILE .ptr LEQA .pend                 ! For each 2.0 LAD parameter
;  1059          1051  2         DO
;  1060          1052  4             BEGIN
;  1061          1053  4             IF .(.ptr) <0,16> NEQ nma$c_pclk_lad ! If not an LAD parameter,
;  1062          1054  4             THEN
;  1063          1055  4                 EXITLOOP;                     ! then get out of loop
;  1064          1056  4             new_buffer [1,0,0,16,0] = .(.ptr+4) <0,16>; ! Copy link address
;  1065          1057  4             new_buffer [6,0,0,32,0] = .(.ptr+7);        ! Copy longword PID
;  1066          1058  4             ncp$sholis(18+CH$RCHAR(new_buffer+17), new_buffer); ! Display it
;  1067          1059  4             ptr = .ptr + 11;                  ! Skip to next LAD parameter
;  1068          1060  3             END;
;  1069          1061  3         RETURN;                               ! Return - all done
;  1070          1062  2         END;
;  1071          1063  2
;  1072          1064  2     IF NOT
;  1073          1065  3         (STATUS = NCP$SHOCOLFMT (.LEN, .BFR, PTR) )! Try for column output
;  1074          1066  2     THEN
;  1075          1067  3         BEGIN
;  1076          1068  3         NCP$FAOSET ();                        ! Set fao pointers
;  1077          1069  3         NCP$SHOENTITY (PTR);                  ! Convert entity
;  1078          1070  3         ADDSTR ('!/ ');                       ! Blank line
;  1079          1071  3         NCP$FAOWRITE ();                      ! Write that much of it
;  1080          1072  3         IF .PTR EQL .PEND                     ! If nothing else returned except
;  1081          1073  3         THEN                                  ! then entity id,
;  1082          1074  3             NCP$WRITESHO(ASCID('No information available'));
;  1083          1075  2         END;
;  1084          1076  2
;  1085          1077  2 !
;  1086          1078  2 !       If we produced column output, we may have quit early if
;  1087          1079  2 !       there was a parameter we did not like to see.  We will print the
;  1088          1080  2 !       remainder of the parameters here in standard form.
;  1089          1081  2 !
;  1090          1082  2
```

```
;  1091          1083  2     WHILE .PTR LSSA .PEND                   ! Scan til the end
;  1092          1084  2     DO
;  1093          1085  3         BEGIN                               ! Setup the output buffer descriptor
;  1094          1086  3         OUTDSC [0] = OUTBUFSIZ;
;  1095          1087  3         OUTDSC [1] = OUTBUF;
;  1096          1088  3
;  1097          1089  3
;  1098          1090  3         IF NOT V2_SHOW_COMPAT(.PTR)             ! Map V2 to V3 parameters; if ignored,
;  1099          1091  3         THEN
;  1100          1092  3             OUTDSC [0] = 0;                 ! Do not pass any result buffer
;  1101          1093  3
;  1102          1094  3         NCP$FORMATPARM(                     ! Format one parameter
;  1103          1095  3                 .NCP$GL_ENTITY,             ! Entity code
;  1104          1096  3                 .PTR,                       ! Point to start
;  1105          1097  3                 TRUE,                       ! We want a name
;  1106          1098  3                 TRUE,                       ! We want data
;  1107          1099  3                 OUTDSC,                     ! Output buffer descriptor
;  1108          1100  3                 OUTDSC [0],                 ! Return length here
;  1109          1101  3                 PTR);                       ! Return pointer here
;  1110          1102  3
;  1111          1103  3         IF .OUTDSC [0] GTRU 0               ! If non-null line,
;  1112          1104  3         THEN
;  1113          1105  3             NCP$WRITESHO (OUTDSC);          ! Write the line out
;  1114          1106  2         END;
;  1115          1107  2
;  1116          1108  2     IF NOT .STATUS                          ! If we did not use column output
;  1117          1109  2     THEN
;  1118          1110  3         BEGIN
;  1119          1111  3         NCP$FAOSET ();                      ! Some blank lines at the end
;  1120          1112  3         ADDSTR ('!/ !/');
;  1121          1113  3         NCP$FAOWRITE ()
;  1122          1114  3         END
;  1123          1115  2     ;
;  1124          1116  2
;  1125          1117  2     RETURN
;  1126          1118  2
;  1127          1119  1     END;


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

                                        20  2F  21  03   007EC P.ACW:  .ASCII  <3>\!/ \                                  ;
20  6E  6F  69  74  61  6D  72  6F  66  6E  69  20  6F  4E   007F0 P.ACY:  .ASCII  \No information available\                 ;
                    65  6C  62  61  6C  69  61  76  61   007FF                                                            ;
                                    00000018   00808 P.ACX:  .LONG   24                                                 ;
                                    00000000'  0080C         .ADDRESS P.ACY                                             ;
                                2F  21  20  2F  21  05   00810 P.ACZ:  .ASCII  <5>\!/ !/\                               ;


                                                        .PSECT  $CODE$,NOWRT,2

                                            0FFC 00000          .ENTRY  NCP$SHOLIS, Save R2,R3,R4,R5,R6,R7,R8,R9,-          ; 0970
                                                                R10,R11                                                     ;
                                5B 00000000'  00  9E 00002          MOVAB   NCP$GQ_CTRDSC, R11                              ;
                                5A 00000000V  00  9E 00009          MOVAB   NCP$FAOSET, R10                                 ;
```

```
                                59 00000000'   00   9E 00010          MOVAB   P.ACW, R9                                       ;
                                58 00000000'   00   9E 00017          MOVAB   OUTDSC, R8                                      ;
                                5E             20   C2 0001E          SUBL2   #32, SP                                         ;
                                         08    AC   DD 00021          PUSHL   BFR                                             ; 1011
                             57       08    AC   04   AC C1 00024     ADDL3   LEN, BFR, PEND                                  ; 1012
                          FFFFFFF9    8F 00000000G   00   D1 0002A     CMPL    NCP$GL_ENTITY, #-7                              ; 1024
                                                   03   13 00035     BEQL    2$                                              ;
                                              0086   31 00037 1$:     BRW     6$                                              ;
                                           00    BE   95 0003A 2$:     TSTB    @PTR                                            ; 1025
                                                 F8   13 0003D         BEQL    1$                                              ;
                20         00                 6E   00   2C 0003F     MOVC5   #0, (SP), #0, #32, NEW_BUFFER                   ; 1031
                                                    04   AE 00044                                                             ;
                                        07    AE   65   8F 9B 00046     MOVZBW  #101, NEW_BUFFER+3                              ; 1034
             09 AE                      02         04   02 F0 0004B     INSV    #2, #4, #2, NEW_BUFFER+5                        ; 1035
             09 AE                      04         00   04 F0 00051     INSV    #4, #0, #4, NEW_BUFFER+5                        ; 1036
                                     0E    AE   66   8F 9B 00057     MOVZBW  #102, NEW_BUFFER+10                             ; 1038
                                     10    AE   C0   8F 88 0005C     BISB2   #192, NEW_BUFFER+12                             ; 1040
             10 AE                      06         00   02 F0 00061     INSV    #2, #0, #6, NEW_BUFFER+12                       ; 1041
             11 AE                      04         00   02 F0 00067     INSV    #2, #0, #4, NEW_BUFFER+13                       ; 1042
                                              56   6E   D0 0006D     MOVL    PTR, R6                                         ; 1043
                                     12    AE   66   B0 00070     MOVW    (R6), NEW_BUFFER+14                                     ;
                                     14    AE   40   8F 88 00074     BISB2   #64, NEW_BUFFER+16                              ; 1044
                                     15    AE   02   A6 90 00079     MOVB    2(R6), NEW_BUFFER+17                            ; 1045
                                              50   02   A6 9A 0007E     MOVZBL  2(R6), R0                                       ; 1046
                          16 AE      03    A6   50   28 00082     MOVC3   R0, 3(R6), NEW_BUFFER+18                                ;
                                              50   02   A6 9A 00088     MOVZBL  2(R6), R0                                       ; 1048
                                              6E   03 A640 9E 0008C     MOVAB   3(R6)[R0], PTR                                  ;
                                              50   6E   D0 00091 3$:     MOVL    PTR, R0                                         ; 1050
                                              57   50   D1 00094         CMPL    R0, PEND                                        ;
                                                   01   1B 00097         BLEQU   4$                                              ;
                                                   04 00099              RET                                                     ;
                                        0069   8F   60   B1 0009A 4$:     CMPW    (R0), #105                                      ; 1053
                                                   01   13 0009F         BEQL    5$                                              ;
                                                   04 000A1              RET                                                     ;
                                     05    AE   04   A0   B0 000A2 5$:     MOVW    4(R0), NEW_BUFFER+1                             ; 1056
                                     0A    AE   07   A0   D0 000A7         MOVL    7(R0), NEW_BUFFER+6                             ; 1057
                                              04   AE   9F 000AC         PUSHAB  NEW_BUFFER                                      ; 1058
                                           7E   19   AE   9A 000AF         MOVZBL  NEW_BUFFER+17, -(SP)                            ;
                                           6E        12   C0 000B3         ADDL2   #18, (SP)                                       ;
                                        FF45   CF        02   FB 000B6         CALLS   #2, NCP$SHOLIS                          ;
                                           6E        0B   C0 000BB         ADDL2   #11, PTR                                        ; 1059
                                                     D1   11 000BE         BRB     3$                                              ; 1050
                                              5E   DD   C0 000C0 6$:     PUSHL   SP                                              ; 1065
                                           7E   04   AC   7D 000C2         MOVQ    LEN, -(SP)                                      ;
                                  00000000V   00        03   FB 000C6         CALLS   #3, NCP$SHOCOLFMT                       ;
                                              52   50   D0 000CD         MOVL    R0, STATUS                                      ;
                                              2D   52   E8 000D0         BLBS    STATUS, 8$                                      ;
                                              6A   00   FB 000D3         CALLS   #0, NCP$FAOSET                                  ; 1068
                                              5E   DD   000D6         PUSHL   SP                                              ; 1069
                                  00000000V   00        01   FB 000D8         CALLS   #1, NCP$SHOENTITY                       ;
                                        0A00   8F   BB   000DF         PUSHR   #^M<R9,R11>                                     ; 1070
                                  00000000V   00        02   FB 000E3         CALLS   #2, NCP$ADDSTR                          ;
                                  00000000V   00        00   FB 000EA         CALLS   #0, NCP$FAOWRITE                        ; 1071
                                              57   6E   D1 000F1         CMPL    PTR, PEND                                       ; 1072
                                                   0A   12 000F4         BNEQ    8$                                              ;
                                           1C   A9   9F 000F6         PUSHAB  P.ACX                                           ; 1074
                                  00000000G   00        01   FB 000F9 7$:     CALLS   #1, NCP$WRITESHO                        ;
```

```
                          57              6E  D1 00100 8$:      CMPL    PTR, PEND                               ; 1083
                                          3A  1E 00103           BGEQU   10$                                     ;
                          68      01F4    8F  3C 00105           MOVZWL  #500, OUTDSC                            ; 1086
                    04    A8        08    A8  9E 0010A           MOVAB   OUTBUF, OUTDSC+4                        ; 1087
                                          6E  DD 0010F           PUSHL   PTR                                     ; 1090
              00000000V   00              01  FB 00111           CALLS   #1, V2_SHOW_COMPAT                      ;
                          02              50  E8 00118           BLBS    R0, 9$                                  ;
                                          68  D4 0011B           CLRL    OUTDSC                                  ; 1092
                                  4100    8F  BB 0011D 9$:       PUSHR   #^M<R8,SP>                              ; 1100
                                          58  DD 00121           PUSHL   R8                                      ; 1094
                                          01  DD 00123           PUSHL   #1                                      ;
                                          01  DD 00125           PUSHL   #1                                      ;
                                      14  AE  DD 00127           PUSHL   PTR                                     ; 1096
                                00000000G 00  DD 0012A           PUSHL   NCP$GL_ENTITY                           ; 1095
              00000000V   00              07  FB 00130           CALLS   #7, NCP$FORMATPARM                      ;
                                          68  D5 00137           TSTL    OUTDSC                                  ; 1103
                                          C5  13 00139           BEQL    8$                                      ;
                                          58  DD 0013B           PUSHL   R8                                      ; 1105
                                          BA  11 0013D           BRB     7$                                      ;
                          16              52  E8 0013F 10$:      BLBS    STATUS, 11$                             ; 1108
                          6A              00  FB 00142           CALLS   #0, NCP$FAOSET                          ; 1111
                                          5B  DD 00145           PUSHL   R11                                     ; 1112
                                      24  A9  9F 00147           PUSHAB  P.ACZ                                   ;
              00000000V   00              02  FB 0014A           CALLS   #2, NCP$ADDSTR                          ;
              00000000V   00              00  FB 00151           CALLS   #0, NCP$FAOWRITE                        ; 1113
                                              04 00158 11$:      RET                                             ; 1119

; Routine Size:  345 bytes,    Routine Base:  $CODE$ + 0375
```

```
;  1129          1120  1 %SBTTL  'NCP$SHOCOLFMT  Produce Column Output'
;  1130          1121  1 ROUTINE NCP$SHOCOLFMT (LEN, BFR, RTNPTR) =      !
;  1131          1122  1
;  1132          1123  1 !++
;  1133          1124  1 ! FUNCTIONAL DESCRIPTION:
;  1134          1125  1 !
;  1135          1126  1 !       Format a message for column output.  We are given the buffer and
;  1136          1127  1 !       a table to use to format the parameters.  Write the header if
;  1137          1128  1 !       not written already, and change the parameter list into a line
;  1138          1129  1 !       of output for the table.
;  1139          1130  1 !       Parameters may appear in any order in the table and they may appear
;  1140          1131  1 !       more than once in the message and the table.  Any multiple entries
;  1141          1132  1 !       are used in order and additional lines are produced if no table entry
;  1142          1133  1 !       remains for the additional parameters.  These features are especially
;  1143          1134  1 !       useful for producing the link display.
;  1144          1135  1 !
;  1145          1136  1 ! FORMAL PARAMETERS:
;  1146          1137  1 !
;  1147          1138  1 !       LEN             Value of the length of the buffer in bytes
;  1148          1139  1 !       BFR             Address of the buffer
;  1149          1140  1 !       RTNPTR          Return pointer here
;  1150          1141  1 !
;  1151          1142  1 ! IMPLICIT INPUTS:
;  1152          1143  1 !
;  1153          1144  1 !       NCP$GL_ENTITY
;  1154          1145  1 !       NCP$A_COLHEAD
;  1155          1146  1 !       NCP$A_COLTBL
;  1156          1147  1 !       NCP$B_COLHEAD
;  1157          1148  1 !
;  1158          1149  1 ! IMPLICIT OUTPUTS:
;  1159          1150  1 !
;  1160          1151  1 !       NONE
;  1161          1152  1 !
;  1162          1153  1 ! ROUTINE VALUE:
;  1163          1154  1 ! COMPLETION CODES:
;  1164          1155  1 !
;  1165          1156  1 !       Success if written as column output
;  1166          1157  1 !       Failure otherwise
;  1167          1158  1 !
;  1168          1159  1 ! SIDE EFFECTS:
;  1169          1160  1 !
;  1170          1161  1 !       NONE
;  1171          1162  1 !
;  1172          1163  1 !--
;  1173          1164  1
;  1174          1165  2     BEGIN
;  1175          1166  2
;  1176          1167  2     LITERAL
;  1177          1168  2         LINSIZ = 128            ! Length of a line can be over 80
;  1178          1169  2         ;
;  1179          1170  2
;  1180          1171  2     LOCAL
;  1181          1172  2         ID,                     ! Parameter id temp
;  1182          1173  2         IDFND,                  ! Bool for id found
;  1183          1174  2         CTR,                    ! Byte counter
;  1184          1175  2         PAD,                    ! Ammount of padding to place before lines in
;  1185          1176  2                                 !  column display
```

```
;  1186          1177  2          PTR:    REF BBLOCK,      ! Pointer into the buffer
;  1187          1178  2          PEND,                    ! End of text
;  1188          1179  2          LIND$C : VECTOR [2],     ! Line descriptor
;  1189          1180  2          LINBUF : VECTOR [LINSIZ, BYTE], ! Line buffer
;  1190          1181  2          TBL : REF BBLOCKVECTOR [1, 4]   ! Table pointer
;  1191          1182  2          ;                        !
;  1192          1183  2
;  1193          1184  2      LABEL
;  1194          1185  2          BLDLINE                  ! Block to escape from
;  1195          1186  2          ;
;  1196          1187  2
;  1197          1188  2      MACRO                        ! Table entry fields
;  1198          1189  2
;  1199          1190  2          PRMID =  0, 0, 16, 0 %, ! Parameter id
;  1200          1191  2          FLDOFF = 2, 0, 8, 0 %,  ! Field offset in line
;  1201          1192  2          FLDSIZ = 3, 0, 8, 0 %   ! Field size in line
;  1202          1193  2          ;
;  1203          1194  2
;  1204          1195  2
;  1205          1196  2  !
;  1206          1197  2  !       Should we be here?
;  1207          1198  2  !
;  1208          1199  2
;  1209          1200  2      IF .NCP$A_COLTBL EQL 0               ! Column output wanted
;  1210          1201  2          OR
;  1211          1202  2          (
;  1212          1203  3          .NCP$GL_ENTITY EQL NMA$C_ENT_NOD ! Check for leading executor case
;  1213          1204  3          AND
;  1214          1205  3          CH$RCHAR (.BFR + 2) GTR 127      ! Executor bit set?
;  1215          1206  3          )
;  1216          1207  2      THEN
;  1217          1208  3          BEGIN
;  1218          1209  3          .RTNPTR = .BFR;                  ! Return false on no column output or
;  1219          1210  3          RETURN FALSE                     ! Executor display is first in a node
;  1220          1211  3          END                              ! display
;  1221          1212  2      ;
;  1222          1213  2
;  1223          1214  2      PTR = .BFR;                          ! Set pointer
;  1224          1215  2      PEND = .BFR + .LEN;                  ! Set end pointer
;  1225          1216  2
;  1226          1217  2      TBL = .NCP$A_COLTBL;                 ! Set table address
;  1227          1218  2      CH$FILL (' ', LINSIZ, LINBUF);       ! Blank the line buffer
;  1228          1219  2      PAD = 2;
;  1229          1220  2
;  1230          1221  2      NCP$FAOSET ();                       ! Set to convert entity
;  1231          1222  2
;  1232          1223  2      SELECTONE .NCP$GL_ENTITY OF          ! Deal with the entity code
;  1233          1224  2      SET
;  1234          1225  2
;  1235          1226  2      [NMA$C_ENT_NOD] :                    ! Node address and name
;  1236          1227  3          BEGIN
;  1237          1228  3          LOCAL
;  1238          1229  3              AREA,
;  1239          1230  3              AREA_ADDR : BBLOCK [4];
;  1240          1231  3
;  1241          1232  3          AREA_ADDR = (.(.PTR) < 0, 16, 0 > );
;  1242          1233  3          AREA = .AREA_ADDR [NMA$V_AREA];
```

```
;  1243          1234  3
;  1244          1235  3          IF .AREA EQL 0
;  1245          1236  3          THEN
;  1246          1237  4              BEGIN                               ! No area so handle normally
;  1247          1238  4              ADDSTR ('!3UW');
;  1248          1239  4              ADDFAO (.AREA_ADDR);
;  1249          1240  4              END
;  1250          1241  3          ELSE
;  1251          1242  4              BEGIN                               ! Non zero area must be separated from address with '.'
;  1252          1243  4              PAD = 0;                            ! Remove padding so it all fits
;  1253          1244  4              ADDSTR ('!2UW.!UW');
;  1254          1245  4              ADDFAO (.AREA);
;  1255          1246  4              ADDFAO (.AREA_ADDR [NMA$V_ADDR]);
;  1256          1247  3              END;
;  1257          1248  3
;  1258          1249  3          PTR = .PTR + 2;
;  1259          1250  3          CTR = CH$RCHAR (.PTR) AND %X'7F'; ! Executor bit
;  1260          1251  3          IF .CTR NEQ 0                           ! Is there a name present
;  1261          1252  3          THEN
;  1262          1253  4              BEGIN
;  1263          1254  4              ADDSTR (' (!AC)');
;  1264          1255  4              ADDFAO (.PTR)
;  1265          1256  4              END
;  1266          1257  3          ;
;  1267          1258  3          PTR = .PTR + .CTR + 1                   ! Over the node entity
;  1268          1259  2          END;
;  1269          1260  2
;  1270          1261  2      [NMA$C_ENT_CIR, NMA$C_ENT_LIN, -NMA$C_SENT_OBJ] :   ! Circuit, Line or object entity
;  1271          1262  3          BEGIN
;  1272          1263  3          ADDSTR ('!AC');                         ! Counted string
;  1273          1264  3          ADDFAO (.PTR);
;  1274          1265  3          PTR = .PTR + CH$RCHAR_A (PTR)
;  1275          1266  2          END;
;  1276          1267  2
;  1277          1268  2      [NMA$C_ENT_MOD]:                            ! MODULE
;  1278          1269  3          BEGIN                                   ! print the entity
;  1279          1270  3          NCP$SHOENTITY (PTR);                    ! Print entity string
;  1280          1271  3          NCP$FAOWRITE ();                        ! Write it to the file
;  1281          1272  3          NCP$FAOSET ();                          ! And set up for next one
;  1282          1273  2          END;
;  1283          1274  2
;  1284          1275  2      [-NMA$C_SENT_LNK]:                          ! Links
;  1285          1276  3          BEGIN
;  1286          1277  3          PTR = .PTR + 1;                         ! Skip by format byte (0 if link #)
;  1287          1278  3          ADDSTR('!UW');                          ! Word link number
;  1288          1279  3          ADDFAO(..PTR);
;  1289          1280  3          PTR = .PTR + 2;
;  1290          1281  2          END;
;  1291          1282  2
;  1292          1283  2      [NMA$C_ENT_ARE]:                            ! Area is a byte of zero
;  1293          1284  3          BEGIN                                   !  followed by byte of area number
;  1294          1285  3          ADDSTR ('!UB');
;  1295          1286  3          PTR = .PTR + 1;
;  1296          1287  3          ADDFAO (..PTR);
;  1297          1288  3          PTR = .PTR + 1;
;  1298          1289  2          END;
;  1299          1290  2
```

```
;  1300     1291  2      [OTHERWISE] :                          ! For all rest including logging
;  1301     1292  3          BEGIN
;  1302     1293  3          LOCAL                              ! Save logging type for a moment
;  1303     1294  3              LOGTYP : BYTE
;  1304     1295  3              ;
;  1305     1296  3          LOGTYP = CH$RCHAR (.PTR);          ! Save the logging type
;  1306     1297  3          ADDSTR ('!/ ');                    ! Blank line
;  1307     1298  3          NCP$SHOENTITY (PTR);               ! Print entity string
;  1308     1299  3          ADDSTR ('!/ ');                    ! Blank line
;  1309     1300  3          IF   .NCP$GL_ENTITY EQL NMA$C_ENT_LOG    ! If logging
;  1310     1301  3              AND
;  1311     1302  3              .NCP$B_LOGTYPE EQL .LOGTYP     ! and the type matches
;  1312     1303  3          THEN
;  1313     1304  4              BEGIN
;  1314     1305  4              NCP$FAOSET ()                  ! Ignore the entity
;  1315     1306  4              END
;  1316     1307  3          ELSE
;  1317     1308  4              BEGIN                          ! Otherwise print the entity
;  1318     1309  4              NCP$FAOWRITE ();               ! Write it to the file
;  1319     1310  4              NCP$FAOSET ();                 ! And set up for next one
;  1320     1311  4              NCP$B_LOGTYPE = .LOGTYP;       ! Save the log type
;  1321     1312  4              NCP$B_COLHEAD = FALSE          ! Column head printed again
;  1322     1313  4              END
;  1323     1314  2          END;
;  1324     1315  2
;  1325     1316  2      TES;
;  1326     1317  2
;  1327     1318  2
;  1328     1319  2      IF NOT .NCP$B_COLHEAD                  ! Write header if not already
;  1329     1320  2      THEN
;  1330     1321  3          BEGIN
;  1331     1322  3          NCP$WRITESHO (.NCP$A_COLHEAD);     ! Write header
;  1332     1323  3          NCP$B_COLHEAD = TRUE               ! None left to write
;  1333     1324  3          END
;  1334     1325  2      ;
;  1335     1326  2
;  1336     1327  2      OUTDSC [0] = OUTBUFSIZ;                ! Setup output buffer
;  1337     1328  2      OUTDSC [1] = OUTBUF;
;  1338     1329  2      NCP$FAOL (OUTDSC);                     ! Convert to text and move it in
;  1339     1330  2
;  1340     1331  2      CH$MOVE (.OUTDSC [0], .OUTDSC [1], LINBUF + .PAD);
;  1341     1332  2
;  1342     1333  2      LINDSC [0] = LINSIZ;                   ! Describe the line for writesho
;  1343     1334  2      LINDSC [1] = LINBUF;
;  1344     1335  2
;  1345     1336  2      BLDLINE:                               ! Block to build the line
;  1346     1337  3      BEGIN
;  1347     1338  3
;  1348     1339  3      IDFND = FALSE;                         ! We have not found any parms as yet
;  1349     1340  3
;  1350     1341  3      WHILE .PTR LSSA .PEND                  ! While there is data
;  1351     1342  3      DO
;  1352     1343  4          BEGIN
;  1353     1344  4          ID = .(.PTR) <0, 16, 0>;           ! Parameter id
;  1354     1345  4          INCR IDX FROM 0                    ! Scan the table for the parm
;  1355     1346  4          DO
;  1356     1347  5              BEGIN
```

```
;  1357     1348  5            IF .TBL [.IDX, PRMID] EQL 65535 ! If not found in table,
;  1358     1349  5            THEN
;  1359     1350  6                BEGIN
;  1360     1351  6                IF .IDFND                   ! If current line non-blank,
;  1361     1352  6                THEN                        ! use multiple lines
;  1362     1353  7                    BEGIN
;  1363     1354  7                    NCP$WRITESHO (LINDSC);      ! Write line so far
;  1364     1355  7                    NCP$FAOSET ();              ! Setup next one
;  1365     1356  7                    CH$FILL (' ', LINSIZ, LINBUF); ! Blank the line buffer
;  1366     1357  7                    IDX = 0;                    ! Reset to restart search
;  1367     1358  7                    IDFND = FALSE               ! Mark line empty again
;  1368     1359  7                    END
;  1369     1360  6                ELSE
;  1370     1361  6                    LEAVE BLDLINE               ! Not found, done here
;  1371     1362  6                END
;  1372     1363  5                ;
;  1373     1364  5                IF .TBL [.IDX, PRMID] EQL .ID ! Do the id's match?
;  1374     1365  5                THEN
;  1375     1366  6                    BEGIN
;  1376     1367  6                    IF CH$EQL                   ! If field in line blank, use it
;  1377     1368  6                        (
;  1378     1369  6                        .TBL [.IDX, FLDSIZ],            ! Size of source1
;  1379     1370  6                        .TBL [.IDX, FLDOFF] + LINBUF,   ! Address of source1
;  1380     1371  6                        0, LINBUF,                      ! Dummy source2
;  1381     1372  6                        ' ',                            ! fill
;  1382     1373  6                        )
;  1383     1374  6                    THEN
;  1384     1375  7                        BEGIN
;  1385     1376  7                        ID = .IDX;              ! Leave with the correct index
;  1386     1377  7                        EXITLOOP
;  1387     1378  7                        END
;  1388     1379  6                    ELSE
;  1389     1380  6                        IDFND = TRUE            ! Just mark line non-empty and go on
;  1390     1381  6                    END
;  1391     1382  5                END
;  1392     1383  4            ;
;  1393     1384  4
;  1394     1385  4            OUTDSC [0] = .TBL [.ID, FLDSIZ];        ! Set an output descriptor
;  1395     1386  4            OUTDSC [1] = .TBL [.ID, FLDOFF] + LINBUF;
;  1396     1387  4
;  1397     1388  4            IF NOT V2_SHOW_COMPAT(.PTR)     ! Map V2 to V3 parameters; if ignored,
;  1398     1389  4            THEN
;  1399     1390  4                OUTDSC [0] = 0;             ! Do not pass any result buffer
;  1400     1391  4
;  1401     1392  4            NCP$FORMATPARM(                 ! Format one parameter
;  1402     1393  4                    .NCP$GL_ENTITY,         ! Entity type
;  1403     1394  4                    .PTR,                   ! Source pointer
;  1404     1395  4                    FALSE,                  ! No name for param
;  1405     1396  4                    TRUE,                   ! Data for param
;  1406     1397  4                    OUTDSC,                 ! Descriptor
;  1407     1398  4                    OUTDSC [0],             ! Return length here
;  1408     1399  4                    PTR);                   ! Return pointer here
;  1409     1400  3            END;
;  1410     1401  2        END; ! Of labeled BLDLINE block
;  1411     1402  2
;  1412     1403  2        NCP$WRITESHO (LINDSC);              ! Write the line so far to output
;  1413     1404  2        .RTNPTR = .PTR;                     ! Return pointer for remainder
```

```
;  1414          1405  2
;  1415          1406  2     RETURN SUCCESS                    ! We wrote a line to output
;  1416          1407  2
;  1417          1408  1     END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                                 57  55  33  21  04  00816 P.ADA:   .ASCII  <4>\!3UW\                                ;
                                             57  55  32  21  08  0081B P.ADB:   .ASCII  <8>\!2UW.!UW\                              ;
                                         57  55  21  2E  29  43  41  21  28  20  06  00824 P.ADC:   .ASCII  <6>\ (!AC)\                              ;
                                                 43  41  21  03  0082B P.ADD:   .ASCII  <3>\!AC\                                  ;
                                                 57  55  21  03  0082F P.ADE:   .ASCII  <3>\!UW\                                  ;
                                                 42  55  21  03  00833 P.ADF:   .ASCII  <3>\!UB\                                  ;
                                                 20  2F  21  03  00837 P.ADG:   .ASCII  <3>\!/ \                                   ;
                                                 20  2F  21  03  0083B P.ADH:   .ASCII  <3>\!/ \                                   ;


                                                                .PSECT  $CODE$,NOWRT,2

                                                     0FFC 00000 NCP$SHOCOLFMT:
                                                                .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11                    ; 1121
                                                 5B 00000000'   00  9E 00002        MOVAB   OUTDSC, R11                            ;
                                                 5E     FF74    CE  9E 00009        MOVAB   -140(SP), SP                           ;
                                                 51       0204  CB  D0 0000E        MOVL    NCP$A_COLTBL, R1                       ; 1200
                                                                13  13 00013        BEQL    1$                                     ;
                                                    00000000G   00  D5 00015        TSTL    NCP$GL_ENTITY                          ; 1203
                                                                13  12 0001B        BNEQ    2$                                     ;
                                                 50       08    AC  D0 0001D        MOVL    BFR, R0                                ; 1205
                                                          7F 8F    02  A0 91 00021   CMPB    2(R0), #127                            ;
                                                                08  1B 00026        BLEQU   2$                                     ;
                                                          0C BC    08  AC  D0 00028 1$:     MOVL    BFR, @RTNPTR                   ; 1209
                                                                02A7  31 0002D      BRW     29$                                    ; 1210
                                                 6E       08    AC  D0 00030 2$:     MOVL    BFR, PTR                               ; 1214
                                             5A       08    AC  04  AC  C1 00034   ADDL3   LEN, BFR, PEND                         ; 1215
                                                 59            51  D0 0003A        MOVL    R1, TBL                                ; 1217
                                         0080 8F      20       6E  00  2C 0003D   MOVC5   #0, (SP), #32, #128, LINBUF               ; 1218
                                                               04  AE  00044                                                    ;
                                                 53            02  D0 00046        MOVL    #2, PAD                                ; 1219
                                                 00000000V     00  00  FB 00049   CALLS   #0, NCP$FAOSET                         ; 1221
                                                 52 00000000G  00  D0 00050        MOVL    NCP$GL_ENTITY, R2                      ; 1223
                                                               7D  12 00057        BNEQ    6$                                     ; 1226
                                                 54       00    BE  3C 00059        MOVZWL  @PTR, AREA_ADDR                        ; 1232
                                             52      54        06  0A  EF 0005D   EXTZV   #10, #6, AREA_ADDR, AREA               ; 1233
                                                               17  12 00062        BNEQ    3$                                     ; 1235
                                                    00000000'   00  9F 00064        PUSHAB  NCP$GQ_CTRDSC                          ; 1238
                                                    00000000'   00  9F 0006A        PUSHAB  P.ADA                                  ;
                                                 00000000V  00      02  FB 00070   CALLS   #2, NCP$ADDSTR                        ;
                                                               54  DD 00077        PUSHL   AREA_ADDR                              ; 1239
                                                               23  11 00079        BRB     4$                                     ;
                                                               53  D4 0007B 3$:     CLRL    PAD                                    ; 1243
                                                    00000000'   00  9F 0007D        PUSHAB  NCP$GQ_CTRDSC                          ; 1244
                                                    00000000'   00  9F 00083        PUSHAB  P.ADB                                  ;
                                                 00000000V  00      02  FB 00089   CALLS   #2, NCP$ADDSTR                        ;
                                                               52  DD 00090        PUSHL   AREA                                   ; 1245
```

