

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	CCC	PPP	
NNN		NNN	CCCCCCCCCCCC	PPP	
NNN		NNN	CCCCCCCCCCCC	PPP	
NNN		NNN	CCCCCCCCCCCC	PPP	



```

NN      NN      CCCCCCCC  PPPPPPPP  SSSSSSSS  TTTTTTTTTT  AAAAAA  MM      MM  TTTTTTTTTT  RRRRRRRR
NN      NN      CCCCCCCC  PPPPPPPP  SSSSSSSS  TTTTTTTTTT  AAAAAA  MM      MM  TTTTTTTTTT  RRRRRRRR
NN      NN      CC          PP        PP  SS          TT          AA      AA  MMMM  MMMM  TT          RR      RR
NN      NN      CC          PP        PP  SS          TT          AA      AA  MMMM  MMMM  TT          RR      RR
NNNN    NN      CC          PP        PP  SS          TT          AA      AA  MM    MM  MM      TT          RR      RR
NNNN    NN      CC          PP        PP  SS          TT          AA      AA  MM    MM  MM      TT          RR      RR
NN      NN      CC          PPPPPPPP  SSSSSS    TT          AA      AA  MM      MM  TT          RRRRRRRR
NN      NN      CC          PPPPPPPP  SSSSSS    TT          AA      AA  MM      MM  TT          RRRRRRRR
NN      NN      CC          PP        SS          TT          AA      AA  MM      MM  TT          RR      RR
NN      NN      CC          PP        SS          TT          AA      AA  MM      MM  TT          RR      RR
NN      NN      CC          PP        SS          TT          AA      AA  MM      MM  TT          RR      RR
NN      NN      CC          PP        SS          TT          AA      AA  MM      MM  TT          RR      RR
NN      NN      CCCCCCCC  PP        SSSSSSSS  TT          AA      AA  MM      MM  TT          RR      RR
NN      NN      CCCCCCCC  PP        SSSSSSSS  TT          AA      AA  MM      MM  TT          RR      RR

```

```

LL      111111  SSSSSSSS
LL      111111  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 111111  SSSSSSSS
LLLLLLLLLL 111111  SSSSSSSS

```

.....

....
....
....
....

```

1 0001 0 %TITLE 'X.25 Trace Module Parsing'
2 0002 0 MODULE NCPSTAMTR (IDENT = 'V04-000',LIST(NOOBJECT)) =
3 0003 1 BEGIN
4 0004 1
5 0005 1
6 0006 1
7 0007 1
8 0008 1 *
9 0009 1 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
10 0010 1 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
11 0011 1 * ALL RIGHTS RESERVED.
12 0012 1 *
13 0013 1 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
14 0014 1 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
15 0015 1 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
16 0016 1 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
17 0017 1 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
18 0018 1 * TRANSFERRED.
19 0019 1 *
20 0020 1 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
21 0021 1 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
22 0022 1 * CORPORATION.
23 0023 1 *
24 0024 1 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
25 0025 1 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
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 X.25 Trace module parameters
36 0036 1
37 0037 1 ENVIRONMENT: VAX/VMS Operating System
38 0038 1
39 0039 1 AUTHOR: Bob Grosso July 1982
40 0040 1
41 0041 1 MODIFIED BY:
42 0042 1
43 0043 1 V03-005 RPG0005 Bob Grosso 04-Nov-1982
44 0044 1 Correct SET X-T K T STATE ON from parsing STA instead
45 0045 1 of TST.
46 0046 1
47 0047 1 V03-004 RPG0004 Bob Grosso 15-Sep-1982
48 0048 1 Flag presence of qualifier on command line so that
49 0049 1 the ALL check in NCPVRBACT will work properly.
50 0050 1 Make a noise word of X25-T in SET/CLEAR tracepoint.
51 0051 1
52 0052 1 V03-003 RPG0003 Bob Grosso 03-Sep-1982
53 0053 1 Fix parameter ranges in some prompt strings.
54 0054 1
55 0055 1 V03-002 TMH0002 Tim Halvorsen 16-Aug-1982
56 0056 1 Fix PCL table so that literal parameters such as STATE
57 0057 1 as coded as NUMB rather than LITB, since that is the

```

: 58
: 59
: 60
: 61
: 62
: 63
: 64
: 65
: 66
: 67
: 68
: 69
: 70
: 71

0058 1 |
0059 1 |
0060 1 |
0061 1 |
0062 1 |
0063 1 |
0064 1 |
0065 1 |
0066 1 |
0067 1 |
0068 1 |
0069 1 |
0070 1 |
0071 1 |--

way byte values are put into the NICE message. (In a
PCL list, LITB means store the parameter code, but no
value - used in CLEAR/PURGE).
Fix so that TRACEPOINT STATE is distinguished from
X25-TRACE STATE, and so that a different parameter code
is sent for each.
Fix CLEAR TPT PBK to correctly indicate that it is a TKN
rather than a LITB, so that the tracepoint name is sent
correctly in the NICE message.

