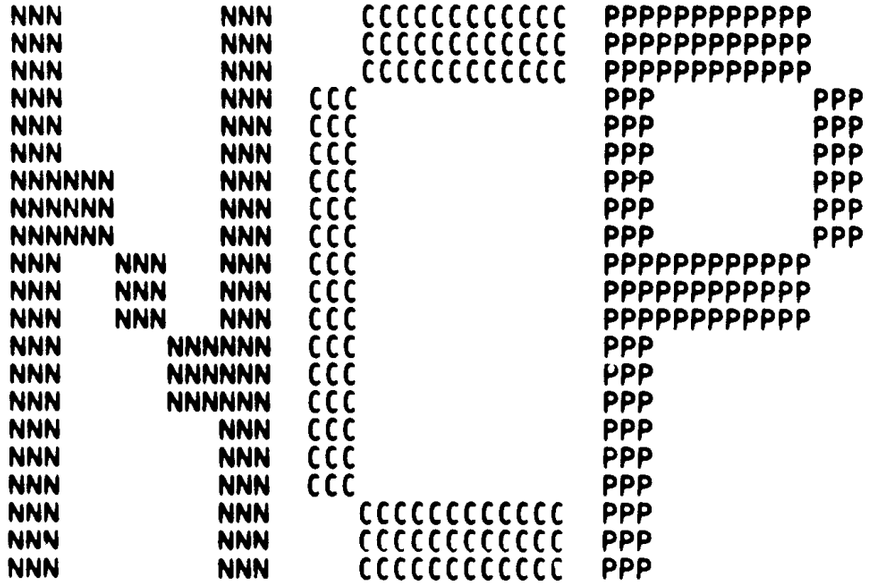
NNN NNN CCCCCCCCCCC PPPPPPPPPPPP
NNN NNN CCCCCCCCCCC PPPPPPPPPPPP
NNN NNN CCCCCCCCCCC PPPPPPPPPPPP
NNN NNN CCC PPP PPP
NNN NNN CCC PPP PPP
NNN NNN CCC PPP PPP
NNNNNN NNN CCC PPP PPP
NNNNNN NNN CCC PPP PPP
NNNNNN NNN CCC PPP PPP
NNN NNN NNN CCC PPPPPPPPPPPP
NNN NNN NNN CCC PPPPPPPPPPPP
NNN NNN NNN CCC 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 CCCCCCCCCCC PPP
NNN NNN CCCCCCCCCCC PPP
NNN NNN CCCCCCCCCCC PPP

```
NN       NN    CCCCCCCC  PPPPPPPP      SSSSSSSS  TTTTTTTTTT    AAAAAA      CCCCCCCC    IIIIII    RRRRRRRR
NN       NN    CCCCCCCC  PPPPPPPP      SSSSSSSS  TTTTTTTTTT    AAAAAA      CCCCCCCC    IIIIII    RRRRRRRR
NN       NN  CC          PP      PP  SS              TT      AA      AA  CC                II      RR      RR
NN       NN  CC          PP      PP  SS              TT      AA      AA  CC                II      RR      RR
NNNN     NN  CC          PP      PP  SS              TT      AA      AA  CC                II      RR      RR
NNNN     NN  CC          PP      PP  SS              TT      AA      AA  CC                II      RR      RR
NN  NN   NN  CC          PPPPPPPP      SSSSSS        TT      AA      AA  CC                II      RRRRRRRR
NN  NN   NN  CC          PPPPPPPP      SSSSSS        TT      AA      AA  CC                II      RRRRRRRR
NN    NNNN  CC          PP                  SS      TT      AAAAAAAAAA  CC                II      RR  RR
NN    NNNN  CC          PP                  SS      TT      AAAAAAAAAA  CC                II      RR  RR
NN      NN  CC          PP                  SS      TT      AA      AA  CC                II      RR    RR        ....
NN      NN  CC          PP                  SS      TT      AA      AA  CC                II      RR    RR        ....
NN      NN    CCCCCCCC  PP          SSSSSSSS        TT      AA      AA    CCCCCCCC    IIIIII    RR      RR        ....
NN      NN    CCCCCCCC  PP          SSSSSSSS        TT      AA      AA    CCCCCCCC    IIIIII    RR      RR        ....

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