```
                00000000V  00              01  FB 00092          CALLS   #1, NCP$ADDFAO
          7E          54                    00  EF 00099          EXTZV   #0, #10, AREA_ADDR, -(SP)                    ; 1246
                00000000V  00              01  FB 0009E 4$:      CALLS   #1, NCP$ADDFAO
                           6E              02  C0 000A5          ADDL2   #2, PTR                                      ; 1249
          52        00   BE               00  EF 000A8          EXTZV   #0, #7, @PTR, CTR                            ; 1250
                           07              1C  13 000AE          BEQL    5$                                           ; 1251
                                00000000'  00  9F 000B0          PUSHAB  NCP$GQ_CTRDSC                                ; 1254
                                00000000'  00  9F 000B6          PUSHAB  P.ADC
                00000000V  00              02  FB 000BC          CALLS   #2, NCP$ADDSTR
                           6E              DD  000C3          PUSHL   PTR                                          ; 1255
                00000000V  00              01  FB 000C5          CALLS   #1, NCP$ADDFAO
                     50    6E              52  C1 000CC 5$:      ADDL3   CTR, PTR, R0                                 ; 1258
                           6E        01    A0  9E 000D0          MOVAB   1(R0), PTR
                                           7D  11 000D4          BRB     10$
                FFFFFFFC   8F              52  D1 000D6 6$:      CMPL    R2, #-4                                      ; 1261
                                           0A  13 000DD          BEQL    7$
                           01              52  D1 000DF          CMPL    R2, #1
                                           05  13 000E2          BEQL    7$
                           03              52  D1 000E4          CMPL    R2, #3
                                           27  12 000E7          BNEQ    8$
                                00000000'  00  9F 000E9 7$:      PUSHAB  NCP$GQ_CTRDSC                                ; 1263
                                00000000'  00  9F 000EF          PUSHAB  P.ADD
                00000000V  00              02  FB 000F5          CALLS   #2, NCP$ADDSTR
                           6E              DD  000FC          PUSHL   PTR                                          ; 1264
                00000000V  00              01  FB 000FE          CALLS   #1, NCP$ADDFAO
                           50        00    BE  9A 00105          MOVZBL  @PTR, R0                                     ; 1265
                           6E              D6  00109          INCL    PTR
                           6E              50  C0 0010B          ADDL2   R0, PTR
                                           6B  11 0010E          BRB     12$
                           04              52  D1 00110 8$:      CMPL    R2, #4                                       ; 1268
                                           13  12 00113          BNEQ    9$
                                           5E  DD 00115          PUSHL   SP                                           ; 1270
     00000000V  00                         01  FB 00117          CALLS   #1, NCP$SHOENTITY
     00000000V  00                         00  FB 0011E          CALLS   #0, NCP$FAOWRITE                             ; 1271
                                      0098 31  00125          BRW     14$                                          ; 1272
     FFFFFFF9   8F                         52  D1 00128 9$:      CMPL    R2, #-7                                      ; 1275
                                           24  12 0012F          BNEQ    11$
                                           6E  D6 00131          INCL    PTR                                          ; 1277
                                00000000'  00  9F 00133          PUSHAB  NCP$GQ_CTRDSC                                ; 1278
                                00000000'  00  9F 00139          PUSHAB  P.ADE
     00000000V  00                         02  FB 0013F          CALLS   #2, NCP$ADDSTR
                           00        BE    DD  00146          PUSHL   @PTR                                         ; 1279
     00000000V  00                         01  FB 00149          CALLS   #1, NCP$ADDFAO
                6E                         02  C0 00150          ADDL2   #2, PTR                                      ; 1280
                                           72  11 00153 10$:     BRB     15$                                          ; 1223
                05                         52  D1 00155 11$:     CMPL    R2, #5                                       ; 1283
                                           23  12 00158          BNEQ    13$
                                00000000'  00  9F 0015A          PUSHAB  NCP$GQ_CTRDSC                                ; 1285
                                00000000'  00  9F 00160          PUSHAB  P.ADF
     00000000V  00                         02  FB 00166          CALLS   #2, NCP$ADDSTR
                                           6E  D6 0016D          INCL    PTR                                          ; 1286
                           00        BE    DD  0016F          PUSHL   @PTR                                         ; 1287
     00000000V  00                         01  FB 00172          CALLS   #1, NCP$ADDFAO
                                           6E  D6 00179          INCL    PTR                                          ; 1288
                                           63  11 0017B 12$:     BRB     17$                                          ; 1223
                52              00         BE  90 0017D 13$:     MOVB    @PTR, LOGTYP                                 ; 1296
                           00000000'       00  9F 00181          PUSHAB  NCP$GQ_CTRDSC                                ; 1297
```

```
                                00000000'    00   9F  00187          PUSHAB  P.ADG                                   ;
                    00000000V   00                02  FB  0018D      CALLS   #2, NCP$ADDSTR                          ;
                                             5E   DD  00194          PUSHL   SP                                      ; 1298
                    00000000V   00               01  FB  00196      CALLS   #1, NCP$SHOENTITY                       ;
                                00000000'    00   9F  0019D          PUSHAB  NCP$GQ_CTRDSC                           ; 1299
                                00000000'    00   9F  001A3          PUSHAB  P.ADH                                   ;
                    00000000V   00                02  FB  001A9      CALLS   #2, NCP$ADDSTR                          ;
                                02 00000000G 00   D1  001B0          CMPL    NCP$GQ_ENTITY, #2                       ; 1300
                                             10   12  001B7          BNEQ    16$                                     ;
                                52      0201 CB   91  001B9          CMPB    NCP$B_LOGTYPE, LOGTYP                   ; 1302
                                             09   12  001BE          BNEQ    16$                                     ;
                    00000000V   00               00  FB  001C0 14$:  CALLS   #0, NCP$FAOSET                         ; 1305
                                             17   11  001C7 15$:     BRB     17$                                     ; 1304
                    00000000V   00               00  FB  001C9 16$:  CALLS   #0, NCP$FAOWRITE                       ; 1309
                    00000000V   00               00  FB  001D0      CALLS   #0, NCP$FAOSET                          ; 1310
                       0201 CB                   52  90  001D7      MOVB    LOGTYP, NCP$B_LOGTYPE                   ; 1311
                                0200 CB          94  001DC          CLRB    NCP$B_COLHEAD                           ; 1312
                                10   0200 CB    E8  001E0 17$:     BLBS    NCP$B_COLHEAD, 18$                      ; 1319
                                01FC CB          DD  001E5          PUSHL   NCP$A_COLHEAD                           ; 1322
                    00000000G   00               01  FB  001E9      CALLS   #1, NCP$WRITESHO                        ;
                       0200 CB                   01  90  001F0      MOVB    #1, NCP$B_COLHEAD                       ; 1323
                                6B      01F4 8F  3C  001F5 18$:     MOVZWL  #500, OUTDSC                            ; 1327
                          04 AB          08 AB  9E  001FA          MOVAB   OUTBUF, OUTDSC+4                        ; 1328
                                             5B   DD  001FF          PUSHL   R11                                     ; 1329
                    00000000V   00               01  FB  00201      CALLS   #1, NCP$FAOL                            ;
                                50           04  AB  D0  00208      MOVL    OUTDSC+4, R0                            ; 1331
                04 AE43         60               6B  28  0020C      MOVC3   OUTDSC, (R0), LINBUF[PAD]               ;
                          F8 AD          80 8F  9A  00212          MOVZBL  #128, LINDSC                            ; 1333
                          FC AD          04 AE  9E  00217          MOVAB   LINBUF, LINDSC+4                        ; 1334
                                             58   D4  0021C          CLRL    IDFND                                   ; 1339
                                5A               6E  D1  0021E 19$:     CMPL    PTR, PEND                               ; 1341
                                                 03  1F  00221          BLSSU   21$                                     ;
                                             009F 31  00223 20$:     BRW     28$                                     ;
                                57           00  BE  3C  00226 21$:     MOVZWL  @PTR, ID                                ; 1344
                                                 56  D4  0022A          CLRL    IDX                                     ; 1345
                                              6946 DF  0022C 22$:     PUSHAL  (TBL)[IDX]                              ; 1348
                             FFFF 8F             9E  B1  0022F          CMPW    @(SP)+, #65535                          ;
                                                 21  12  00234          BNEQ    23$                                     ;
                                EA               58  E9  00236          BLBC    IDFND, 20$                              ; 1351
                                             F8  AD  9F  00239          PUSHAB  LINDSC                                  ; 1354
                    00000000G   00               01  FB  0023C      CALLS   #1, NCP$WRITESHO                        ;
                    00000000V   00               00  FB  00243      CALLS   #0, NCP$FAOSET                          ; 1355
0080 8F         20              6E               00  2C  0024A      MOVC5   #0, (SP), #32, #128, LINBUF             ; 1356
                                             04  AE      00251      ;
                                                 56  D4  00253          CLRL    IDX                                     ; 1357
                                                 58  D4  00255          CLRL    IDFND                                   ; 1358
                                50               6946 DE  00257 23$:     MOVAL   (TBL)[IDX], R0                          ; 1364
          57                60               10   00  ED  0025B      CMPZV   #0, #16, (R0), ID                       ;
                                                 1B  12  00260          BNEQ    25$                                     ;
                                51           03  A0  9A  00262          MOVZBL  3(R0), R1                               ; 1369
                                50           02  A0  9A  00266          MOVZBL  2(R0), R0                               ; 1370
          00                20        04 AE40  51  2D  0026A          CMPC5   R1, LINBUF[R0], #32, #0, LINBUF         ; 1368
                                             04  AE      00271      ;
                                                 05  12  00273          BNEQ    24$                                     ;
                                57               56  D0  00275          MOVL    IDX, ID                                 ; 1376
                                                 0B  11  00278          BRB     26$                                     ; 1375
                                58               01  D0  0027A 24$:     MOVL    #1, IDFND                               ; 1380
```

```
                A7      .       56 7FFFFFFF    8F   F3 0027D 25$:     AOBLEQ  #2147483647, IDX, 22$                  ; 1364
                                50         6947   DE 00285 26$:     MOVAL   (TBL)[ID], R0                           ; 1385
                                6B         03  A0  9A 00289         MOVZBL  3(R0), OUTDSC                           ;
                                51         04  AE  9E 0028D         MOVAB   LINBUF, R1                              ; 1386
                                52         02  A0  9A 00291         MOVZBL  2(R0), R2                               ;
            04  AB              51                 52   C1 00295         ADDL3   R2, R1, OUTDSC+4                    ;
                                                   6E   DD 0029A         PUSHL   PTR                                 ; 1388
                00000000V       00                 01   FB 0029C         CALLS   #1, V2_SHOW_COMPAT                  ;
                                02                 50   E8 002A3         BLBS    R0, 27$                             ;
                                                   6B   D4 002A6         CLRL    OUTDSC                              ; 1390
                                             4800  8F   BB 002A8 27$:     PUSHR   #^M<R11,SP>                         ; 1398
                                                   5B   DD 002AC         PUSHL   R11                                 ; 1392
                                                   01   DD 002AE         PUSHL   #1                                  ;
                                                   7E   D4 002B0         CLRL    -(SP)                               ;
                                               14  AE   DD 002B2         PUSHL   PTR                                 ; 1394
                                        00000000G  00   DD 002B5         PUSHL   NCP$GL_ENTITY                       ; 1393
                00000000V       00                 07   FB 002BB         CALLS   #7, NCP$FORMATPARM                  ;
                                                 FF59   31   002C2         BRW     19$                                 ; 1341
                                               F8  AD   9F   002C5 28$:     PUSHAB  LINDSC                          ; 1403
                00000000G       00                 01   FB 002C8         CALLS   #1, NCP$WRITESHO                    ;
                          0C    BC                 6E   D0 002CF         MOVL    PTR, @RTNPTR                        ; 1404
                                50                 01   D0 002D3         MOVL    #1, R0                              ; 1406
                                                        04 002D6         RET                                         ;
                                                   50   D4 002D7 29$:     CLRL    R0                                  ; 1408
                                                        04 002D9         RET                                         ;

; Routine Size:  730 bytes,    Routine Base:  $CODE$ + 04CE
```

```
%SBTTL  'NCP$SHOENTITY  Convert Entity in Return'
GLOBAL ROUTINE NCP$SHOENTITY (PTRADR) :NOVALUE =         !

!++
! FUNCTIONAL DESCRIPTION:
!
!       Convert an entity code in a return to fao control string and
!       parameters.
!
! FORMAL PARAMETERS:
!
!       PTRADR           Address of pointer to entity encoding in buffer
!                        Return pointer past entity here
!
! IMPLICIT INPUTS:
!
!       NCP$GL_ENTITY   Entity code
!
! IMPLICIT OUTPUTS:
!
!       NONE
!
! ROUTINE VALUE:
! COMPLETION CODES:
!
!       NONE
!
! SIDE EFFECTS:
!
!       NONE
!
!--

    BEGIN

    LOCAL
        CTR,                                ! Local counter
        PTR                                 ! Local pointer
        ;               !

    PTR = ..PTRADR;                         ! Setup pointer

    SELECTONE .NCP$GL_ENTITY OF             ! Format the header for the block
    SET

    [NMA$C_ENT_NOD] :                       ! Report address and name for a node
        BEGIN
        LOCAL
            NOD_ADR,                        ! Address of node
            AREA,                           ! Area
            AREA_ADDR : BBLOCK [4];         ! Node address in low 10 bits, Area in upper 6 bits

        AREA_ADDR = (.(.PTR) < 0, 16, 0 > );
        AREA = .AREA_ADDR [NMA$V_AREA]; ! Extract the area from the upper 6 bits
        NOD_ADR = .AREA_ADDR [NMA$V_ADDR];

        IF .AREA EQL 0
```

```
;  1476          1466  3          THEN                                ! No area so handle normally
;  1477          1467  3              ADDFAO (.NOD_ADR)
;  1478          1468  3          ELSE                                ! Non zero area must be separated from address with '.'
;  1479          1469  4              BEGIN
;  1480          1470  4              ADDFAO (.AREA);
;  1481          1471  4              ADDFAO (.NOD_ADR);
;  1482          1472  3              END;
;  1483          1473  3
;  1484          1474  3          PTR = .PTR + 2;
;  1485          1475  3          IF .BBLOCK [.PTR, NMA$V_ENT_EXE]        ! This the executor?
;  1486          1476  3          THEN
;  1487          1477  3              IF .AREA EQL 0
;  1488          1478  3              THEN
;  1489          1479  3                  ADDSTR     ('Executor node = !UW')      ! Executor node
;  1490          1480  3              ELSE
;  1491          1481  3                  ADDSTR     ('Executor node = !UW.!UW')  ! Executor node area and address
;  1492          1482  3          ELSE
;  1493          1483  4              BEGIN
;  1494          1484  4              IF .NOD_ADR EQL 0                   ! Is this a loop node?
;  1495          1485  4              THEN
;  1496          1486  5                  BEGIN
;  1497          1487  5                  IF .AREA EQL 0
;  1498          1488  5                  THEN
;  1499          1489  5                      ADDSTR ('Loop node =     !UW')      ! Loop node
;  1500          1490  5                  ELSE
;  1501          1491  5                      ADDSTR ('Loop node =     !UW.!UW')  ! Loop node area and address
;  1502          1492  5                  END
;  1503          1493  4              ELSE
;  1504          1494  4                  IF .AREA EQL 0
;  1505          1495  4                  THEN
;  1506          1496  4                      ADDSTR     ('Remote node =   !UW')  ! Other node
;  1507          1497  4                  ELSE
;  1508          1498  4                      ADDSTR     ('Remote node =   !UW.!UW')      ! Other node
;  1509          1499  4              END
;  1510          1500  3              ;
;  1511          1501  3              CTR = CH$RCHAR (.PTR) AND %X'7F'; ! Get count and drop exec node bit
;  1512          1502  3              CH$WCHAR (.CTR, .PTR);           ! Put the real count back
;  1513          1503  3              IF .CTR NEQ 0                    ! Is there a name
;  1514          1504  3              THEN
;  1515          1505  4                  BEGIN                        ! Report the name
;  1516          1506  4                  ADDSTR (' (!AC)');
;  1517          1507  4                  ADDFAO (.PTR);
;  1518          1508  4                  END
;  1519          1509  3              ;
;  1520          1510  3              PTR = .PTR + .CTR + 1            ! Advance the pointer
;  1521          1511  2              END;
;  1522          1512  2
;  1523          1513  2          [NMA$C_ENT_LIN] :                    ! The line is a counted string
;  1524          1514  3              (
;  1525          1515  3              ADDSTR ('Line = !AC');
;  1526          1516  3              ADDFAO (.PTR);
;  1527          1517  3              PTR = .PTR + .(.PTR) <0,8,0> + 1
;  1528          1518  2              );
;  1529          1519  2
;  1530          1520  2          [-NMA$C_SENT_LNK]:                   ! Link address
;  1531          1521  3              BEGIN
;  1532          1522  3              PTR = .PTR + 1;                  ! Skip by format byte (0=link #)
```