V03-001 RPG0001 Bob Grosso 03-Aug-1982
Enhance prompting for ALL

```
.. 73      0072 1 %SBTTL 'Definitions'
.. 74      0073 1
.. 75      0074 1
.. 76      0075 1  ! INCLUDE FILES:
.. 77      0076 1  !
.. 78      0077 1
.. 79      0078 1      LIBRARY 'LIBS:NMALIBRY';
.. 80      0079 1      LIBRARY 'LIBS:NCPLIBRY';
.. 81      0080 1      LIBRARY 'SYSSLIBRARY:TPAMAC';
.. 82      0081 1
.. 83      0082 1
.. 84      0083 1  ! EXTERNAL REFERENCES:
.. 85      0084 1  !
.. 86      0085 1
.. 87      0086 1      ACT_DFN          ! Action routine externals
.. 88      0087 1
.. 89      0088 1  EXTERNAL
.. 90      0089 1      NCP$GL_QUALPRS;    ! Flag presence of qualifier
.. 91      0090 1
.. 92      0091 1  !
.. 93      0092 1  ! OWN storage
.. 94      0093 1  !
.. 95      0094 1
.. 96      0095 1  OWN
.. 97      0096 1      TPT_PARAMS;        ! True if tracepoint parameters
.. 98      0097 1      ! False if trace-wide parameters
.. 99      0098 1
.. 100     0099 1  !
.. 101     0100 1  ! LITERALS
.. 102     0101 1  !
.. 103     0102 1
.. 104     0103 1  LITERAL
.. 105     0104 1      QUALPRESENT = 1;    ! Flag presence of qualifier on command line
.. 106     0105 1
```

```

: 108      0106 1 %SBT!L 'Set Parameter blocks'
: 109      0107 1
: 110      0108 1
: 111      0109 1      Set X25-TRACE Parameter Blocks
: 112      0110 1
: 113      0111 1
: 114      P 0112 1      BUILD_PCL
: 115      P P 0113 1
: 116      P P 0114 1      (MTR,                               ! Module X25-TRACE
: 117      P P 0115 1
: 118      P P 0116 1      TPT, TKN,          PCXT_TPT, .
: 119      P P 0117 1
: 120      P P 0118 1      STA, NUMB,          PCXT_STA, .
: 121      P P 0119 1      BSZ, NUMW,          PCXT_BSZ, .
: 122      P P 0120 1      MBK, NUMW,          PCXT_MBK, .
: 123      P P 0121 1      FNM, TKN,          PCXT_FNM, .
: 124      P P 0122 1      MBF, NUMW,          PCXT_MBF, .
: 125      P P 0123 1      CPL, NUMW,          PCXT_CPL, .
: 126      P P 0124 1      MVR, NUMW,          PCXT_MVR, .
: 127      P P 0125 1
: 128      P P 0126 1      CPS, NUMW,          PCXT_CPS, .
: 129      P P 0127 1      TST, NUMB,          PCXT_TST, .
: 130      P P 0128 1
: 131      P P 0129 1      , END, . .
: 132      P 0130 1
: 133      P 0131 1      )
: 134      P 0132 1
: 135      P 0133 1      BUILD_PBK
: 136      P P 0134 1
: 137      P P 0135 1      (MTR,                               ! Module X25-TRACE
: 138      P P 0136 1
: 139      P P 0137 1      STAON, LITB, NMA$C_STATE_ON, MTR_STA,
: 140      P P 0138 1      STAOFF, LITB, NMA$C_STATE_OFF, MTR_STA,
: 141      P P 0139 1      BSZ, NUMW, . .
: 142      P P 0140 1      MBK, NUMW, . .
: 143      P P 0141 1      FNM, TKN, . .
: 144      P P 0142 1      MBF, NUMW, . .
: 145      P P 0143 1      CPL, NUMW, . .
: 146      P P 0144 1      MVR, NUMW, . .
: 147      P P 0145 1      TPT, TKN,
: 148      P P 0146 1      KTP, LITB, NMA$C_ENT_KNO, MTR_TPT,      ! Known trace points
: 149      P P 0147 1      CPS, NUMW,
: 150      P P 0148 1      TSTON, LITB, NMA$C_STATE_ON, MTR_TST,
: 151      P P 0149 1      TSTOFF, LITB, NMA$C_STATE_OFF, MTR_TST,
: 152      P 0150 1
: 153      P 0151 1      )
: 154      P 0152 1
: 155      P 0153 1      BIND      PDB$G_MTR_ENT = UPLIT BYTE(0, %ASCIC 'X25-TRACE');
: 156      P 0154 1
: 157      P 0155 1      BUILD_SDB
: 158      P 0156 1      (MTR, NMA$C_ENT_MOD, MTR_ENT, MTR)