```
;  1          0001  0 %TITLE  'Circuit Parameter Parse States and Data'
;  2          0002  0 MODULE NCPSTACIR (IDENT = 'V04-000', LIST(NOOBJECT)) =
;  3          0003  1 BEGIN
;  4          0004  1
;  5          0005  1 !
;  6          0006  1 !*****************************************************************************
;  7          0007  1 !*                                                                           *
;  8          0008  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
;  9          0009  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
; 10          0010  1 !*  ALL RIGHTS RESERVED.                                                     *
; 11          0011  1 !*                                                                           *
; 12          0012  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
; 13          0013  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
; 14          0014  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
; 15          0015  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
; 16          0016  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
; 17          0017  1 !*  TRANSFERRED.                                                             *
; 18          0018  1 !*                                                                           *
; 19          0019  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
; 20          0020  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
; 21          0021  1 !*  CORPORATION.                                                             *
; 22          0022  1 !*                                                                           *
; 23          0023  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
; 24          0024  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
; 25          0025  1 !*                                                                           *
; 26          0026  1 !*                                                                           *
; 27          0027  1 !*****************************************************************************
; 28          0028  1 !
; 29          0029  1
; 30          0030  1 !++
; 31          0031  1 ! FACILITY:     Network Control Program (NCP)
; 32          0032  1 !
; 33          0033  1 ! ABSTRACT:
; 34          0034  1 !
; 35          0035  1 !       States and data for the parsing of NCP circuit parameters
; 36          0036  1 !
; 37          0037  1 ! ENVIRONMENT:  VAX/VMS Operating System
; 38          0038  1 !
; 39          0039  1 ! AUTHOR:       Tim Halvorsen, June 1981
; 40          0040  1 !
; 41          0041  1 ! MODIFIED BY:
; 42          0042  1 !
; 43          0043  1 !       V03-013 RPG0012         Bob Grosso              06-Oct-1982
; 44          0044  1 !               Change Transport type PHASE II to ROUTING III.
; 45          0045  1 !
; 46          0046  1 !       V03-012 RPG0012         Bob Grosso              21-Sep-1982
; 47          0047  1 !               Alter prompting if circuit type is X25.
; 48          0048  1 !
; 49          0049  1 !       V03-011 RPG0011         Bob Grosso              03-Sep-1982
; 50          0050  1 !               Add new transport types.
; 51          0051  1 !
; 52          0052  1 !       V03-010 RPG0010         Bob Grosso              28-Jun-1982
; 53          0053  1 !               Change MAX BLOCK to MAX DATA.
; 54          0054  1 !
; 55          0055  1 !       V009    TMH0009         Tim Halvorsen   10-May-1982
; 56          0056  1 !               Add circuit MRT and RPR parameters for NI support.
; 57          0057  1 !               Add OWNER parameter.
```

```
;    58          0058  1 !
;    59          0059  1 !       V008    TMH0008         Tim Halvorsen   08-Mar-1982
;    60          0060  1 !               Only prompt for "essential" and "important" parameters.
;    61          0061  1 !
;    62          0062  1 !       V007    TMH0007         Tim Halvorsen   20-Jan-1982
;    63          0063  1 !               Add TRANSPORT TYPE parameter.
;    64          0064  1 !               Disable prompting of X25 parameters.
;    65          0065  1 !
;    66          0066  1 !       V006    TMH0006         Tim Halvorsen   08-Jan-1982
;    67          0067  1 !               Remove TMH0003, thus restoring RETRANSMIT TIMER
;    68          0068  1 !               back to a line parameter, which is what NM V3.0
;    69          0069  1 !               finally came up with.
;    70          0070  1 !
;    71          0071  1 !       V005    TMH0005         Tim Halvorsen   15-Aug-1981
;    72          0072  1 !               Add CIRCUIT VERIFICATION parameter.
;    73          0073  1 !
;    74          0074  1 !       V004    TMH0004         Tim Halvorsen   11-Aug-1981
;    75          0075  1 !               Remove fix to TYPE keywords in TMH0002, since
;    76          0076  1 !               only the X25 value may be written using that
;    77          0077  1 !               parameter - all the rest of the values are read-only.
;    78          0078  1 !
;    79          0079  1 !       V003    TMH0003         Tim Halvorsen   05-Aug-1981
;    80          0080  1 !               Make RETRANSMIT TIMER a circuit parameter rather
;    81          0081  1 !               than a line parameter.
;    82          0082  1 !
;    83          0083  1 !       V002    TMH0002         Tim Halvorsen   30-Jul-1981
;    84          0084  1 !               Fix keywords accepted for TYPE parameter.
;    85          0085  1 !               Fix parameter code used for TRIBUTARY.
;    86          0086  1 !
;    87          0087  1 !       V001    TMH0001         Tim Halvorsen   07-Jul-1981
;    88          0088  1 !               Add MAXIMUM TRANSMITS
;    89          0089  1 !--
```

```
;    91          0090  1 %SBTTL  'Definitions'
;    92          0091  1
;    93          0092  1 !
;    94          0093  1 ! INCLUDE FILES:
;    95          0094  1 !
;    96          0095  1
;    97          0096  1         LIBRARY 'LIB$:NMALIBRY';
;    98          0097  1         LIBRARY 'LIB$:NCPLIBRY';
;    99          0098  1         LIBRARY 'SYS$LIBRARY:TPAMAC';
;   100          0099  1
;   101          0100  1 !
;   102          0101  1 ! EXTERNAL REFERENCES:
;   103          0102  1 !
;   104          0103  1
;   105          0104  1         ACT_DFN                 ! Action routine externals
```