```
;  1533          1523  3          ADDSTR('Link = !UW');
;  1534          1524  3          ADDFAO(..PTR);
;  1535          1525  3          PTR = .PTR + 2;
;  1536          1526  2          END;
;  1537          1527  2
;  1538          1528  2      [NMA$C_ENT_LOG] :                   ! Logging is a byte of sink type
;  1539          1529  3          (
;  1540          1530  3          ADDSTR ('Logging sink type = ');
;  1541          1531  3          SELECTONE .(.PTR) <0, 8, 0> OF  ! Obtain the proper code
;  1542          1532  3          SET
;  1543          1533  3
;  1544          1534  3          [NMA$C_SNK_CON] : ADDSTR ('console');
;  1545          1535  3          [NMA$C_SNK_FIL] : ADDSTR ('file');
;  1546          1536  3          [NMA$C_SNK_MON] : ADDSTR ('monitor');
;  1547          1537  3
;  1548          1538  3          TES
;  1549          1539  3          ;
;  1550          1540  3          PTR = .PTR + 1
;  1551          1541  2          );
;  1552          1542  2
;  1553          1543  2      [-NMA$C_SENT_OBJ] :                 ! Object is a counted string
;  1554          1544  3          (
;  1555          1545  3          ADDSTR ('Object = !AC');
;  1556          1546  3          ADDFAO (.PTR);
;  1557          1547  3          PTR = .PTR + .(.PTR) <0, 8, 0> + 1
;  1558          1548  2          );
;  1559          1549  2
;  1560          1550  2      [NMA$C_ENT_CIR]:                    ! Circuit name is a counted string
;  1561          1551  3          BEGIN
;  1562          1552  3          ADDSTR ('Circuit = !AC');
;  1563          1553  3          ADDFAO (.PTR);
;  1564          1554  3          PTR = .PTR + CH$RCHAR(.PTR) + 1;
;  1565          1555  2          END;
;  1566          1556  2
;  1567          1557  2      [NMA$C_ENT_MOD]:                    ! MODULE type should be skipped past
;  1568          1558  3          BEGIN
;  1569          1559  3          PTR = .PTR + CH$RCHAR(.PTR) + 1;
;  1570          1560  2          END;
;  1571          1561  2
;  1572          1562  2      [NMA$C_ENT_ARE]:                    ! Area is a byte of zero
;  1573          1563  3          BEGIN                           !  followed by byte of area number
;  1574          1564  3          ADDSTR ('Area = !UB');
;  1575          1565  3          PTR = .PTR + 1;
;  1576          1566  3          ADDFAO (..PTR);
;  1577          1567  3          PTR = .PTR + 1;
;  1578          1568  2          END;
;  1579          1569  2
;  1580          1570  2      TES;
;  1581          1571  2
;  1582          1572  2      .PTRADR = .PTR;                     ! Return pointer beyond entity
;  1583          1573  2
;  1584          1574  2      RETURN
;  1585          1575  2
;  1586          1576  1      END;
```

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

20 65 64 6F 6E 20 72 6F 74 75 63 65 78 45 13  0083F P.ADI:   .ASCII  <19>\Executor node = !UW\
                                                 57 55 21 20 3D 0084E
20 65 64 6F 6E 20 72 6F 74 75 63 65 78 45 17  00853 P.ADJ:   .ASCII  <23>\Executor node = !UW.!UW\
                                        57 55 21 2E 57 55 21 20 3D 00862
20 20 20 3D 20 65 64 6F 6E 20 70 6F 6F 4C 13  0086B P.ADK:   .ASCII  <19>\Loop node =     !UW\
                                                 57 55 21 20 20 0087A
20 20 20 3D 20 65 64 6F 6E 20 70 6F 6F 4C 17  0087F P.ADL:   .ASCII  <23>\Loop node =     !UW.!UW\
                                        57 55 21 2E 57 55 21 20 20 0088E
20 3D 20 65 64 6F 6E 20 65 74 6F 6D 65 52 13  00897 P.ADM:   .ASCII  <19>\Remote node =   !UW\
                                                 57 55 21 20 20 008A6
20 3D 20 65 64 6F 6E 20 65 74 6F 6D 65 52 17  008AB P.ADN:   .ASCII  <23>\Remote node =   !UW.!UW\
                                        57 55 21 2E 57 55 21 20 20 008BA
                                        29 43 41 21 28 20 06  008C3 P.ADO:   .ASCII  <6>\ (!AC)\
                  43 41 21 20 3D 20 65 6E 69 4C 0A  008CA P.ADP:   .ASCII  <10>\Line = !AC\
                  57 55 21 20 3D 20 6B 6E 69 4C 0A  008D5 P.ADQ:   .ASCII  <10>\Link = !UW\
74 20 6B 6E 69 73 20 67 6E 69 67 67 6F 4C 14  008E0 P.ADR:   .ASCII  <20>\Logging sink type = \
                                        20 3D 20 65 70 79 008EF
                              65 6C 6F 73 6E 6F 63 07  008F5 P.ADS:   .ASCII  <7>\console\
                                    65 6C 69 66 04  008FD P.ADT:   .ASCII  <4>\file\
                              72 6F 74 69 6E 6F 6D 07  00902 P.ADU:   .ASCII  <7>\monitor\
         43 41 21 20 3D 20 74 63 65 6A 62 4F 0C  0090A P.ADV:   .ASCII  <12>\Object = !AC\
      43 41 21 20 3D 20 74 69 75 63 72 69 43 0D  00917 P.ADW:   .ASCII  <13>\Circuit = !AC\
               42 55 21 20 3D 20 61 65 72 41 0A  00925 P.ADX:   .ASCII  <10>\Area = !UB\


                                                                                .PSECT  $CODE$,NOWRT,2

                                    01FC 00000          .ENTRY  NCP$SHOENTITY, Save R2,R3,R4,R5,R6,R7,R8    ; 1410
                              58 00000000V  00  9E 00002        MOVAB   NCP$ADDFAO, R8                      ;
                              57 00000000V  00  9E 00009        MOVAB   NCP$ADDSTR, R7                      ;
                              56 00000000'  00  9E 00010        MOVAB   P.ADI, R6                           ;
                              55 00000000'  00  9E 00017        MOVAB   NCP$GQ_CTRDSC, R5                   ;
                                    52    04  BC  D0 0001E        MOVL    @PTRADR, PTR                        ; 1449
                              53 00000000G  00  D0 00022        MOVL    NCP$GL_ENTITY, R3                   ; 1451
                                          03  13 00029        BEQL    1$                                  ; 1454
                                        0080  31 0002B        BRW     11$                                 ;
                                          50  62 3C 0002E 1$:  MOVZWL  (PTR), AREA_ADDR                   ; 1461
51                                        06  0A EF 00031        EXTZV   #10, #6, AREA_ADDR, AREA            ; 1462
53                50                      0A  00 EF 00036        EXTZV   #0, #10, AREA_ADDR, NOD_ADR         ; 1463
                                              54 D4 0003B        CLRL    R4                                  ; 1465
                                              51 D5 0003D        TSTL    AREA                                ;
                                              04 12 0003F        BNEQ    2$                                  ;
                                              54 D6 00041        INCL    R4                                  ;
                                              05 11 00043        BRB     3$                                  ; 1467
                                              51 DD 00045 2$:    PUSHL   AREA                                ; 1470
                                    68        01 FB 00047        CALLS   #1, NCP$ADDFAO                      ;
                                              53 DD 0004A 3$:    PUSHL   NOD_ADR                             ; 1471
                                    68        01 FB 0004C        CALLS   #1, NCP$ADDFAO                      ;
                                    52        02 C0 0004F        ADDL2   #2, PTR                             ; 1474
                                              62 95 00052        TSTB    (PTR)                               ; 1475
                                              10 18 00054        BGEQ    5$                                  ;
                                    06        54 E9 00056        BLBC    R4, 4$                              ; 1477
                                              55 DD 00059        PUSHL   R5                                  ; 1479
                                              56 DD 0005B        PUSHL   R6                                  ;
```

```
                                  2B  11 0005D       BRB     9$                      ;
                                  55  DD 0005F 4$:   PUSHL   R5                      ; 1481
                          14      A6  9F 00061       PUSHAB  P.ADJ                   ;
                                  24  11 00064       BRB     9$                      ;
                                  53  D5 00066 5$:   TSTL    NOD_ADR                 ; 1484
                                  11  12 00068       BNEQ    7$                      ;
                    07            54  E9 0006A       BLBC    R4, 6$                  ; 1487
                                  55  DD 0006D       PUSHL   R5                      ; 1489
                          2C      A6  9F 0006F       PUSHAB  P.ADK                   ;
                                  16  11 00072       BRB     9$                      ;
                                  55  DD 00074 6$:   PUSHL   R5                      ; 1491
                          40      A6  9F 00076       PUSHAB  P.ADL                   ;
                                  0F  11 00079       BRB     9$                      ;
                    07            54  E9 0007B 7$:   BLBC    R4, 8$                  ; 1494
                                  55  DD 0007E       PUSHL   R5                      ; 1496
                          58      A6  9F 00080       PUSHAB  P.ADM                   ;
                                  05  11 00083       BRB     9$                      ;
                                  55  DD 00085 8$:   PUSHL   R5                      ; 1498
                          6C      A6  9F 00087       PUSHAB  P.ADN                   ;
                    67            02  FB 0008A 9$:   CALLS   #2, NCP$ADDSTR          ;
   53          62   07            00  EF 0008D       EXTZV   #0, #7, (PTR), CTR      ; 1501
                    62            53  90 00092       MOVB    CTR, (PTR)              ; 1502
                                  53  D5 00095       TSTL    CTR                     ; 1503
                                  0E  13 00097       BEQL    10$                     ;
                                  55  DD 00099       PUSHL   R5                      ; 1506
                        0084      C6  9F 0009B       PUSHAB  P.ADO                   ;
                    67            02  FB 0009F       CALLS   #2, NCP$ADDSTR          ;
                                  52  DD 000A2       PUSHL   PTR                     ; 1507
                    68            01  FB 000A4       CALLS   #1, NCP$ADDFAO          ;
                    52         01 A342 9E 000A7 10$: MOVAB   1(CTR)[PTR], PTR        ; 1510
                                  29  11 000AC       BRB     13$                     ;
                    01            53  D1 000AE 11$:  CMPL    R3, #1                  ; 1513
                                  08  12 000B1       BNEQ    12$                     ;
                                  55  DD 000B3       PUSHL   R5                      ; 1515
                        008B      C6  9F 000B5       PUSHAB  P.ADP                   ;
                                  72  11 000B9       BRB     20$                     ;
               FFFFFFF9   8F      53  D1 000BB 12$:  CMPL    R3, #-7                 ; 1520
                                  15  12 000C2       BNEQ    14$                     ;
                                  52  D6 000C4       INCL    PTR                     ; 1522
                                  55  DD 000C6       PUSHL   R5                      ; 1523
                        0096      C6  9F 000C8       PUSHAB  P.ADQ                   ;
                    67            02  FB 000CC       CALLS   #2, NCP$ADDSTR          ;
                                  62  DD 000CF       PUSHL   (PTR)                   ; 1524
                    68            01  FB 000D1       CALLS   #1, NCP$ADDFAO          ;
                    52            02  C0 000D4       ADDL2   #2, PTR                 ; 1525
                                  6B  11 000D7 13$:  BRB     23$                     ; 1451
                    02            53  D1 000D9 14$:  CMPL    R3, #2                  ; 1528
                                  33  12 000DC       BNEQ    18$                     ;
                                  55  DD 000DE       PUSHL   R5                      ; 1530
                        00A1      C6  9F 000E0       PUSHAB  P.ADR                   ;
                    67            02  FB 000E4       CALLS   #2, NCP$ADDSTR          ;
                    01            62  91 000E7       CMPB    (PTR), #1               ; 1534
                                  08  12 000EA       BNEQ    15$                     ;
                                  55  DD 000EC       PUSHL   R5                      ;
                        00B6      C6  9F 000EE       PUSHAB  P.ADS                   ;
                                  18  11 000F2       BRB     17$                     ;
                    02            62  91 000F4 15$:  CMPB    (PTR), #2               ; 1535
```

```
                                    08  12 000F7          BNEQ    16$                                     ;
                                    55  DD 000F9          PUSHL   R5                                      ;
                              00BE  C6  9F 000FB          PUSHAB  P.ADT                                   ;
                                    0B  11 000FF          BRB     17$                                     ;
                           03       62  91 00101 16$:     CMPB    (PTR), #3                               ; 1536
                                    55  12 00104          BNEQ    25$                                     ;
                                    55  DD 00106          PUSHL   R5                                      ;
                              00C3  C6  9F 00108          PUSHAB  P.ADU                                   ;
                           67       02  FB 0010C 17$:     CALLS   #2, NCP$ADDSTR                          ;
                                    4A  11 0010F          BRB     25$                                     ; 1540
                     FFFFFFFC  8F       53  D1 00111 18$:     CMPL    R3, #-4                             ; 1543
                                    08  12 00118          BNEQ    19$                                     ;
                                    55  DD 0011A          PUSHL   R5                                      ; 1545
                              00CB  C6  9F 0011C          PUSHAB  P.ADV                                   ;
                                    0B  11 00120          BRB     20$                                     ;
                           03       53  D1 00122 19$:     CMPL    R3, #3                              ; 1550
                                    10  12 00125          BNEQ    21$                                     ;
                                    55  DD 00127          PUSHL   R5                                      ; 1552
                              00D8  C6  9F 00129          PUSHAB  P.ADW                                   ;
                           67       02  FB 0012D 20$:     CALLS   #2, NCP$ADDSTR                          ;
                                    52  DD 00130          PUSHL   PTR                                     ; 1553
                           68       01  FB 00132          CALLS   #1, NCP$ADDFAO                          ;
                                    05  11 00135          BRB     22$                                     ; 1554
                           04       53  D1 00137 21$:     CMPL    R3, #4                              ; 1557
                                    0A  12 0013A          BNEQ    24$                                     ;
                           50       62  9A 0013C 22$:     MOVZBL  (PTR), R0                               ; 1559
                           52    01 A042  9E 0013F          MOVAB   1(R0)[PTR], PTR                         ;
                                    17  11 00144 23$:     BRB     26$                                     ; 1451
                           05       53  D1 00146 24$:     CMPL    R3, #5                              ; 1562
                                    12  12 00149          BNEQ    26$                                     ;
                                    55  DD 0014B          PUSHL   R5                                      ; 1564
                              00E6  C6  9F 0014D          PUSHAB  P.ADX                                   ;
                           67       02  FB 00151          CALLS   #2, NCP$ADDSTR                          ;
                                    52  D6 00154          INCL    PTR                                     ; 1565
                                    62  DD 00156          PUSHL   (PTR)                                   ; 1566
                           68       01  FB 00158          CALLS   #1, NCP$ADDFAO                          ;
                                    52  D6 0015B 25$:     INCL    PTR                                     ; 1567
                        04  BC          52  D0 0015D 26$:     MOVL    PTR, @PTRADR                        ; 1572
                                        04 00161          RET                                             ; 1576

; Routine Size:  354 bytes,    Routine Base:  $CODE$ + 07A8
```

```
;  1588     1577  1 %SBTTL  'V2_SHOW_COMPAT  Convert V2.0 NICE SHOW response'
;  1589     1578  1 ROUTINE V2_SHOW_COMPAT (PARM: REF BBLOCK) =
;  1590     1579  1
;  1591     1580  1 !---
;  1592     1581  1 !
;  1593     1582  1 !       This routine is called before each parameter is formatted
;  1594     1583  1 !       to convert the parameter into the appropriate V3.0 NICE
;  1595     1584  1 !       parameter, if we are currently connected to a V2.0 NML.
;  1596     1585  1 !
;  1597     1586  1 ! Inputs:
;  1598     1587  1 !
;  1599     1588  1 !       parm = Address of parameter (starting with parameter ID word)
;  1600     1589  1 !
;  1601     1590  1 ! Outputs:
;  1602     1591  1 !
;  1603     1592  1 !       Routine = True if mapped into V3 parameter; False if does not
;  1604     1593  1 !                       apply to V3 and should be ignored.
;  1605     1594  1 !---
;  1606     1595  1
;  1607     1596  2 BEGIN
;  1608     1597  2
;  1609     1598  2 EXTERNAL
;  1610     1599  2     NCP$GL_EXELCB: REF BBLOCK;          ! Address of current LCB
;  1611     1600  2
;  1612     1601  2 IF CH$NEQ(3, NCP$GL_EXELCB [LCB$B_NMLVERS],     ! If not NML V2.0,
;  1613     1602  2         3, UPLIT BYTE(2,0,0), 0)
;  1614     1603  2 THEN
;  1615     1604  2     RETURN TRUE;                        ! then leave the message stand as is
;  1616     1605  2
;  1617     1606  2 IF .NCP$GL_OPTION [NMA$V_OPT_ENT] NEQ NMA$C_ENT_LIN     ! If not SHOW LINE response,
;  1618     1607  2 THEN
;  1619     1608  2     RETURN TRUE;                        ! then leave it stand as is
;  1620     1609  2
;  1621     1610  2 IF .PARM [NMA$V_CNT_COU]                ! If its a counter,
;  1622     1611  2 THEN
;  1623     1612  2     RETURN TRUE;                        ! then format it as is
;  1624     1613  2
;  1625     1614  2 SELECTONEU .NCP$GL_ENTITY               ! Based on original entity requested,
;  1626     1615  2 OF
;  1627     1616  2     SET
;  1628     1617  2     [NMA$C_ENT_CIR]:                    ! Map V2 lines -> V3 circuits
;  1629     1618  3         BEGIN
;  1630     1619  3         BIND LINE_TO_CIRCUIT = UPLIT WORD(
;  1631     1620  3                 NMA$C_PCLI_STA, NMA$C_PCCI_STA,
;  1632     1621  3                 NMA$C_PCLI_SER, NMA$C_PCCI_SER,
;  1633     1622  3                 NMA$C_PCLI_LCT, NMA$C_PCCI_LCT,
;  1634     1623  3                 NMA$C_PCLI_LOO, NMA$C_PCCI_LOO,
;  1635     1624  3                 NMA$C_PCLI_ADJ, NMA$C_PCCI_ADJ,
;  1636     1625  3                 NMA$C_PCLI_BLO, NMA$C_PCCI_BLO,
;  1637     1626  3                 NMA$C_PCLI_COS, NMA$C_PCCI_COS,
;  1638     1627  3                 NMA$C_PCLI_LTY, NMA$C_PCCI_TYP,
;  1639     1628  3                 NMA$C_PCLI_TRI, NMA$C_PCCI_TRI,
;  1640     1629  3                 -1, -1): VECTOR [,WORD,SIGNED];
;  1641     1630  3
;  1642     1631  3         INCRU I FROM 0 BY 2             ! For each entry in conversion table,
;  1643     1632  3         DO
;  1644     1633  4             BEGIN
```

```
;  1645          1634  4              IF .LINE_TO_CIRCUIT [.I] EQL -1     ! If not found in table,
;  1646          1635  4              THEN
;  1647          1636  4                  RETURN FALSE;                   ! Then ignore it
;  1648          1637  4
;  1649          1638  4              IF .PARM [NMA$V_PTY_TYP]            ! If found in table,
;  1650          1639  4                  EQL .LINE_TO_CIRCUIT [.I]
;  1651          1640  4              THEN
;  1652          1641  5                  BEGIN
;  1653          1642  5                  PARM [NMA$V_PTY_TYP] = .LINE_TO_CIRCUIT [.I+1]; ! Convert ID
;  1654          1643  5                  RETURN TRUE;                    ! Format the converted parameter
;  1655          1644  4                  END;
;  1656          1645  3              END;
;  1657          1646  2          END;
;  1658          1647  2
;  1659          1648  2      [NMA$C_ENT_LIN]:                            ! Map V2 lines -> V3 lines
;  1660          1649  2          BEGIN
;  1661          1650  2          BIND LINE_TO_LINE = UPLIT WORD(
;  1662          1651  2                  NMA$C_PCLI_STA, NMA$C_PCLI_STA,
;  1663          1652  2                  NMA$C_PCLI_SER, NMA$C_PCLI_SER,
;  1664          1653  2                  NMA$C_PCLI_LCT, NMA$C_PCLI_LCT,
;  1665          1654  2                  NMA$C_PCLI_CON, NMA$C_PCLI_CON,
;  1666          1655  2                  NMA$C_PCLI_DUP, NMA$C_PCLI_DUP,
;  1667          1656  2                  NMA$C_PCLI_LTY, NMA$C_PCLI_PRO,
;  1668          1657  2                  NMA$C_PCLI_STI, NMA$C_PCLI_STI,
;  1669          1658  2                  NMA$C_PCLI_NTI, NMA$C_PCLI_RTT,
;  1670          1659  2                  2700, NMA$C_PCLI_BFN,
;  1671          1660  2                  -1, -1): VECTOR [,WORD,SIGNED];
;  1672          1661  2
;  1673          1662  2          INCRU I FROM 0 BY 2                     ! For each entry in conversion table,
;  1674          1663  3          DO
;  1675          1664  4              BEGIN
;  1676          1665  4              IF .LINE_TO_LINE [.I] EQL -1        ! If not found in table,
;  1677          1666  4              THEN
;  1678          1667  4                  RETURN FALSE;                   ! Then ignore it
;  1679          1668  4
;  1680          1669  4              IF .PARM [NMA$V_PTY_TYP]            ! If found in table,
;  1681          1670  4                  EQL .LINE_TO_LINE [.I]
;  1682          1671  4              THEN
;  1683          1672  5                  BEGIN
;  1684          1673  5                  PARM [NMA$V_PTY_TYP] = .LINE_TO_LINE [.I+1]; ! Convert ID
;  1685          1674  5                  RETURN TRUE;                    ! Format the converted parameter
;  1686          1675  4                  END;
;  1687          1676  3              END;
;  1688          1677  2          END;
;  1689          1678  2      TES;
;  1690          1679  2
;  1691          1680  2  RETURN FALSE;                                   ! If we get here, ignore parameter
;  1692          1681  2
;  1693          1682  1  END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                                00  00  02  00930 P.ADY:  .BYTE   2, 0, 0                                                ;
                                                            00933          .BLKB   1
0320  0320  0190  0190  006E  006E  0064  0064  0000  0000  00934 P.ADZ:  .WORD   0, 0, 100, 100, 110, 110, 400, 400, 800, -  ;
```

```
FFFF  FFFF  0474  0474  0458  0458  0384  0384  032A  032A  00948                    800, 810, 810, 900, 900, 1112, 1112, -      ;
                                                                                   1140, 1140, -1, -1                          ;
0457  0457  0456  0456  006E  006E  0064  0064  0000  0000  0095C P.AEA:  .WORD    0, 0, 100, 100, 110, 110, 1110, 1110, -     ;
FFFF  FFFF  0451  0A8C  0461  0461  0460  0460  0458  0458  00970                   1111, 1111, 1112, 1112, 1120, 1120, 1121, - ;
                                                                                   1121, 2700, 1105, -1, -1                    ;

                                                                LINE_TO_CIRCUIT=    P.ADZ
                                                                LINE_TO_LINE=       P.AEA
                                                                        .EXTRN  NCP$GL_EXELCB

                                                                        .PSECT  $CODE$,NOWRT,2

                                                        001C 00000 V2_SHOW_COMPAT:
                                                                        .WORD   Save R2,R3,R4                               ; 1578
                                                    54 00000000'  00    9E 00002        MOVAB   P.ADY, R4                   ;
                                                    50 00000000G  00    D0 00009        MOVL    NCP$GL_EXELCB, R0           ; 1601
                                64          06      A0                  03 29 00010     CMPC3   #3, 6(R0), P.ADY            ;
                                                                        62 12 00015     BNEQ    6$                          ;
                01 00000000G  00                    03                  00 ED 00017     CMPZV   #0, #3, NCP$GL_OPTION, #1   ; 1606
                                                                        57 12 00020     BNEQ    6$                          ;
                                                    52          04      AC D0 00022     MOVL    PARM, R2                    ; 1610
                                                                        62 B5 00026     TSTW    (R2)                        ;
                                                                        4F 19 00028     BLSS    6$                          ;
                                                    50 00000000G  00    D0 0002A        MOVL    NCP$GL_ENTITY, R0           ; 1614
                                                    03                  50 D1 00031     CMPL    R0, #3                      ; 1617
                                                                        20 12 00034     BNEQ    3$                          ;
                                                                        50 D4 00036     CLRL    I                           ; 1631
                                                    51          04  A440  32 00038 1$:   CVTWL   LINE_TO_CIRCUIT[I], R1      ; 1634
                                        FFFF        8F                  51 B1 0003D     CMPW    R1, #-1                     ;
                                                                        3E 13 00042     BEQL    8$                          ;
                51              62                  0F                  00 ED 00044     CMPZV   #0, #15, (R2), R1           ; 1639
                                                                        06 12 00049     BNEQ    2$                          ;
                                                                06  A440  3F 0004B      PUSHAW  LINE_TO_CIRCUIT+2[I]        ; 1642
                                                                    23  11 0004F       BRB     5$                          ;
                                                    50                  02 C0 00051 2$: ADDL2   #2, I                       ; 1631
                                                                        E2 11 00054     BRB     1$                          ;
                                                    01                  50 D1 00056 3$: CMPL    R0, #1                      ; 1648
                                                                        27 12 00059     BNEQ    8$                          ;
                                                                        50 D4 0005B     CLRL    I                           ; 1662
                                                    51          2C  A440  32 0005D 4$:   CVTWL   LINE_TO_LINE[I], R1        ; 1665
                                        FFFF        8F                  51 B1 00062     CMPW    R1, #-1                     ;
                                                                        19 13 00067     BEQL    8$                          ;
                51              62                  0F                  00 ED 00069     CMPZV   #0, #15, (R2), R1           ; 1670
                                                                        0D 12 0006E     BNEQ    7$                          ;
                                                                2E  A440  3F 00070      PUSHAW  LINE_TO_LINE+2[I]           ; 1673
                62              0F                  00                  9E F0 00074 5$: INSV    @(SP)+, #0, #15, (R2)       ;
                                                    50                  01 D0 00079 6$: MOVL    #1, R0                      ; 1674
                                                                            04 0007C    RET                                 ;
                                                    50                  02 C0 0007D 7$: ADDL2   #2, I                       ; 1662
                                                                        DB 11 00080     BRB     4$                          ;
                                                                        50 D4 00082 8$: CLRL    R0                          ; 1682
                                                                            04 00084    RET                                 ;

; Routine Size:  133 bytes,    Routine Base:  $CODE$ + 090A
```

```
;  1695          1683  1 %SBTTL  'NCP$FORMATPARM  Return Text Format for Parameter'
;  1696          1684  1 GLOBAL ROUTINE NCP$FORMATPARM (ENTY, PDID, NAMQ, DATAQ, BDSC, RLEN, RPTR)
;  1697          1685  1                 :NOVALUE =
;  1698          1686  1
;  1699          1687  1 !++
;  1700          1688  1 ! FUNCTIONAL DESCRIPTION:
;  1701          1689  1 !
;  1702          1690  1 !       This routine parses a parameter in binary format and returns
;  1703          1691  1 !       the text of a description of the parameter in a buffer.
;  1704          1692  1 !
;  1705          1693  1 !       The parameters are the entity type of the parameter, and a pointer
;  1706          1694  1 !       to its binary representation.  Flags are passed indicating whether
;  1707          1695  1 !       to include the parameter name and the parameter value.
;  1708          1696  1 !
;  1709          1697  1 !       The returns are the length of the text data returned in the buffer,
;  1710          1698  1 !       and a pointer beyond the parameter as it was parsed.
;  1711          1699  1 !
;  1712          1700  1 !
;  1713          1701  1 ! FORMAL PARAMETERS:
;  1714          1702  1 !
;  1715          1703  1 !       ENTY            Entity number corresponding to the parameter
;  1716          1704  1 !                       If negative, then system-specific entity
;  1717          1705  1 !       PDID            Address of the parameter ID word
;  1718          1706  1 !       NAMQ            True for include name in text output
;  1719          1707  1 !       DATAQ           True for include value in text output
;  1720          1708  1 !                       Also pass over data and return pointer beyond it
;  1721          1709  1 !       BDSC            Address of buffer descriptor for text
;  1722          1710  1 !       RLEN            Address for returned length of text
;  1723          1711  1 !       RPTR            Address for returned address beyond parameter
;  1724          1712  1 !
;  1725          1713  1 ! COMPLETION CODES:
;  1726          1714  1 !
;  1727          1715  1 !       Error is signaled by a called routine if it is not safe to
;  1728          1716  1 !       continue parsing parameters.
;  1729          1717  1 !--
```



```
; R
```

```
;  1731          1718  1
;  1732          1719  2      BEGIN
;  1733          1720  2
;  1734          1721  2      LOCAL
;  1735          1722  2          STATUS,                         ! General status value
;  1736          1723  2          TBL,                            ! Address of table to search
;  1737          1724  2          TYP,                            ! Type code from table lookup
;  1738          1725  2          LST,                            ! Keyword list from table lookup
;  1739          1726  2          TXT,                            ! Parameter name from table lookup
;  1740          1727  2          ID : BBLOCK [LONG],             ! Parameter id code
;  1741          1728  2          DTY,                            ! Data type code
;  1742          1729  2          PTR                             ! Pointer to parameter data
;  1743          1730  2          ;                   !
;  1744          1731  2
;  1745          1732  2      NCP$FAOSET ();                      ! Setup fao parameters
;  1746          1733  2
;  1747          1734  2      PTR = .PDID;                        ! Set pointer to parameter
;  1748          1735  2      ID = .(.PTR) <0, 16, 0>;            ! The parameter id
;  1749          1736  2      PTR = .PTR + 2;                     ! And advance over it
;  1750          1737  2
;  1751          1738  2  !
;  1752          1739  2  !       There is no real choice about getting the name or data for
;  1753          1740  2  !       counters.  We never print counters in a list with headers so its
;  1754          1741  2  !       not necessary.  You always get the name and the data for a counter.
;  1755          1742  2  !
;  1756          1743  2
;  1757          1744  2      IF .ID [NMA$V_CNT_COU]              ! If its a counter
;  1758          1745  2      THEN
;  1759          1746  2          NCP$PARSECOUNTER (.ID, PTR)     ! This routine does it all
;  1760          1747  2      ELSE
;  1761          1748  3          BEGIN
;  1762          1749  3
;  1763          1750  3  !
;  1764          1751  3  !       For parameters, we select a table of data and then act based
;  1765          1752  3  !       on the data type code
;  1766          1753  3  !
;  1767          1754  3
;  1768          1755  3          TBL =                           ! Select the parameter table
;  1769          1756  4              BEGIN                       ! Based on the entity passed
;  1770          1757  4              SELECTONE .ENTY OF          ! Entity code
;  1771          1758  4              SET
;  1772          1759  4
;  1773          1760  4              [NMA$C_ENT_NOD] : NCP$GA_PRM_NOD ;
;  1774          1761  4              [NMA$C_ENT_CIR] : NCP$GA_PRM_CIR ;
;  1775          1762  4              [NMA$C_ENT_LIN] : NCP$GA_PRM_LIN ;
;  1776          1763  4              [NMA$C_ENT_LOG] : NCP$GA_PRM_LOG ;
;  1777          1764  4              [NMA$C_ENT_MOD] :
;  1778          1765  5                  BEGIN
;  1779          1766  5                  SELECTONE .NCP$GL_MODTYP OF
;  1780          1767  5                  SET
;  1781          1768  5
;  1782          1769  5                  [NCP$C_ENT_MODCNF] : NCP$GA_PRM_MODCNF; ! module Configurator
;  1783          1770  5  !!              [NCP$C_ENT_MODCNS] : NCP$GA_PRM_MODCNS; ! module Console
;  1784          1771  5  !!              [NCP$C_ENT_MODLOA] : NCP$GA_PRM_MODLOA; ! module Loader
;  1785          1772  5  !!              [NCP$C_ENT_MODLOO] : NCP$GA_PRM_MODLOO; ! module Looper
;  1786          1773  5                  [NCP$C_ENT_MODACC] : NCP$GA_PRM_MODACC; ! module X25-Access
;  1787          1774  5                  [NCP$C_ENT_MODPRO] : NCP$GA_PRM_MODPRO; ! module X25-Protocol
```

```
;  1788          1775  5                  [NCP$C_ENT_MODSER] : NCP$GA_PRM_MODSER; ! module X25-Server
;  1789          1776  5                  [NCP$C_ENT_MODTRC] : NCP$GA_PRM_MODTRC; ! module X25-Trace
;  1790          1777  5                  [NCP$C_ENT_MOD29S] : NCP$GA_PRM_MODSER; ! module X29-Server
;  1791          1778  5                  [OTHERWISE] : SIGNAL_STOP
;  1792          1779  5                          (NCP$_BADMSG, 1, ASCIC ('invalid component code') );
;  1793          1780  5
;  1794          1781  5                  TES
;  1795          1782  4                  END;
;  1796          1783  4              [NMA$C_ENT_ARE] : NCP$GA_PRM_ARE ;
;  1797          1784  4              [-NMA$C_ENT_OBJ] : NCP$GA_PRM_OBJ ;
;  1798          1785  4              [-NMA$C_ENT_LNK] : NCP$GA_PRM_LNK ;
;  1799          1786  4              [OTHERWISE] :
;  1800          1787  4                  SIGNAL_STOP (NCP$_BADMSG, 1,
;  1801          1788  4                          ASCIC ('invalid component code') )
;  1802          1789  4                  ;
;  1803          1790  4
;  1804          1791  4              TES
;  1805          1792  4              END
;  1806          1793  3          ;
```

```
;  1808     1794  3
;  1809     1795  !
;  1810     1796  !       Search for the parameter to get its type and
;  1811     1797  !       a list of keywords if any and the text of its name
;  1812     1798  !
;  1813     1799
;  1814     1800          STATUS =
;  1815     1801              NCP$PTBSEARCH
;  1816     1802                  (
;  1817     1803                  .TBL,
;  1818     1804                  .ID [NMA$V_PTY_TYP],
;  1819     1805                  TYP,
;  1820     1806                  LST,
;  1821     1807                  TXT
;  1822     1808                  )
;  1823     1809          ;
;  1824     1810          IF NOT .STATUS                  ! Not known?
;  1825     1811          THEN
;  1826     1812              TYP = PBK$K_END             ! Set for 'we don't know'
;  1827     1813          ;
;  1828     1814
;  1829     1815          IF .DATAQ AND .NAMQ             ! If data wanted too, use a fixed
;  1830     1816          THEN                            ! length field for the name
;  1831     1817              ADDSTR ('!24<')
;  1832     1818          ;
;  1833     1819
;  1834     1820          IF .NAMQ                        ! If the user wants the name
;  1835     1821          THEN                            ! Of the parameter
;  1836     1822  4           BEGIN
;  1837     1823  4           IF NOT .STATUS              ! We don't know its name, so
;  1838     1824  4           THEN                        ! Concoct a name based on its id
;  1839     1825  5               BEGIN
;  1840     1826  5               ADDSTR ('Parameter #!UW');
;  1841     1827  5               ADDFAO (.ID [NMA$V_PTY_TYP])
;  1842     1828  5               END
;  1843     1829  4           ELSE
;  1844     1830  5               BEGIN                   ! Use the params given name
;  1845     1831  5               ADDSTR ('!AC');
;  1846     1832  5               ADDFAO (.TXT)
;  1847     1833  5               END
;  1848     1834  4           END
;  1849     1835  3       ;
;  1850     1836  3
;  1851     1837  3       IF .DATAQ AND .NAMQ             ! If the name and the data
;  1852     1838  3       THEN
;  1853     1839  3           ADDSTR ('!> = ')            ! Set the data off in some way
;  1854     1840  3                                       ! and close the fixed size field
;  1855     1841  3       ;
;  1856     1842  3
;  1857     1843  3       IF .DATAQ                       ! If the user wants the data too
;  1858     1844  3       THEN
;  1859     1845  4           BEGIN
;  1860     1846  4           DTY = .(.PTR) <0, 8, 0>;    ! Obtain the data type code
;  1861     1847  4           PTR = .PTR + 1;             ! And skip over it
;  1862     1848  4
;  1863     1849  4 !
;  1864     1850  4 !       As you might guess, this routine does all the real work of parsing
```

```
;  1865          1851  4 !       a parameter. It builds the text of the data and the fao list.
;  1866          1852  4 !       This routine signals if it cannot interpret the param in any way.
;  1867          1853  4 !
;  1868          1854  4
;  1869          1855  4             NCP$PARSEPARM (.DTY, PTR, .TYP, .LST)
;  1870          1856  4
;  1871          1857  4             END
;  1872          1858  3         END
;  1873          1859  2     ;
;  1874          1860  2
;  1875          1861  2 !
;  1876          1862  2 !       Now Make the text to return to the caller
;  1877          1863  2 !
;  1878          1864  2
;  1879        P 1865  2     $FAOL                           ! The magic text maker
;  1880        P 1866  2         (
;  1881        P 1867  2         CTRSTR = CTRDSC,            ! control string
;  1882        P 1868  2         OUTLEN = .RLEN,             ! Return the length here
;  1883        P 1869  2         OUTBUF = .BDSC,             ! Put the text here
;  1884        P 1870  2         PRMLST = FAOLST             ! Use this arg list
;  1885          1871  2         );
;  1886          1872  2
;  1887          1873  2     .RPTR = .PTR;                   ! Return the pointer past data
;  1888          1874  2
;  1889          1875  2     RETURN
;  1890          1876  2
;  1891          1877  1     END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

6E 6F 70 6D 6F 63 20 64 69 6C 61 76 6E 69 16  00984 P.AEB:      .ASCII  <22>\invalid component code\                    ;
                     65 64 6F 63 20 74 6E 65  00993
6E 6F 70 6D 6F 63 20 64 69 6C 61 76 6E 69 16  0099B P.AEC:      .ASCII  <22>\invalid component code\                    ;
                     65 64 6F 63 20 74 6E 65  009AA
                              3C 34 32 21 04  009B2 P.AED:      .ASCII  <4>\!24<\                                       ;
57 55 21 23 20 72 65 74 65 6D 61 72 61 50 0E  009B7 P.AEE:      .ASCII  <14>\Parameter #!UW\                            ;
                                 43 41 21 03  009C6 P.AEF:      .ASCII  <3>\!AC\                                        ;
                           20 3D 20 3E 21 05  009CA P.AEG:      .ASCII  <5>\!> = \                                      ;

                                                                .EXTRN  SYS$FAOL

                                                                .PSECT  $CODE$,NOWRT,2

                                          007C 00000            .ENTRY  NCP$FORMATPARM, Save R2,R3,R4,R5,R6             ; 1684
                              56 00000000V 00  9E 00002         MOVAB   NCP$ADDSTR, R6                                  ;
                              55 00000000' 00  9E 00009         MOVAB   NCP$GQ_CTRDSC, R5                               ;
                              54 00000000' 00  9E 00010         MOVAB   P.AEB, R4                                       ;
                                 5E        10  C2 00017         SUBL2   #16, SP                                         ;
                  00000000V      00        00  FB 0001A         CALLS   #0, NCP$FAOSET                                  ; 1732
                         0C      AE     08 AC  D0 00021         MOVL    PDID, PTR                                       ; 1734
                                 53     0C BE  3C 00026         MOVZWL  @PTR, ID                                        ; 1735
                         0C      AE        02  C0 0002A         ADDL2   #2, PTR                                         ; 1736
                                           53  B5 0002E         TSTW    ID                                              ; 1744
                                           0F  18 00030         BGEQ    1$                                              ;
                                        0C AE  9F 00032         PUSHAB  PTR                                             ; 1746
```

```
                                  53  DD 00035       PUSHL   ID                                   :
              00000000V  00       02  FB 00037       CALLS   #2, NCP$PARSECOUNTER                 :
                               015C   31 0003E       BRW     28$                                  :
                         52      04   AC  D0 00041 1$:  MOVL    ENTY, R2                          : 1757
                                  09  12 00045       BNEQ    2$                                   : 1760