```

```
160      0157 1 %SBTTL 'Clear Parameter blocks'
161      0158 1
162      0159 1
163      0160 1      Clear X25-TRACE Parameter Blocks
164      0161 1
165      0162 1
166      P 0163 1      BUILD_PCL
167      P P 0164 1
168      P P 0165 1      (CTR,                      ! Module X25-TRACE
169      P P 0166 1
170      P P 0167 1      TPT, TKN,          PCXT_TPT, .
171      P P 0168 1
172      P P 0169 1      STA, LITB,          PCXT_STA, .
173      P P 0170 1      BSZ, LITB,          PCXT_BSZ, .
174      P P 0171 1      MBK, LITB,          PCXT_MBK, .
175      P P 0172 1      FNM, LITB,          PCXT_FNM, .
176      P P 0173 1      MBF, LITB,          PCXT_MBF, .
177      P P 0174 1      CPL, LITB,          PCXT_CPL, .
178      P P 0175 1      MVR, LITB,          PCXT_MVR, .
179      P P 0176 1
180      P P 0177 1      CPS, LITB,          PCXT_CPS, .
181      P P 0178 1      TST, LITB,          PCXT_TST, .
182      P P 0179 1
183      P P 0180 1      . END, . . .
184      P 0181 1
185      0182 1      )
186      0183 1
187      P 0184 1      BUILD_PBK
188      P P 0185 1
189      P P 0186 1      (CTR,                      ! Module X25-TRACE
190      P P 0187 1
191      P P 0188 1      ALL, LITB, 0, VRB_ALL,
192      P P 0189 1
193      P P 0190 1      STA, LITB, 0, .
194      P P 0191 1      BSZ, LITB, 0, .
195      P P 0192 1      MBK, LITB, 0, .
196      P P 0193 1      FNM, LITB, 0, .
197      P P 0194 1      MBF, LITB, 0, .
198      P P 0195 1      CPL, LITB, 0, .
199      P P 0196 1      MVR, LITB, 0, .
200      P P 0197 1      TPT, TKN, 0, .
201      P P 0198 1      KTP, LITB, NMA$C_ENT_KNO, CTR_TPT, ! Known tracepoints
202      P P 0199 1      CPS, LITB, 0, .
203      P P 0200 1      TST, LITB, 0, .
204      P 0201 1
205      0202 1      )
206      0203 1
207      P 0204 1      BUILD_SDB
208      P 0205 1
209      0206 1      (CTR, NMA$C_ENT_MOD, MTR_ENT, CTR)
```

```
.. 211      0207 1 %SBTTL 'Prompt strings'
.. 212      0208 1
.. 213      0209 1
.. 214      0210 1      Build prompt strings
.. 215      0211 1
.. 216      0212 1
.. 217      0213 1      BIND
.. 218      0214 1
.. 219      P 0215 1      PROMPT_STRINGS
.. 220      PP 0216 1      (MTR,
.. 221      PP 0217 1
.. 222      PP 0218 1      DAT, '      (TRACEPOINT name, or KNOWN): ',
.. 223      PP 0219 1      KWN, '      (TRACEPOINTS): ',
.. 224      PP 0220 1
.. 225      PP 0221 1      STA, 'State      (ON or OFF): ',
.. 226      PP 0222 1      BSZ, 'Buffer size      (1-4096 bytes): ',
.. 227      PP 0223 1      MBK, 'Maximum blocks      (1-65535): ',
.. 228      PP 0224 1      FNM, 'Filename      (1-253 characters): ',
.. 229      PP 0225 1      MBF, 'Maximum number of buffers      (1-255): ',
.. 230      PP 0226 1      CPL, 'Global data capture limit      (1-65535): ',
.. 231      P 0227 1      MVR, 'Maximum trace file version      (1-255): ',
.. 232      P 0228 1      TPT, 'Trace point name (tracepointname.channel): ',
.. 233      P 0229 1      CPS, 'Per-trace capture size (1-65535 bytes): ',
.. 234      P 0230 1      TST, 'Per-trace state      (ON or OFF): ',
.. 235      0231 1      ),
.. 236      0232 1
.. 237      P 0233 1      PROMPT_STRINGS
.. 238      PP 0234 1      (CTR,
.. 239      PP 0235 1
.. 240      PP 0236 1      DAT, '      (TRACEPOINT name, or KNOWN): ',
.. 241      PP 0237 1      KWN, '      (TRACEPOINTS): ',
.. 242      PP 0238 1      ALL, 'All X.25 Trace parameters      (Y, N): ',
.. 243      PP 0239 1      ALL2, 'All X.25 Trace tracepoint parameters      (Y, N): ',
.. 244      PP 0240 1
.. 245      PP 0241 1      STA, 'State      (Y, N): ',
.. 246      PP 0242 1      BSZ, 'Buffer size      (Y, N): ',
.. 247      PP 0243 1      MBK, 'Maximum blocks      (Y, N): ',
.. 248      PP 0244 1      FNM, 'Filename      (Y, N): ',
.. 249      PP 0245 1      MBF, 'Maximum number of buffers      (Y, N): ',
.. 250      PP 0246 1      CPL, 'Global data capture limit      (Y, N): ',
.. 251      PP 0247 1      MVR, 'Maximum trace file version      (Y, N): ',
.. 252      PP 0248 1      TPT, 'Trace point name      (Y, N): ',
.. 253      PP 0249 1      CPS, 'Per-trace capture size      (Y, N): ',
.. 254      P 0250 1      TST, 'Per-trace state      (Y, N): ',
.. 255      P 0251 1
.. 256      0252 1      );
```



```

: 258 0253 1 %SBTTL 'Declare entry points to TPARSE tables'
: 259 0254 1
: 260 0255 1
: 261 0256 1          Declare entry points to this TPARSE table
: 262 0257 1
: 263 0258 1
: 264 0259 1 $INIT_STATE (NCP$G_STTBL_MTR, NCP$G_KYTBL_MTR);
: 265 0260 1
: 266 0261 1 FORWARD
: 267 0262 1          ST_MTRTPT:      VECTOR [0],      ! Set X25-TRACE Tracepoint
: 268 0263 1          ST_CTRTPT:      VECTOR [0],      ! Clear X25-TRACE Tracepoint
: 269 0264 1          ST_CTR:         VECTOR [0],      ! Clear X25-TRACE
: 270 0265 1
: 271 0266 1 GLOBAL BIND
: 272 0267 1          NCP$G_STTBL_MTRTPT = ST_MTRTPT,
: 273 0268 1          NCP$G_KYTBL_MTRTPT = NCP$G_KYTBL_MTR,
: 274 0269 1          NCP$G_STTBL_CTR = ST_CTR,
: 275 0270 1          NCP$G_KYTBL_CTR = NCP$G_KYTBL_MTR,
: 276 0271 1          NCP$G_STTBL_CTRTPT = ST_CTRTPT,
: 277 0272 1          NCP$G_KYTBL_CTRTPT = NCP$G_KYTBL_MTR;