```
;  107          0105  1 %SBTTL  'Parameter blocks'
;  108          0106  1
;  109          0107  1 !
;  110          0108  1 ! BIND DATA:
;  111          0109  1 !
;  112          0110  1
;  113          0111  1 !
;  114          0112  1 !       Parameter Blocks
;  115          0113  1 !
;  116          0114  1
;  117        P 0115  1         BUILD_PCL
;  118        P 0116  1
;  119        P 0117  1         (CIR,
;  120        P 0118  1
;  121        P 0119  1         STA, NUMB,      PCCI_STA, ,
;  122        P 0120  1         SER, NUMB,      PCCI_SER, ,
;  123        P 0121  1         CTM, NUMW,      PCCI_LCT, ,
;  124        P 0122  1         COS, NUMB,      PCCI_COS, ,
;  125        P 0123  1         MRT, NUMB,      PCCI_MRT, ,
;  126        P 0124  1         RPR, NUMB,      PCCI_RPR, ,
;  127        P 0125  1         HET, NUMW,      PCCI_HET, ,
;  128        P 0126  1         LIT, NUMW,      PCCI_LIT, ,
;  129        P 0127  1         BLK, NUMB,      PCCI_BLK, ,
;  130        P 0128  1         MRC, NUMB,      PCCI_MRC, ,
;  131        P 0129  1         RCT, NUMW,      PCCI_RCT, ,
;  132        P 0130  1         NUM, TKN,       PCCI_NUM, ,
;  133        P 0131  1         POL, NUMB,      PCCI_POL, ,
;  134        P 0132  1         OWN, ENT,       PCCI_OWN, ,
;  135        P 0133  1         LIN, TKN,       PCCI_LIN, ,
;  136        P 0134  1         USE, NUMB,      PCCI_USE, ,
;  137        P 0135  1         TYP, NUMB,      PCCI_TYP, ,
;  138        P 0136  1         DTE, TKN,       PCCI_DTE, ,
;  139        P 0137  1         CHN, NUMW,      PCCI_CHN, ,
;  140        P 0138  1         MBL, NUMW,      PCCI_MBL, ,
;  141        P 0139  1         MWI, NUMB,      PCCI_MWI, ,
;  142        P 0140  1         TRI, NUMB,      PCCI_TRI, ,
;  143        P 0141  1         BBT, NUMW,      PCCI_BBT, ,
;  144        P 0142  1         TRT, NUMW,      PCCI_TRT, ,
;  145        P 0143  1         MRB, NUMB,      PCCI_MRB, ,
;  146        P 0144  1         MTR, NUMB,      PCCI_MTR, ,
;  147        P 0145  1         ACB, NUMB,      PCCI_ACB, ,
;  148        P 0146  1         ACI, NUMB,      PCCI_ACI, ,
;  149        P 0147  1         IAB, NUMB,      PCCI_IAB, ,
;  150        P 0148  1         IAI, NUMB,      PCCI_IAI, ,
;  151        P 0149  1         IAT, NUMB,      PCCI_IAT, ,
;  152        P 0150  1         DYB, NUMB,      PCCI_DYB, ,
;  153        P 0151  1         DYI, NUMB,      PCCI_DYI, ,
;  154        P 0152  1         DYT, NUMB,      PCCI_DYT, ,
;  155        P 0153  1         DTH, NUMB,      PCCI_DTH, ,
;  156        P 0154  1         VER, NUMB,      PCCI_VER, ,
;  157        P 0155  1         XPT, NUMB,      PCCI_XPT, ,
;  158        P 0156  1
;  159        P 0157  1         . END. . .
;  160        P 0158  1
;  161          0159  1         )
;  162          0160  1
;  163        P 0161  1         BUILD_PBK
```

```
;  164          P 0162  1
;  165          P 0163  1          (CIR,
;  166          P 0164  1
;  167          P 0165  1          STAON, LITB, NMA$C_STATE_ON, CIR_STA,
;  168          P 0166  1          STAOFF, LITB, NMA$C_STATE_OFF, CIR_STA,
;  169          P 0167  1          STASVC, LITB, NMA$C_STATE_SER, CIR_STA,
;  170          P 0168  1          SERENA, LITB, NMA$C_LINSV_ENA, CIR_SER,
;  171          P 0169  1          SERDIS, LITB, NMA$C_LINSV_DIS, CIR_SER,
;  172          P 0170  1          CTM, NUMW, , ,
;  173          P 0171  1          COS, NUMB, , ,
;  174          P 0172  1          MRT, NUMB, , ,
;  175          P 0173  1          RPR, NUMB, , ,
;  176          P 0174  1          HET, NUMW, , ,
;  177          P 0175  1          LIT, NUMW, , ,
;  178          P 0176  1          BLKENA, LITB, NMA$C_CIRBLK_ENA, CIR_BLK,
;  179          P 0177  1          BLKDIS, LITB, NMA$C_CIRBLK_DIS, CIR_BLK,
;  180          P 0178  1          MRC, NUMB, , ,
;  181          P 0179  1          RCT, NUMW, , ,
;  182          P 0180  1          NUM, TKN, , ,
;  183          P 0181  1          POLAUT, LITB, NMA$C_CIRPST_AUT, CIR_POL,
;  184          P 0182  1          POLACT, LITB, NMA$C_CIRPST_ACT, CIR_POL,
;  185          P 0183  1          POLINA, LITB, NMA$C_CIRPST_INA, CIR_POL,
;  186          P 0184  1          POLDIE, LITB, NMA$C_CIRPST_DIE, CIR_POL,
;  187          P 0185  1          POLDED, LITB, NMA$C_CIRPST_DED, CIR_POL,
;  188          P 0186  1          OWNEXE, LITL, NMA$C_ENT_NOD OR (0 ^ 8), CIR_OWN,
;  189          P 0187  1          LIN, TKN, , ,
;  190          P 0188  1          USEPER, LITB, NMA$C_CIRUS_PER, CIR_USE,
;  191          P 0189  1          USEINC, LITB, NMA$C_CIRUS_INC, CIR_USE,
;  192          P 0190  1          USEOUT, LITB, NMA$C_CIRUS_OUT, CIR_USE,
;  193          P 0191  1          TYPX25, LITB, NMA$C_CIRTY_X25, CIR_TYP,
;  194          P 0192  1          DTE, TKN, , ,
;  195          P 0193  1          CHN, NUMW, , ,
;  196          P 0194  1          MBL, NUMW, , ,
;  197          P 0195  1          MWI, NUMB, , ,
;  198          P 0196  1          TRI, NUMB, , ,
;  199          P 0197  1          BBT, NUMW, , ,
;  200          P 0198  1          TRT, NUMW, , ,
;  201          P 0199  1          MRB, NUMB, , ,
;  202          P 0200  1          MRBUNL, LITB, 255, CIR_MRB,
;  203          P 0201  1          MTR, NUMB, , ,
;  204          P 0202  1          ACB, NUMB, , ,
;  205          P 0203  1          ACI, NUMB, , ,
;  206          P 0204  1          IAB, NUMB, , ,
;  207          P 0205  1          IAI, NUMB, , ,
;  208          P 0206  1          IAT, NUMB, , ,
;  209          P 0207  1          DYB, NUMB, , ,
;  210          P 0208  1          DYI, NUMB, , ,
;  211          P 0209  1          DYT, NUMB, , ,
;  212          P 0210  1          DTH, NUMB, , ,
;  213          P 0211  1          VERENA, LITB, NMA$C_CIRVE_ENA, CIR_VER,
;  214          P 0212  1          VERDIS, LITB, NMA$C_CIRVE_DIS, CIR_VER,
;  215          P 0213  1          XPTPH2, LITB, NMA$C_CIRXPT_PH2, CIR_XPT,
;  216          P 0214  1          XPTRO3, LITB, NMA$C_CIRXPT_RO3, CIR_XPT,
;  217          P 0215  1          XPTNR4, LITB, NMA$C_CIRXPT_NR4, CIR_XPT,
;  218          P 0216  1
;  219            0217  1          )
;  220            0218  1
```