                         50 00000000G 00  9E 00047   MOVAB   NCP$GA_PRM_NOD, TBL                  :
                                  7F  11 0004E       BRB     11$                                  :
                         03       52  D1 00050 2$:   CMPL    R2, #3                               : 1761
                                  09  12 00053       BNEQ    3$                                   :
                         50 00000000G 00  9E 00055   MOVAB   NCP$GA_PRM_CIR, TBL                  :
                                  71  11 0005C       BRB     11$                                  :
                         01       52  D1 0005E 3$:   CMPL    R2, #1                               : 1762
                                  09  12 00061       BNEQ    4$                                   :
                         50 00000000G 00  9E 00063   MOVAB   NCP$GA_PRM_LIN, TBL                  :
                                  75  11 0006A       BRB     14$                                  :
                         02       52  D1 0006C 4$:   CMPL    R2, #2                               : 1763
                                  09  12 0006F       BNEQ    5$                                   :
                         50 00000000G 00  9E 00071   MOVAB   NCP$GA_PRM_LOG, TBL                  :
                                  79  11 00078       BRB     16$                                  :
                         04       52  D1 0007A 5$:   CMPL    R2, #4                               : 1764
                                  56  12 0007D       BNEQ    13$                                  :
                         51 00000000G 00  D0 0007F   MOVL    NCP$GL_MODTYP, R1                    : 1766
                         01       51  D1 00086       CMPL    R1, #1                               : 1769
                                  09  12 00089       BNEQ    6$                                   :
                         50 00000000G 00  9E 0008B   MOVAB   NCP$GA_PRM_MODCNF, TBL               :
                                  71  11 00092       BRB     18$                                  :
                         05       51  D1 00094 6$:   CMPL    R1, #5                               : 1773
                                  09  12 00097       BNEQ    7$                                   :
                         50 00000000G 00  9E 00099   MOVAB   NCP$GA_PRM_MODACC, TBL               :
                                  77  11 000A0       BRB     21$                                  :
                         06       51  D1 000A2 7$:   CMPL    R1, #6                               : 1774
                                  09  12 000A5       BNEQ    8$                                   :
                         50 00000000G 00  9E 000A7   MOVAB   NCP$GA_PRM_MODPRO, TBL               :
                                  69  11 000AE       BRB     21$                                  :
                         07       51  D1 000B0 8$:   CMPL    R1, #7                               : 1775
                                  13  13 000B3       BEQL    10$                                  :
                         08       51  D1 000B5       CMPL    R1, #8                               : 1776
                                  09  12 000B8       BNEQ    9$                                   :
                         50 00000000G 00  9E 000BA   MOVAB   NCP$GA_PRM_MODTRC, TBL               :
                                  56  11 000C1       BRB     21$                                  :
                         09       51  D1 000C3 9$:   CMPL    R1, #9                               : 1777
                                  09  12 000C6       BNEQ    12$                                  :
                         50 00000000G 00  9E 000C8 10$:  MOVAB   NCP$GA_PRM_MODSER, TBL           :
                                  48  11 000CF 11$:  BRB     21$                                  :
                                  54  DD 000D1 12$:  PUSHL   R4                                   : 1779
                                  35  11 000D3       BRB     20$                                  :
                         05       52  D1 000D5 13$:  CMPL    R2, #5                               : 1783
                                  09  12 000D8       BNEQ    15$                                  :
                         50 00000000G 00  9E 000DA   MOVAB   NCP$GA_PRM_ARE, TBL                  :
                                  36  11 000E1 14$:  BRB     21$                                  :
              FFFFFFFC   8F       52  D1 000E3 15$:  CMPL    R2, #-4                              : 1784
                                  09  12 000EA       BNEQ    17$                                  :
                         50 00000000G 00  9E 000EC   MOVAB   NCP$GA_PRM_OBJ, TBL                  :
                                  24  11 000F3 16$:  BRB     21$                                  :
              FFFFFFF9   8F       52  D1 000F5 17$:  CMPL    R2, #-7                              : 1785
                                  09  12 000FC       BNEQ    19$                                  :
                         50 00000000G 00  9E 000FE   MOVAB   NCP$GA_PRM_LNK, TBL                  :
```

```
                                              12  11 00105 18$:    BRB     21$                               ;
                                        17    A4  9F 00107 19$:    PUSHAB  P.AEC                             ; 1788
                                              01  DD 0010A 20$:    PUSHL   #1                                ; 1787
                               00000000G      8F  DD 0010C         PUSHL   #NCP$_BADMSG                      ;
                   00000000G  00              03  FB 00112         CALLS   #3, LIB$STOP                      ;
                                              5E  DD 00119 21$:    PUSHL   SP                                ; 1802
                                        08    AE  9F 0011B         PUSHAB  LST                               ;
                                        10    AE  9F 0011E         PUSHAB  TYP                               ;
   7E          53             0F              00  EF 00121         EXTZV   #0, #15, ID, -(SP)                ; 1804
                                              50  DD 00126         PUSHL   TBL                               ; 1803
                   00000000V  00              05  FB 00128         CALLS   #5, NCP$PTBSEARCH                 ;
                              52              50  D0 0012F         MOVL    R0, STATUS                        ;
                              04              52  E8 00132         BLBS    STATUS, 22$                       ; 1810
                        08    AE              17  D0 00135         MOVL    #23, TYP                          ; 1812
                              0C        10    AC  E9 00139 22$:    BLBC    DATAQ, 23$                        ; 1815
                              2F        0C    AC  E9 0013D         BLBC    NAMQ, 26$                         ;
                                              55  DD 00141         PUSHL   R5                                ; 1817
                                        2E    A4  9F 00143         PUSHAB  P.AED                             ;
                              66              02  FB 00146         CALLS   #2, NCP$ADDSTR                    ;
                              23        0C    AC  E9 00149 23$:    BLBC    NAMQ, 26$                         ; 1820
                              0F              52  E8 0014D         BLBS    STATUS, 24$                       ; 1823
                                              55  DD 00150         PUSHL   R5                                ; 1826
                                        33    A4  9F 00152         PUSHAB  P.AEE                             ;
                              66              02  FB 00155         CALLS   #2, NCP$ADDSTR                    ;
   7E          53             0F              00  EF 00158         EXTZV   #0, #15, ID, -(SP)                ; 1827
                                              0A  11 0015D         BRB     25$                               ;
                                              55  DD 0015F 24$:    PUSHL   R5                                ; 1831
                                        42    A4  9F 00161         PUSHAB  P.AEF                             ;
                              66              02  FB 00164         CALLS   #2, NCP$ADDSTR                    ;
                                              6E  DD 00167         PUSHL   TXT                               ; 1832
                   00000000V  00              01  FB 00169 25$:    CALLS   #1, NCP$ADDFAO                    ;
                              29        10    AC  E9 00170 26$:    BLBC    DATAQ, 28$                        ; 1837
                              08        0C    AC  E9 00174         BLBC    NAMQ, 27$                         ;
                                              55  DD 00178         PUSHL   R5                                ; 1839
                                        46    A4  9F 0017A         PUSHAB  P.AEG                             ;
                              66              02  FB 0017D         CALLS   #2, NCP$ADDSTR                    ;
                              19        10    AC  E9 00180 27$:    BLBC    DATAQ, 28$                        ; 1843
                              50        0C    BE  9A 00184         MOVZBL  @PTR, DTY                         ; 1846
                                        0C    AE  D6 00188         INCL    PTR                               ; 1847
                                        04    AE  DD 0018B         PUSHL   LST                               ; 1855
                                        0C    AE  DD 0018E         PUSHL   TYP                               ;
                                        14    AE  9F 00191         PUSHAB  PTR                               ;
                                              50  DD 00194         PUSHL   DTY                               ;
                   00000000V  00              04  FB 00196         CALLS   #4, NCP$PARSEPARM                 ;
                               00000000'      00  9F 0019D 28$:    PUSHAB  FAOLST                            ; 1871
                                        14    AC  DD 001A3         PUSHL   BDSC                              ;
                                        18    AC  DD 001A6         PUSHL   RLEN                              ;
                                              55  DD 001A9         PUSHL   R5                                ;
                   00000000G  00              04  FB 001AB         CALLS   #4, SYS$FAOL                      ;
                          1C  BC        0C    AE  D0 001B2         MOVL    PTR, @RPTR                        ; 1873
                                                  04 001B7         RET                                       ; 1877

; Routine Size:  440 bytes,    Routine Base:  $CODE$ + 098F
```

```
;  1893     1878  1 %SBTTL  'NCP$PARSECOUNTER  Parse a Counter'
;  1894     1879  1 ROUTINE NCP$PARSECOUNTER (ID, CPTR) :NOVALUE =  !
;  1895     1880  1
;  1896     1881  1 !++
;  1897     1882  1 ! FUNCTIONAL DESCRIPTION:
;  1898     1883  1 !
;  1899     1884  1 !       This routine parses a counter and produces the necessary fao
;  1900     1885  1 !       control string and list entries to format the counter into text.
;  1901     1886  1 !
;  1902     1887  1 ! FORMAL PARAMETERS:
;  1903     1888  1 !
;  1904     1889  1 !       ID              Value of the ID code for the counter
;  1905     1890  1 !       CPTR            Pointer to the counter data block,
;  1906     1891  1 !                       returned as a pointer beyond the block
;  1907     1892  1 !
;  1908     1893  1 ! IMPLICIT INPUTS:
;  1909     1894  1 !
;  1910     1895  1 !       NONE
;  1911     1896  1 !
;  1912     1897  1 ! IMPLICIT OUTPUTS:
;  1913     1898  1 !
;  1914     1899  1 !       NONE
;  1915     1900  1 !
;  1916     1901  1 ! ROUTINE VALUE:
;  1917     1902  1 ! COMPLETION CODES:
;  1918     1903  1 !
;  1919     1904  1 !       Data inconsistancies are signalled as errors
;  1920     1905  1 !
;  1921     1906  1 ! SIDE EFFECTS:
;  1922     1907  1 !
;  1923     1908  1 !       NONE
;  1924     1909  1 !
;  1925     1910  1 !--
;  1926     1911  1
;  1927     1912  2     BEGIN
;  1928     1913  2
;  1929     1914  2     MAP
;  1930     1915  2         ID : BBLOCK [4]                 ! Id has fields in it
;  1931     1916  2         ;
;  1932     1917  2
;  1933     1918  2     LOCAL
;  1934     1919  2         PTR,                            ! Pointer to counter data
;  1935     1920  2         COUID,                          ! Id of the counter
;  1936     1921  2         COUVAL,                         ! Value of the counter
;  1937     1922  2         BITMAP,                         ! Things that got counted
;  1938     1923  2         COUTYP,                         ! Data type of counter (ignored)
;  1939     1924  2         COULST,                         ! List of counted strings
;  1940     1925  2         COUTXT
;  1941     1926  2         ;               !
;  1942     1927  2
;  1943     1928  2
;  1944     1929  2     PTR = ..CPTR;                       ! Set the local pointer to the data
;  1945     1930  2
;  1946     1931  2     COUID = .ID [NMA$V_CNT_TYP];        ! Counter id
;  1947     1932  2     IF .ID [NMA$V_CNT_MAP]              ! Look at the bit map bit
;  1948     1933  2     THEN
;  1949     1934  3         BEGIN                           ! We have a bitmap
```

```
;  1950     1935  3          BITMAP = .(.PTR) <0, 16, 0>;    ! Eat it up
;  1951     1936  3          PTR = .PTR + 2                  ! And skip it
;  1952     1937  3          END
;  1953     1938  2      ELSE
;  1954     1939  2          BITMAP = 0                      ! No bits in map
;  1955     1940  2      ;
;  1956     1941  2
;  1957     1942  2      SELECTONEU .ID [NMA$V_CNT_WID] OF   ! Obtain the counter value by width
;  1958     1943  2      SET
;  1959     1944  2
;  1960     1945  2      [0] :                               ! NML blew it, with a reserved code
;  1961     1946  3          (
;  1962     1947  3          SIGNAL_STOP (NCP$_BADMSG, 1, ASCIC ('reserved counter width') )
;  1963     1948  3          );
;  1964     1949  2
;  1965     1950  2      [1] :                               ! Its a byte
;  1966     1951  3          (
;  1967     1952  3          COUVAL = .(.PTR) <0, 8, 0>;
;  1968     1953  3          IF .COUVAL <0, 8, 1> EQL -1     ! Retain -1 to mean overflow
;  1969     1954  3          THEN
;  1970     1955  4              BEGIN
;  1971     1956  4              COUVAL = -1;
;  1972     1957  4              ADDSTR ('!14<        >254!>')
;  1973     1958  4              END
;  1974     1959  3          ;
;  1975     1960  3          PTR = .PTR + 1
;  1976     1961  3          );
;  1977     1962  2
;  1978     1963  2      [2] :                               ! Its a word
;  1979     1964  3          (
;  1980     1965  3          COUVAL = .(.PTR) <0, 16, 0>;
;  1981     1966  3          IF .COUVAL <0, 16, 1> EQL -1
;  1982     1967  3          THEN
;  1983     1968  4              BEGIN
;  1984     1969  4              COUVAL = -1;
;  1985     1970  4              ADDSTR ('!14<       >65534!>')
;  1986     1971  4              END
;  1987     1972  3          ;
;  1988     1973  3          PTR = .PTR + 2
;  1989     1974  2          );
;  1990     1975  2
;  1991     1976  2      [3] :                               ! Its a longword counter
;  1992     1977  3          (
;  1993     1978  3          COUVAL = .(.PTR) <0, 32, 0>;
;  1994     1979  3          IF .COUVAL EQL -1
;  1995     1980  3          THEN
;  1996     1981  4              BEGIN
;  1997     1982  4              ADDSTR ('!14< >4294967294!>')
;  1998     1983  4              END
;  1999     1984  3          ;
;  2000     1985  3          PTR = .PTR + 4
;  2001     1986  2          );
;  2002     1987  2
;  2003     1988  2      TES
;  2004     1989  2      ;
;  2005     1990  2
;  2006     1991  2      IF .COUVAL EQL -1                   ! Is it really overflow?
```

```
;  2007          1992  2      THEN
;  2008          1993  2          0                                   ! Already taken care of
;  2009          1994  2      ELSE
;  2010          1995  3          BEGIN
;  2011          1996  3          ADDSTR ('!14<!12UL!>');             ! The counter value here
;  2012          1997  3          ADDFAO (.COUVAL)
;  2013          1998  3          END
;  2014          1999  2      ;
;  2015          2000  2
;  2016          2001  2      IF
;  2017          2002  2          NCP$PTBSEARCH                       ! Look up the counter name, etc.
;  2018          2003  2              (
;  2019          2004  3                  BEGIN
;  2020          2005  3                  SELECTONE .NCP$GL_ENTITY OF     ! The type of entity
;  2021          2006  3                      SET
;  2022          2007  3                      [NMA$C_ENT_NOD] : NCP$GA_PRM_NODCNTR;
;  2023          2008  3                      [NMA$C_ENT_LIN] : NCP$GA_PRM_LINCNTR;
;  2024          2009  3                      [NMA$C_ENT_CIR] : NCP$GA_PRM_CIRCNTR;
;  2025          2010  3                      [NMA$C_ENT_MOD] :
;  2026          2011  4                          BEGIN
;  2027          2012  4                          SELECTONE .NCP$GL_MODTYP OF
;  2028          2013  4                              SET
;  2029          2014  4                              [NCP$C_ENT_MODPRO] : NCP$GA_PRM_MPRCNTR;
;  2030          2015  4                              [NCP$C_ENT_MODSER] : NCP$GA_PRM_MSECNTR;
;  2031          2016  4                              [NCP$C_ENT_MOD29S] : NCP$GA_PRM_MSECNTR;
;  2032          2017  4                              [OTHERWISE] :
;  2033          2018  4                                  SIGNAL_STOP (NCP$_BADMSG, 1,
;  2034          2019  4                                      ASCIC ('invalid component code') );
;  2035          2020  4                              TES
;  2036          2021  3                          END;
;  2037          2022  3                      [OTHERWISE] :
;  2038          2023  3                          SIGNAL_STOP (NCP$_BADMSG, 1,
;  2039          2024  3                              ASCIC ('invalid component code') );
;  2040          2025  3                      TES
;  2041          2026  2                  END,
;  2042          2027  2              .COUID,
;  2043          2028  2              COUTYP,                             ! Type code
;  2044          2029  2              COULST,                             ! List of bitmap items
;  2045          2030  2              COUTXT                              ! Text of name of counter
;  2046          2031  2              )
;  2047          2032  2      THEN
;  2048          2033  3          BEGIN
;  2049          2034  3          ADDSTR ('!AC');                     ! We found it return the name
;  2050          2035  3          ADDFAO (.COUTXT)
;  2051          2036  3          END
;  2052          2037  2      ELSE
;  2053          2038  3          BEGIN
;  2054          2039  3          ADDSTR ('Counter #!UW');            ! Generic name for we don't know
;  2055          2040  3          ADDFAO (.COUID);
;  2056          2041  3          COULST = 0                          ! We have no list either
;  2057          2042  3          END
;  2058          2043  2      ;
;  2059          2044  2
;  2060          2045  2      IF .BITMAP NEQ 0                        ! Items in the bitmap
;  2061          2046  2      THEN
;  2062          2047  3          BEGIN
;  2063          2048  3          ADDSTR (', including:');            ! A title for them
```

```
;  2064          2049  3          INCRU NBIT FROM 0 TO 15          ! Scan all the bits
;  2065          2050  3          DO
;  2066          2051  4              BEGIN
;  2067          2052  4              IF .BITMAP <.NBIT, 1, 0>     ! If its here
;  2068          2053  4              THEN
;  2069          2054  5                  BEGIN
;  2070          2055  5                  IF .COULST EQL 0         ! Is there a list of bits?
;  2071          2056  5                  THEN
;  2072          2057  6                      BEGIN                ! No use some standard title
;  2073          2058  6                      ADDSTR ('!/!16* Qualifier #!UB');
;  2074          2059  6                      ADDFAO (.NBIT)       ! And report the bit number
;  2075          2060  6                      END
;  2076          2061  5                  ELSE
;  2077          2062  6                      BEGIN                ! Report the qualifier string
;  2078          2063  6                      ADDSTR ('!/!16* !AC');
;  2079          2064  6                      ADDFAO (.COULST);
;  2080          2065  6                      END
;  2081          2066  4                  END;
;  2082          2067  4
;  2083          2068  4                  IF .COULST NEQ 0
;  2084          2069  4                  THEN
;  2085          2070  5                      BEGIN
;  2086          2071  5                      COULST =                         ! Advance to the next qualifier string
;  2087          2072  5                          .(.COULST) <0, 8, 0> + .COULST + 1;
;  2088          2073  5                      IF .(.COULST) <0, 8, 0> EQL 0
;  2089          2074  5                      THEN                             ! No string left, report rest as bits
;  2090          2075  5                          COULST = 0;
;  2091          2076  4                      END;
;  2092          2077  4              END
;  2093          2078  3          END
;  2094          2079  2      ;
;  2095          2080  2
;  2096          2081  2      .CPTR = .PTR;                        ! Return the pointer beyond data
;  2097          2082  2
;  2098          2083  2      RETURN
;  2099          2084  2
;  2100          2085  1      END;


                                                                    .PSECT  $PLIT$,NOWRT,NOEXE,2

74  6E  75  6F  63  20  64  65  76  72  65  73  65  72  16  009D0 P.AEH:  .ASCII  <22>\reserved counter width\                    ;
                            68  74  64  69  77  20  72  65  009DF                                                               ;
32  3E  20  20  20  20  20  20  20  20  3C  34  31  21  12  009E7 P.AEI:  .ASCII  <18>\!14<         >254!>\                    ;
                                                    3E  21  34  35  009F6                                                       ;
35  35  36  3E  20  20  20  20  20  20  3C  34  31  21  12  009FA P.AEJ:  .ASCII  <18>\!14<       >65534!>\                    ;
                                                    3E  21  34  33  00A09                                                       ;
32  37  36  39  34  39  32  34  3E  20  3C  34  31  21  12  00A0D P.AEK:  .ASCII  <18>\!14< >4294967294!>\                    ;
                                                    3E  21  34  39  00A1C                                                       ;
                3E  21  4C  55  32  31  21  3C  34  31  21  0B  00A20 P.AEL:  .ASCII  <11>\!14<!12UL!>\                        ;
6E  6F  70  6D  6F  63  20  64  69  6C  61  76  6E  69  16  00A2C P.AEM:  .ASCII  <22>\invalid component code\                  ;
                            65  64  6F  63  20  74  6E  65  00A3B                                                               ;
6E  6F  70  6D  6F  63  20  64  69  6C  61  76  6E  69  16  00A43 P.AEN:  .ASCII  <22>\invalid component code\                  ;
                            65  64  6F  63  20  74  6E  65  00A52                                                               ;
                                                43  41  21  03  00A5A P.AEO:  .ASCII  <3>\!AC\                                  ;
        57  55  21  23  20  72  65  74  6E  75  6F  43  0C  00A5E P.AEP:  .ASCII  <12>\Counter #!UW\                            ;
```

```
               3A 67 6E 69 64 75 6C 63 6E 69 20 2C 0C 00A6B P.AEQ:  .ASCII  <12>\, including:\                    ;
 69 66 69 6C 61 75 51 20 2A 36 31 21 2F 21 15 00A78 P.AER:  .ASCII  <21>\!/!16* Qualifier #!UB\             ;
                         42 55 21 23 20 72 65 00A87
               43 41 21 20 2A 36 31 21 2F 21 0A 00A8E P.AES:  .ASCII  <10>\!/!16* !AC\                        ;

                                                             .PSECT  $CODE$,NOWRT,2

                                          0FFC 00000 NCP$PARSECOUNTER:
                                                             .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11  ; 1879
                       5B 00000000G  00  9E 00002           MOVAB   LIB$STOP, R11                         ;
                       5A 00000000G  8F  D0 00009           MOVL    #NCP$_BADMSG, R10                     ;
                       59 00000000V  00  9E 00010           MOVAB   NCP$ADDFAO, R9                        ;
                       58 00000000V  00  9E 00017           MOVAB   NCP$ADDSTR, R8                        ;
                       57 00000000'  00  9E 0001E           MOVAB   NCP$GQ_CTRDSC, R7                     ;
                       56 00000000'  00  9E 00025           MOVAB   P.AEH, R6                             ;
                       5E            0C  C2 0002C           SUBL2   #12, SP                               ;
                       53        08  BC  D0 0002F           MOVL    @CPTR, PTR                            ; 1929
      55     04  AC    0C            00  EF 00033           EXTZV   #0, #12, ID, COUID                    ; 1931
                 05        05  AC    04  E1 00039           BBC     #4, ID+1, 1$                          ; 1932
                       54            83  3C 0003E           MOVZWL  (PTR)+, BITMAP                        ; 1935
                                     02  11 00041           BRB     2$                                    ; 1936
                                     54  D4 00043 1$:       CLRL    BITMAP                                ; 1939
      50     05  AC    02            05  EF 00045 2$:       EXTZV   #5, #2, ID+1, R0                      ; 1942
                                     0B  12 0004B           BNEQ    3$                                    ; 1945
                                     56  DD 0004D           PUSHL   R6                                    ; 1947
                                     01  DD 0004F           PUSHL   #1                                    ;
                                     5A  DD 00051           PUSHL   R10                                   ;
                       6B            03  FB 00053           CALLS   #3, LIB$STOP                          ;
                                     58  11 00056           BRB     9$                                    ; 1946
                       01            50  D1 00058 3$:       CMPL    R0, #1                                ; 1950
                                     18  12 0005B           BNEQ    5$                                    ;
                       52            63  9A 0005D           MOVZBL  (PTR), COUVAL                         ; 1952
                 FF    8F            52  91 00060           CMPB    COUVAL, #-1                           ; 1953
                                     0B  12 00064           BNEQ    4$                                    ;
                       52            01  CE 00066           MNEGL   #1, COUVAL                            ; 1956
                                     57  DD 00069           PUSHL   R7                                    ; 1957
                             17  A6  9F 0006B           PUSHAB  P.AEI                                 ;
                       68            02  FB 0006E           CALLS   #2, NCP$ADDSTR                        ;
                                     53  D6 00071 4$:       INCL    PTR                                   ; 1960
                                     3B  11 00073           BRB     9$                                    ;
                       02            50  D1 00075 5$:       CMPL    R0, #2                                ; 1963
                                     1A  12 00078           BNEQ    7$                                    ;
                       52            63  3C 0007A           MOVZWL  (PTR), COUVAL                         ; 1965
               FFFF    8F            52  B1 0007D           CMPW    COUVAL, #-1                           ; 1966
                                     0B  12 00082           BNEQ    6$                                    ;
                       52            01  CE 00084           MNEGL   #1, COUVAL                            ; 1969
                                     57  DD 00087           PUSHL   R7                                    ; 1970
                             2A  A6  9F 00089           PUSHAB  P.AEJ                                 ;
                       68            02  FB 0008C           CALLS   #2, NCP$ADDSTR                        ;
                       53            02  C0 0008F 6$:       ADDL2   #2, PTR                               ; 1973
                                     1C  11 00092           BRB     9$                                    ;
                       03            50  D1 00094 7$:       CMPL    R0, #3                                ; 1976
                                     17  12 00097           BNEQ    9$                                    ;
                       52            63  D0 00099           MOVL    (PTR), COUVAL                         ; 1978
           FFFFFFFF    8F            52  D1 0009C           CMPL    COUVAL, #-1                           ; 1979
```

```
                                        08  12  000A3          BNEQ    8$                                      ;
                                        57  DD  000A5          PUSHL   R7                                      ; 1982
                                    3D  A6  9F  000A7          PUSHAB  P.AEK                                   ;
                                68      02  FB  000AA          CALLS   #2, NCP$ADDSTR                          ;
                                53      04  C0  000AD  8$:     ADDL2   #4, PTR                                 ; 1985
                         FFFFFFFF   8F      52  D1  000B0  9$:  CMPL    COUVAL, #-1                             ; 1991
                                        0D  13  000B7          BEQL    10$                                     ;
                                        57  DD  000B9          PUSHL   R7                                      ; 1996
                                    50  A6  9F  000BB          PUSHAB  P.AEL                                   ;
                                68      02  FB  000BE          CALLS   #2, NCP$ADDSTR                          ;
                                        52  DD  000C1          PUSHL   COUVAL                                  ; 1997
                                69      01  FB  000C3          CALLS   #1, NCP$ADDFAO                          ;
                                        5E  DD  000C6  10$:     PUSHL   SP                                      ; 2003
                                    08  AE  9F  000C8          PUSHAB  COULST                                  ;
                                    10  AE  9F  000CB          PUSHAB  COUTYP                                  ;
                                        55  DD  000CE          PUSHL   COUID                                   ; 2027
                                51  00000000G  00  D0  000D0  MOVL    NCP$GL_ENTITY, R1                       ; 2005
                                        09  12  000D7          BNEQ    11$                                     ; 2007
                                50  00000000G  00  9E  000D9  MOVAB   NCP$GA_PRM_NODCNTR, R0                  ;
                                        58  11  000E0          BRB     19$                                     ;
                                01              51  D1  000E2  11$:    CMPL    R1, #1                          ; 2008
                                        09  12  000E5          BNEQ    12$                                     ;
                                50  00000000G  00  9E  000E7  MOVAB   NCP$GA_PRM_LINCNTR, R0                  ;
                                        4A  11  000EE          BRB     19$                                     ;
                                03              51  D1  000F0  12$:    CMPL    R1, #3                          ; 2009
                                        09  12  000F3          BNEQ    13$                                     ;
                                50  00000000G  00  9E  000F5  MOVAB   NCP$GA_PRM_CIRCNTR, R0                  ;
                                        3C  11  000FC          BRB     19$                                     ;
                                04              51  D1  000FE  13$:    CMPL    R1, #4                          ; 2010
                                        2D  12  00101          BNEQ    17$                                     ;
                                51  00000000G  00  D0  00103  MOVL    NCP$GL_MODTYP, R1                       ; 2012
                                06              51  D1  0010A  CMPL    R1, #6                                  ; 2014
                                        09  12  0010D          BNEQ    14$                                     ;
                                50  00000000G  00  9E  0010F  MOVAB   NCP$GA_PRM_MPRCNTR, R0                  ;
                                        22  11  00116          BRB     19$                                     ;
                                07              51  D1  00118  14$:    CMPL    R1, #7                          ; 2015
                                        05  13  0011B          BEQL    15$                                     ;
                                09              51  D1  0011D  CMPL    R1, #9                                  ; 2016
                                        09  12  00120          BNEQ    16$                                     ;
                                50  00000000G  00  9E  00122  15$:    MOVAB   NCP$GA_PRM_MSECNTR, R0          ;
                                        0F  11  00129          BRB     19$                                     ;
                                        5C  A6  9F  0012B  16$:    PUSHAB  P.AEM                               ; 2019
                                        03  11  0012E          BRB     18$                                     ; 2018
                                        73  A6  9F  00130  17$:    PUSHAB  P.AEN                               ; 2024
                                        01  DD  00133  18$:    PUSHL   #1                                      ; 2023
                                        5A  DD  00135          PUSHL   R10                                     ;
                                68      03  FB  00137          CALLS   #3, LIB$STOP                            ;
                                        50  DD  0013A  19$:    PUSHL   R0                                      ;
                         00000000V  00      05  FB  0013C          CALLS   #5, NCP$PTBSEARCH                   ; 2004
                                10      50  E9  00143          BLBC    R0, 20$                                 ;
                                        57  DD  00146          PUSHL   R7                                      ; 2034
                                    008A  C6  9F  00148          PUSHAB  P.AEO                                   ;
                                68      02  FB  0014C          CALLS   #2, NCP$ADDSTR                          ;
                                        6E  DD  0014F          PUSHL   COUTXT                                  ; 2035
                                69      01  FB  00151          CALLS   #1, NCP$ADDFAO                          ;
                                        11  11  00154          BRB     21$                                     ;
                                        57  DD  00156  20$:    PUSHL   R7                                      ; 2039
```

```
                              008E     C6   9F  00158        PUSHAB  P.AEP                          :
                        68             02   FB  0015C        CALLS   #2, NCP$ADDSTR                 :
                                       55   DD  0015F        PUSHL   COUID                          : 2040
                        69             01   FB  00161        CALLS   #1, NCP$ADDFAO                 :
                                 04    AE   D4  00164        CLRL    COULST                         : 2041
                                       54   D5  00167 21$:   TSTL    BITMAP                         : 2045
                                       4E   13  00169        BEQL    27$                            :
                                       57   DD  0016B        PUSHL   R7                             : 2048
                              009B     C6   9F  0016D        PUSHAB  P.AEQ                          :
                        68             02   FB  00171        CALLS   #2, NCP$ADDSTR                 :
                                       52   D4  00174        CLRL    NBIT                           : 2049
            21          54             52   E1  00176 22$:   BBC     NBIT, BITMAP, 25$              : 2052
                                 04    AE   D5  0017A        TSTL    COULST                         : 2055
                                       0D   12  0017D        BNEQ    23$                            :
                                       57   DD  0017F        PUSHL   R7                             : 2058
                              00A8     C6   9F  00181        PUSHAB  P.AER                          :
                        68             02   FB  00185        CALLS   #2, NCP$ADDSTR                 :
                                       52   DD  00188        PUSHL   NBIT                           : 2059
                                       0C   11  0018A        BRB     24$                            :
                                       57   DD  0018C 23$:   PUSHL   R7                             : 2063
                              00BE     C6   9F  0018E        PUSHAB  P.AES                          :
                        68             02   FB  00192        CALLS   #2, NCP$ADDSTR                 :
                                 04    AE   DD  00195        PUSHL   COULST                         : 2064
                        69             01   FB  00198 24$:   CALLS   #1, NCP$ADDFAO                 :
                        51       04    AE   D0  0019B 25$:   MOVL    COULST, R1                     : 2068
                                       11   13  0019F        BEQL    26$                            :
                        50             61   9A  001A1        MOVZBL  (R1), R0                       : 2072
                  04    AE         01 A140  9E  001A4        MOVAB   1(R1)[R0], COULST              :
                                 04    BE   95  001AA        TSTB    @COULST                        : 2073
                                       03   12  001AD        BNEQ    26$                            :
                                 04    AE   D4  001AF        CLRL    COULST                         : 2075
                                       52   D6  001B2 26$:   INCL    NBIT                           : 2049
                        0F             52   D1  001B4        CMPL    NBIT, #15                      :
                                       BD   1B  001B7        BLEQU   22$                            :
                  08    BC             53   D0  001B9 27$:   MOVL    PTR, @CPTR                     : 2081
                                            04  001BD        RET                                    : 2085

; Routine Size:  446 bytes,    Routine Base:  $CODE$ + 0B47
```

```
;  2102          2086  1 %SBTTL  'NCP$PARSEPARM  Parse a Parameter'
;  2103          2087  1 ROUTINE NCP$PARSEPARM (DTY, PTR, TYP, LST) :NOVALUE =   !
;  2104          2088  1
;  2105          2089  1 !++
;  2106          2090  1 ! FUNCTIONAL DESCRIPTION:
;  2107          2091  1 !
;  2108          2092  1 !       This routine parses a parameter given its data type code and
;  2109          2093  1 !       puts text in the control string and entries in the fao list
;  2110          2094  1 !       to make the text for the data portion of the parameter.
;  2111          2095  1 !
;  2112          2096  1 !       Parameters are decoded in one of two ways.  By the format implicit
;  2113          2097  1 !       in their ID code.  This type is passed as TYP and comes from a PTB.
;  2114          2098  1 !       The other way is by explicit coding in the DTY code.  Even if we
;  2115          2099  1 !       do not know the format in the implicit case, we know the length of the
;  2116          2100  1 !       data so we can report the data in hexadecimal and skip the
;  2117          2101  1 !       data.  If the format is not consistant in some way, we signal an
;  2118          2102  1 !       error with a detail of the reason for the inconsistancy.
;  2119          2103  1 !
;  2120          2104  1 !       This routine is fully recursive because multiple fields contain
;  2121          2105  1 !       several data type codes followed by parameter data.  Runaway
;  2122          2106  1 !       recursion is prevented, because we reject multiple fields inside
;  2123          2107  1 !       of multiple fields.
;  2124          2108  1 !
;  2125          2109  1 ! FORMAL PARAMETERS:
;  2126          2110  1 !
;  2127          2111  1 !       DTY             Value of the data type field
;  2128          2112  1 !       PTR             Address of the pointer to the data
;  2129          2113  1 !       TYP             Value of the type code from the PTB
;  2130          2114  1 !       LST             Address of the keyword list for keyword parameters
;  2131          2115  1 !
;  2132          2116  1 ! IMPLICIT INPUTS:
;  2133          2117  1 !
;  2134          2118  1 !       NCP$A_COLTBL    To decide about width of integers
;  2135          2119  1 !       NCP$B_COLHEAD
;  2136          2120  1 !
;  2137          2121  1 ! IMPLICIT OUTPUTS:
;  2138          2122  1 !
;  2139          2123  1 !       FAOLST          List of fao entries
;  2140          2124  1 !       CTRBUF          Control string buffer
;  2141          2125  1 !
;  2142          2126  1 ! ROUTINE VALUE:
;  2143          2127  1 ! COMPLETION CODES:
;  2144          2128  1 !
;  2145          2129  1 !       NONE
;  2146          2130  1 !
;  2147          2131  1 ! SIDE EFFECTS:
;  2148          2132  1 !
;  2149          2133  1 !       NONE
;  2150          2134  1 !
;  2151          2135  1 !--
```

```
;  2153          2136  1
;  2154          2137  2      BEGIN
;  2155          2138  2
;  2156          2139  2      MAP
;  2157          2140  2          DTY : BBLOCK [LONG]     ! Data type
;  2158          2141  2          ;
;  2159          2142  2
;  2160          2143  2      LOCAL
;  2161          2144  2          PSTRT,                  ! Address of start of parameter data
;  2162          2145  2          MULCTR,                 ! Multiple field counter
;  2163          2146  2          LEN,                    ! Length of data
;  2164          2147  2          NDTY : BBLOCK [LONG],   ! New data type code, for recursion
;  2165          2148  2          TXT,                    ! Address of text of name
;  2166          2149  2          SOURTYP,                ! Source type code for events
;  2167          2150  2          ECLS                    ! Event class code
;  2168          2151  2          ;
;  2169          2152  2
;  2170          2153  2
;  2171          2154  2  !
;  2172          2155  2  !       The NICE return message is self descriptive.
;  2173          2156  2  !       The data returned is decribed in the DTY field.
;  2174          2157  2  !       If the parameter is coded, then NCPSHOLIS must know
;  2175          2158  2  !       about the specific parameter type cause they usually