```

```

: 279      0273 1 XSBTTL 'SET X25-TRACE Module Parameters'
: 280      0274 1
: 281      0275 1
: 282      0276 1
: 283      0277 1
: 284      0278 1
: 285      P 0279 1 $STATE (ST_MTR,
: 286      P 0280 1 ((ST_MTR_INIT))
: 287      0281 1 );
: 288      P 0282 1 $STATE (
: 289      P 0283 1 (TPAS_EOS, ST_MTR_PMT_DAT),
: 290      P 0284 1 (TPAS_LAMBDA, ST_MTR_DAT)
: 291      0285 1 );
: 292      0286 1
: 293      0287 1
: 294      0288 1
: 295      0289 1
: 296      P 0290 1 $STATE (ST_MTR_TPT,
: 297      P 0291 1 ((ST_MTR_INIT), ST_MTR_DAT_TPT)
: 298      0292 1 );
: 299      P 0293 1
: 300      P 0294 1 $STATE (ST_MTR_INIT,
: 301      0295 1 (TPAS_LAMBDA, TPAS_EXIT, ACT$CLRLONG,,, TPT_PARAMS)); ! Assume trace-wide params
: 302      0296 1
: 303      0297 1
: 304      0298 1
: 305      0299 1
: 306      0300 1
: 307      0301 1
: 308      0302 1
: 309      0303 1
: 310      P 0304 1 $STATE (ST_MTR_PMT_DAT,
: 311      0305 1 (TPAS_LAMBDA,, ACT$PRMPT,,, PMT$G_MTR_DAT));
: 312      0306 1
: 313      P 0307 1 $STATE (ST_MTR_DAT, ! Determine whether TRACEPOINT parameters
: 314      P 0308 1 ('TRACEPOINT', ST_MTR_DAT_TPT),
: 315      P 0309 1 ('KNOWN', ST_MTR_DAT_KWN),
: 316      P 0310 1 (TPAS_LAMBDA) ! Else assume TRACE-wide parameters
: 317      0311 1 );
: 318      0312 1
: 319      0313 1
: 320      0314 1
: 321      0315 1
: 322      P 0316 1 $STATE (
: 323      P 0317 1 (TPAS_EOS), ! Start prompting if EOS
: 324      0318 1 (TPAS_LAMBDA, ST_MTR_PRC)); ! Else try parsing parameters
: 325      0319 1
: 326      P 0320 1 PROMPT_STATES
: 327      P 0321 1 (MTR,
: 328      P 0322 1
: 329      0323 1 STA, BSZ, MBK, FNM, MBF, CPL, MVR)
: 330      0324 1
: 331      P 0325 1 $STATE (
: 332      0326 1 (TPAS_LAMBDA, ST_MTR_DOIT));
: 333      0327 1
: 334      0328 1
: 335      0329 1