```
;  221          P 0219  1          BUILD_SDB
;  222          P 0220  1
;  223            0221  1          (CIR, NMA$C_ENT_CIR, VRB_ENT, CIR)
```

```
;  225          0222  1 %SBTTL  'Prompt strings'
;  226          0223  1
;  227          0224  1 !
;  228          0225  1 !       Build prompt strings
;  229          0226  1 !
;  230          0227  1
;  231          0228  1 BIND
;  232        P 0229  1         PROMPT_STRINGS
;  233        P 0230  1         (CIR,
;  234        P 0231  1
;  235        P 0232  1         STA, 'Circuit state        (ON, OFF, SERVICE): ',
;  236        P 0233  1 !       SER, 'Service mode        (ENABLED, DISABLED): ',
;  237        P 0234  1 !       CTM, 'Counter timer         (1-65535 seconds): ',
;  238        P 0235  1         COS, 'Cost of circuit                (1-125): ',
;  239        P 0236  1 !       HET, 'Hello timer           (1-65535 seconds): ',
;  240        P 0237  1 !       LIT, 'Listen timer          (1-65535 seconds): ',
;  241        P 0238  1 !       BLK, 'Blocking            (ENABLED, DISABLED): ',
;  242        P 0239  1         MRC, 'Maximum recalls                (0-255): ',
;  243        P 0240  1         RCT, 'Recall timer          (1-65535 seconds): ',
;  244        P 0241  1         NUM, 'Number                   (1-16 digits): ',
;  245        L 0242  1         POL, %STRING('Polling state (AUTOMATIC, ACTIVE,', CRLF,
;  246        P 0243  1                      ' INACTIVE, DYING, DEAD): '),
;  247        P 0244  1         OWN, 'Owner                  (EXECUTOR NODE): ',
;  248        P 0245  1 !!      LIN, 'Line ID                (1-16 characters): ',
;  249        P 0246  1         USE, 'Usage   (INCOMING, OUTGOING, PERMANENT): ',
;  250        P 0247  1         TYP, 'Type                      (X25 or <CR>): ',
;  251        P 0248  1 !       VER, 'Verification        (ENABLED, DISABLED): ',
;  252        P 0249  1 !       XPT, 'Transport type  (PHASE II, ROUTING III): ',
;  253        P 0250  1         DTE, 'X.25 DTE address         (1-16 digits): ',
;  254        P 0251  1         CHN, 'X.25 channel number          (0-4095): ',
;  255        P 0252  1         MBL, 'X.25 maximum data size      (1-65535): ',
;  256        P 0253  1         MWI, 'X.25 maximum window size (1-255 blocks): ',
;  257        P 0254  1         TRI, 'Tributary address              (0-255): ',
;  258        P 0255  1 !       BBT, 'Babble timer     (1-65535 milliseconds): ',
;  259        P 0256  1 !       TRT, 'Transmit timer   (0-65535 milliseconds): ',
;  260        P 0257  1 !       MRB, 'Maximum buffers      (1-254, UNLIMITED): ',
;  261        P 0258  1 !       MTR, 'Maximum transmits      (1-255 messages): ',
;  262        P 0259  1 !       ACB, 'Active base priority           (0-255): ',
;  263        P 0260  1 !       ACI, 'Active priority increment      (0-255): ',
;  264        P 0261  1 !       IAB, 'Inactive base priority         (0-255): ',
;  265        P 0262  1 !       IAI, 'Inactive priority increment    (0-255): ',
;  266        P 0263  1 !       IAT, 'Inactive threshold             (0-255): ',
;  267        P 0264  1 !       DYB, 'Dying base priority            (0-255): ',
;  268        P 0265  1 !       DYI, 'Dying priority increment       (0-255): ',
;  269        P 0266  1 !       DYT, 'Dying threshold                (0-255): ',
;  270        P 0267  1 !       DTH, 'Dead threshold                 (0-255): ',
;  271        P 0268  1
;  272          0269  1         );
```