;  2176          2159  2  !       require special formatting, such as insertion of ':'s or
;  2177          2160  2  !       '-'s.
;  2178          2161  2  !       If the parameter is not coded, then the format of the data is
;  2179          2162  2  !       probably explicit in the data type code, expcept for a few
;  2180          2163  2  !       exceptions such as HEX and NIADR which although are not coded
;  2181          2164  2  !       require some special formatting treatment.
;  2182          2165  2  !
;  2183          2166  2
;  2184          2167  2      IF NOT .DTY [NMA$V_PTY_COD]
;  2185          2168  2      THEN
;  2186          2169  3          BEGIN
;  2187          2170  3
;  2188          2171  3          IF .TYP EQL PBK$K_HEX                   ! Not coded and is type HEX
;  2189          2172  3          THEN
;  2190          2173  4              BEGIN                               ! data type is HEX and must be formatted
;  2191          2174  4              LEN = .(..PTR) <0, 8, 0>;
;  2192          2175  4              .PTR = ..PTR + 1;
;  2193          2176  4
;  2194          2177  4              INCRU IDX FROM 0 TO .LEN -1
;  2195          2178  4              DO
;  2196          2179  5                  BEGIN
;  2197          2180  5                  ADDSTR ('!XB');             ! They are always in hex and not reversed
;  2198          2181  5                  ADDFAO (.(..PTR + .IDX) <0, 8, 0>);
;  2199          2182  4                  END;
;  2200          2183  4
;  2201          2184  4              .PTR = ..PTR + .LEN;                ! Skip the parameter data
;  2202          2185  4              RETURN
;  2203          2186  3              END;                                ! Not coded and type HEX
;  2204          2187  3
;  2205          2188  3  !
;  2206          2189  3  !       Type NIADR is a 12 digit hex number formatted as 6 hyphen seperated
;  2207          2190  3  !       pairs, nn-nn-nn-nn-nn-nn
;  2208          2191  3  !
;  2209          2192  3          IF .TYP EQL PBK$K_NIADR                 ! Not coded and is type NIADR
```

```
;  2210          2193  3          THEN
;  2211          2194  4              BEGIN                                   ! data type is NIADR and must be formatted
;  2212          2195  4              LEN = .(..PTR) <0, 8, 0>;
;  2213          2196  4              .PTR = ..PTR + 1;
;  2214          2197  4
;  2215          2198  4              ADDSTR ('!XB');                         ! They are always in hex
;  2216          2199  4              ADDFAO (.(..PTR) <0, 8, 0>);
;  2217          2200  4
;  2218          2201  4              INCRU IDX FROM 1 TO .LEN -1 DO
;  2219          2202  5                  BEGIN
;  2220          2203  5                  ADDSTR ('-');                       ! They are always in hex
;  2221          2204  5                  ADDSTR ('!XB');                     ! They are always in hex
;  2222          2205  5                  ADDFAO (.(..PTR + .IDX) <0, 8, 0>)
;  2223          2206  4                  END;
;  2224          2207  4
;  2225          2208  4              .PTR = ..PTR + .LEN;                    ! Skip the parameter data
;  2226          2209  4              RETURN
;  2227          2210  3              END;                                    ! Not coded and type NIADR
```

```
;  2229          2211  3          IF .DTY [NMA$V_PTY_ASC]                    ! Is the coding ascii image?
;  2230          2212  3          THEN
;  2231          2213  4              BEGIN
;  2232          2214  4              ADDSTR ('!AC');                        ! Formatting is easy
;  2233          2215  4              ADDFAO (..PTR);                        ! And skip the data
;  2234          2216  4              CH$WCHAR                               ! Rewrite count minus the exec bit
;  2235          2217  4                  (
;  2236          2218  4                  CH$RCHAR (..PTR) AND %X'7F',
;  2237          2219  4                  ..PTR
;  2238          2220  4                  );
;  2239          2221  4              .PTR = ..PTR + 1 + .(..PTR) <0, 8, 0>;
;  2240          2222  4              RETURN                                 ! We are done very quickly
;  2241          2223  4              END
;  2242          2224  3          ELSE
;  2243          2225  4              BEGIN                                  ! Not ascii
;  2244          2226  4              LEN = .DTY [NMA$V_PTY_NLE];            ! Obtain the length of binary data
;  2245          2227  4              IF .LEN EQL 0                          ! Any length?
;  2246          2228  4              THEN
;  2247          2229  5                  BEGIN                              ! Its an image field
;  2248          2230  5                  LEN = .(..PTR) <0, 8, 0>;
;  2249          2231  5                  .PTR = ..PTR + 1
;  2250          2232  4                  END;
;  2251          2233  4
;  2252          2234  4              IF .LEN LEQ 4                          ! If the length is less than longword
;  2253          2235  4              THEN
;  2254          2236  5                  BEGIN                              ! Obtain the numeric format code
;  2255          2237  5                  SELECTONE .DTY [NMA$V_PTY_NTY] OF
;  2256          2238  5                  SET
;  2257          2239  5
;  2258          2240  5                  [0] :                              ! Unsigned decimal
;  2259          2241  6                      BEGIN
;  2260          2242  6                      IF  .NCP$B_COLHEAD AND         ! Header printed and
;  2261          2243  6                        .NCP$A_COLTBL NEQ 0          ! We are writting column output
;  2262          2244  6                      THEN
;  2263          2245  6                          ADDSTR ('!5U')             ! Column format, right justify
;  2264          2246  6                      ELSE
;  2265          2247  6                          ADDSTR ('!U')
;  2266          2248  5                      END;
;  2267          2249  5                  [1] : ADDSTR ('!S');               ! Signed decimal
;  2268          2250  5                  [2] : ADDSTR ('!X');               ! Hex
;  2269          2251  5                  [3] : ADDSTR ('!O');               ! Octal
;  2270          2252  5
;  2271          2253  5                  TES
;  2272          2254  4                  END;
;  2273          2255  4
;  2274          2256  4          SELECTONE .LEN OF                          ! Do the appropriate thing for length
;  2275          2257  4          SET
;  2276          2258  4
;  2277          2259  4          [0] :                                      ! If length is zero, NML screwed up
;  2278          2260  5              (
;  2279          2261  5              SIGNAL_STOP (NCP$_BADMSG, 1,
;  2280          2262  5                  ASCIC ('length of parameter is zero') )
;  2281          2263  4              );
;  2282          2264  4
;  2283          2265  4          [1] :                                      ! One is a byte? right?
;  2284          2266  5              (
;  2285          2267  5              ADDSTR ('B');
```

```
;  2286          2268  5                 ADDFAO (...PTR)
;  2287          2269  4                 );
;  2288          2270  4
;  2289          2271  4             [2] :                                   ! Two for a word, buckle my shoe
;  2290          2272  5                 (
;  2291          2273  5                 ADDSTR ('W');
;  2292          2274  5                 ADDFAO (...PTR)
;  2293          2275  4                 );
;  2294          2276  4
;  2295          2277  4             [3] :                                   ! Three for a long, wait a minute
;  2296          2278  5                 (
;  2297          2279  5                 ADDSTR ('L');
;  2298          2280  5                 ADDFAO (.(..PTR) <0, 24, 0>)    ! Sure, just use 24 bits
;  2299          2281  4                 );
;  2300          2282  4
;  2301          2283  4             [4] :                                   ! Four for a long, shut the door
;  2302          2284  5                 (                                   ! We lost the rhyming somewhere
;  2303          2285  5                 ADDSTR ('L');
;  2304          2286  5                 ADDFAO (...PTR)
;  2305          2287  4                 );
;  2306          2288  4
;  2307          2289  4             [5 TO 32] :                             ! For reasonable length binary images
;  2308          2290  5                 BEGIN
;  2309          2291  5                 INCRU IDX FROM 1 TO .LEN
;  2310          2292  5                 DO
;  2311          2293  6                     BEGIN
;  2312          2294  6                     ADDSTR ('!XB');                 ! They are always in hex
;  2313          2295  6                     ADDFAO (.(..PTR + .LEN - .IDX) <0, 8, 0>)
;  2314          2296  6                     END
;  2315          2297  5                 END
;  2316          2298  4                 ;
;  2317          2299  4
;  2318          2300  4             [OTHERWISE] :                           ! Now lets be reasonable,
;  2319          2301  5                 (                                   ! Binary data longer than 32 is dumb
;  2320          2302  5                 SIGNAL_STOP (NCP$_BADMSG, 1,
;  2321          2303  5                         ASCIC ('numeric parameter too long') )
;  2322          2304  4                 );
;  2323          2305  4
;  2324          2306  4             TES;
;  2325          2307  4
;  2326          2308  4             .PTR = ..PTR + .LEN;                    ! Skip the parameter data
;  2327          2309  4             RETURN                                  ! We are done
;  2328          2310  4             END
;  2329          2311  2         END;                                        ! Not a coded parameter
```

```
;  2331     2312  2
;  2332     2313  2 !
;  2333     2314  2 !       We have finished with explicitly formatted data both
;  2334     2315  2 !       ascii image and binary fields
;  2335     2316  2 !
;  2336     2317  2 !       Now we are going to decode implicitly formatted fields.
;  2337     2318  2 !       The format of the data is known in the PTB for the parameter
;  2338     2319  2 !       and is passed here by the TYP value.
;  2339     2320  2 !       These types of data may be multiplepleplepleply (sic) coded so
;  2340     2321  2 !       we may call ourselves to decode the subfields
;  2341     2322  2 !
;  2342     2323  2
;  2343     2324  2       IF .DTY [NMA$V_PTY_MUL]                 ! Is the field multiply coded
;  2344     2325  2       THEN
;  2345     2326  2           BEGIN
;  2346     2327  2           MULCTR = .DTY [NMA$V_PTY_CLE];      ! Grab the count of fields
;  2347     2328  2           IF .MULCTR GTR NMA$C_PTY_MAX        ! Is it ridiculous?
;  2348     2329  2           THEN                                ! Signal a parameter error
;  2349     2330  2               SIGNAL_STOP (NCP$_BADMSG, 1,
;  2350     2331  2                   ASCIC ('too many multiply coded fields') )
;  2351     2332  2           ;
;  2352     2333  2
;  2353     2334  2 !
;  2354     2335  2 !       Now we are going to handle some special formatting for multiply
;  2355     2336  2 !       coded fields.  There are three fields to worry about.
;  2356     2337  2 !       version triples, node names -addresses and logging events.
;  2357     2338  2 !       These parameters are passed as coded multiple fields so we
;  2358     2339  2 !       must add some special formatting around them.
;  2359     2340  2 !
;  2360     2341  3
;  2361     2342  3           SELECTONE .TYP OF                   ! Select action by expected type
;  2362     2343  3           SET
;  2363     2344  3
;  2364     2345  3           [PBK$K_TRIPL] :                     ! Triples are easy, just a V
;  2365     2346  4               BEGIN                           ! and . between the numbers
;  2366     2347  4               ADDSTR ('V');
;  2367     2348  4               INCR IDX FROM 1 TO .MULCTR      ! For each of the subfields
;  2368     2349  4               DO
;  2369     2350  5                   BEGIN
;  2370     2351  5                   IF .IDX NEQ 1               ! Separate after the first
;  2371     2352  5                   THEN
;  2372     2353  5                       ADDSTR ('.')
;  2373     2354  5                   ;
;  2374     2355  5                   NCP$PARSEPARM               ! Parse each following subfield
;  2375     2356  5                       (
;  2376     2357  5                       CH$RCHAR_A (.PTR),      ! Pickup data type
;  2377     2358  5                       .PTR,
;  2378     2359  5                       PBK$K_END,              ! We do not know the data
;  2379     2360  5                       0
;  2380     2361  5                       )
;  2381     2362  5                   END
;  2382     2363  4               ;
;  2383     2364  4               MULCTR = 0                      ! No more fields to process
;  2384     2365  3               END;
```

```
;  2386     2366  3
;  2387     2367  3     !
;  2388     2368  3     !       For a subaddress range coded multiple, put a dash between the values.
;  2389     2369  3     !
;  2390     2370  3
;  2391     2371  3             [PBK$K_SAD]:
;  2392     2372  4                 BEGIN
;  2393     2373  4                 INCR IDX FROM 1 TO .MULCTR  ! For each of the subfields
;  2394     2374  4                 DO
;  2395     2375  5                     BEGIN
;  2396     2376  5                     IF .IDX NEQ 1           ! Separate after the first
;  2397     2377  5                     THEN
;  2398     2378  5                         ADDSTR ('-');
;  2399     2379  5                     NCP$PARSEPARM(          ! Parse each following subfield
;  2400     2380  5                         CH$RCHAR_A (.PTR),  ! Pickup data type
;  2401     2381  5                         .PTR,
;  2402     2382  5                         PBK$K_END,          ! We do not know the data
;  2403     2383  5                         0);
;  2404     2384  4                     END;
;  2405     2385  4                 MULCTR = 0;                 ! No more fields to process
;  2406     2386  3                 END;
;  2407     2387  3
;  2408     2388  3
;  2409     2389  3     !
;  2410     2390  3     !       For a range list coded multiple, put a dash between the values
;  2411     2391  3     !       of a value pair, and place a comma between value pairs
;  2412     2392  3     !
;  2413     2393  3
;  2414     2394  3             [PBK$K_RNGL]:
;  2415     2395  4                 BEGIN
;  2416     2396  4                 INCR IDX FROM 1 TO .MULCTR  ! For each of the subfields
;  2417     2397  4                 DO
;  2418     2398  5                     BEGIN
;  2419     2399  5                     IF .IDX NEQ 1           ! Separate after the first
;  2420     2400  5                     THEN
;  2421     2401  5                         ADDSTR ('-');
;  2422     2402  5                     NCP$PARSEPARM(          ! Parse each following subfield
;  2423     2403  5                         CH$RCHAR_A (.PTR),  ! Pickup data type
;  2424     2404  5                         .PTR,
;  2425     2405  5                         PBK$K_END,          ! We do not know the data
;  2426     2406  5                         0);
;  2427     2407  4                     END;
;  2428     2408  4                 MULCTR = 0;                 ! No more fields to process
;  2429     2409  3                 END;
```

```
;  2431     2410  3
;  2432     2411  3 !
;  2433     2412  3 !       For node addresses names we need to surround the names in ()
;  2434     2413  3 !
;  2435     2414  3
;  2436     2415  3         [PBK$K_NADR] :
;  2437     2416  4             BEGIN
;  2438     2417  4             LOCAL
;  2439     2418  4                 AREA,
;  2440     2419  4                 AREA_ADDR : BBLOCK [4];
;  2441     2420  4
;  2442     2421  4             IF .MULCTR GTR 2                        ! We expect only an address
;  2443     2422  4             THEN                                    ! and a name
;  2444     2423  4                 SIGNAL_STOP (NCP$_BADMSG, 1,
;  2445     2424  4                       ASCIC ('too many multiply coded fields') );
;  2446     2425  4
;  2447     2426  4
;  2448     2427  4             .PTR = ..PTR + 1;
;  2449     2428  4             AREA_ADDR = (.(..PTR) < 0, 16, 0 > );
;  2450     2429  4             AREA = .AREA_ADDR [NMA$V_AREA];
;  2451     2430  4
;  2452     2431  4             IF .AREA EQL 0
;  2453     2432  4                 THEN
;  2454     2433  5                 BEGIN                               ! No area so handle normally
;  2455     2434  5                 ADDSTR ('!3UW');
;  2456     2435  5                 ADDFAO (.AREA_ADDR);
;  2457     2436  5                 END
;  2458     2437  4             ELSE
;  2459     2438  5                 BEGIN                               ! Non zero area must be separated from address with '.'
;  2460     2439  5                 ADDSTR ('!2UW.!UW');
;  2461     2440  5                 ADDFAO (.AREA);
;  2462     2441  5                 ADDFAO (.AREA_ADDR [NMA$V_ADDR] );
;  2463     2442  4                 END;
;  2464     2443  4
;  2465     2444  4             .PTR = ..PTR + 2;
;  2466     2445  4
;  2467     2446  4             IF .MULCTR EQL 2                        ! We expect only an address
;  2468     2447  4             THEN
;  2469     2448  5                 BEGIN
;  2470     2449  5                 ADDSTR (' (');                      ! Place parens around Node name
;  2471     2450  5                 NCP$PARSEPARM
;  2472     2451  5                     (
;  2473     2452  5                     CH$RCHAR_A (.PTR),
;  2474     2453  5                     .PTR,
;  2475     2454  5                     PBK$K_NADR,
;  2476     2455  5                     0);
;  2477     2456  5
;  2478     2457  5                 ADDSTR (')');
;  2479     2458  4                 END;
;  2480     2459  4
;  2481     2460  4             MULCTR = 0                              ! No more fields to process
;  2482     2461  3             END;
```

```
;  2484     2462  3 !
;  2485     2463  3 !       For a DELTIM coded multiple, put a colon between the values.
;  2486     2464  3 !
;  2487     2465  3
;  2488     2466  3         [PBK$K_DELTIM]:
;  2489     2467  4             BEGIN
;  2490     2468  4             INCR IDX FROM 1 TO .MULCTR  ! For each of the subfields
;  2491     2469  4             DO
;  2492     2470  5                 BEGIN
;  2493     2471  5                 LOCAL
;  2494     2472  5                     L_DTY : BBLOCK [1],
;  2495     2473  5                     LEN;
;  2496     2474  5
;  2497     2475  5                 IF .IDX NEQ 1           ! Separate after the first
;  2498     2476  5                 THEN
;  2499     2477  5                     ADDSTR (':');
;  2500     2478  5
;  2501     2479  5                 L_DTY = CH$RCHAR_A (.PTR);
;  2502     2480  5                 LEN = .L_DTY [NMA$V_PTY_NLE];
;  2503     2481  5                 ADDSTR (UPLIT('!2Z'));
;  2504     2482  5                 SELECTONE .LEN OF
;  2505     2483  5                 SET
;  2506     2484  5                 [1] :
;  2507     2485  6                     BEGIN
;  2508     2486  6                     ADDSTR ('B');
;  2509     2487  6                     ADDFAO (...PTR);
;  2510     2488  5                     END;
;  2511     2489  5                 [2] :
;  2512     2490  6                     BEGIN
;  2513     2491  6                     ADDSTR ('W');
;  2514     2492  6                     ADDFAO (...PTR);
;  2515     2493  5                     END;
;  2516     2494  5                 [OTHERWISE] :
;  2517     2495  5                     SIGNAL_STOP (NCP$_BADMSG, 1, ASCIC (' Delta Time not decoded'));
;  2518     2496  5                 TES;
;  2519     2497  5
;  2520     2498  5                 .PTR = ..PTR + .LEN;
;  2521     2499  4                 END;
;  2522     2500  4
;  2523     2501  4             MULCTR = 0;                 ! No more fields to process
;  2524     2502  3             END;
;  2525     2503  3
;  2526     2504  3
;  2527     2505  3
;  2528     2506  3 !
;  2529     2507  3 !       For Daytime specification, we must format it specially.
;  2530     2508  3 !
;  2531     2509  3
;  2532     2510  3         [PBK$K_DAYTIM]:
;  2533     2511  4             BEGIN
;  2534     2512  4             LOCAL
;  2535     2513  4                 COLTBL;
;  2536     2514  4
;  2537     2515  4             COLTBL = .NCP$A_COLTBL;     ! Save it
;  2538     2516  4             NCP$A_COLTBL = 0;           ! Blank it to override !5U formatting
;  2539     2517  4                                         !  used if COLTBL is set.
;  2540     2518  4
```

```
;  2541     2519  4              NCP$PARSEPARM (
;  2542     2520  4                  CH$RCHAR_A (.PTR),
;  2543     2521  4                  .PTR,
;  2544     2522  4                  PBK$K_END,                      ! Assume we do not know
;  2545     2523  4                  .LST);                          !  its format
;  2546     2524  4              ADDSTR ('=');
;  2547     2525  4              MULCTR = .MULCTR - 1;               ! One less subfield
;  2548     2526  4
;  2549     2527  4              NDTY = CH$RCHAR (..PTR);            ! The second data type
;  2550     2528  4              .PTR = ..PTR + 1;
;  2551     2529  4              IF .NDTY [NMA$V_PTY_COD]            ! If it's coded
;  2552     2530  4                  AND NOT .NDTY [NMA$V_PTY_MUL]   ! and not multiple
;  2553     2531  4              THEN
;  2554     2532  5                  BEGIN
;  2555     2533  5                  NCP$PARSEPARM (
;  2556     2534  5                      .NDTY,
;  2557     2535  5                      .PTR,
;  2558     2536  5                      PBK$K_LITB,                 ! Decode Month field
;  2559     2537  5                      .LST);                      !
;  2560     2538  5
;  2561     2539  5                  MULCTR = .MULCTR - 1;           ! One less subfield
;  2562     2540  5                  ADDSTR (' ');
;  2563     2541  4                  END;
;  2564     2542  4
;  2565     2543  4              INCR IDX FROM 1 TO .MULCTR  ! For each of the subfields
;  2566     2544  4              DO
;  2567     2545  5                  BEGIN
;  2568     2546  5                  LOCAL
;  2569     2547  5                      L_DTY : BBLOCK [1],
;  2570     2548  5                      LEN;
;  2571     2549  5
;  2572     2550  5                  IF .IDX NEQ 1               ! Separate after the first
;  2573     2551  5                  THEN
;  2574     2552  5                      ADDSTR (':');
;  2575     2553  5
;  2576     2554  5                  L_DTY = CH$RCHAR_A (.PTR);
;  2577     2555  5                  LEN = .L_DTY [NMA$V_PTY_NLE];
;  2578     2556  5                  ADDSTR (^!2Z');
;  2579     2557  5                  SELECTONE .LEN OF
;  2580     2558  5                  SET
;  2581     2559  5                  [1] :
;  2582     2560  6                      BEGIN
;  2583     2561  6                      ADDSTR ('B');
;  2584     2562  6                      ADDFAO (...PTR);
;  2585     2563  5                      END;
;  2586     2564  5                  [2] :
;  2587     2565  6                      BEGIN
;  2588     2566  6                      ADDSTR ('W');
;  2589     2567  6                      ADDFAO (...PTR);
;  2590     2568  5                      END;
;  2591     2569  5                  [OTHERWISE] :
;  2592     2570  5                      SIGNAL_STOP (NCP$_BADMSG, 1, ASCIC (' Daytime not decoded'));
;  2593     2571  5                  TES;
;  2594     2572  5
;  2595     2573  5                  .PTR = ..PTR + .LEN;
;  2596     2574  4                  END;
;  2597     2575  4
```

```
;  2598          2576  4            NCP$A_COLTBL = .COLTBL;      ! Restore it
;  2599          2577  4            MULCTR = 0;                  ! No more fields to process
;  2600          2578  3            END;
;  2601          2579  3
;  2602          2580  3
;  2603          2581  3 !
;  2604          2582  3 !       For a LITLST coded multiple, put a dash between the values.
;  2605          2583  3 !
;  2606          2584  3
;  2607          2585  3        [PBK$K_LITLST]:
;  2608          2586  4            BEGIN
;  2609          2587  4            INCR IDX FROM 1 TO .MULCTR  ! For each of the subfields
;  2610          2588  4            DO
;  2611          2589  5                BEGIN
;  2612          2590  5                IF .IDX NEQ 1           ! Separate after the first
;  2613          2591  5                THEN
;  2614          2592  5                    ADDSTR (', ');
;  2615          2593  5                NCP$PARSEPARM(          ! Parse each following subfield
;  2616          2594  5                    CH$RCHAR_A (.PTR),  ! Pickup data type
;  2617          2595  5                    .PTR,
;  2618          2596  5                    PBK$K_LITB,         ! We know they'll be coded
;  2619          2597  5                    .LST);              !  from the same table
;  2620          2598  4                END;
;  2621          2599  4            MULCTR = 0;                 ! No more fields to process
;  2622          2600  3            END;
```

```
;  2624          2601  3
;  2625          2602  3 !
;  2626          2603  3 !       For entity specifications, we must format it specially.
;  2627          2604  3 !
;  2628          2605  3
;  2629          2606  3         [PBK$K_ENT]:
;  2630          2607  4             BEGIN
;  2631          2608  4             NDTY = CH$RCHAR (..PTR);                ! The first data type
;  2632          2609  4             IF .NDTY [NMA$V_PTY_COD]                ! If it's coded
;  2633          2610  4                 AND NOT .NDTY [NMA$V_PTY_MUL]       ! and not multiple
;  2634          2611  4             THEN
;  2635          2612  5                 BEGIN
;  2636          2613  5                 SOURTYP = CH$RCHAR (..PTR + 1); ! Save the entity ID
;  2637          2614  5                 .PTR = ..PTR + 2;               ! Skip CM byte and entity ID
;  2638          2615  5
;  2639          2616  5                 MULCTR = .MULCTR - 1;           ! One less subfield
;  2640          2617  5                 IF .MULCTR GTR 0                ! If there are more left
;  2641          2618  5                 THEN
;  2642          2619  6                     BEGIN
;  2643          2620  6                     NDTY = CH$RCHAR (..PTR);    ! Next data type code
;  2644          2621  6                     IF .SOURTYP EQL NMA$C_ENT_NOD ! If it's a node,
;  2645          2622  6                         AND NOT .NDTY [NMA$V_PTY_COD] ! and not coded,
;  2646          2623  6                         AND NOT .NDTY [NMA$V_PTY_ASC] ! and not ascii,
;  2647          2624  6                         AND .(..PTR+1) < 0, 16, 0 > EQL 0 ! and it's node number 0,
;  2648          2625  6                     THEN
;  2649          2626  7                         BEGIN
;  2650          2627  7                         ADDSTR('Executor');     ! then print it specially
;  2651          2628  7                         .PTR = ..PTR + 3;       ! Skip over node address
;  2652          2629  7                         END
;  2653          2630  6                     ELSE
;  2654          2631  7                         BEGIN
;  2655          2632  7                         SELECTONEU .SOURTYP     ! Based on entity type,
;  2656          2633  7                         OF
;  2657          2634  7                             SET
;  2658          2635  7                             [NMA$C_ENT_LIN]: ADDSTR('Line ');
;  2659          2636  7                             [NMA$C_ENT_CIR]: ADDSTR('Circuit ');
;  2660          2637  7                             [NMA$C_ENT_NOD]: ADDSTR('Node ');
;  2661          2638  7                             [NMA$C_ENT_MOD]: ADDSTR('Module ');
;  2662          2639  7                             TES;
;  2663          2640  7                         NCP$PARSEPARM(          ! Parse the string or number
;  2664          2641  7                             CH$RCHAR_A (.PTR),
;  2665          2642  7                             .PTR,
;  2666          2643  7                             PBK$K_END,          ! Assume we do not know its
;  2667          2644  7                             0);                 ! format
;  2668          2645  6                         END;
;  2669          2646  6                 MULCTR = .MULCTR - 1;           ! One less subfield
;  2670          2647  5                 END;
;  2671          2648  5
;  2672          2649  5             NDTY = CH$RCHAR (..PTR);            ! Next data type code
;  2673          2650  5             IF .SOURTYP EQL NMA$C_ENT_NOD       ! Source is a node
;  2674          2651  5                 AND .MULCTR GTR 0               ! Fields are left
;  2675          2652  5                 AND .NDTY [NMA$V_PTY_ASC]       ! Its ascii
;  2676          2653  5                 AND NOT .NDTY [NMA$V_PTY_COD]   ! And not coded
;  2677          2654  5             THEN
;  2678          2655  6                 BEGIN
;  2679          2656  6                 NCP$PARSEPARM(                  ! Parse as a node name
;  2680          2657  6                     CH$RCHAR_A (.PTR),
```

```
;  2681          2658  6                          .PTR,
;  2682          2659  6                          PBK$K_NADR,
;  2683          2660  6                          0);
;  2684          2661  6                      MULCTR = .MULCTR - 1;       ! One less parameter
;  2685          2662  5                      END;
;  2686          2663  4                  END;
;  2687          2664  3              END;
```

```
;  2689          2665  3
;  2690          2666  3 !
;  2691          2667  3 !       Events are very difficult however.  We must format the source
;  2692          2668  3 !       type and follow it by the event class (possibly wild) and then the
;  2693          2669  3 !       event mask.
;  2694          2670  3 !
;  2695          2671  3
;  2696          2672  3
;  2697          2673  3         [PBK$K_ESET TO PBK$K_ESNO, PBK$K_ESCI] :
;  2698          2674  4             BEGIN
;  2699          2675  4             NDTY = CH$RCHAR (..PTR);                ! The first data type
;  2700          2676  4             IF  .NDTY [NMA$V_PTY_COD]               ! If its coded and not mult
;  2701          2677  4                 AND NOT
;  2702          2678  4                 .NDTY [NMA$V_PTY_MUL]               ! Then its the source type
;  2703          2679  4             THEN
;  2704          2680  5                 BEGIN
;  2705          2681  5                 SOURTYP = CH$RCHAR (..PTR + 1); ! Save the source type for
;  2706          2682  5                                                 ! Later formatting
;  2707          2683  5                 IF .NCP$A_COLTBL NEQ 0          ! Formatting for columns?
;  2708          2684  5                 THEN
;  2709          2685  6                     BEGIN
;  2710          2686  6                     ADDSTR ('!19<');            ! Source in a field
;  2711          2687  6                     .PTR = ..PTR + 2            ! Skip the data type byte
;  2712          2688  6                     END                         ! and the code byte too
;  2713          2689  5                 ELSE
;  2714          2690  5                     NCP$PARSEPARM               ! Write source type
;  2715          2691  5                         (                       ! parse the parameter to
;  2716          2692  5                         CH$RCHAR_A (.PTR),      ! Look up the keyword for the
;  2717          2693  5                         .PTR,                   ! source type.
;  2718          2694  5                         PBK$K_LITB,
;  2719          2695  5                         NCP$GA_TBL_EVESOUR      ! Event source keyword table
;  2720          2696  5                         )
;  2721          2697  5                 ;
;  2722          2698  5                 MULCTR = .MULCTR - 1;           ! One less subfield
;  2723          2699  5                 IF .SOURTYP NEQ %X'FF'          ! If it's not a null source
;  2724          2700  5                     AND
;  2725          2701  5                     .MULCTR GTR 0               ! And there are more left
;  2726          2702  5                 THEN
;  2727          2703  6                     BEGIN
;  2728          2704  6                     SELECTONEU .SOURTYP         ! Based on source type,
;  2729          2705  6                     OF
;  2730          2706  6                         SET
;  2731          2707  6                         [NMA$C_ENT_LIN]: ADDSTR('Line ');
;  2732          2708  6                         [NMA$C_ENT_CIR]: ADDSTR('Circuit ');
;  2733          2709  6                         [NMA$C_ENT_NOD]: ADDSTR('Node ');
;  2734          2710  6                         TES;
;  2735          2711  6                     IF .SOURTYP NEQ             ! If it's not a node
;  2736          2712  6                       NMA$C_ENT_NOD
;  2737          2713  6                     THEN
;  2738          2714  6                         NCP$PARSEPARM           ! Parse the string or number
;  2739          2715  6                         (
;  2740          2716  6                         CH$RCHAR_A (.PTR),
;  2741          2717  6                         .PTR,
;  2742          2718  6                         PBK$K_END,              ! Assume we do not know its
;  2743          2719  6                         0                       ! format
;  2744          2720  6                         )
;  2745          2721  6                     ELSE
```

```
;  2746     2722  7                    BEGIN
;  2747     2723  7                    LOCAL
;  2748     2724  7                        AREA,
;  2749     2725  7                        AREA_ADDR : BBLOCK [4];
;  2750     2726  7
;  2751     2727  7                    .PTR = ..PTR + 1;
;  2752     2728  7                    AREA_ADDR = (.(..PTR) < 0, 16, 0 > );
;  2753     2729  7                    AREA = .AREA_ADDR [NMA$V_AREA];
;  2754     2730  7
;  2755     2731  7                    IF .AREA EQL 0
;  2756     2732  7                    THEN
;  2757     2733  8                        BEGIN                   ! No area so handle normally
;  2758     2734  8                        ADDSTR ('!3UW');
;  2759     2735  8                        ADDFAO (.AREA_ADDR);
;  2760     2736  8                        END
;  2761     2737  7                    ELSE
;  2762     2738  8                        BEGIN                   ! Non zero area must be
;  2763     2739  8                        ADDSTR ('!2UW.!UW');    !  separated from address
;  2764     2740  8                        ADDFAO (.AREA);         !  with '.'
;  2765     2741  8                        ADDFAO (.AREA_ADDR [NMA$V_ADDR] );
;  2766     2742  7                        END;
;  2767     2743  7
;  2768     2744  7                    .PTR = ..PTR + 2;
;  2769     2745  6                    END;
;  2770     2746  6
;  2771     2747  6                MULCTR = .MULCTR - 1            ! One less subfield
;  2772     2748  6                END
;  2773     2749  5            ELSE
;  2774     2750  5                ADDSTR('(All sources)');
;  2775     2751  5
;  2776     2752  5            NDTY = CH$RCHAR (..PTR);            ! Next data type code
;  2777     2753  5            IF .SOURTYP EQL NMA$C_ENT_NOD       ! If source is a node
;  2778     2754  5                AND
;  2779     2755  5                .MULCTR GTR 0                   ! Fields are left
;  2780     2756  5                AND
;  2781     2757  5                .NDTY [NMA$V_PTY_ASC]           ! Its ascii
;  2782     2758  5                AND NOT
;  2783     2759  5                .NDTY [NMA$V_PTY_COD]           ! And not coded
;  2784     2760  5            THEN
;  2785     2761  6                BEGIN
;  2786     2762  6                ADDSTR (' ('):                  ! Place parens around Node name
;  2787     2763  6                NCP$PARSEPARM                   ! Parse as a node name
;  2788     2764  6                    (
;  2789     2765  6                    CH$RCHAR_A (.PTR),
;  2790     2766  6                    .PTR,
;  2791     2767  6                    PBK$K_NADR,
;  2792     2768  6                    0
;  2793     2769  6                    );
;  2794     2770  6                ADDSTR (')');
;  2795     2771  6                MULCTR = .MULCTR - 1            ! One less parameter
;  2796     2772  6                END
;  2797     2773  5            ;
;  2798     2774  5            IF .NCP$A_COLTBL NEQ 0              ! If column output
;  2799     2775  5            THEN
;  2800     2776  5                ADDSTR ('!> ')                  ! Close off the source column
;  2801     2777  5            END
;  2802     2778  4        ;
```

```
;  2803     2779  4              IF .MULCTR GTR 0                        ! If there are still more
;  2804     2780  4              THEN                                    ! Subfields
;  2805     2781  5                  BEGIN
;  2806     2782  5                  NDTY = CH$RCHAR (..PTR);            ! Look at the data type
;  2807     2783  5                  IF .NDTY EQL 2                      ! Two byte unsigned decimal?
;  2808     2784  5                  THEN                                ! Its the class code
;  2809     2785  6                      BEGIN
;  2810     2786  6                      .PTR = ..PTR + 1;               ! Advance to value
;  2811     2787  6                      ECLS = (.(..PTR) <0, 9, 0>);    ! Get the class (9 bits)
;  2812     2788  6                      .PTR = ..PTR + 2;               ! Advance over it
;  2813     2789  6                      MULCTR = .MULCTR - 1;           ! Reduce the number of fields
;  2814     2790  6                      SELECTONEU .(..PTR-2)<14, 2, 0> OF! Look at the wild codes
;  2815     2791  6                      SET
;  2816     2792  6
;  2817     2793  6                      [0] :                           ! Single event class, and mask
;  2818     2794  7                          (
;  2819     2795  7                          ADDFAO (.ECLS);
;  2820     2796  7                          ADDSTR (' !UW.');           ! for the class
;  2821     2797  7                          NDTY = CH$RCHAR (..PTR);    ! Get the data type
;  2822     2798  7                          IF .NDTY EQL %X'20'         ! Hex image?
;  2823     2799  7                              AND
;  2824     2800  7                              .MULCTR GTR 0           ! And there is one
;  2825     2801  7                          THEN
;  2826     2802  8                              BEGIN
;  2827     2803  8                              .PTR = ..PTR + 1;       ! Advance to hex image
;  2828     2804  8                              NCP$PARSEVENTS (.PTR);! Decode the mask
;  2829     2805  8                                                      ! With a little help from our
;  2830     2806  8                                                      ! friends
;  2831     2807  8                              MULCTR = .MULCTR - 1 ! One less subfield
;  2832     2808  8                              END
;  2833     2809  6                          );
;  2834     2810  6
;  2835     2811  6                      [1] :                           ! This is a reserved type code
;  2836     2812  7                          (
;  2837     2813  7                          SIGNAL_STOP (NCP$_BADMSG, 1,
;  2838     2814  7                          ASCIC ('reserved event class code') )
;  2839     2815  6                          );
;  2840     2816  6
;  2841     2817  6                      [2] :                           ! wild Mask
;  2842     2818  7                          (
;  2843     2819  7                          ADDFAO (.ECLS);
;  2844     2820  7                          ADDSTR (' !UW.*')
;  2845     2821  6                          );
;  2846     2822  6
;  2847     2823  6                      [3] :                           ! Wild class and events
;  2848     2824  7                          (
;  2849     2825  7                          ADDSTR (' *.*')
;  2850     2826  6                          );
;  2851     2827  6                      TES
;  2852     2828  6                      END
;  2853     2829  5                  END
;  2854     2830  3              END;
;  2855     2831  3          TES
;  2856     2832  3          ;
```

```
;  2858          2833  3
;  2859          2834  3 !
;  2860          2835  3 !       Take care of all the subfields that are left after we perform
;  2861          2836  3 !       the special actions.
;  2862          2837  3 !       If the special formats are wrong this will print the remainder
;  2863          2838  3 !       without leaving us in the middle of a multiple coded field.
;  2864          2839  3 !
;  2865          2840  3
;  2866          2841  3         INCR IDX FROM 1 TO .MULCTR      ! For each of the subfields
;  2867          2842  3         DO
;  2868          2843  4             BEGIN
;  2869          2844  4             IF .IDX NEQ 1               ! Separate after the first
;  2870          2845  4             THEN
;  2871          2846  4                 ADDSTR (', ')
;  2872          2847  4             ;
;  2873          2848  4             NDTY = .(..PTR) <0, 8, 0>;  ! The new data type
;  2874          2849  4             IF  .NDTY [NMA$V_PTY_MUL]   ! If its multiply coded,
;  2875          2850  4                 AND
;  2876          2851  4                 .NDTY [NMA$V_PTY_COD]
;  2877          2852  4             THEN                        ! Blow away, for multiple recursion
;  2878          2853  4                 SIGNAL_STOP (NCP$_BADMSG, 1,
;  2879          2854  4                         ASCIC ('recursive multiple fields') )
;  2880          2855  4             ;
;  2881          2856  4             .PTR = ..PTR + 1;           ! Skip beyond the data type
;  2882          2857  4             NCP$PARSEPARM (.NDTY, .PTR, PBK$K_END, 0);
;  2883          2858  4             END
;  2884          2859  3         ;
;  2885          2860  3         RETURN
;  2886          2861  3         END
;  2887          2862  2     ;
```

```
;  2889     2863  2
;  2890     2864  2 !
;  2891     2865  2 !       Its not a multiply coded field, so decode the data based on
;  2892     2866  2 !       the type code passed by the TYP parameter.
;  2893     2867  2 !
;  2894     2868  2
;  2895     2869  2 !
;  2896     2870  2 !       Dispatch on the data type to handle each separately
;  2897     2871  2 !
;  2898     2872  2
;  2899     2873  2       LEN = .DTY [NMA$V_PTY_CLE];             ! The length of the parameter data
;  2900     2874  2       PSTRT = ..PTR;                          ! Save the start of the parameter data
;  2901     2875  2
;  2902     2876  2       SELECTONE .TYP OF                       ! Dispatch on the type code
;  2903     2877  2       SET
;  2904     2878  2
;  2905     2879  2       [PBK$K_LITB] :                          ! This means we want to search a table
;  2906     2880  3           (                                   ! For a keyword based on a byte code
;  2907     2881  3           IF .LST EQL 0                       ! If the list address is zero,
;  2908     2882  3           THEN                                ! Thats our fault
;  2909     2883  3               SIGNAL_STOP (NCP$_LOGIC, 1, ASCIC ('invalid keyword list') )
;  2910     2884  3           ;
;  2911     2885  3           IF
;  2912     2886  3               NCP$TABLESEARCH             ! General routine for searching tables
;  2913     2887  3                   (
;  2914     2888  3                   .(..PTR) <0, 8, 0>,     ! Byte code
;  2915     2889  3                   .LST,                   ! Table address is word offset
;  2916     2890  3                   TXT                     ! Put address of text here
;  2917     2891  3                   )
;  2918     2892  3           THEN
;  2919     2893  4               BEGIN
;  2920     2894  4               ADDFAO (.TXT);              ! Write the text we know
;  2921     2895  4               ADDSTR ('!AC')
;  2922     2896  4               END
;  2923     2897  3           ELSE