```

B
C
D
E
F
G
H
I
J
K
L
M
N
O
P
Q
R
S
T
U
V
W
X
Y
Z
[
\
]
^
_
`
a
b
c
d
e
f
g
h
i
j
k
l
m
n
o
p
q
r
s
t
u
v
w
x
y
z
{
|
}
~
_

```

336      0330 1 !
337      0331 1
338      P 0332 1 $STATE (ST_MTR_DAT_TPT,
339      0333 1 ((SE_TRCPNT_NAME),, ACT$SAVPRM,QUALPRESENT, NCP$GL_QUALPRS, PBK$G_MTR_TPT));
340      0334 1
341      P 0335 1 $STATE (ST_MTR_PMT_TPT,
342      P 0336 1 (TPAS_EOC), ! Start prompting if EOS
343      P 0337 1 (TPAS_LAMBDA, ST_MTR_PRC, ! Else try parsing parameters while
344      0338 1 TRUE, TPT_PARAMS)); ! remembering that we are parsing
345      0339 1 ! tracepoint-specific parameters
346      0340 1
347      P 0341 1 PROMPT_STATES
348      P 0342 1 (MTR,
349      0343 1
350      0344 1 (CPS, TST)
351      0345 1
352      P 0346 1 $STATE (ST_MTR_DOIT,
353      P 0347 1 (TPAS_EOS, TPAS_EXIT, ACT$VRB_UTILITY, , , SDB$G_MTR),
354      0348 1 );
355      0349 1
356      0350 1 !
357      0351 1 ! Dispatch on KNOWN keyword during prompting
358      0352 1 !
359      0353 1
360      P 0354 1 $STATE (ST_MTR_DAT_KWN,
361      0355 1 (TPAS_LAMBDA));
362      0356 1
363      P 0357 1 COMMAND PROMPT
364      P 0358 1 (MTR, KWN, NCP$_INVKEY,
365      P 0359 1
366      P 0360 1 ('TRACEPOINTS', ST_MTR_PMT_TPT, ACT$SAVPRM,,, PBK$G_MTR_KTP),
367      0361 1 )