```
;  274          0270  1 %SBTTL  'State table for circuit parameters'
;  275          0271  1
;  276          0272  1 $INIT_STATE(NCP$G_STTBL_CIR, NCP$G_KYTBL_CIR);
;  277          0273  1
;  278          0274  1 !
;  279          0275  1 !       SET/DEFINE CIRCUIT parameters
;  280          0276  1 !
;  281          0277  1
;  282        P 0278  1 $STATE  (ST_CIR,
;  283        P 0279  1          ((SE_ALL), ST_CIR_DOIT),                 ! ALL parameter
;  284        P 0280  1          (TPA$_EOS, , ACT$PMT_ON),                ! Prompt if no keywords
;  285        P 0281  1          (TPA$_LAMBDA, ST_CIR_PRC, ACT$PMT_OFF)   ! Process keywords
;  286          0282  1          );
;  287          0283  1
;  288          0284  1 !
;  289          0285  1 ! Find out whether circuit type is X25
;  290          0286  1 !
;  291          0287  1
;  292        P 0288  1 $STATE  (ST_CIR_PMT_TYP,
;  293          0289  1          (TPA$_LAMBDA, , ACT$PRMPT, , , PMT$G_CIR_TYP));
;  294        P 0290  1 $STATE  (,
;  295        P 0291  1          (TPA$_SYMBOL, ST_CIR_DOIT, ACT$PMTDONEQ),
;  296        P 0292  1          ((ST_CIR_TYP), ST_CIR_PMT_TYP_X25),
;  297        P 0293  1          (TPA$_EOS, ST_CIR_PMT_TYP_CR),
;  298        P 0294  1          (TPA$_LAMBDA, ST_CIR_PMT_TYP, ACT$SIGNAL, , , NCP$_INVVAL)
;  299          0295  1          );
;  300          0296  1
;  301          0297  1 !
;  302          0298  1 !       Prompting when TYP was <CR>
;  303          0299  1 !
;  304        P 0300  1 $STATE  (ST_CIR_PMT_TYP_CR,
;  305          0301  1          (TPA$_LAMBDA));
;  306          0302  1
;  307        P 0303  1          PROMPT_STATES
;  308        P 0304  1          (CIR,
;  309        P 0305  1
;  310        P 0306  1          STA, COS, TRI,
;  311        P 0307  1 !        BLK, LIN
;  312          0308  1          )
;  313          0309  1
;  314        P 0310  1 $STATE  (,
;  315        P 0311  1          (TPA$_LAMBDA, ST_CIR_DOIT)
;  316          0312  1          );
;  317          0313  1
;  318          0314  1 !
;  319          0315  1 !       Prompting when TYP was X25
;  320          0316  1 !
;  321        P 0317  1 $STATE  (ST_CIR_PMT_TYP_X25,
;  322          0318  1          (TPA$_LAMBDA));
;  323          0319  1
;  324        P 0320  1          PROMPT_STATES
;  325        P 0321  1          (CIR,
;  326        P 0322  1
;  327        P 0323  1          USE, OWN, POL, DTE, CHN, NUM, MRC, RCT, MBL, MWI,
;  328          0324  1          )
;  329          0325  1
;  330        P 0326  1 $STATE  (ST_CIR_DOIT,
```

```
;  331        P 0327  1          (TPA$_EOS, TPA$_EXIT, ACT$VRB_UTILITY, , , SDB$G_CIR)
;  332          0328  1          );
```