;  2924     2898  4               BEGIN
;  2925     2899  4               ADDFAO (...PTR);            ! Write in standard form
;  2926     2900  4               ADDSTR ('type #!UB')
;  2927     2901  4               END
;  2928     2902  3           ;
;  2929     2903  3           .PTR = ..PTR + 1                ! Advance over code
;  2930     2904  2           );
```

```
;  2932     2905  2
;  2933     2906  2
;  2934     2907  2      !   Privilege mask
;  2935     2908  2      !
;  2936     2909  2
;  2937     2910  2          [PBK$K_PRVC, PBK$K_PRVL] :
;  2938     2911  2              BEGIN
;  2939     2912  2              LOCAL
;  2940     2913  2                  PRVMSK : VECTOR [8, BYTE],          ! Full 8 byte mask
;  2941     2914  2                  CTR                                 ! Temp counter
;  2942     2915  2                  ;
;  2943     2916  2
;  2944     2917  2                  CTR = .LEN;                         ! Obtain bytes in mask
;  2945     2918  2                  IF .CTR GTR 8                       ! If the mask is too long
;  2946     2919  2                  THEN                                ! Signal an error
;  2947     2920  2                      SIGNAL_STOP (NCP$_BADMSG, 1, ASCIC ('invalid privilege mask') )
;  2948     2921  2                  ;
;  2949     2922  2                  CH$FILL (0, 8, PRVMSK);             ! Clear the full mask size
;  2950     2923  2                  INCR IDX FROM 0 TO MINU (8, .CTR)   ! Copy the message mask
;  2951     2924  2                  DO
;  2952     2925  2                      PRVMSK [.IDX] = .(..PTR + .IDX) <0, 8, 0>
;  2953     2926  2                  ;
;  2954     2927  2
;  2955     2928  2                  .PTR = ..PTR + .CTR;                ! Step pointer beyond mask
;  2956     2929  2
;  2957     2930  2                  NCP$PRIVBITS (PRVMSK, CTR);         ! Make bits into text entries
;  2958     2931  2                                                      ! Return how many strings
;  2959     2932  2
;  2960     2933  2                  INCRU IDX FROM 1 TO .CTR            ! Add the control string
;  2961     2934  2                  DO                                  ! Entries to print them out
;  2962     2935  4                      BEGIN
;  2963     2936  5                      IF  ( (.IDX MOD 6) EQL 1)       ! Every 6 get a new line
;  2964     2937  4                          AND
;  2965     2938  5                          (.IDX NEQ 1)                ! Except the first
;  2966     2939  4                      THEN
;  2967     2940  4                          ADDSTR ('!/!27* ')          ! A new line and indent
;  2968     2941  4                      ;
;  2969     2942  4                      ADDSTR ('!AC ')                 ! A string pointer
;  2970     2943  4                      END
;  2971     2944  3              END
;  2972     2945  2              ;
```

```
;  2974          2946  2     [OTHERWISE] :
;  2975          2947  3         (
;  2976          2948  3         ADDSTR ('%X''');                    ! Say the data is hex
;  2977          2949  3         INCRU IDX FROM 0 TO .LEN - 1
;  2978          2950  3         DO
;  2979          2951  4             BEGIN                           ! Write the data reversed
;  2980          2952  4             ADDSTR ('!XB');
;  2981          2953  4             ADDFAO (.(..PTR+.IDX) <0, 8, 0>)
;  2982          2954  4             END
;  2983          2955  3         ;
;  2984          2956  3         ADDSTR ('''');
;  2985          2957  3         .PTR = ..PTR + .LEN                 ! Assume the length is correct
;  2986          2958  2         );
;  2987          2959  2
;  2988          2960  2     TES
;  2989          2961  2     ;
;  2990          2962  2
;  2991          2963  2     IF (.PSTRT + .LEN) NEQU ..PTR           ! Check the coded length and
;  2992          2964  2     THEN                                    ! Our decoding
;  2993          2965  2         SIGNAL_STOP (NCP$_BADMSG, 1,        ! They did not match
;  2994          2966  2                 ASCIC ('invalid length in parameter') )
;  2995          2967  2     ;
;  2996          2968  2
;  2997          2969  2     RETURN
;  2998          2970  1     END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

                                                  42  58  21  03  00A99 P.AET:  .ASCII  <3>\!XB\                                  ;
                                                  42  58  21  03  00A9D P.AEU:  .ASCII  <3>\!XB\                                  ;
                                                      2D  01  00AA1 P.AEV:  .ASCII  <1>\-\                                        ;
                                                  42  58  21  03  00AA3 P.AEW:  .ASCII  <3>\!XB\                                  ;
                                                  43  41  21  03  00AA7 P.AEX:  .ASCII  <3>\!AC\                                  ;
                                                  55  35  21  03  00AAB P.AEY:  .ASCII  <3>\!5U\                                  ;
                                                      55  21  02  00AAF P.AEZ:  .ASCII  <2>\!U\                                   ;
                                                      53  21  02  00AB2 P.AFA:  .ASCII  <2>\!S\                                   ;
                                                      58  21  02  00AB5 P.AFB:  .ASCII  <2>\!X\                                   ;
                                                      4F  21  02  00AB8 P.AFC:  .ASCII  <2>\!O\                                   ;
61  72  61  70  20  66  6F  20  68  74  67  6E  65  6C  1B  00ABB P.AFD:  .ASCII  <27>\length of parameter is zero\                  ;
        6F  72  65  7A  20  73  69  20  72  65  74  65  6D  00ACA                                                                    ;
                                                      42  01  00AD7 P.AFE:  .ASCII  <1>\B\                                       ;
                                                      57  01  00AD9 P.AFF:  .ASCII  <1>\W\                                       ;
                                                      4C  01  00ADB P.AFG:  .ASCII  <1>\L\                                       ;
                                                      4C  01  00ADD P.AFH:  .ASCII  <1>\L\                                       ;
                                                  42  58  21  03  00ADF P.AFI:  .ASCII  <3>\!XB\                                  ;
65  6D  61  72  61  70  20  63  69  72  65  6D  75  6E  1A  00AE3 P.AFJ:  .ASCII  <26>\numeric parameter too long\                 ;
            67  6E  6F  6C  20  6F  6F  74  20  72  65  74  00AF2                                                                   ;
69  74  6C  75  6D  20  79  6E  61  6D  20  6F  6F  74  1E  00AFE P.AFK:  .ASCII  <30>\too many multiply coded fields\             ;
64  6C  65  69  66  20  64  65  64  6F  63  20  79  6C  70  00B0D                                                                   ;
                                                          73  00B1C                                                               ;
                                                      56  01  00B1D P.AFL:  .ASCII  <1>\V\                                       ;
                                                      2E  01  00B1F P.AFM:  .ASCII  <1>\.\                                       ;
                                                      2D  01  00B21 P.AFN:  .ASCII  <1>\-\                                       ;
                                                      2D  01  00B23 P.AFO:  .ASCII  <1>\-\                                       ;
69  74  6C  75  6D  20  79  6E  61  6D  20  6F  6F  74  1E  00B25 P.AFP:  .ASCII  <30>\too many multiply coded fields\             ;
```

```
64 6C 65 69 66 20 64 65 64 6F 63 20 79 6C 70 00B34
                                  73 00B43
                              57 55 33 21 04 00B44 P.AFQ:  .ASCII  <4>\!3UW\
                        57 55 21 2E 57 55 32 21 08 00B49 P.AFR:  .ASCII  <8>\!2UW.!UW\
                                    28 20 02 00B52 P.AFS:  .ASCII  <2>\ (\
                                    29 01 00B55 P.AFT:  .ASCII  <1>\)\
                                    3A 01 00B57 P.AFU:  .ASCII  <1>\:\
                                 5A 32 21 03 00B59 P.AFV:  .ASCII  <3>\!2Z\
                                    42 01 00B5D P.AFW:  .ASCII  <1>\B\
                                    57 01 00B5F P.AFX:  .ASCII  <1>\W\
6F 6E 20 65 6D 69 54 20 61 74 6C 65 44 20 17 00B61 P.AFY:  .ASCII  <23>\ Delta Time not decoded\
                  64 65 64 6F 63 65 64 20 74 00B70
                                    2D 01 00B79 P.AFZ:  .ASCII  <1>\-\
                                    20 01 00B7B P.AGA:  .ASCII  <1>\ \
                                    3A 01 00B7D P.AGB:  .ASCII  <1>\:\
                                 5A 32 21 03 00B7F P.AGC:  .ASCII  <3>\!2Z\
                                    42 01 00B83 P.AGD:  .ASCII  <1>\B\
                                    57 01 00B85 P.AGE:  .ASCII  <1>\W\
64 20 74 6F 6E 20 65 6D 69 74 79 61 44 20 14 00B87 P.AGF:  .ASCII  <20>\ Daytime not decoded\
                        64 65 64 6F 63 65 00B96
                                    20 2C 02 00B9C P.AGG:  .ASCII  <2>\, \
                        72 6F 74 75 63 65 78 45 08 00B9F P.AGH:  .ASCII  <8>\Executor\
                           20 65 6E 69 4C 05 00BA8 P.AGI:  .ASCII  <5>\Line \
                        20 74 69 75 63 72 69 43 08 00BAE P.AGJ:  .ASCII  <8>\Circuit \
                           20 65 64 6F 4E 05 00BB7 P.AGK:  .ASCII  <5>\Node \
                              20 65 6C 75 64 6F 4D 07 00BBD P.AGL:  .ASCII  <7>\Module \
                                 3C 39 31 21 04 00BC5 P.AGM:  .ASCII  <4>\!19<\
                           20 65 6E 69 4C 05 00BCA P.AGN:  .ASCII  <5>\Line \
                        20 74 69 75 63 72 69 43 08 00BD0 P.AGO:  .ASCII  <8>\Circuit \
                           20 65 64 6F 4E 05 00BD9 P.AGP:  .ASCII  <5>\Node \
                              57 55 33 21 04 00BDF P.AGQ:  .ASCII  <4>\!3UW\
                        57 55 21 2E 57 55 32 21 08 00BE4 P.AGR:  .ASCII  <8>\!2UW.!UW\
   29 73 65 63 72 75 6F 73 20 6C 6C 41 28 0D 00BED P.AGS:  .ASCII  <13>\(All sources)\
                                    28 20 02 00BFB P.AGT:  .ASCII  <2>\ (\
                                    29 01 00BFE P.AGU:  .ASCII  <1>\)\
                                 20 3E 21 03 00C00 P.AGV:  .ASCII  <3>\!> \
                           2E 57 55 21 20 05 00C04 P.AGW:  .ASCII  <5>\ !UW.\
74 6E 65 76 65 20 64 65 76 72 65 73 65 72 19 00C0A P.AGX:  .ASCII  <25>\reserved event class code\
                  65 64 6F 63 20 73 73 61 6C 63 20 00C19
                              2A 2E 57 55 21 20 06 00C24 P.AGY:  .ASCII  <6>\ !UW.*\
                                 2A 2E 2A 20 04 00C2B P.AGZ:  .ASCII  <4>\ *.*\
                                    20 2C 02 00C30 P.AHA:  .ASCII  <2>\, \
74 6C 75 6D 20 65 76 69 73 72 75 63 65 72 19 00C33 P.AHB:  .ASCII  <25>\recursive multiple fields\
                  73 64 6C 65 69 66 20 65 6C 70 69 00C42
72 6F 77 79 65 6B 20 64 69 6C 61 76 6E 69 14 00C4D P.AHC:  .ASCII  <20>\invalid keyword list\
                           74 73 69 6C 20 64 00C5C
                                    43 41 21 03 00C62 P.AHD:  .ASCII  <3>\!AC\
                     42 55 21 23 20 65 70 79 74 09 00C66 P.AHE:  .ASCII  <9>\type #!UB\
6C 69 76 69 72 70 20 64 69 6C 61 76 6E 69 16 00C70 P.AHF:  .ASCII  <22>\invalid privilege mask\
                        6B 73 61 6D 20 65 67 65 00C7F
                           20 2A 37 32 21 2F 21 07 00C87 P.AHG:  .ASCII  <7>\!/!27* \
                                 20 43 41 21 04 00C8F P.AHH:  .ASCII  <4>\!AC \
                                    27 58 25 03 00C94 P.AHI:  .ASCII  <3>\%X'\
                                    42 58 21 03 00C98 P.AHJ:  .ASCII  <3>\!XB\
                                    27 01 00C9C P.AHK:  .ASCII  <1>\'\
68 74 67 6E 65 6C 20 64 69 6C 61 76 6E 69 1B 00C9E P.AHL:  .ASCII  <27>\invalid length in parameter\
   72 65 74 65 6D 61 72 61 70 20 6E 69 20 00CAD
```

```
                                                .PSECT  $CODE$,NOWRT,2

                                    0FFC 00000 NCP$PARSEPARM:
                                                .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11            ; 2087
                5B 00000000V   00    9E 00002           MOVAB   NCP$ADDSTR, R11
                5A 00000000'   00    9E 00009           MOVAB   NCP$GQ_CTRDSC, R10
                59 00000000'   00    9E 00010           MOVAB   P.AET, R9
                5E                10  C2 00017           SUBL2   #16, SP
                         04       AC  95 0001A           TSTB    DTY                                    ; 2167
                                  03  18 0001D           BGEQ    1$
                               01B9  31 0001F           BRW     29$
                1C       0C       AC  D1 00022 1$:       CMPL    TYP, #28                               ; 2171
                                  31  12 00026           BNEQ    4$
                50       08       BC  D0 00028           MOVL    @PTR, R0                               ; 2174
                56                60  9A 0002C           MOVZBL  (R0), LEN
                         08       BC  D6 0002F           INCL    @PTR                                   ; 2175
                53       FF       A6  9E 00032           MOVAB   -1(R6), R3                             ; 2177
                                  52  D4 00036           CLRL    IDX
                                  18  11 00038           BRB     3$
                              0600  8F  BB 0003A 2$:       PUSHR   #^M<R9,R10>                            ; 2180
                6B                02  FB 0003E           CALLS   #2, NCP$ADDSTR
                50       08       BC  D0 00041           MOVL    @PTR, R0                               ; 2181
                7E                6240 9A 00045          MOVZBL  (IDX)[R0], -(SP)
    00000000V   00                01  FB 00049           CALLS   #1, NCP$ADDFAO
                                  52  D6 00050           INCL    IDX                                    ; 2177
                53                52  D1 00052 3$:       CMPL    IDX, R3
                                  E3  1B 00055           BLEQU   2$
                                  51  11 00057           BRB     7$                                     ; 2184
                1F       0C       AC  D1 00059 4$:       CMPL    TYP, #31                               ; 2192
                                  4E  12 0005D           BNEQ    8$
                50       08       BC  D0 0005F           MOVL    @PTR, R0                               ; 2195
                56                60  9A 00063           MOVZBL  (R0), LEN
                         08       BC  D6 00066           INCL    @PTR                                   ; 2196
                                  5A  DD 00069           PUSHL   R10                                    ; 2198
                         04       A9  9F 0006B           PUSHAB  P.AEU
                6B                02  FB 0006E           CALLS   #2, NCP$ADDSTR
                52       08       BC  D0 00071           MOVL    @PTR, R2                               ; 2199
                7E                62  9A 00075           MOVZBL  (R2), -(SP)
    00000000V   00                01  FB 00078           CALLS   #1, NCP$ADDFAO
                53       FF       A6  9E 0007F           MOVAB   -1(R6), R3                             ; 2201
                54                01  D0 00083           MOVL    #1, IDX
                                  1D  11 00086           BRB     6$
                                  5A  DD 00088 5$:       PUSHL   R10                                    ; 2203
                         08       A9  9F 0008A           PUSHAB  P.AEV
                6B                02  FB 0008D           CALLS   #2, NCP$ADDSTR
                                  5A  DD 00090           PUSHL   R10                                    ; 2204
                         0A       A9  9F 00092           PUSHAB  P.AEW
                6B                02  FB 00095           CALLS   #2, NCP$ADDSTR
                7E                6442 9A 00098          MOVZBL  (IDX)[R2], -(SP)                       ; 2205
    00000000V   00                01  FB 0009C           CALLS   #1, NCP$ADDFAO
                                  54  D6 000A3           INCL    IDX
                53                54  D1 000A5 6$:       CMPL    IDX, R3
                                  DE  1B 000A8           BLEQU   5$
                               0129  31 000AA 7$:       BRW     28$                                    ; 2208
```

```
                27      04  AC              06  E1 000AD 8$:    BBC     #6, DTY, 9$                     ;2211
                                            5A  DD 000B2        PUSHL   R10                             ;2214
                                        0E  A9  9F 000B4        PUSHAB  P.AEX                           ;
                                        6B  02  FB 000B7        CALLS   #2, NCP$ADDSTR                  ;
                                        52  08  BC D0 000BA     MOVL    @PTR, R2                        ;2215
                                            52  DD 000BE      PUSHL   R2                              ;
                00000000V               00      01  FB 000C0   CALLS   #1, NCP$ADDFAO                  ;
        50              62              07      00  EF 000C7   EXTZV   #0, #7, (R2), R0               ;2220
                                        62      50  90 000CC   MOVB    R0, (R2)                        ;
                                        50      62  9A 000CF   MOVZBL  (R2), R0                       ;2221
                                08  BC      01 A042 9E 000D2   MOVAB   1(R0)[R2], @PTR                 ;
                                            04  000D8 RET                                             ;2225
        56      04  AC              04          00  EF 000D9 9$:    EXTZV   #0, #4, DTY, LEN            ;2226
                                            0A  12  000DF       BNEQ    10$                             ;2227
                                        50  08  AC  D0 000E1    MOVL    PTR, R0                         ;2230
                                        56  00  B0  9A 000E5    MOVZBL  @0(R0), LEN                     ;
                                            60  D6  000E9       INCL    (R0)                            ;2231
                                        04          56  D1 000EB 10$:   CMPL    LEN, #4                 ;2234
                                                    4A  14 000EE    BGTR    16$                         ;
        50      04  AC              02          04  EF 000F0   EXTZV   #4, #2, DTY, R0                 ;2237
                                                1D  12 000F6    BNEQ    12$                             ;2240
                                    0F  00000000'   00  E9 000F8    BLBC    NCP$B_COLHEAD, 11$          ;2242
                                        00000000'   00  D5 000FF    TSTL    NCP$A_COLTBL                ;2243
                                                07  13 00105    BEQL    11$                             ;
                                            5A  DD 00107        PUSHL   R10                             ;2245
                                        12  A9  9F 00109        PUSHAB  P.AEY                           ;
                                                29  11 0010C    BRB     15$                             ;
                                            5A  DD 0010E 11$:   PUSHL   R10                             ;2247
                                        16  A9  9F 00110        PUSHAB  P.AEZ                           ;
                                                22  11 00113    BRB     15$                             ;
    01                                          50  D1 00115 12$:   CMPL    R0, #1                  ;2249
                                                07  12 00118    BNEQ    13$                             ;
                                            5A  DD 0011A        PUSHL   R10                             ;
                                        19  A9  9F 0011C        PUSHAB  P.AFA                           ;
                                                16  11 0011F    BRB     15$                             ;
    02                                          50  D1 00121 13$:   CMPL    R0, #2                  ;2250
                                                07  12 00124    BNEQ    14$                             ;
                                            5A  DD 00126        PUSHL   R10                             ;
                                        1C  A9  9F 00128        PUSHAB  P.AFB                           ;
                                                0A  11 0012B    BRB     15$                             ;
    03                                          50  D1 0012D 14$:   CMPL    R0, #3                  ;2251
                                                08  12 00130    BNEQ    16$                             ;
                                            5A  DD 00132        PUSHL   R10                             ;
                                        1F  A9  9F 00134        PUSHAB  P.AFC                           ;
    6B                                  02  FB 00137 15$:   CALLS   #2, NCP$ADDSTR                   ;
                                            56  D5 0013A 16$:   TSTL    LEN                             ;2259
                                            06  12 0013C        BNEQ    17$                             ;
                                        22  A9  9F 0013E        PUSHAB  P.AFD                           ;2262
                                    0083    31  00141        BRW     27$                             ;2261
    01                                      56  D1 00144 17$:   CMPL    LEN, #1                     ;2265
                                            07  12 00147        BNEQ    18$                             ;
                                            5A  DD 00149        PUSHL   R10                             ;2267
                                        3E  A9  9F 0014B        PUSHAB  P.AFE                           ;
                                            2E  11 0014E        BRB     21$                             ;
    02                                      56  D1 00150 18$:   CMPL    LEN, #2                     ;2271
                                            07  12 00153        BNEQ    19$                             ;
                                            5A  DD 00155        PUSHL   R10                             ;2273
```

```
                                  40  A9 9F 00157          PUSHAB  P.AFF                          ;
                                      22 11 0015A          BRB     21$                            ;
                             03       56 D1 0015C  19$:    CMPL    LEN, #3                        ; 2277
                                      13 12 0015F          BNEQ    20$                            ;
                                      5A DD 00161          PUSHL   R10                            ; 2279
                                  42  A9 9F 00163          PUSHAB  P.AFG                          ;
                             6B       02 FB 00166          CALLS   #2, NCP$ADDSTR                 ;
                             50   08  BC D0 00169          MOVL    @PTR, R0                       ; 2280
       7E          60             18  00 EF 0016D          EXTZV   #0, #24, (R0), -(SP)           ;
                                      13 11 00172          BRB     22$                            ;
                             04       56 D1 00174  20$:    CMPL    LEN, #4                        ; 2283
                                      17 12 00177          BNEQ    23$                            ;
                                      5A DD 00179          PUSHL   R10                            ; 2285
                                  44  A9 9F 0017B          PUSHAB  P.AFH                          ;
                             6B       02 FB 0017E  21$:    CALLS   #2, NCP$ADDSTR                 ;
                             50   08  BC D0 00181          MOVL    @PTR, R0                       ; 2286
                                      60 DD 00185          PUSHL   (R0)                           ;
                   00000000V 00       01 FB 00187  22$:    CALLS   #1, NCP$ADDFAO                 ;
                                      46 11 0018E          BRB     28$                            ;
                             05       56 D1 00190  23$:    CMPL    LEN, #5                        ; 2289
                                      2F 19 00193          BLSS    26$                            ;
                             20       56 D1 00195          CMPL    LEN, #32                       ;
                                      2A 14 00198          BGTR    26$                            ;
                             52       01 D0 0019A          MOVL    #1, IDX                        ; 2295
                                      1E 11 0019D          BRB     25$                            ;
                                      5A DD 0019F  24$:    PUSHL   R10                            ; 2294
                                  46  A9 9F 001A1          PUSHAB  P.AFI                          ;
                             6B       02 FB 001A4          CALLS   #2, NCP$ADDSTR                 ;
                             50   08  BC D0 001A7          MOVL    @PTR, R0                       ; 2295
                             50       56 C0 001AB          ADDL2   LEN, R0                        ;
                             50       52 C2 001AE          SUBL2   IDX, R0                        ;
                             7E       60 9A 001B1          MOVZBL  (R0), -(SP)                    ;
                   00000000V 00       01 FB 001B4          CALLS   #1, NCP$ADDFAO                 ;
                                      52 D6 001BB          INCL    IDX                            ;
                             56       52 D1 001BD  25$:    CMPL    IDX, LEN                       ;
                                      DD 1B 001C0          BLEQU   24$                            ;
                                      12 11 001C2          BRB     28$                            ; 2290
                                  4A  A9 9F 001C4  26$:    PUSHAB  P.AFJ                          ; 2303
                                      01 DD 001C7  27$:    PUSHL   #1                             ; 2302
                           00000000G  8F DD 001C9          PUSHL   #NCP$_BADMSG                   ;
                   00000000G 00       03 FB 001CF          CALLS   #3, LIB$STOP                   ;
                          08 BC       56 C0 001D6  28$:    ADDL2   LEN, @PTR                      ; 2308
                                      04 001DA             RET                                    ; 2225
              03      04  AC          06 E0 001DB  29$:    BBS     #6, DTY, 30$                   ; 2324
                                    0583 31 001E0          BRW     118$                           ;
       53        04  AC       06      00 EF 001E3  30$:    EXTZV   #0, #6, DTY, MULCTR            ; 2327
                             0F       53 D1 001E9          CMPL    MULCTR, #15                    ; 2328
                                      12 15 001EC          BLEQ    31$                            ;
                                  65  A9 9F 001EE          PUSHAB  P.AFK                          ; 2331
                                      01 DD 001F1          PUSHL   #1                             ; 2330
                           00000000G  8F DD 001F3          PUSHL   #NCP$_BADMSG                   ;
                   00000000G 00       03 FB 001F9          CALLS   #3, LIB$STOP                   ;
                             52   0C  AC D0 00200  31$:    MOVL    TYP, R2                        ; 2342
                             0A       52 D1 00204          CMPL    R2, #10                        ; 2345
                                      36 12 00207          BNEQ    35$                            ;
                                      5A DD 00209          PUSHL   R10                            ; 2347
                                0084  C9 9F 0020B          PUSHAB  P.AFL                          ;
```

```
                    6B              02  FB 0020F          CALLS   #2, NCP$ADDSTR                          ;
                                    54  D4 00212          CLRL    IDX                                    ; 2358
                                    23  11 00214          BRB     34$                                    ;
                    01              54  D1 00216 32$:     CMPL    IDX, #1                                ; 2351
                                    09  13 00219          BEQL    33$                                    ;
                                    5A  DD 0021B          PUSHL   R10                                    ; 2353
                  0086              C9  9F 0021D          PUSHAB  P.AFM                                  ;
                    6B              02  FB 00221          CALLS   #2, NCP$ADDSTR                         ;
                    7E              17  7D 00224 33$:     MOVQ    #23, -(SP)                             ; 2356
                      08            AC  DD 00227          PUSHL   PTR                                    ; 2358
                    50  08          BC  D0 0022A          MOVL    @PTR, R0                               ; 2357
                    7E              60  9A 0022E          MOVZBL  (R0), -(SP)                            ;
                      08            BC  D6 00231          INCL    @PTR                                   ;
                FDC7  CF            04  FB 00234          CALLS   #4, NCP$PARSEPARM                      ;
            D9      54              53  F3 00239 34$:     AOBLEQ  MULCTR, IDX, 32$                       ; 2356
                                    62  11 0023D          BRB     43$                                    ; 2364
                    18              52  D1 0023F 35$:     CMPL    R2, #24                                ; 2371
                                    2D  12 00242          BNEQ    39$                                    ;
                                    54  D4 00244          CLRL    IDX                                    ; 2381
                                    23  11 00246          BRB     38$                                    ;
                    01              54  D1 00248 36$:     CMPL    IDX, #1                                ; 2376
                                    09  13 0024B          BEQL    37$                                    ;
                                    5A  DD 0024D          PUSHL   R10                                    ; 2378
                  0088              C9  9F 0024F          PUSHAB  P.AFN                                  ;
                    6B              02  FB 00253          CALLS   #2, NCP$ADDSTR                         ;
                    7E              17  7D 00256 37$:     MOVQ    #23, -(SP)                             ; 2379
                      08            AC  DD 00259          PUSHL   PTR                                    ; 2381
                    50  08          BC  D0 0025C          MOVL    @PTR, R0                               ; 2380
                    7E              60  9A 00260          MOVZBL  (R0), -(SP)                            ;
                      08            BC  D6 00263          INCL    @PTR                                   ;
                FD95  CF            04  FB 00266          CALLS   #4, NCP$PARSEPARM                      ;
            D9      54              53  F3 0026B 38$:     AOBLEQ  MULCTR, IDX, 36$                       ; 2373
                                    30  11 0026F          BRB     43$                                    ; 2385
                    1B              52  D1 00271 39$:     CMPL    R2, #27                                ; 2394
                                    2E  12 00274          BNEQ    44$                                    ;
                                    54  D4 00276          CLRL    IDX                                    ; 2404
                                    23  11 00278          BRB     42$                                    ;
                    01              54  D1 0027A 40$:     CMPL    IDX, #1                                ; 2399
                                    09  13 0027D          BEQL    41$                                    ;
                                    5A  DD 0027F          PUSHL   R10                                    ; 2401
                  008A              C9  9F 00281          PUSHAB  P.AFO                                  ;
                    6B              02  FB 00285          CALLS   #2, NCP$ADDSTR                         ;
                    7E              17  7D 00288 41$:     MOVQ    #23, -(SP)                             ; 2402
                      08            AC  DD 0028B          PUSHL   PTR                                    ; 2404
                    50  08          BC  D0 0028E          MOVL    @PTR, R0                               ; 2403
                    7E              60  9A 00292          MOVZBL  (R0), -(SP)                            ;
                      08            BC  D6 00295          INCL    @PTR                                   ;
                FD63  CF            04  FB 00298          CALLS   #4, NCP$PARSEPARM                      ;
            D9      54              53  F3 0029D 42$:     AOBLEQ  MULCTR, IDX, 40$                       ; 2396
                                01FA 31  00A1 43$:     BRW     72$                                    ; 2408
                    07              52  D1 002A4 44$:     CMPL    R2, #7                                 ; 2415
                                    03  13 002A7          BEQL    45$                                    ;
                                0080 31  002A9          BRW     49$                                    ;
                    02              53  D1 002AC 45$:     CMPL    MULCTR, #2                             ; 2421
                                    13  15 002AF          BLEQ    46$                                    ;
                  008C              C9  9F 002B1          PUSHAB  P.AFP                                  ; 2424
                                    01  DD 002B5          PUSHL   #1                                     ; 2423
```

```
                                00000000G  8F  DD 002B7          PUSHL   #NCP$_BADMSG                                    ;
                     00000000G  00         03  FB 002BD          CALLS   #3, LIB$STOP                                    ;
                                52         08  AC  D0 002C4 46$:  MOVL    PTR, R2                                     ; 2427
                                                62  D6 002C8          INCL    (R2)                                        ;
                                54         00  B2  3C 002CA          MOVZWL  @0(R2), AREA_ADDR                           ; 2428
                55              54         06  0A  EF 002CE          EXTZV   #10, #6, AREA_ADDR, AREA                    ; 2429
                                                0D  12 002D3          BNEQ    47$                                         ; 2431
                                                5A  DD 002D5          PUSHL   R10                                         ; 2434
                                           00AB  C9  9F 002D7          PUSHAB  P.AFQ                                       ;
                                6B              02  FB 002DB          CALLS   #2, NCP$ADDSTR                              ;
                                                54  DD 002DE          PUSHL   AREA_ADDR                                   ; 2435
                                                17  11 002E0          BRB     48$                                         ;
                                                5A  DD 002E2 47$:     PUSHL   R10                                         ; 2439
                                           00B0  C9  9F 002E4          PUSHAB  P.AFR                                       ;
                                6B              02  FB 002E8          CALLS   #2, NCP$ADDSTR                              ;
                                                55  DD 002EB          PUSHL   AREA                                        ; 2440
                     00000000V  00         01  FB 002ED          CALLS   #1, NCP$ADDFAO                              ;
                7E              54         0A  00  EF 002F4          EXTZV   #0, #10, AREA_ADDR, -(SP)                   ; 2441
                     00000000V  00         01  FB 002F9 48$:     CALLS   #1, NCP$ADDFAO                              ;
                                62         02  C0 00300          ADDL2   #2, (R2)                                    ; 2444
                                02         53  D1 00303          CMPL    MULCTR, #2                                  ; 2446
                                                99  12 00306          BNEQ    43$                                         ;
                                                5A  DD 00308          PUSHL   R10                                         ; 2449
                                           00B9  C9  9F 0030A          PUSHAB  P.AFS                                       ;
                                6B              02  FB 0030E          CALLS   #2, NCP$ADDSTR                              ;
                                7E              07  7D 00311          MOVQ    #7, -(SP)                                   ; 2451
                                                52  DD 00314          PUSHL   R2                                          ; 2453
                                7E         00  B2  9A 00316          MOVZBL  @0(R2), -(SP)                               ; 2452
                                                62  D6 0031A          INCL    (R2)                                        ;
                          FCDF  CF              04  FB 0031C          CALLS   #4, NCP$PARSEPARM                           ;
                                                5A  DD 00321          PUSHL   R10                                         ; 2457
                                           00BC  C9  9F 00323          PUSHAB  P.AFT                                       ;
                                6B              02  FB 00327          CALLS   #2, NCP$ADDSTR                              ;
                                                70  11 0032A          BRB     57$                                         ; 2460
                                20              52  D1 0032C 49$:     CMPL    R2, #32                                     ; 2466
                                                6E  12 0032F          BNEQ    58$                                         ;
                                54         08  AC  D0 00331          MOVL    PTR, R4                                     ; 2479
                                                55  D4 00335          CLRL    IDX                                         ;
                                                5F  11 00337          BRB     56$                                         ;
                                01              55  D1 00339 50$:     CMPL    IDX, #1                                     ; 2475
                                                09  13 0033C          BEQL    51$                                         ;
                                                5A  DD 0033E          PUSHL   R10                                         ; 2477
                                           00BE  C9  9F 00340          PUSHAB  P.AFU                                       ;
                                6B              02  FB 00344          CALLS   #2, NCP$ADDSTR                              ;
                                50         00  B4  90 00347 51$:     MOVB    @0(R4), L_DTY                               ; 2479
                                                64  D6 0034B          INCL    (R4)                                        ;
                52              50         04  00  EF 0034D          EXTZV   #0, #4, L_DTY, LEN                          ; 2480
                                                5A  DD 00352          PUSHL   R10                                         ; 2481
                                           00C0  C9  9F 00354          PUSHAB  P.AFV                                       ;
                                6B              02  FB 00358          CALLS   #2, NCP$ADDSTR                              ;
                                01              52  D1 0035B          CMPL    LEN, #1                                     ; 2484
                                                08  12 0035E          BNEQ    52$                                         ;
                                                5A  DD 00360          PUSHL   R10                                         ; 2486
                                           00C4  C9  9F 00362          PUSHAB  P.AFW                                       ;
                                                0B  11 00366          BRB     53$                                         ;
                                02              52  D1 00368 52$:     CMPL    LEN, #2                                     ; 2489
                                                15  12 0036B          BNEQ    54$                                         ;
```

```
                                    5A  DD 0036D          PUSHL   R10                                             ; 2491
                                00C6    C9 9F 0036F          PUSHAB  P.AFX
                                    6B  02 FB 00373 53$:     CALLS   #2, NCP$ADDSTR
                                    00  B4 DD 00376          PUSHL   @0(R4)                                          ; 2492
                          00000000V 00  01 FB 00379          CALLS   #1, NCP$ADDFAO
                                    13  11 00380          BRB     55$                                             ; 2482
                                00C8    C9 9F 00382 54$:     PUSHAB  P.AFY                                           ; 2495
                                    01  DD 00386          PUSHL   #1
                            00000000G   8F DD 00388          PUSHL   #NCP$_BADMSG
                          00000000G 00  03 FB 0038E          CALLS   #3, LIB$STOP
                                    64  52 C0 00395 55$:     ADDL2   LEN, (R4)                                       ; 2498
                                9D  55  53 F3 00398 56$:     AOBLEQ  MULCTR, IDX, 50$                                ; 2468
                                    00FF 31 00039C 57$:      BRW     72$                                             ; 2501
                                    21  52 D1 00039F 58$:    CMPL    R2, #33                                         ; 2510
                                        03 13 003A2          BEQL    59$
                                    00C5 31 003A4            BRW     68$
                                    58 00000000' 00 D0 003A7 59$:    MOVL    NCP$A_COLTBL, COLTBL                    ; 2515
                                    00000000' 00 D4 003AE            CLRL    NCP$A_COLTBL                            ; 2516
                                        10  AC DD 003B4          PUSHL   LST                                             ; 2523
                                        17  DD 003B7          PUSHL   #23                                             ; 2519
                                    52  08  AC D0 003B9          MOVL    PTR, R2                                         ; 2521
                                        52  DD 003BD          PUSHL   R2
                                    7E  00  B2 9A 003BF          MOVZBL  @0(R2), -(SP)                                   ; 2520