```

```

369      0362 1
370      0363 1
371      0364 1
372      0365 1
373      0366 1
374      0367 1
375      P 0368 1 $STATE (ST_MTR_PRC,
376      P 0369 1 (TPAS_LAMBDA, ST_TPT_PRC, ACT$TESTLONG,,, TPT_PARAMS),
377      0370 1 (TPAS_LAMBDA));
378      0371 1
379      0372 1
380      0373 1
381      0374 1
382      0375 1
383      P 0376 1 $STATE (
384      P 0377 1 ((SE_ALL), ST_MTR_DOIT),
385      P 0378 1
386      P 0379 1 DISPATCH_STATES
387      P 0380 1 (MTR,
388      P 0381 1
389      P 0382 1 BSZ, 'BUFFER',
390      P 0383 1 CAP, 'CAPTURE',
391      P 0384 1 FNM, 'FILE',
392      P 0385 1 MAX, 'MAXIMUM',
393      P 0386 1 STA, 'STATE',
394      P 0387 1
395      P 0388 1 )
396      P 0389 1
397      P 0390 1 (TPAS_EOS, ST_MTR_DOIT)
398      0391 1 };
399      0392 1
400      0393 1
401      0394 1
402      0395 1
403      0396 1
404      P 0397 1 $STATE (ST_TPT_PRC,
405      P 0398 1
406      P 0399 1 ((SE_ALL), ST_MTR_DOIT),
407      P 0400 1
408      P 0401 1 DISPATCH_STATES
409      P 0402 1 (MTR,
410      P 0403 1
411      P 0404 1 CAP, 'CAPTURE',
412      P 0405 1 TST, 'STATE',
413      P 0406 1
414      P 0407 1 )
415      P 0408 1
416      P 0409 1 (TPAS_EOS, ST_MTR_DOIT)
417      0410 1 };

```

```

419      0411 1
420      0412 1
421      0413 1
422      0414 1
423      0415 1
424      P 0416 1 $STATE (ST_MTR_PRC_KWN,
425      0417 1 ((SE_MTR_KWN), ST_MTR_PRC));
426      0418 1
427      P 0419 1 $STATE (SE_MTR_KWN,
428      P 0420 1
429      P 0421 1 KEYWORD_STATE
430      P 0422 1 (MTR,
431      P 0423 1
432      P 0424 1 KTP, 'TRACEPOINTS',
433      P 0425 1
434      0426 1 ));
435      0427 1
436      0428 1
437      0429 1
438      0430 1
439      0431 1
440      P 0432 1 $STATE (ST_MTR_PRC_MAX,
441      P 0433 1
442      P 0434 1 DISPATCH_STATES
443      P 0435 1 (MTR,
444      P 0436 1
445      P 0437 1 MBK, 'BLOCKS',
446      P 0438 1 MBF, 'BUFFERS',
447      P 0439 1 MVR, 'VERSIONS',
448      P 0440 1
449      0441 1 ));
450      0442 1
451      0443 1
452      0444 1
453      0445 1
454      0446 1
455      P 0447 1 $STATE (ST_MTR_PRC_CAP,
456      P 0448 1
457      P 0449 1 DISPATCH_STATES
458      P 0450 1 (MTR,
459      P 0451 1
460      P 0452 1 CPL, 'LIMIT',
461      P 0453 1 CPS, 'SIZE',
462      P 0454 1
463      0455 1 ));
464      0456 1
465      0457 1
466      0458 1
467      0459 1
468      0460 1
469      P 0461 1 $STATE (ST_MTR_STA,
470      P 0462 1
471      P 0463 1 KEYWORD_STATE
472      P 0464 1 (MTR,
473      P 0465 1
474      P 0466 1 STAON, 'ON',
475      P 0467 1 STAOFF, 'OFF',

```

```

: 476 P 0468 .
: 477 0469 1 )));
: 478 0470 1
: 479 0471 1 |
: 480 0472 1 | Tracing state
: 481 0473 1 |
: 482 0474 1 |
: 483 P 0475 1 $STATE (ST_MTR_TST,
: 484 P 0476 1
: 485 P 0477 1 KEYWORD_STATE
: 486 P 0478 1 (MTR,
: 487 P 0479 1
: 488 P 0480 1 TSTON, 'ON',
: 489 P 0481 1 TSTOFF, 'OFF',
: 490 P 0482 1
: 491 0483 1 )));
```

```

: 493      0484 1
: 494      04 5 1
: 495      0486 1
: 496      0487 1
: 497      0488 1
: 498      P 0489 1
: 499      P 0490 1
: 500      P 0491 1
: 501      P 0492 1
: 502      P 0493 1
: 503      P 0494 1
: 504      P 0495 1
: 505      P 0496 1
: 506      P 0497 1
: 507      P 0498 1
: 508      P 0499 1
: 509      P 0500 1
: 510      P 0501 1
: 511      0502 1