```
;   334          0329  1 %SBTTL  'Dispatch on parameter keywords'
;   335          0330  1
;   336          0331  1 !
;   337          0332  1 !       Dispatch on circuit parameter keywords
;   338          0333  1 !
;   339          0334  1
;   340        P 0335  1 $STATE  (ST_CIR_PRC,
;   341        P 0336  1
;   342        P 0337  1          DISPATCH_STATES
;   343        P 0338  1          (CIR,
;   344        P 0339  1
;   345        P 0340  1          ACT, 'ACTIVE',
;   346        P 0341  1          BBT, 'BABBLE',
;   347        P 0342  1          BLK, 'BLOCKING',
;   348        P 0343  1          CHN, 'CHANNEL',
;   349        P 0344  1          COS, 'COST',
;   350        P 0345  1          CTM, 'COUNTER',
;   351        P 0346  1          DTH, 'DEAD',
;   352        P 0347  1          DTE, 'DTE',
;   353        P 0348  1          DYE, 'DYING',
;   354        P 0349  1          HET, 'HELLO',
;   355        P 0350  1          IAC, 'INACTIVE',
;   356        P 0351  1          LIN, 'LINE',
;   357        P 0352  1          LIT, 'LISTEN',
;   358        P 0353  1          MAX, 'MAXIMUM',
;   359        P 0354  1          NUM, 'NUMBER',
;   360        P 0355  1          OWN, 'OWNER',
;   361        P 0356  1          POL, 'POLLING',
;   362        P 0357  1          RCT, 'RECALL',
;   363        P 0358  1          RPR, 'ROUTER',
;   364        P 0359  1          SER, 'SERVICE',
;   365        P 0360  1          STA, 'STATE',
;   366        P 0361  1          TRT, 'TRANSMIT',
;   367        P 0362  1          XPT, 'TRANSPORT',
;   368        P 0363  1          TRI, 'TRIBUTARY',
;   369        P 0364  1          TYP, 'TYPE',
;   370        P 0365  1          USE, 'USAGE',
;   371        P 0366  1          VER, 'VERIFICATION',
;   372        P 0367  1
;   373        P 0368  1          ),
;   374        P 0369  1
;   375        P 0370  1          (TPA$_EOS, ST_CIR_DOIT)
;   376          0371  1          );
```

```
;  378          0372  1
;  379          0373  1 !
;  380          0374  1 !       Dispatch on MAXIMUM keyword
;  381          0375  1 !
;  382          0376  1
;  383        P 0377  1 $STATE  (ST_CIR_PRC_MAX,
;  384        P 0378  1
;  385        P 0379  1          DISPATCH_STATES
;  386        P 0380  1          (CIR,
;  387        P 0381  1
;  388        P 0382  1          MBL, 'DATA',
;  389        P 0383  1          MRB, 'BUFFERS',
;  390        P 0384  1          MRC, 'RECALLS',
;  391        P 0385  1          MRT, 'ROUTERS',
;  392        P 0386  1          MTR, 'TRANSMITS',
;  393        P 0387  1          MWI, 'WINDOW',
;  394        P 0388  1
;  395          0389  1          ));
;  396          0390  1
;  397          0391  1 !
;  398          0392  1 !       Dispatch on ACTIVE keyword
;  399          0393  1 !
;  400          0394  1
;  401        P 0395  1 $STATE  (ST_CIR_PRC_ACT,
;  402        P 0396  1
;  403        P 0397  1          DISPATCH_STATES
;  404        P 0398  1          (CIR,
;  405        P 0399  1
;  406        P 0400  1          ACB, 'BASE',
;  407        P 0401  1          ACI, 'INCREMENT',
;  408        P 0402  1
;  409          0403  1          ));
;  410          0404  1
;  411          0405  1 !
;  412          0406  1 !       Dispatch for INACTIVE keyword
;  413          0407  1 !
;  414          0408  1
;  415        P 0409  1 $STATE  (ST_CIR_PRC_IAC,
;  416        P 0410  1
;  417        P 0411  1          DISPATCH_STATES
;  418        P 0412  1          (CIR,
;  419        P 0413  1
;  420        P 0414  1          IAB, 'BASE',
;  421        P 0415  1          IAI, 'INCREMENT',
;  422        P 0416  1          IAT, 'THRESHOLD',
;  423        P 0417  1
;  424          0418  1          ));
;  425          0419  1
;  426          0420  1 !
;  427          0421  1 !       Dispatch for DYING keyword
;  428          0422  1 !
;  429          0423  1
;  430        P 0424  1 $STATE  (ST_CIR_PRC_DYE,
;  431        P 0425  1
;  432        P 0426  1          DISPATCH_STATES
;  433        P 0427  1          (CIR,
;  434        P 0428  1
```

```
;  435          P 0429  1          DYB, 'BASE',
;  436          P 0430  1          DYI, 'INCREMENT',
;  437          P 0431  1          DYT, 'THRESHOLD',
;  438          P 0432  1
;  439            0433  1          ));
```

```
;  441          0434  1
;  442          0435  1 !
;  443          0436  1 !       These states take care of noise words
;  444          0437  1 !       and call the subexpressions to do the work
;  445          0438  1 !
;  446          0439  1
;  447        P 0440  1         PROCESS_STATES
;  448        P 0441  1         (CIR,
;  449        P 0442  1
;  450        P 0443  1         STA, ,
;  451        P 0444  1         SER, ,
;  452        P 0445  1         CTM, 'TIMER',
;  453        P 0446  1         COS, ,
;  454        P 0447  1         MRT, ,                          ! From MAX
;  455        P 0448  1         RPR, 'PRIORITY',
;  456        P 0449  1         HET, 'TIMER',
;  457        P 0450  1         LIT, 'TIMER',
;  458        P 0451  1         BLK, ,
;  459        P 0452  1         MRC, ,                          ! From MAX
;  460        P 0453  1         RCT, 'TIMER',
;  461        P 0454  1         NUM,
;  462        P 0455  1         POL, 'STATE',
;  463        P 0456  1         OWN, ,
;  464        P 0457  1         LIN, ,
;  465        P 0458  1         USE, ,
;  466        P 0459  1         TYP, ,
;  467        P 0460  1         DTE, ,
;  468        P 0461  1         CHN, ,
;  469        P 0462  1         MBL, ,                          ! From MAX
;  470        P 0463  1         MWI, ,                          ! From MAX
;  471        P 0464  1         TRI,
;  472        P 0465  1         BBT, 'TIMER',
;  473        P 0466  1         TRT, 'TIMER',
;  474        P 0467  1         MRB, ,                          ! From MAX
;  475        P 0468  1         MTR, ,                          ! From MAX
;  476        P 0469  1         ACB, ,                          ! From ACT
;  477        P 0470  1         ACI, ,                          ! From ACT
;  478        P 0471  1         IAB, ,                          ! From IAC
;  479        P 0472  1         IAI, ,                          ! From IAC
;  480        P 0473  1         IAT, ,                          ! From IAC
;  481        P 0474  1         DYB, ,                          ! From DYE
;  482        P 0475  1         DYI, ,                          ! From DYE
;  483        P 0476  1         DYT, ,                          ! From DYE
;  484        P 0477  1         DTH, 'THRESHOLD',
;  485        P 0478  1         VER,
;  486        P 0479  1         XPT, 'TYPE',
;  487        P 0480  1
;  488          0481  1         )
```