                                        62  D6 003C3          INCL    (R2)
                                FC36    CF  04 FB 003C5          CALLS   #4, NCP$PARSEPARM
                                        5A  DD 003CA          PUSHL   R10                                             ; 2524
                                00E0    C9 9F 003CC          PUSHAB  P.AFZ
                                    6B  02 FB 003D0          CALLS   #2, NCP$ADDSTR
                                        53  D7 003D3          DECL    MULCTR                                          ; 2525
                                    55  00  B2 9A 003D5          MOVZBL  @0(R2), NDTY                                    ; 2527
                                        62  D6 003D9          INCL    (R2)                                            ; 2528
                                        55  95 003DB          TSTB    NDTY                                            ; 2529
                                        1D  18 003DD          BGEQ    60$
                                19      55  06 E0 003DF          BBS     #6, NDTY, 60$                               ; 2530
                                        10  AC DD 003E3          PUSHL   LST                                             ; 2537
                                        01  DD 003E6          PUSHL   #1                                              ; 2533
                                        52  DD 003E8          PUSHL   R2                                              ; 2535
                                        55  DD 003EA          PUSHL   NDTY                                            ; 2534
                                FC0F    CF  04 FB 003EC          CALLS   #4, NCP$PARSEPARM
                                        53  D7 003F1          DECL    MULCTR                                          ; 2539
                                        5A  DD 003F3          PUSHL   R10                                             ; 2540
                                00E2    C9 9F 003F5          PUSHAB  P.AGA
                                    6B  02 FB 003F9          CALLS   #2, NCP$ADDSTR
                                        57  D4 003FC 60$:     CLRL    IDX                                             ; 2543
                                        5F  11 003FE          BRB     67$
                                    01  57 D1 00400 61$:     CMPL    IDX, #1                                         ; 2550
                                        09  13 00403          BEQL    62$
                                        5A  DD 00405          PUSHL   R10                                             ; 2552
                                00E4    C9 9F 00407          PUSHAB  P.AGB
                                    6B  02 FB 0040B          CALLS   #2, NCP$ADDSTR
                                    50  00  B2 90 0040E 62$:     MOVB    @0(R2), L_DTY                               ; 2554
                                        62  D6 00412          INCL    (R2)
                            54      50  04 00 EF 00414          EXTZV   #0, #4, L_DTY, LEN                          ; 2555
                                        5A  DD 00419          PUSHL   R10                                             ; 2556
                                00E6    C9 9F 0041B          PUSHAB  P.AGC
                                    6B  02 FB 0041F          CALLS   #2, NCP$ADDSTR
                                    01  54 D1 00422          CMPL    LEN, #1                                         ; 2559
```

```
                                        08  12 00425          BNEQ    63$                                     ;
                                        5A  DD 00427          PUSHL   R10                                     ; 2561
                                  00EA  C9  9F 00429          PUSHAB  P.AGD                                   ;
                                        0B  11 0042D          BRB     64$                                     ;
                               02       54  D1 0042F 63$:     CMPL    LEN, #2                                 ; 2564
                                        15  12 00432          BNEQ    65$                                     ;
                                        5A  DD 00434          PUSHL   R10                                     ; 2566
                                  00EC  C9  9F 00436          PUSHAB  P.AGE                                   ;
                               6B       02  FB 0043A 64$:     CALLS   #2, NCP$ADDSTR                          ;
                                    00  B2  DD 0043D          PUSHL   @0(R2)                                  ; 2567
                  00000000V    00       01  FB 00440          CALLS   #1, NCP$ADDFAO                          ;
                                        13  11 00447          BRB     66$                                     ; 2557
                                  00EE  C9  9F 00449 65$:     PUSHAB  P.AGF                                   ; 2570
                                        01  DD 0044D          PUSHL   #1                                      ;
                             00000000G  8F  DD 0044F          PUSHL   #NCP$_BADMSG                            ;
                  00000000G    00       03  FB 00455          CALLS   #3, LIB$STOP                            ;
                               62       54  C0 0045C 66$:     ADDL2   LEN, (R2)                               ; 2573
                            9D 57       53  F3 0045F 67$:     AOBLEQ  MULCTR, IDX, 61$                        ; 2543
                  00000000'    00       58  D0 00463          MOVL    COLTBL, NCP$A_COLTBL                    ; 2576
                                        32  11 0046A          BRB     72$                                     ; 2577
                               22       52  D1 0046C 68$:     CMPL    R2, #34                                 ; 2585
                                        32  12 0046F          BNEQ    74$                                     ;
                                        52  D4 00471          CLRL    IDX                                     ; 2595
                                        25  11 00473          BRB     71$                                     ;
                               01       52  D1 00475 69$:     CMPL    IDX, #1                                 ; 2590
                                        09  13 00478          BEQL    70$                                     ;
                                        5A  DD 0047A          PUSHL   R10                                     ; 2592
                                  0103  C9  9F 0047C          PUSHAB  P.AGG                                   ;
                               6B       02  FB 00480          CALLS   #2, NCP$ADDSTR                          ;
                                    10  AC  DD 00483 70$:     PUSHL   LST                                     ; 2597
                                        01  DD 00486          PUSHL   #1                                      ; 2593
                                    08  AC  DD 00488          PUSHL   PTR                                     ; 2595
                               50       08  BC  D0 0048B          MOVL    @PTR, R0                            ; 2594
                               7E       60  9A 0048F          MOVZBL  (R0), -(SP)                             ;
                                    08  BC  D6 00492          INCL    @PTR                                    ;
                         FB66  CF       04  FB 00495          CALLS   #4, NCP$PARSEPARM                       ;
                      D7       52       53  F3 0049A 71$:     AOBLEQ  MULCTR, IDX, 69$                        ; 2587
                                        53  D4 0049E 72$:     CLRL    MULCTR                                  ; 2599
                                  027A  31  00 004A0 73$:     BRW     113$                                    ; 2342
                               16       52  D1 004A3 74$:     CMPL    R2, #22                                 ; 2606
                                        03  13 004A6          BEQL    75$                                     ;
                                  00B2  31  00 004A8          BRW     86$                                     ;
                               54       08  AC  D0 004AB 75$:     MOVL    PTR, R4                             ; 2608
                               50       64  D0 004AF          MOVL    (R4), R0                                ;
                               55       60  9A 004B2          MOVZBL  (R0), NDTY                              ;
                                        55  95 004B5          TSTB    NDTY                                    ; 2609
                                        E7  18 004B7          BGEQ    73$                                     ;
                      E3       55       06  E0 004B9          BBS     #6, NDTY, 73$                           ; 2610
                               57       01  A0 9A 004BD          MOVZBL  1(R0), SOURTYP                      ; 2613
                               64       02  C0 004C1          ADDL2   #2, (R4)                                ; 2614
                                        53  D7 004C4          DECL    MULCTR                                  ; 2616
                                        6B  15 004C6          BLEQ    83$                                     ; 2617
                               50       64  D0 004C8          MOVL    (R4), R0                                ; 2620
                               55       60  9A 004CB          MOVZBL  (R0), NDTY                              ;
                                        57  D5 004CE          TSTL    SOURTYP                                 ; 2621
                                        1B  12 004D0          BNEQ    76$                                     ;
                                        55  95 004D2          TSTB    NDTY                                    ; 2622
```

```
                                 17   19 004D4           BLSS    76$                        ;
        13          55           06   E0 004D6           BBS     #6, NDTY, 76$              ; 2623
                            01   A0   B5 004DA           TSTW    1(R0)                      ; 2624
                                 0E   12 004DD           BNEQ    76$                        ;
                                 5A   DD 004DF           PUSHL   R10                        ; 2627
                          0106   C9   9F 004E1           PUSHAB  P.AGH                      ;
                    6B           02   FB 004E5           CALLS   #2, NCP$ADDSTR             ;
                    64           03   C0 004E8           ADDL2   #3, (R4)                   ; 2628
                                 44   11 004EB           BRB     82$                        ; 2621
                    01           57   D1 004ED 76$:      CMPL    SOURTYP, #1                ; 2635
                                 08   12 004F0           BNEQ    77$                        ;
                                 5A   DD 004F2           PUSHL   R10                        ;
                          010F   C9   9F 004F4           PUSHAB  P.AGI                      ;
                                 24   11 004F8           BRB     80$                        ;
                    03           57   D1 004FA 77$:      CMPL    SOURTYP, #3                ; 2636
                                 08   12 004FD           BNEQ    78$                        ;
                                 5A   DD 004FF           PUSHL   R10                        ;
                          0115   C9   9F 00501           PUSHAB  P.AGJ                      ;
                                 17   11 00505           BRB     80$                        ;
                                 57   D5 00507 78$:      TSTL    SOURTYP                    ; 2637
                                 08   12 00509           BNEQ    79$                        ;
                                 5A   DD 0050B           PUSHL   R10                        ;
                          011E   C9   9F 0050D           PUSHAB  P.AGK                      ;
                                 0B   11 00511           BRB     80$                        ;
                    04           57   D1 00513 79$:      CMPL    SOURTYP, #4                ; 2638
                                 09   12 00516           BNEQ    81$                        ;
                                 5A   DD 00518           PUSHL   R10                        ;
                          0124   C9   9F 0051A           PUSHAB  P.AGL                      ;
                    6B           02   FB 0051E 80$:      CALLS   #2, NCP$ADDSTR             ;
                    7E           17   7D 00521 81$:      MOVQ    #23, -(SP)                 ; 2640
                                 54   DD 00524           PUSHL   R4                         ; 2642
                    7E      00   B4   9A 00526           MOVZBL  @0(R4), -(SP)              ; 2641
                                 64   D6 0052A           INCL    (R4)                       ;
              FACF  CF           04   FB 0052C           CALLS   #4, NCP$PARSEPARM          ;
                                 53   D7 00531 82$:      DECL    MULCTR                     ; 2646
                    55      00   B4   9A 00533 83$:      MOVZBL  @0(R4), NDTY               ; 2649
                                 57   D5 00537           TSTL    SOURTYP                    ; 2650
                                 04   12 00539           BNEQ    84$                        ;
                                 53   D5 0053B           TSTL    MULCTR                     ; 2651
                                 03   14 0053D           BGTR    85$                        ;
                               01DB   31 0053F 84$:      BRW     113$                       ;
        F9          55           06   E1 00542 85$:      BBC     #6, NDTY, 84$              ; 2652
                                 55   95 00546           TSTB    NDTY                       ; 2653
                                 F5   19 00548           BLSS    84$                        ;
                    7E           07   7D 0054A           MOVQ    #7, -(SP)                  ; 2656
                                 54   DD 0054D           PUSHL   R4                         ; 2658
                    7E      00   B4   9A 0054F           MOVZBL  @0(R4), -(SP)              ; 2657
                                 64   D6 00553           INCL    (R4)                       ;
              FAA6  CF           04   FB 00555           CALLS   #4, NCP$PARSEPARM          ;
                               017E   31 0055A           BRW     106$                       ; 2661
                    0E           52   D1 0055D 86$:      CMPL    R2, #14                    ; 2673
                                 05   19 00560           BLSS    87$                        ;
                    13           52   D1 00562           CMPL    R2, #19                    ;
                                 05   15 00565           BLEQ    88$                        ;
                    1A           52   D1 00567 87$:      CMPL    R2, #26                    ;
                                 D3   12 0056A           BNEQ    84$                        ;
                    52      08   AC   D0 0056C 88$:      MOVL    PTR, R2                    ; 2675
```

```
                                50              62  D0 00570          MOVL    (R2), R0                                        ;
                                55              60  9A 00573          MOVZBL  (R0), NDTY                                      ;
                                                55  95 00576          TSTB    NDTY                                            ; 2676
                                                03  19 00578          BLSS    90$                                             ;
                                              0110  31 0057A 89$:       BRW     105$                                            ;
                             F9                 55  E0 0057D 90$:       BBS     #6, NDTY, 89$                                   ; 2678
                                                57  A0 00581          MOVZBL  1(R0), SOURTYP                                  ; 2681
                                    00000000'   00  D5 00585          TSTL    NCP$A_COLTBL                                    ; 2683
                                                0E  13 00588          BEQL    91$                                             ;
                                                5A  DD 0058D          PUSHL   R10                                             ; 2686
                                          012C  C9  9F 0058F          PUSHAB  P.AGM                                           ;
                                50              6B  02 FB 00593          CALLS   #2, NCP$ADDSTR                                  ;
                                50              62  02 C0 00596          ADDL2   #2, (R2)                                        ; 2687
                                                15  11 00599          BRB     92$                                             ;
                                    00000000G   00  9F 0059B 91$:       PUSHAB  NCP$GA_TBL_EVESOUR                              ; 2691
                                                01  DD 005A1          PUSHL   #1                                              ;
                                                52  DD 005A3          PUSHL   R2                                              ; 2693
                                50              7E  B2 9A 005A5          MOVZBL  @0(R2), -(SP)                                   ; 2692
                                                62  D6 005A9          INCL    (R2)                                            ;
                                          FA50  CF  04 FB 005AB          CALLS   #4, NCP$PARSEPARM                               ;
                                                53  D7 005B0 92$:       DECL    MULCTR                                          ; 2698
                                  000000FF      8F  57 D1 005B2          CMPL    SOURTYP, #255                                   ; 2699
                                                02  13 005B9          BEQL    93$                                             ;
                                                53  D5 005BB          TSTL    MULCTR                                          ; 2701
                                                7C  15 005BD 93$:       BLEQ    102$                                            ;
                                50              01  57 D1 005BF          CMPL    SOURTYP, #1                                     ; 2707
                                                08  12 005C2          BNEQ    94$                                             ;
                                                5A  DD 005C4          PUSHL   R10                                             ;
                                          0131  C9  9F 005C6          PUSHAB  P.AGN                                           ;
                                                17  11 005CA          BRB     96$                                             ;
                                50              03  57 D1 005CC 94$:       CMPL    SOURTYP, #3                                     ; 2708
                                                08  12 005CF          BNEQ    95$                                             ;
                                                5A  DD 005D1          PUSHL   R10                                             ;
                                          0137  C9  9F 005D3          PUSHAB  P.AGO                                           ;
                                                0A  11 005D7          BRB     96$                                             ;
                                                57  D5 005D9 95$:       TSTL    SOURTYP                                         ; 2709
                                                09  12 005DB          BNEQ    97$                                             ;
                                                5A  DD 005DD          PUSHL   R10                                             ;
                                          0140  C9  9F 005DF          PUSHAB  P.AGP                                           ;
                                50              6B  02 FB 005E3 96$:       CALLS   #2, NCP$ADDSTR                                  ;
                                                57  D5 005E6 97$:       TSTL    SOURTYP                                         ; 2711
                                                12  13 005E8          BEQL    98$                                             ;
                                50              7E  17 7D 005EA          MOVQ    #23, -(SP)                                      ; 2715
                                                52  DD 005ED          PUSHL   R2                                              ; 2717
                                50              7E  B2 9A 005EF          MOVZBL  @0(R2), -(SP)                                   ; 2716
                                                62  D6 005F3          INCL    (R2)                                            ;
                                          FA06  CF  04 FB 005F5          CALLS   #4, NCP$PARSEPARM                               ;
                                                3B  11 005FA          BRB     101$                                            ; 2715
                                                62  D6 005FC 98$:       INCL    (R2)                                            ; 2727
                                54              00  B2 3C 005FE          MOVZWL  @0(R2), AREA_ADDR                               ; 2728
58                              54              06  0A EF 00602          EXTZV   #10, #6, AREA_ADDR, AREA                        ; 2729
                                                0D  12 00607          BNEQ    99$                                             ; 2731
                                                5A  DD 00609          PUSHL   R10                                             ; 2734
                                          0146  C9  9F 0060B          PUSHAB  P.AGQ                                           ;
                                50              6B  02 FB 0060F          CALLS   #2, NCP$ADDSTR                                  ;
                                                54  DD 00612          PUSHL   AREA_ADDR                                       ; 2735
                                                17  11 00614          BRB     100$                                            ;
```

```
                                                5A   DD 00616 99$:     PUSHL   R10                                     ; 2739
                                         014B   C9   9F 00618          PUSHAB  P.AGR
                                          6B    02   FB 0061C          CALLS   #2, NCP$ADDSTR
                                                58   DD 0061F          PUSHL   AREA                                    ; 2740
                                00000000V  00    01   FB 00621          CALLS   #1, NCP$ADDFAO
                                                00   EF 00628          EXTZV   #0, #10, AREA_ADDR, -(SP)               ; 2741
                                00000000V  00    01   FB 0062D 100$:    CALLS   #1, NCP$ADDFAO
                                          62    02   C0 00634          ADDL2   #2, (R2)                                ; 2744
                                                53   D7 00637 101$:    DECL    MULCTR                                  ; 2747
                                                09   11 00639          BRB     103$
                                                5A   DD 0063B 102$:    PUSHL   R10                                     ; 2750
                                         0154   C9   9F 0063D          PUSHAB  P.AGS
                                          6B    02   FB 00641          CALLS   #2, NCP$ADDSTR
                                          55     00  B2   9A 00644 103$:    MOVZBL  @0(R2), NDTY                    ; 2752
                                                57   D5 00648          TSTL    SOURTYP                                 ; 2753
                                                30   12 0064A          BNEQ    104$
                                                53   D5 0064C          TSTL    MULCTR                                  ; 2755
                                                2C   15 0064E          BLEQ    104$
                                    28          55   06   E1 00650          BBC     #6, NDTY, 104$                  ; 2757
                                                55   95 00654          TSTB    NDTY                                    ; 2759
                                                24   19 00656          BLSS    104$
                                                5A   DD 00658          PUSHL   R10                                     ; 2762
                                         0162   C9   9F 0065A          PUSHAB  P.AGT
                                          6B    02   FB 0065E          CALLS   #2, NCP$ADDSTR
                                          7E    07   7D 00661          MOVQ    #7, -(SP)                               ; 2764
                                                52   DD 00664          PUSHL   R2                                      ; 2766
                                          7E     00  B2   9A 00666          MOVZBL  @0(R2), -(SP)                   ; 2765
                                                6?   D6 0066A          INCL    (R2)
                                    F98F    CF   04   FB 0066C          CALLS   #4, NCP$PARSEPARM
                                                5A   DD 00671          PUSHL   R10                                     ; 2770
                                         0165   C9   9F 00673          PUSHAB  P.AGU
                                          6B    02   FB 00677          CALLS   #2, NCP$ADDSTR
                                                53   D7 0067A          DECL    MULCTR                                  ; 2771
                                00000000'  00    D5   0067C 104$:    TSTL    NCP$A_COLTBL                            ; 2774
                                                09   13 00682          BEQL    105$
                                                5A   DD 00684          PUSHL   R10                                     ; 2776
                                         0167   C9   9F 00686          PUSHAB  P.AGV
                                          6B    02   FB 0068A          CALLS   #2, NCP$ADDSTR
                                                53   D5 0068D 105$:    TSTL    MULCTR                                  ; 2779
                                                66   15 0068F          BLEQ    108$
                                          55     00  B2   9A 00691          MOVZBL  @0(R2), NDTY                    ; 2782
                                          02    55   D1 00695          CMPL    NDTY, #2                                ; 2783
                                                78   12 00698          BNEQ    111$
                                                62   D6 0069A          INCL    (R2)                                    ; 2786
58        00    B2                      09    00   EF 0069C          EXTZV   #0, #9, @0(R2), ECLS                    ; 2787
                                          62    02   C0 006A2          ADDL2   #2, (R2)                                ; 2788
                                                53   D7 006A5          DECL    MULCTR                                  ; 2789
                                          54    62   D0 006A7          MOVL    (R2), R4                                ; 2790
57        FF    A4                      02    06   EF 006AA          EXTZV   #6, #2, -1(R4), R7
                                                2D   12 006B0          BNEQ    107$                                    ; 2793
                                                58   DD 006B2          PUSHL   ECLS                                    ; 2795
                                00000000V  00    01   FB 006B4          CALLS   #1, NCP$ADDFAO
                                                5A   DD 006BB          PUSHL   R10                                     ; 2796
                                         016B   C9   9F 006BD          PUSHAB  P.AGW
                                          6B    02   FB 006C1          CALLS   #2, NCP$ADDSTR
                                          55    64   9A 006C4          MOVZBL  (R4), NDTY                              ; 2797
                                          20    55   D1 006C7          CMPL    NDTY, #32                               ; 2798
```

```
                                                 51  12 006CA          BNEQ    113$                                    ;
                                                 53  D5 006CC          TSTL    MULCTR                                  ; 2800
                                                 4D  15 006CE          BLEQ    113$                                    ;
                                                 62  D6 006D0          INCL    (R2)                                    ; 2803
                                                 52  DD 006D2          PUSHL   R2                                      ; 2804
                00000000V  00                    01  FB 006D4          CALLS   #1, NCP$PARSEVENTS                      ;
                                                 53  D7 006DB 106$:     DECL    MULCTR                                  ; 2807
                                                 3E  11 006DD          BRB     113$                                    ; 2798
                           01                    57  D1 006DF 107$:     CMPL    R7, #1                                  ; 2811
                                                 15  12 006E2          BNEQ    109$                                    ;
                                   0171          C9  9F 006E4          PUSHAB  P.AGX                                   ; 2814
                                                 01  DD 006E8          PUSHL   #1                                      ; 2813
                                00000000G        8F  DD 006EA          PUSHL   #NCP$_BADMSG                            ;
                00000000G  00                    03  FB 006F0          CALLS   #3, LIB$STOP                            ;
                                                 24  11 006F7 108$:     BRB     113$                                    ; 2812
                           02                    57  D1 006F9 109$:     CMPL    R7, #2                                  ; 2817
                                                 11  12 006FC          BNEQ    110$                                    ;
                                                 58  DD 006FE          PUSHL   ECLS                                    ; 2819
                00000000V  00                    01  FB 00700          CALLS   #1, NCP$ADDFAO                          ;
                                                 5A  DD 00707          PUSHL   R10                                     ; 2820
                                   018B          C9  9F 00709          PUSHAB  P.AGY                                   ;
                                                 0B  11 0070D          BRB     112$                                    ;
                           03                    57  D1 0070F 110$:     CMPL    R7, #3                                  ; 2823
                                                 09  12 00712 111$:     BNEQ    113$                                    ;
                                                 5A  DD 00714          PUSHL   R10                                     ; 2825
                                   0192          C9  9F 00716          PUSHAB  P.AGZ                                   ;
                           6B                    02  FB 0071A 112$:     CALLS   #2, NCP$ADDSTR                          ;
                                                 52  D4 0071D 113$:     CLRL    IDX                                     ; 2848
                                                 40  11 0071F          BRB     117$                                    ;
                           01                    52  D1 00721 114$:     CMPL    IDX, #1                                 ; 2844
                                                 09  13 00724          BEQL    115$                                    ;
                                                 5A  DD 00726          PUSHL   R10                                     ; 2846
                                   0197          C9  9F 00728          PUSHAB  P.AHA                                   ;
                           6B                    02  FB 0072C          CALLS   #2, NCP$ADDSTR                          ;
                           50            08      BC  D0 0072F 115$:     MOVL    @PTR, R0                                ; 2848
                           55                    60  9A 00733          MOVZBL  (R0), NDTY                              ;
                     17    55                    06  E1 00736          BBC     #6, NDTY, 116$                          ; 2849
                                                 55  95 0073A          TSTB    NDTY                                    ; 2851
                                                 13  18 0073C          BGEQ    116$                                    ;
                                   019A          C9  9F 0073E          PUSHAB  P.AHB                                   ; 2854
                                                 01  DD 00742          PUSHL   #1                                      ; 2853
                                00000000G        8F  DD 00744          PUSHL   #NCP$_BADMSG                            ;
                     00000000G  00               03  FB 0074A          CALLS   #3, LIB$STOP                            ;
                                08               BC  D6 00751 116$:     INCL    @PTR                                    ; 2856
                           7E                    17  7D 00754          MOVQ    #23, -(SP)                              ; 2857
                                08               AC  DD 00757          PUSHL   PTR                                     ;
                                                 55  DD 0075A          PUSHL   NDTY                                    ;
                      F89F  CF                   04  FB 0075C          CALLS   #4, NCP$PARSEPARM                       ;
                     BC     52                    53  F3 00761 117$:     AOBLEQ  MULCTR, IDX, 114$                       ; 2841
                                                 04  00765          RET                                             ; 2326
56        04   AC          06                    00  EF 00766 118$:     EXTZV   #0, #6, DTY, LEN                        ; 2873
                           58            08      AC  D0 0076C          MOVL    PTR, R8                                 ; 2874
                           57                    68  D0 00770          MOVL    (R8), PSTRT                             ;
                           50            0C      AC  D0 00773          MOVL    TYP, R0                                 ; 2876
                           01                    50  D1 00777          CMPL    R0, #1                                  ; 2879
                                                 54  12 0077A          BNEQ    122$                                    ;
                                                 10  AC  D5 0077C          TSTL    LST                                     ; 2881
```

```
                                      13  12 0077F         BNEQ    119$                                   ;
                            01B4      C9  9F 00781         PUSHAB  P.AHC                                  ; 2883
                                      01  DD 00785         PUSHL   #1                                     ;
                       00000000G      8F  DD 00787         PUSHL   #NCP$_LOGIC                            ;
           00000000G  00              03  FB 0078D         CALLS   #3, LIB$STOP                           ;
                                      5E  DD 00794 119$:   PUSHL   SP                                     ; 2887
                              10      AC  DD 00796         PUSHL   LST                                    ; 2889
                  7E          00      B8  9A 00799         MOVZBL  @0(R8), -(SP)                          ; 2888
           00000000G  00              03  FB 0079D         CALLS   #3, NCP$TABLESEARCH                    ;
                      11              50  E9 007A4         BLBC    R0, 120$                               ;
                                      6E  DD 007A7         PUSHL   TXT                                    ; 2894
           0000000V   00              01  FB 007A9         CALLS   #1, NCP$ADDFAO                         ;
                                      5A  DD 007B0         PUSHL   R10                                    ; 2895
                            01C9      C9  9F 007B2         PUSHAB  P.AHD                                  ;
                                      10  11 007B6         BRB     121$                                   ;
                              00      B8  DD 007B8 120$:   PUSHL   @0(R8)                                 ; 2899
           0000000V   00              01  FB 007BB         CALLS   #1, NCP$ADDFAO                         ;
                                      5A  DD 007C2         PUSHL   R10                                    ; 2900
                            01CD      C9  9F 007C4         PUSHAB  P.AHE                                  ;
                      6B              02  FB 007C8 121$:   CALLS   #2, NCP$ADDSTR                         ;
                                      68  D6 007CB         INCL    (R8)                                   ; 2903
                                    00CC  31 007CD         BRW     135$                                   ;
                      0C              50  D1 007D0 122$:   CMPL    R0, #12                                ; 2910
                                      03  18 007D3         BGEQ    124$                                   ;
                                    008B  31 007D5 123$:   BRW     132$                                   ;
                      0D              50  D1 007D8 124$:   CMPL    R0, #13                                ;
                                      F8  14 007DB         BGTR    123$                                   ;
                 04   AE              56  D0 007DD         MOVL    LEN, CTR                               ; 2917
                      08      04      AE  D1 007E1         CMPL    CTR, #8                                ; 2918
                                      13  15 007E5         BLEQ    125$                                   ;
                            01D7      C9  9F 007E7         PUSHAB  P.AHF                                  ; 2920
                                      01  DD 007EB         PUSHL   #1                                     ;
                       00000000G      8F  DD 007ED         PUSHL   #NCP$_BADMSG                           ;
           00000000G  00              03  FB 007F3         CALLS   #3, LIB$STOP                           ;
  08          00      6E              00  2C 007FA 125$:   MOVC5   #0, (SP), #0, #8, PRVMSK               ; 2922
                              08      AE     007FF                                                        ;
                      51      04      AE  D0 00801         MOVL    CTR, R1                                ; 2923
                      08              51  D1 00805         CMPL    R1, #8                                 ;
                                      03  1B 00808         BLEQU   126$                                   ;
                      51              08  D0 0080A         MOVL    #8, R1                                 ;
                      50              01  CE 0080D 126$:   MNEGL   #1, IDX                                ;
                                      07  11 00810         BRB     128$                                   ;
                 08 AE40         00 B840  90 00812 127$:   MOVB    @0(R8)[IDX], PRVMSK[IDX]               ; 2925
              F5      50              51  F3 00819 128$:   AOBLEQ  R1, IDX, 127$                          ;
                      68      04      AE  C0 0081D         ADDL2   CTR, (R8)                              ; 2928
                              04      AE  9F 00821         PUSHAB  CTR                                    ; 2930
                              0C      AE  9F 00824         PUSHAB  PRVMSK                                 ;
           0000000V   00              02  FB 00827         CALLS   #2, NCP$PRIVBITS                       ;
                      52              01  D0 0082E         MOVL    #1, IDX                                ; 2933
                                      28  11 00831         BRB     131$                                   ;
  7E          00      52              01  7A 00833 129$:   EMUL    #1, IDX, #0, -(SP)                     ; 2936
  50          50      8E              06  7B 00838         EDIV    #6, (SP)+, R0, R0                      ;
                      01              50  D1 0083D         CMPL    R0, #1                                 ;
                                      0E  12 00840         BNEQ    130$                                   ;
                      01              52  D1 00842         CMPL    IDX, #1                                ; 2938
                                      09  13 00845         BEQL    130$                                   ;
                                      5A  DD 00847         PUSHL   R10                                    ; 2940
```

```
                          01EE   C9  9F 00849          PUSHAB  P.AHG                                ;
                       6B        02  FB 0084D          CALLS   #2, NCP$ADDSTR                      ;
                                 5A  DD 00850 130$:     PUSHL   R10                                 ; 2942
                          01F6   C9  9F 00852          PUSHAB  P.AHH                               ;
                       6B        02  FB 00856          CALLS   #2, NCP$ADDSTR                      ;
                                 52  D6 00859          INCL    IDX                                 ;
                    04   AE        52  D1 0085B 131$:     CMPL    IDX, CTR                        ;
                                 D2  1B 0085F          BLEQU   129$                                ;
                                 39  11 00861          BRB     135$                                ; 2933
                                 5A  DD 00863 132$:     PUSHL   R10                                 ; 2948
                          01FB   C9  9F 00865          PUSHAB  P.AHI                               ;
                       6B        02  FB 00869          CALLS   #2, NCP$ADDSTR                      ;
                       53     FF  A6  9E 0086C          MOVAB   -1(R6), R3                          ; 2949
                                 52  D4 00870          CLRL    IDX                                 ;
                                 17  11 00872          BRB     134$                                ;
                                 5A  DD 00874 133$:     PUSHL   R10                                 ; 2952
                          01FF   C9  9F 00876          PUSHAB  P.AHJ                               ;
                       6B        02  FB 0087A          CALLS   #2, NCP$ADDSTR                      ;
                       7E     00 B842  9A 0087D          MOVZBL  @0(R8)[IDX], -(SP)                  ; 2953
             00000000V 00            01  FB 00882          CALLS   #1, NCP$ADDFAO                      ;