```

Process states

```

PROCESS_STATES
(MTR,
BSZ, 'SIZE',
CPL, 'LIMIT',
CPS, 'SIZE',
FNM,
MBK, 'BLOCKS',
MBF, 'BUFFERS',
MVR, 'VERSIONS',
STA, ;
TST, ;
)

```

```
.. 513      0503  1
.. 514      0504  1
.. 515      0505  1
.. 516      0506  1
.. 517      0507  1
.. 518      P 0508  1
.. 519      P 0509  1
.. 520      P 0510  1
.. 521      P 0511  1
.. 522      P 0512  1
.. 523      P 0513  1
.. 524      P 0514  1
.. 525      P 0515  1
.. 526      P 0516  1
.. 527      P 0517  1
.. 528      P 0518  1
.. 529      0519  1
```

Subexpression states

SUB EXPRESSIONS
(MTR,

BSZ, TPAS_DECIMAL,
CPL, TPAS_DECIMAL,
CPS, TPAS_DECIMAL,
FNM, (SE_FILE_ID),
MBK, TPAS_DECIMAL,
MBF, TPAS_DECIMAL,
MVR, TPAS_DECIMAL,
)


```

531 0520 1 %SBTTL 'CLEAR X25-TRACE Module Parameters'
532 0521 1
533 0522 1
534 0523 1
535 0524 1
536 0525 1
537 P 0526 1 $STATE (ST_CTR,
538 P 0527 1 ((ST_CTR_INIT))
539 0528 1 );
540 P 0529 1 $STATE (
541 P 0530 1 (TPAS_EOS, ST_CTR_PMT_DAT),
542 P 0531 1 (TPAS_LAMBDA, ST_CTR_DAT)
543 0532 1 );
544 0533 1
545 0534 1
546 0535 1
547 0536 1
548 P 0537 1 $STATE (ST_CTRTPT,
549 P 0538 1 ((ST_CTR_INIT), ST_CTR_DAT_TPT)
550 0539 1 );
551 0540 1
552 0541 1
553 P 0542 1 $STATE (ST_CTR_INIT,
554 0543 1 (TPAS_LAMBDA, TPAS_EXIT, ACT$CLRLONG,,, TPT_PARAMS)); ! Assume trace-wide params
555 0544 1
556 0545 1
557 0546 1
558 0547 1
559 0548 1
560 0549 1
561 0550 1
562 0551 1
563 P 0552 1 $STATE (ST_CTR_PMT_DAT,
564 0553 1 (TPAS_LAMBDA,,, ACT$PRMPT,,, PMT$G_CTR_DAT));
565 0554 1
566 P 0555 1 $STATE (ST_CTR_DAT, ! Determine whether TRACEPOINT parameters
567 P 0556 1 ('TRACEPOINT', ST_CTR_DAT_TPT),
568 P 0557 1 ('KNOWN', ST_CTR_DAT_KWN),
569 0558 1 (TPAS_LAMBDA)); ! Else, assume TRACE-wide
570 0559 1
571 0560 1
572 0561 1
573 0562 1
574 0563 1
575 P 0564 1 $STATE (
576 P 0565 1 (TPAS_EOS), ! Start prompting if EOS
577 0566 1 (TPAS_LAMBDA, ST_CTR_PRC)); ! Else, try parsing parameters
578 0567 1
579 P 0568 1 QUERY_STATES
580 P 0569 1 (CTR,
581 P 0570 1
582 0571 1 ALL, STA, BSZ, MBK, FNM, MBF, CPL, MVR)
583 0572 1
584 P 0573 1 $STATE (
585 0574 1 (TPAS_LAMBDA, ST_CTR_DOIT));
586 0575 1
587 0576 1