```
;  490          0482  1
;  491          0483  1 !
;  492          0484  1 !       These subexpressions store away the parameter values
;  493          0485  1 !
;  494          0486  1
;  495        P 0487  1          SUB_EXPRESSIONS
;  496        P 0488  1          (CIR,
;  497        P 0489  1
;  498        P 0490  1          CTM, TPA$_DECIMAL,
;  499        P 0491  1          COS, TPA$_DECIMAL,
;  500        P 0492  1          MRT, TPA$_DECIMAL,
;  501        P 0493  1          RPR, TPA$_DECIMAL,
;  502        P 0494  1          HET, TPA$_DECIMAL,
;  503        P 0495  1          LIT, TPA$_DECIMAL,
;  504        P 0496  1          MRC, TPA$_DECIMAL,
;  505        P 0497  1          RCT, TPA$_DECIMAL,
;  506        P 0498  1          NUM, (SE_DTE_NUMBER),
;  507        P 0499  1          LIN, (SE_LINE_ID),
;  508        P 0500  1          DTE, (SE_DTE_NUMBER),
;  509        P 0501  1          CHN, TPA$_DECIMAL,
;  510        P 0502  1          MBL, TPA$_DECIMAL,
;  511        P 0503  1          MWI, TPA$_DECIMAL,
;  512        P 0504  1          TRI, TPA$_DECIMAL,
;  513        P 0505  1          BBT, TPA$_DECIMAL,
;  514        P 0506  1          TRT, TPA$_DECIMAL,
;  515        P 0507  1          MTR, TPA$_DECIMAL,
;  516        P 0508  1          ACB, TPA$_DECIMAL,
;  517        P 0509  1          ACI, TPA$_DECIMAL,
;  518        P 0510  1          IAB, TPA$_DECIMAL,
;  519        P 0511  1          IAI, TPA$_DECIMAL,
;  520        P 0512  1          IAT, TPA$_DECIMAL,
;  521        P 0513  1          DYB, TPA$_DECIMAL,
;  522        P 0514  1          DYI, TPA$_DECIMAL,
;  523        P 0515  1          DYT, TPA$_DECIMAL,
;  524        P 0516  1          DTH, TPA$_DECIMAL,
;  525        P 0517  1
;  526        P 0518  1          XPTPH2, TPA$_LAMBDA,
;  527        P 0519  1          XPTRO3, TPA$_LAMBDA,
;  528        P 0520  1          XPTNR4, TPA$_LAMBDA,
;  529          0521  1          )
```

```
;  531          0522  1
;  532          0523  1 !
;  533          0524  1 !       Circuit state
;  534          0525  1 !
;  535          0526  1
;  536        P 0527  1 $STATE  (ST_CIR_STA,
;  537        P 0528  1
;  538        P 0529  1          KEYWORD_STATE
;  539        P 0530  1          (CIR,
;  540        P 0531  1          STAOFF, 'OFF',
;  541        P 0532  1          STAON, 'ON',
;  542        P 0533  1          STASVC, 'SERVICE',
;  543          0534  1          ));
;  544          0535  1
;  545          0536  1 !
;  546          0537  1 !       Circuit service mode
;  547          0538  1 !
;  548          0539  1
;  549        P 0540  1 $STATE  (ST_CIR_SER,
;  550        P 0541  1
;  551        P 0542  1          KEYWORD_STATE
;  552        P 0543  1          (CIR,
;  553        P 0544  1          SERENA, 'ENABLED',
;  554        P 0545  1          SERDIS, 'DISABLED',
;  555          0546  1          ));
;  556          0547  1
;  557          0548  1 !
;  558          0549  1 !       Circuit blocking mode
;  559          0550  1 !
;  560          0551  1
;  561        P 0552  1 $STATE  (ST_CIR_BLK,
;  562        P 0553  1
;  563        P 0554  1          KEYWORD_STATE
;  564        P 0555  1          (CIR,
;  565        P 0556  1          BLKENA, 'ENABLED',
;  566        P 0557  1          BLKDIS, 'DISABLED',
;  567          0558  1          ));
;  568          0559  1
;  569          0560  1 !
;  570          0561  1 !       Circuit polling state
;  571          0562  1 !
;  572          0563  1
;  573        P 0564  1 $STATE  (ST_CIR_POL,
;  574        P 0565  1
;  575        P 0566  1          KEYWORD_STATE
;  576        P 0567  1          (CIR,
;  577        P 0568  1          POLAUT, 'AUTOMATIC',
;  578        P 0569  1          POLACT, 'ACTIVE',
;  579        P 0570  1          POLINA, 'INACTIVE',
;  580        P 0571  1          POLDIE, 'DYING',
;  581        P 0572  1          POLDED, 'DEAD',
;  582          0573  1          ));
;  583          0574  1
;  584          0575  1 !
;  585          0576  1 !       Circuit usage
;  586          0577  1 !
;  587          0578  1
```