                                 52  D6 00889          INCL    IDX                                 ;
                       53        52  D1 0088B 134$:     CMPL    IDX, R3                             ;
                                 E4  1B 0088E          BLEQU   133$                                ;
                                 5A  DD 00890          PUSHL   R10                                 ; 2956
                          0203   C9  9F 00892          PUSHAB  P.AHK                               ;
                       6B        02  FB 00896          CALLS   #2, NCP$ADDSTR                      ;
                       68        56  C0 00899          ADDL2   LEN, (R8)                           ; 2957
                       57        56  C0 0089C 135$:     ADDL2   LEN, R7                             ; 2963
                       68        57  D1 0089F          CMPL    R7, (R8)                            ;
                                 13  13 008A2          BEQL    136$                                ;
                          0205   C9  9F 008A4          PUSHAB  P.AHL                               ; 2966
                                 01  DD 008A8          PUSHL   #1                                  ; 2965
                    00000000G    8F  DD 008AA          PUSHL   #NCP$_BADMSG                        ;
          00000000G 00           03  FB 008B0          CALLS   #3, LIB$STOP                        ;
                                 04 008B7 136$:     RET                                         ; 2970

; Routine Size:  2232 bytes,    Routine Base:  $CODE$ + 0D05
```

```
;  3000     2971  1 %SBTTL  'NCP$PARSEVENTS  Parse Event Parameter'
;  3001     2972  1 ROUTINE NCP$PARSEVENTS (PTR) :NOVALUE = !
;  3002     2973  1
;  3003     2974  1 !++
;  3004     2975  1 ! FUNCTIONAL DESCRIPTION:
;  3005     2976  1 !
;  3006     2977  1 !       Parses the bit mask for event types and creates the fao list
;  3007     2978  1 !       entries and appents entries to the fao control string to do the
;  3008     2979  1 !       proper conversions.  This routine properly formats both single bits
;  3009     2980  1 !       and ranges.
;  3010     2981  1 !
;  3011     2982  1 ! FORMAL PARAMETERS:
;  3012     2983  1 !
;  3013     2984  1 !       PTR             Address of a pointer to the event mask
;  3014     2985  1 !
;  3015     2986  1 ! IMPLICIT INPUTS:
;  3016     2987  1 !
;  3017     2988  1 !       NONE
;  3018     2989  1 !
;  3019     2990  1 ! IMPLICIT OUTPUTS:
;  3020     2991  1 !
;  3021     2992  1 !       ADDSTR          Macro called to add strings to fao control string
;  3022     2993  1 !       ADDFAO          Macro called to add items to fao list
;  3023     2994  1 !
;  3024     2995  1 ! ROUTINE VALUE:
;  3025     2996  1 ! COMPLETION CODES:
;  3026     2997  1 !
;  3027     2998  1 !       NONE
;  3028     2999  1 !
;  3029     3000  1 ! SIDE EFFECTS:
;  3030     3001  1 !
;  3031     3002  1 !       NONE
;  3032     3003  1 !
;  3033     3004  1 !--
```

```
;  3035     3005  2     BEGIN
;  3036     3006  2
;  3037     3007  2     BUILTIN
;  3038     3008  2         FFS, FFC                        ! Find bit operations
;  3039     3009  2                                         ! Remember that these are true if
;  3040     3010  2                                         ! No bit is found
;  3041     3011  2         ;
;  3042     3012  2
;  3043     3013  2     LOCAL
;  3044     3014  2         BITS,                           ! How many bits
;  3045     3015  2         BASE,                           ! Base address of bits
;  3046     3016  2         LOBIT,                          ! Low bit number
;  3047     3017  2         HIBIT,                          ! High bit number
;  3048     3018  2         SIZE                            ! Size if field to search
;  3049     3019  2         ;               !
;  3050     3020  2
;  3051     3021  2     BITS = ( .(..PTR) <0, 8, 0> ) * 8;  ! Number of bits
;  3052     3022  2     BASE = ..PTR + 1;
;  3053     3023  2     .PTR = ..PTR + .(..PTR) <0, 8, 0> + 1; ! Point beyond the bits
;  3054     3024  2
;  3055     3025  2     IF .BITS EQL 0                      ! Any bits
;  3056     3026  2     THEN
;  3057     3027  2         RETURN                          ! No bits, then nothing
;  3058     3028  2     ;
;  3059     3029  2
;  3060     3030  2     SIZE = 32;                          ! Size of field to search
;  3061     3031  2     LOBIT = 0;                          ! Lowest bit to look for (POS)
;  3062     3032  2
;  3063     3033  2     WHILE TRUE                          ! Forever look for bits
;  3064     3034  2     DO
;  3065     3035  3         BEGIN                           ! Any bits set?
;  3066     3036  3         WHILE .LOBIT LSS .BITS          ! While there are bits to look for
;  3067     3037  3             AND
;  3068     3038  3             FFS (LOBIT, SIZE, .BASE, LOBIT) ! Find first set until it is
;  3069     3039  3         DO
;  3070     3040  3         ;
;  3071     3041  3
;  3072     3042  3         IF .LOBIT GEQ .BITS             ! First in our mask?
;  3073     3043  3         THEN
;  3074     3044  3             RETURN                      ! Nope
;  3075     3045  3         ;
;  3076     3046  3
;  3077     3047  3         FFC (LOBIT, SIZE, .BASE, HIBIT); ! If no bit is found, lobit + size
;  3078     3048  3                                         ! is returned as high bit
```

```
;  3080          3049  3
;  3081          3050  3          IF .HIBIT GEQ .BITS                ! Clamp the top of the range too
;  3082          3051  3          THEN
;  3083          3052  3              HIBIT = .BITS
;  3084          3053  3          ;
;  3085          3054  3          HIBIT = .HIBIT - 1;                ! Adjust hibit
;  3086          3055  3          IF .LOBIT EQL .HIBIT               ! A range of bits found
;  3087          3056  3          THEN
;  3088          3057  4              BEGIN                          ! Not a range
;  3089          3058  4              ADDFAO (.LOBIT);               ! Stuff one bit
;  3090          3059  4              ADDSTR ('!UB ')
;  3091          3060  4              END
;  3092          3061  3          ELSE
;  3093          3062  4              BEGIN                          ! Stuff a range of bits
;  3094          3063  4              ADDFAO (.LOBIT);
;  3095          3064  4              ADDFAO (.HIBIT);
;  3096          3065  4              ADDSTR ('!UB-!UB ')
;  3097          3066  4              END
;  3098          3067  3          ;
;  3099          3068  3          LOBIT = .HIBIT + 1                 ! Continue searching from here
;  3100          3069  3          END
;  3101          3070  3
;  3102          3071  1      END;


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

                                20  42  55  21  04  00CBA P.AHM:  .ASCII  <4>\!UB \                                    ;
                            20  42  55  21  2D  42  55  21  08  00CBF P.AHN:  .ASCII  <8>\!UB-!UB \                    ;


                                                        .PSECT  $CODE$,NOWRT,2

                                                    01FC 00000 NCP$PARSEVENTS:
                                                               .WORD   Save R2,R3,R4,R5,R6,R7,R8                        ; 2972
                                            58 00000000'   00   9E 00002        MOVAB   NCP$GQ_CTRDSC, R8            ;
                                            57 00000000V   00   9E 00009        MOVAB   NCP$ADDFAO, R7               ;
                                            53         04   AC   D0 00010        MOVL    PTR, R3                      ; 3021
                                            50              63   D0 00014        MOVL    (R3), R0                     ;
                                            51              60   9A 00017        MOVZBL  (R0), R1                     ;
                                        55  51              03   78 0001A        ASHL    #3, R1, BITS                 ;
                                            52         01   A0   9E 0001E        MOVAB   1(R0), BASE                  ; 3022
                                            63         01 A140   9E 00022        MOVAB   1(R1)[R0], (R3)              ; 3023
                                                            55   D5 00027        TSTL    BITS                         ; 3025
                                                            58   13 00029        BEQL    6$                           ;
                                            56              20   D0 0002B        MOVL    #32, SIZE                    ; 3030
                                                            53   D4 0002E        CLRL    LOBIT                        ; 3031
                                            55              53   D1 00030 1$:    CMPL    LOBIT, BITS                  ; 3036
                                                            07   18 00033        BGEQ    2$                           ;
                                53          62              56   EA 00035        FFS     LOBIT, SIZE, (BASE), LOBIT   ; 3038
                                                            F4   13 0003A        BEQL    1$                           ;
                                            55              53   D1 0003C 2$:    CMPL    LOBIT, BITS                  ; 3042
                                                            42   18 0003F        BGEQ    6$                           ;
                                54          62              56   EB 00041        FFC     LOBIT, SIZE, (BASE), HIBIT   ; 3047
                                            55              54   D1 00046        CMPL    HIBIT, BITS                  ; 3050
```

```
                                    03  19 00049          BLSS    3$                                          ;
                                54  55  D0 0004B          MOVL    BITS, HIBIT                                 ; 3052
                                54  D7 0004E 3$:        DECL    HIBIT                                       ; 3054
                                54  53  D1 00050          CMPL    LOBIT, HIBIT                                ; 3055
                                    0F  12 00053          BNEQ    4$                                          ;
                                    53  DD 00055          PUSHL   LOBIT                                       ; 3058
                                67  01  FB 00057          CALLS   #1, NCP$ADDFAO                              ;
                                    58  DD 0005A          PUSHL   R8                                          ; 3059
                            00000000'   00  9F 0005C          PUSHAB  P.AHM                                       ;
                                    12  11 00062          BRB     5$                                          ;
                                    53  DD 00064 4$:       PUSHL   LOBIT                                       ; 3063
                                67  01  FB 00066          CALLS   #1, NCP$ADDFAO                              ;
                                    54  DD 00069          PUSHL   HIBIT                                       ; 3064
                                67  01  FB 0006B          CALLS   #1, NCP$ADDFAO                              ;
                                    58  DD 0006E          PUSHL   R8                                          ; 3065
                            00000000'   00  9F 00070          PUSHAB  P.AHN                                       ;
                00000000V   00      02  FB 00076 5$:       CALLS   #2, NCP$ADDSTR                              ;
                                53  01  A4  9E 0007D          MOVAB   1(R4), LOBIT                                ; 3068
                                    AD  11 00081          BRB     1$                                          ;
                                        04 00083 6$:       RET                                                 ; 3071

; Routine Size:  132 bytes,    Routine Base:  $CODE$ + 15BD
```

```
;  3104          3072  1 %SBTTL  'NCP$PTBSEARCH  Search a PTB for an Entry'
;  3105          3073  1 ROUTINE NCP$PTBSEARCH (TBL, ID, TYP, LST, TXT) =       !
;  3106          3074  1
;  3107          3075  1 !++
;  3108          3076  1 ! FUNCTIONAL DESCRIPTION:
;  3109          3077  1 !
;  3110          3078  1 !       Search a Parameter Table Block for an entry specified by
;  3111          3079  1 !       ID.  Return the parameter type code, address of a list entry, and
;  3112          3080  1 !       address of the text for the parameter name.
;  3113          3081  1 !
;  3114          3082  1 ! FORMAL PARAMETERS:
;  3115          3083  1 !
;  3116          3084  1 !       TBL             Address of the PTB to search
;  3117          3085  1 !       ID              Value of the id to find
;  3118          3086  1 !       TYP             Address to return the type code
;  3119          3087  1 !       LST             Address to return the address of the associated list
;  3120          3088  1 !       TXT             Address to return the address of the name
;  3121          3089  1 !
;  3122          3090  1 ! IMPLICIT INPUTS:
;  3123          3091  1 !
;  3124          3092  1 !       NONE
;  3125          3093  1 !
;  3126          3094  1 ! IMPLICIT OUTPUTS:
;  3127          3095  1 !
;  3128          3096  1 !       NONE
;  3129          3097  1 !
;  3130          3098  1 ! ROUTINE VALUE:
;  3131          3099  1 ! COMPLETION CODES:
;  3132          3100  1 !
;  3133          3101  1 !       Success or failure
;  3134          3102  1 !
;  3135          3103  1 ! SIDE EFFECTS:
;  3136          3104  1 !
;  3137          3105  1 !       NONE
;  3138          3106  1 !
;  3139          3107  1 !--
;  3140          3108  1
;  3141          3109  2     BEGIN
;  3142          3110  2
;  3143          3111  2
;  3144          3112  2     FIELD                                   ! Fields for Parameter Text Blocks
;  3145          3113  2         PTBFLDS =
;  3146          3114  2         SET
;  3147          3115  2         PTB$W_ID        = [0, 0, 16, 0],        ! Id is the NMA code
;  3148          3116  2         PTB$B_TYP       = [2, 0, 8, 0],         ! Type is the PBK type code
;  3149          3117  2         PTB$W_LST       = [3, 0, 16, 1],        ! Word offset to a text block
;  3150          3118  2         PTB$W_TXT       = [5, 0, 16, 1]         ! Word offset to a c string
;  3151          3119  2         TES
;  3152          3120  2         ;
;  3153          3121  2
;  3154          3122  2     LITERAL
;  3155          3123  2         PTB$C_SIZE      = 7                     ! Size of the block
;  3156          3124  2         ;
;  3157          3125  2
;  3158          3126  2     LOCAL
;  3159          3127  2                                             ! Pointer to the table
;  3160          3128  2         PLIST : REF BBLOCKVECTOR [1, PTB$C_SIZE] FIELD (PTBFLDS)
```

```
        ;                   !

    PLIST = .TBL;                                   ! Set the table start

    INCRU IDX FROM 0                                ! Look at the table from
    DO                                              ! The beginning
        BEGIN
        IF .PLIST [.IDX, PTB$W_ID] EQL 65535        ! All done
        THEN
            RETURN FAILURE                          ! We didn't find it here
        ;
!
!       If id's match, return the type, list, and text
!

        IF .PLIST [.IDX, PTB$W_ID] EQL .ID <0, 16, 0>
        THEN
            BEGIN
            .TYP = .PLIST [.IDX, PTB$B_TYP];        ! Return the type code
            IF .PLIST [.IDX, PTB$W_LST] EQL 0       ! If there is no list
            THEN
                .LST = 0                            ! Then return no list
            ELSE                                    ! Else, return real list addr
                .LST = .PLIST [.IDX, PTB$W_LST] + PLIST [.IDX, PTB$W_LST]
            ;
            .TXT = .PLIST [.IDX, PTB$W_TXT] + PLIST [.IDX, PTB$W_TXT];
            RETURN SUCCESS
            END
        END
    END;


                                    001C 00000 NCP$PTBSEARCH:
                                                      .WORD   Save R2,R3,R4              ; 3073
                  53       04  AC  D0 00002           MOVL    TBL, PLIST                 ; 3131
                               50  D4 00006           CLRL    IDX                        ; 3145
          51      50           07  C5 00008 1$:       MULL3   #7, IDX, R1                ; 3136
                  51           53  C0 0000C           ADDL2   PLIST, R1                  ;
            FFFF  8F           61  B1 0000F           CMPW    (R1), #65535               ;
                               03  12 00014           BNEQ    2$                         ;
                               50  D4 00016           CLRL    R0                         ; 3138
                                   04 00018           RET                                ;
              08  AC           61  B1 00019 2$:       CMPW    (R1), ID                   ; 3145
                               2A  12 0001D           BNEQ    5$                         ;
              0C  BC       02  A1  9A 0001F           MOVZBL  2(R1), @TYP                ; 3148
                  52       03  A1  9E 00024           MOVAB   3(R1), R2                  ; 3149
                               62  B5 00028           TSTW    (R2)                       ;
                               05  12 0002A           BNEQ    3$                         ;
                           10  BC  D4 0002C           CLRL    @LST                       ; 3151
                               08  11 0002F           BRB     4$                         ;
                  54           62  32 00031 3$:       CVTWL   (R2), R4                   ; 3153
      10  BC      54           52  C1 00034           ADDL3   R2, R4, @LST               ;
```

```
                        52      05  A1  9E 00039 4$:    MOVAB   5(R1), R2                       ; 3155
                        51          62  32 0003D        CVTWL   (R2), R1                        ;
                    14  BC          51  C1 00040        ADDL3   R2, R1, @TXT                    ;
                        50          01  D0 00045        MOVL    #1, R0                          ; 3156
                                        04 00048        RET                                     ;
                                    50  D6 00049 5$:    INCL    IDX                             ; 3145
                                    BB  11 0004B        BRB     1$                              ;

; Routine Size:  77 bytes,    Routine Base:  $CODE$ + 1641
```

```
;  3194          3161  1 %SBTTL  'NCP$FAOSET  Setup parameters for fao list'
;  3195          3162  1 GLOBAL ROUTINE NCP$FAOSET :NOVALUE =    !
;  3196          3163  1
;  3197          3164  1 !++
;  3198          3165  1 ! FUNCTIONAL DESCRIPTION:
;  3199          3166  1 !
;  3200          3167  1 !       Setup control string buffer descriptor and fao list pointer.
;  3201          3168  1 !
;  3202          3169  1 ! FORMAL PARAMETERS:
;  3203          3170  1 !
;  3204          3171  1 !       NONE
;  3205          3172  1 !
;  3206          3173  1 ! IMPLICIT INPUTS:
;  3207          3174  1 !
;  3208          3175  1 !       NONE
;  3209          3176  1 !
;  3210          3177  1 ! IMPLICIT OUTPUTS:
;  3211          3178  1 !
;  3212          3179  1 !       NONE
;  3213          3180  1 !
;  3214          3181  1 ! ROUTINE VALUE:
;  3215          3182  1 ! COMPLETION CODES:
;  3216          3183  1 !
;  3217          3184  1 !       NONE
;  3218          3185  1 !
;  3219          3186  1 ! SIDE EFFECTS:
;  3220          3187  1 !
;  3221          3188  1 !       NONE
;  3222          3189  1 !
;  3223          3190  1 !--
;  3224          3191  1
;  3225          3192  2     BEGIN
;  3226          3193  2
;  3227          3194  2     FAOPTR = 0;                         ! Initialize pointer to fao list
;  3228          3195  2     CTRDSC [0] = 0;                     ! Control string descriptor
;  3229          3196  2     CTRDSC [1] = CTRBUF;                ! for FAO call later, set for ADDSTR
;  3230          3197  2     CTRDSC [2] = CTRBUFSIZ;             ! Set for size check
;  3231          3198  2     RETURN
;  3232          3199  2
;  3233          3200  1     END;


                                        0004 00000      .ENTRY  NCP$FAOSET, Save R2                             ; 3162
                              52 00000000'  00   9E 00002 MOVAB   CTRDSC, R2                                  ;
                                 00000000'  00   D4 00009 CLRL    FAOPTR                                         ; 3194
                                            62   D4 0000F CLRL    CTRDSC                                         ; 3195
                           04    A2 00000000'  00   9E 00011 MOVAB   CTRBUF, CTRDSC+4                        ; 3196
                           08    A2      01F4  8F   3C 00019 MOVZWL  #500, CTRDSC+8                          ; 3197
                                               04   0001F RET                                                 ; 3200

; Routine Size:  32 bytes,    Routine Base:  $CODE$ + 168E
```

```
;  3235          3201  1 %SBTTL  'NCP$ADDSTR  Add a String to a Descriptor'
;  3236          3202  1 GLOBAL ROUTINE NCP$ADDSTR (CSTR, DSC) :NOVALUE =       !
;  3237          3203  1
;  3238          3204  1 !++
;  3239          3205  1 ! FUNCTIONAL DESCRIPTION:
;  3240          3206  1 !
;  3241          3207  1 !       This routine adds a counted string to the string described by
;  3242          3208  1 !       a descriptor.  A length check is made against the maximum length
;  3243          3209  1 !       of the string which is in the third longword of the descriptor.
;  3244          3210  1 !
;  3245          3211  1 ! FORMAL PARAMETERS:
;  3246          3212  1 !
;  3247          3213  1 !       CSTR            Address of a counted string, ASCIC ('txt')
;  3248          3214  1 !       DSC             Address of a string descriptor and following longword
;  3249          3215  1 !                       max length
;  3250          3216  1 !
;  3251          3217  1 ! IMPLICIT INPUTS:
;  3252          3218  1 !
;  3253          3219  1 !       NONE
;  3254          3220  1 !
;  3255          3221  1 ! IMPLICIT OUTPUTS:
;  3256          3222  1 !
;  3257          3223  1 !       NONE
;  3258          3224  1 !
;  3259          3225  1 ! ROUTINE VALUE:
;  3260          3226  1 ! COMPLETION CODES:
;  3261          3227  1 !
;  3262          3228  1 !       NONE, error signaled if string does not fit
;  3263          3229  1 !
;  3264          3230  1 ! SIDE EFFECTS:
;  3265          3231  1 !
;  3266          3232  1 !       NONE
;  3267          3233  1 !
;  3268          3234  1 !--
```

```
;  3270          3235  1
;  3271          3236  2      BEGIN
;  3272          3237  2
;  3273          3238  2      MAP
;  3274          3239  2          DSC : REF VECTOR                ! Descriptor of string
;  3275          3240  2          ;
;  3276          3241  2
;  3277          3242  2      IF (.(.CSTR) <0, 8, 0> + .DSC [0])  ! Will string fit?
;  3278          3243  2          GTR
;  3279          3244  2          .DSC [2]
;  3280          3245  2      THEN
;  3281          3246  2          SIGNAL_STOP (NCP$_BADMSG, 1, ASCIC ('too many parameters') )
;  3282          3247  2      ;
;  3283          3248  2
;  3284          3249  2      CH$MOVE                             ! Copy the string
;  3285          3250  2          (
;  3286          3251  2          .(.CSTR) <0, 8, 0>,
;  3287          3252  2          .CSTR + 1,
;  3288          3253  2          .DSC [1] + .DSC [0]
;  3289          3254  2          );
;  3290          3255  2
;  3291          3256  2      DSC [0] = .DSC [0] + .(.CSTR) <0, 8, 0>; ! Update descriptor
;  3292          3257  2
;  3293          3258  2      RETURN
;  3294          3259  2
;  3295          3260  1      END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2
6D 61 72 61 70 20 79 6E 61 6D 20 6F 6F 74 13  00CC8 P.AHO:    .ASCII  <19>\too many parameters\                       ;
                                 73 72 65 74 65  00CD7                                                              ;


                                                                .PSECT  $CODE$,NOWRT,2

                                                     00FC 00000         .ENTRY  NCP$ADDSTR, Save R2,R3,R4,R5,R6,R7      ; 3202
                                          57      04  AC   D0 00002    MOVL    CSTR, R7                                ; 3242
                                          56      08  AC   D0 00006    MOVL    DSC, R6                                 ;
                                          50          67   9A 00009    MOVZBL  (R7), R0                                ;
                                          50          66   C0 0000C    ADDL2   (R6), R0                                ;
                                    08    A6          50   D1 0000F    CMPL    R0, 8(R6)                               ; 3244
                                                      15   15 00013    BLEQ    1$                                      ;
                                00000000'             00   9F 00015    PUSHAB  P.AHO                                   ; 3246
                                                      01   DD 0001B    PUSHL   #1                                      ;
                                00000000G             8F   DD 0001D    PUSHL   #NCP$_BADMSG                            ;
                          00000000G   00              03   FB 00023    CALLS   #3, LIB$STOP                            ;
                                      51              67   9A 00026    MOVZBL  (R7), R1                        1$:     ; 3251
                          50          04  A6          66   C1 00029    ADDL3   (R6), 4(R6), R0                         ; 3253
                          60          01  A7          51   28 0002D    MOVC3   R1, 1(R7), (R0)                         ;
                                      50              67   9A 00032    MOVZBL  (R7), R0                                ; 3256
                                      66              50   C0 00035    ADDL2   R0, (R6)                                ;
                                                          04 00038    RET                                             ; 3260

; Routine Size:  63 bytes,    Routine Base:  $CODE$ + 16AE
```

```
;  3297          3261  1 %SBTTL  'NCP$ADDFAO  Add an Entry to the FAO List'
;  3298          3262  1 GLOBAL ROUTINE NCP$ADDFAO (ITEM) :NOVALUE =     !
;  3299          3263  1
;  3300          3264  1 !++
;  3301          3265  1 ! FUNCTIONAL DESCRIPTION:
;  3302          3266  1 !
;  3303          3267  1 !       Add a longword entry to the fao list we are building.
;  3304          3268  1 !       If the list overflows, signal an error for too many parameters.
;  3305          3269  1 !
;  3306          3270  1 ! FORMAL PARAMETERS:
;  3307          3271  1 !
;  3308          3272  1 !       ITEM            Value of the list entry
;  3309          3273  1 !
;  3310          3274  1 ! IMPLICIT INPUTS:
;  3311          3275  1 !
;  3312          3276  1 !       FAOPTR          Index into the FAOLST
;  3313          3277  1 !
;  3314          3278  1 ! IMPLICIT OUTPUTS:
;  3315          3279  1 !
;  3316          3280  1 !       FAOPTR          Incremented
;  3317          3281  1 !
;  3318          3282  1 ! ROUTINE VALUE:
;  3319          3283  1 ! COMPLETION CODES:
;  3320          3284  1 !
;  3321          3285  1 !       NONE
;  3322          3286  1 !
;  3323          3287  1 ! SIDE EFFECTS:
;  3324          3288  1 !
;  3325          3289  1 !       NONE
;  3326          3290  1 !
;  3327          3291  1 !--
;  3328          3292  1
;  3329          3293  2     BEGIN
;  3330          3294  2
;  3331          3295  2     IF .FAOPTR GEQ FAOLSTSIZ            ! Check the size of the list
;  3332          3296  2     THEN
;  3333          3297  2         SIGNAL_STOP (NCP$_BADMSG, 1,    ! Too large, bump us off
;  3334          3298  2                 ASCIC ('too many parameters') )
;  3335          3299  2     ;
;  3336          3300  2
;  3337          3301  2     FAOLST [.FAOPTR] = .ITEM;           ! Add the item to the list
;  3338          3302  2     FAOPTR = .FAOPTR + 1;               ! Bump the index into the list
;  3339          3303  2     RETURN
;  3340          3304  2
;  3341          3305  1     END;


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

6D  61  72  61  70  20  79  6E  61  6D  20  6F  6F  74  13  00CDC P.AHP:  .ASCII  <19>\too many parameters\                ;
                                        73  72  65  74  65  00CEB                                                       ;


                                                                .PSECT  $CODE$,NOWRT,2
```

```
                                                0004 00000          .ENTRY  NCP$ADDFAO, Save R2                     ; 3262
                                    52 00000000'  00   9E 00002     MOVAB   FAOPTR, R2                              ;
                                00000064    8F        62   D1 00009     CMPL    FAOPTR, #100                    ; 3295
                                                      15   19 00010     BLSS    1$                              ;
                                           00000000'  00   9F 00012     PUSHAB  P.AHP                           ; 3298
                                                      01   DD 00018     PUSHL   #1                              ; 3297
                                           00000000G  8F   DD 0001A     PUSHL   #NCP$_BADMSG                    ;
                                00000000G     00        03   FB 00020     CALLS   #3, LIB$STOP                    ;
                                        50            62   D0 00027 1$:  MOVL    FAOPTR, R0                      ; 3301
                                   04 A240       04   AC   D0 0002A     MOVL    ITEM, FAOLST[R0]                ;
                                                      62   D6 00030     INCL    FAOPTR                          ; 3302
                                                      04   00032     RET                                     ; 3305

; Routine Size:  51 bytes,    Routine Base:  $CODE$ + 16ED
```

```
;  3343          3306  1 %SBTTL  'NCP$FAOWRITE  Convert and Write Fao Parameters'
;  3344          3307  1 GLOBAL ROUTINE NCP$FAOWRITE :NOVALUE =  !
;  3345          3308  1
;  3346          3309  1 !++
;  3347          3310  1 ! FUNCTIONAL DESCRIPTION:
;  3348          3311  1 !
;  3349          3312  1 !       This routine calls $FAOL to convert the parameters in the
;  3350          3313  1 !       Fao control string and fao list and then calls the write
;  3351          3314  1 !       routine to write the string to the output device for
;  3352          3315  1 !       show and list.
;  3353          3316  1 !
;  3354          3317  1 ! FORMAL PARAMETERS:
;  3355          3318  1 !
;  3356          3319  1 !       NONE
;  3357          3320  1 !
;  3358          3321  1 ! IMPLICIT INPUTS:
;  3359          3322  1 !
;  3360          3323  1 !       CTRDSC
;  3361          3324  1 !       FAOLST
;  3362          3325  1 !
;  3363          3326  1 ! IMPLICIT OUTPUTS:
;  3364          3327  1 !
;  3365          3328  1 !       NONE
;  3366          3329  1 !
;  3367          3330  1 ! ROUTINE VALUE:
;  3368          3331  1 ! COMPLETION CODES:
;  3369          3332  1 !
;  3370          3333  1 !       NONE
;  3371          3334  1 !
;  3372          3335  1 ! SIDE EFFECTS:
;  3373          3336  1 !
;  3374          3337  1 !       NONE
;  3375          3338  1 !
;  3376          3339  1 !--
;  3377          3340  1
;  3378          3341  2     BEGIN
;  3379          3342  2
;  3380          3343  2     OUTDSC [0] = OUTBUFSIZ;             ! Set the output buffer descriptor
;  3381          3344  2     OUTDSC [1] = OUTBUF;
;  3382          3345  2
;  3383          3346  2     NCP$FAOL (OUTDSC);                  ! Convert the fao lists
;  3384          3347  2
;  3385          3348  2     NCP$WRITESHO (OUTDSC);              ! Write the output
;  3386          3349  2
;  3387          3350  2     RETURN
;  3388          3351  2
;  3389          3352  1     END;


                                     0004 00000         .ENTRY  NCP$FAOWRITE, Save R2                                  ; 3307
                                52 00000000'  00  9E 00002  MOVAB   OUTDSC, R2                                         ;
                                62       01F4 8F  3C 00009  MOVZWL  #500, OUTDSC                                        ; 3343
                             04 A2         08 A2  9E 0000E  MOVAB   OUTBUF, OUTDSC+4                                    ; 3344
                                                52  DD 00013  PUSHL   R2                                               ; 3346
```

```
                    00000000V  00          01  FB 00015        CALLS   #1, NCP$FAOL                ;
                               52  DD 0001C                    PUSHL   R2                          ; 3348
                    00000000G  00          01  FB 0001E        CALLS   #1, NCP$WRITESHO            ;
                               04 00025                        RET                                 ; 3352

; Routine Size:  38 bytes,    Routine Base:  $CODE$ + 1720
```

```
;  3391          3353  1 %SBTTL  'NCP$FAOL  Convert fao list'
;  3392          3354  1 GLOBAL ROUTINE NCP$FAOL (OUTDSC) :NOVALUE =     !
;  3393          3355  1
;  3394          3356  1 !++
;  3395          3357  1 ! FUNCTIONAL DESCRIPTION:
;  3396          3358  1 !
;  3397          3359  1 !       Convert the fao list we have been building.  Return the string in
;  3398          3360  1 !       a string whose descriptor is passed.
;  3399          3361  1 !
;  3400          3362  1 ! FORMAL PARAMETERS:
;  3401          3363  1 !
;  3402          3364  1 !       OUTDSC          Address of descriptor of buffer for text.
;  3403          3365  1 !                       Descriptor is modified to contain the new length.
;  3404          3366  1 !
;  3405          3367  1 ! IMPLICIT INPUTS:
;  3406          3368  1 !
;  3407          3369  1 !       CTRDSC          Descriptor of control string
;  3408          3370  1 !       FAOLST          Fao parameter list
;  3409          3371  1 !
;  3410          3372  1 ! IMPLICIT OUTPUTS:
;  3411          3373  1 !
;  3412          3374  1 !       NONE
;  3413          3375  1 !
;  3414          3376  1 ! ROUTINE VALUE:
;  3415          3377  1 ! COMPLETION CODES:
;  3416          3378  1 !
;  3417          3379  1 !       NONE
;  3418          3380  1 !
;  3419          3381  1 ! SIDE EFFECTS:
;  3420          3382  1 !
;  3421          3383  1 !       NONE
;  3422          3384  1 !
;  3423          3385  1 !--
;  3424          3386  1
;  3425          3387  2     BEGIN
;  3426          3388  2
;  3427          3389  2     MAP
;  3428          3390  2         OUTDSC : REF VECTOR [2]         ! Local description of descriptor
;  3429          3391  2         ;               !
;  3430          3392  2
;  3431        P 3393  2     $FAOL
;  3432        P 3394  2         (
;  3433        P 3395  2         CTRSTR = CTRDSC,                ! Module wide control string
;  3434        P 3396  2         OUTLEN = OUTDSC [0],            ! Descriptor parameter
;  3435        P 3397  2         OUTBUF = .OUTDSC,
;  3436        P 3398  2         PRMLST = FAOLST                 ! Module wide parameter list
;  3437          3399  2         );
;  3438          3400  2
;  3439          3401  2     RETURN
;  3440          3402  2
;  3441          3403  1     END;


                                0000 00000          .ENTRY  NCP$FAOL, Save nothing                          ; 3354
```

```
                         00000000'  00  9F  00002          PUSHAB  FAOLST                          ; 3399
                                04  AC  DD  00008          PUSHL   OUTDSC                          ;
                                04  AC  DD  0000B          PUSHL   OUTDSC                          ;
                         00000000'  00  9F  0000E          PUSHAB  CTRDSC                          ;
               00000000G  00        04  FB  00014          CALLS   #4, SYS$FAOL                    ;
                                    04  00  0001B          RET                                     ; 3403

; Routine Size:  28 bytes,    Routine Base:  $CODE$ + 1746
```

```
;  3443          3404  1 %SBTTL  'NCP$PRIVBITS  Convert Privilege Mask to Text Entries'
;  3444          3405  1 ROUTINE NCP$PRIVBITS (MASK, RLEN) :NOVALUE =    !
;  3445          3406  1
;  3446          3407  1 !++
;  3447          3408  1 ! FUNCTIONAL DESCRIPTION:
;  3448          3409  1 !
;  3449          3410  1 !       Convert a privilege mask to a of bit names entries in the fao list.
;  3450          3411  1 !       Store the list in the fao list and return its length.
;  3451          3412  1 !
;  3452          3413  1 ! FORMAL PARAMETERS:
;  3453          3414  1 !
;  3454          3415  1 !       MASK            Address of privilege mask
;  3455          3416  1 !       RLEN            Address to return length of the fao list
;  3456          3417  1 !
;  3457          3418  1 ! IMPLICIT INPUTS:
;  3458          3419  1 !
;  3459          3420  1 !       PRV$AB_NAMES            Table of bit names and values
;  3460          3421  1 !                               Format :        BYTE minimum name size
;  3461          3422  1 !                                               BYTE bit value
;  3462          3423  1 !                                               ASCIC bit name
;  3463          3424  1 !                                               .
;  3464          3425  1 !                                               .
;  3465          3426  1 !                                               BYTE 0
;  3466          3427  1 !
;  3467          3428  1 ! IMPLICIT OUTPUTS:
;  3468          3429  1 !
;  3469          3430  1 !       NONE
;  3470          3431  1 !
;  3471          3432  1 ! ROUTINE VALUE:
;  3472          3433  1 ! COMPLETION CODES:
;  3473          3434  1 !
;  3474          3435  1 !       NONE
;  3475          3436  1 !
;  3476          3437  1 ! SIDE EFFECTS:
;  3477          3438  1 !
;  3478          3439  1 !       NONE
;  3479          3440  1 !--
```

```
;  3481          3441  1
;  3482          3442  2      BEGIN
;  3483          3443  2
;  3484          3444  2      LOCAL
;  3485          3445  2          NBIT,                           ! Number of the current bit
;  3486          3446  2          PTR,                            ! Pointer to bit
;  3487          3447  2          CTR                             ! Counter for list
;  3488          3448  2          ;                   !
;  3489          3449  2
;  3490          3450  2      EXTERNAL
;  3491          3451  2          PRV$AB_NAMES;                   ! Magic table of privilege names
;  3492          3452  2
;  3493          3453  2      CTR = 0;                            ! No entries yet
;  3494          3454  2      PTR = PRV$AB_NAMES;                 ! Scan the table, bit by bit
;  3495          3455  2
;  3496          3456  2      WHILE .(.PTR) <0, 8, 0> NEQ 0       ! While the table is not done
;  3497          3457  2      DO
;  3498          3458  3          BEGIN
;  3499          3459  3          PTR = .PTR + 1;                 ! Skip the minimum string size
;  3500          3460  3          NBIT = .(.PTR) <0, 8, 0>;       ! Bit number
;  3501          3461  3          PTR = .PTR + 1;                 ! -> ascic name of bit
;  3502          3462  3          IF .(.MASK) <.NBIT, 1, 0>       ! Look at the bit in the mask
;  3503          3463  3          THEN
;  3504          3464  4              BEGIN                       ! If so, add the name
;  3505          3465  4              ADDFAO (.PTR);              ! Add the entry to the fao list
;  3506          3466  4              CTR = .CTR + 1              ! Count the entry
;  3507          3467  4              END
;  3508          3468  3          ;
;  3509          3469  3
;  3510          3470  3          PTR = .PTR + .(.PTR) <0, 8, 0> + 1 ! Skip the string
;  3511          3471  3          END
;  3512          3472  2      ;
;  3513          3473  2
;  3514          3474  2      .RLEN = .CTR;                       ! Return the length of the data
;  3515          3475  2      RETURN
;  3516          3476  2
;  3517          3477  1      END;


                                                    .EXTRN  PRV$AB_NAMES

                                        001C 00000 NCP$PRIVBITS:
                                                    .WORD   Save R2,R3,R4                                   ;  3405
                                             53  D4 00002           CLRL    CTR                             ;  3453
                                   52 00000000G  00  9E 00004           MOVAB   PRV$AB_NAMES, PTR               ;  3454
                                             62  95 00008 1$:        TSTB    (PTR)                           ;  3456
                                             1D  13 0000A           BEQL    3$
                                             52  D6 0000C           INCL    PTR                             ;  3459
                                        54       82  9A 0000E           MOVZBL  (PTR)+, NBIT                    ;  3460
                                   09       04  BC  E1 00011           BBC     NBIT, @MASK, 2$                 ;  3462
                                             52  DD 00016           PUSHL   PTR                             ;  3465
                                        FF6B     CF  FB 00018           CALLS   #1, NCP$ADDFAO
                                             53  D6 0001F           INCL    CTR                             ;  3466
                                        50       62  9A 00021 2$:        MOVZBL  (PTR), R0                       ;  3470
                                        52  01 A042  9E 00024           MOVAB   1(R0)[PTR], PTR
                                             DF  11 00029           BRB     1$                              ;
```

```
                          08   BC            53  D0 0002C 3$:     MOVL    CTR, @RLEN                      ; 3474
                                                 04 00030         RET                                     ; 3477

; Routine Size:  49 bytes,    Routine Base:  $CODE$ + 1762
```

```
; 3519          3478  1 END                                   !End of module
; 3520          3479  0 ELUDOM


                                                              .EXTRN  LIB$STOP

;                                PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $PLIT$                        3312  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $OWN$                         1424  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $GLOBAL$                        12  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                        6035  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)


;                             Library Statistics
;
;                                      -------- Symbols --------      Pages       Processing
;       File                           Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1     9776         3         0         581       00:01.0
;  _$255$DUA28:[NCP.OBJ]NMALIBRY.L32;1    887       111        12          47       00:00.7
;  _$255$DUA28:[NCP.OBJ]NCPLIBRY.L32;1    373        35         9          52       00:00.3


;                                  COMMAND QUALIFIERS

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

; Size:         6035 code + 4748 data bytes
; Run Time:       01:39.6
; Elapsed Time:   05:20.1
; Lines/CPU Min:   2096
; Lexemes/CPU-Min: 13799
; Memory Used:  621 pages
; Compilation Complete
```

0268 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

NCPPRSACT
LIS

NCPSHOLIS
LIS

NCPSHOIO
LIS

NCPPDBS
LIS

NCPSTACNO
LIS

NCPSTALIN
LIS

NCPSTACIR
LIS

NCPSTADUM
LIS

NCPSTADIS
LIS

NCPSTACON
LIS

NCPSTACLR
LIS