```
;  588        P 0579  1 $STATE  (ST_CIR_USE,
;  589        P 0580  1
;  590        P 0581  1          KEYWORD_STATE
;  591        P 0582  1          (CIR,
;  592        P 0583  1          USEINC, 'INCOMING',
;  593        P 0584  1          USEOUT, 'OUTGOING',
;  594        P 0585  1          USEPER, 'PERMANENT',
;  595          0586  1          ));
;  596          0587  1
;  597          0588  1 !
;  598          0589  1 !        Circuit type
;  599          0590  1 !
;  600          0591  1
;  601        P 0592  1 $STATE  (ST_CIR_TYP,
;  602        P 0593  1
;  603        P 0594  1          KEYWORD_STATE
;  604        P 0595  1          (CIR,
;  605        P 0596  1          TYPX25, 'X25',
;  606          0597  1          ));
;  607          0598  1
;  608          0599  1 !
;  609          0600  1 !        Circuit verification mode
;  610          0601  1 !
;  611          0602  1
;  612        P 0603  1 $STATE  (ST_CIR_VER,
;  613        P 0604  1
;  614        P 0605  1          KEYWORD_STATE
;  615        P 0606  1          (CIR,
;  616        P 0607  1          VERENA, 'ENABLED',
;  617        P 0608  1          VERDIS, 'DISABLED',
;  618          0609  1          ));
;  619          0610  1
;  620          0611  1 !
;  621          0612  1 !        Circuit desired transport type
;  622          0613  1 !
;  623          0614  1
;  624      P 0615  1 $STATE  (ST_CIR_XPT,
;  625      P 0616  1
;  626      P 0617  1          ('NONROUTING',  ST_CIR_XPTNON),
;  627      P 0618  1          ('PHASE',       ST_CIR_XPTPHA),
;  628      P 0619  1          ('ROUTING',     ST_CIR_XPTROU),
;  629        0620  1          );
;  630        0621  1
;  631      P 0622  1 $STATE  (ST_CIR_XPTNON,
;  632      P 0623  1          ('IV',          ST_CIR_XPTNR4),
;  633        0624  1          );
;  634        0625  1
;  635      P 0626  1 $STATE  (ST_CIR_XPTPHA,
;  636      P 0627  1          ('II',          ST_CIR_XPTPH2),
;  637      P 0628  1          ('2',           ST_CIR_XPTPH2),
;  638        0629  1          );
;  639        0630  1
;  640      P 0631  1 $STATE  (ST_CIR_XPTROU,
;  641      P 0632  1          ('III',         ST_CIR_XPTRO3),
;  642      P 0633  1          ('3',           ST_CIR_XPTRO3),
;  643        0634  1          );
;  644        0635  1
```

```
;  645          0636  1 !
;  646          0637  1 !       Maximum circuit receive buffers
;  647          0638  1 !
;  648          0639  1
;  649        P 0640  1 $STATE  (ST_CIR_MRB,
;  650        P 0641  1          ('UNLIMITED', TPA$_EXIT, ACT$SAVPRM,,, PBK$G_CIR_MRBUNL),
;  651        P 0642  1          (TPA$_DECIMAL, TPA$_EXIT, ACT$SAVPRM,,, PBK$G_CIR_MRB)
;  652          0643  1          );
;  653          0644  1
;  654          0645  1 !
;  655          0646  1 !       Circuit owner (only EXECUTOR value allowed)
;  656          0647  1 !
;  657          0648  1
;  658        P 0649  1 $STATE  (ST_CIR_OWN,
;  659        P 0650  1
;  660        P 0651  1          KEYWORD_STATE
;  661        P 0652  1          (CIR,
;  662        P 0653  1          OWNEXE, 'EXECUTOR',
;  663          0654  1          ));
```

```
;  665     0655  1 %SBTTL  'Define shared subexpressions'
;  666     0656  1
;  667     0657  1 !
;  668     0658  1 !       Define Subexpressions from Library
;  669     0659  1 !
;  670     0660  1
;  671     0661  1         SEM_ALL                 ! All parameter
;  672     0662  1         SEM_LINE_ID             ! Device name string
;  673     0663  1         SEM_NODE_ID             ! Circuit owner id
;  674     0664  1         SEM_DTE_NUMBER          ! DTE call number
```

```
;  676          0665  1 %SBTTL  'Object Listing of Parse Table'
;  677          0666  1
;  678          0667  1 END                                   !End of module
;  679          0668  0 ELUDOM
```

NCPSTACNO
LIS

NCPSTALIN
LIS

NCPSTACIR
LIS

NCPSTADUM
LIS

NCPSTADIS
LIS

NCPSTACON
LIS

NCPSTACLR
LIS