```

```

: 588      0577 1 !      Prompt for tracepoint parameters
: 589      0578 1 !
: 590      0579 1 !
: 591      P 0580 1 $STATE (ST_CTR_DAT_TPT,
: 592      0581 1          (TPAS_LAMBDA,,, TRUE, TPT_PARAMS)); ! Remember that we are parsing
: 593      0582 1          ! tracepoint-specific parameters
: 594      P 0583 1 $STATE (
: 595      0584 1          ((SE_TRCPNT_NAME),, ACT$SAVPRM,,, PBK$G_CTR_TPT));
: 596      0585 1
: 597      P 0586 1 $STATE (ST_CTR_PMT_TPT,
: 598      P 0587 1          (TPAS_EOS), ! Start prompting if EOS
: 599      0588 1          (TPAS_LAMBDA,ST_CTR_PRC)); ! Else, try parsing parameters
: 600      0589 1
: 601      P 0590 1          QUERY_STATES_S
: 602      P 0591 1          (CTR,
: 603      P 0592 1
: 604      0593 1          ALL, ALL2, CPS, CPS, TST, TST)
: 605      0594 1
: 606      P 0595 1 $STATE (ST_CTR_DOIT,
: 607      P 0596 1          (TPAS_EOS, TPAS_EXIT, ACT$VRB_UTILITY, , , SDB$G_CTR),
: 608      0597 1          );
: 609      0598 1
: 610      0599 1 !
: 611      0600 1 !      Dispatch on KNOWN keyword during prompting
: 612      0601 1 !
: 613      0602 1 !
: 614      P 0603 1 $STATE (ST_CTR_DAT_KWN,
: 615      0604 1          (TPAS_LAMBDA));
: 616      0605 1
: 617      P 0606 1          COMMAND PROMPT
: 618      P 0607 1          (CTR, KWN, NCPS_INVKEY,
: 619      P 0608 1
: 620      P 0609 1          ('TRACEPOINTS', ST_CTR_PMT_TPT, ACT$SAVPRM,,, PBK$G_CTR_KTP),
: 621      0610 1          )

```

```
623 0611 1
624 0612 1
625 0613 1
626 0614 1
627 0615 1
628 0616 1
629 P 0617 1 $STATE (ST_CTR_PRC,
630 P 0618 1 (TPAS_LAMBDA, ST_CTP_PRC, ACT$TESTLONG,,, TPT_PARAMS),
631 0619 1 (TPAS_LAMBDA));
632 0620 1
633 0621 1
634 0622 1
635 0623 1
636 0624 1
637 0625 1
638 P 0626 1 $STATE (,
639 P 0627 1
640 P 0628 1 DISPATCH_STATES
641 P 0629 1 (CTR,
642 P 0630 1
643 P 0631 1 ALL, 'ALL',
644 P 0632 1 BSZ, 'BUFFER',
645 P 0633 1 CAP, 'CAPTURE',
646 P 0634 1 FNM, 'FILE',
647 P 0635 1 MAX, 'MAXIMUM',
648 P 0636 1 STA, 'STATE',
649 P 0637 1 )
650 P 0638 1
651 P 0639 1 ,(TPAS_EOS, ST_CTR_DOIT)
652 0640 1 );
653 0641 1
654 0642 1
655 0643 1
656 0644 1
657 0645 1
658 P 0646 1 $STATE (ST_CTP_PRC,
659 P 0647 1
660 P 0648 1 DISPATCH_STATES
661 P 0649 1 (CTR,
662 P 0650 1
663 P 0651 1 ALL, 'ALL',
664 P 0652 1 CAP, 'CAPTURE',
665 P 0653 1 TST, 'STATE',
666 P 0654 1 )
667 P 0655 1
668 P 0656 1 ,(TPAS_EOS, ST_CTR_DOIT)
669 0657 1 );
```

```

: 671      0658 1
: 672      0659 1
: 673      0660 1      :      Dispatch on KNOWN keyword
: 674      0661 1
: 675      0662 1
: 676      P 0663 1  $STATE (ST_CTR_PRC_KWN,
: 677      0664 1      ((SE_CTR_KWN), ST_CTR_PRC));
: 678      0665 1
: 679      P 0666 1  $STATE (SE_CTR_KWN,
: 680      P 0667 1
: 681      P 0668 1      KEYWORD_STATE
: 682      P 0669 1      (CTR,
: 683      P 0670 1
: 684      P 0671 1      KTP, 'TRACEPOINTS',
: 685      P 0672 1
: 686      0673 1      ));
: 687      0674 1
: 688      0675 1      :
: 689      0676 1      :      Dispatch on MAXIMUM keyword
: 690      0677 1
: 691      0678 1
: 692      P 0679 1  $STATE (ST_CTR_PRC_MAX,
: 693      P 0680 1
: 694      P 0681 1      DISPATCH_STATES
: 695      P 0682 1      (CTR,
: 696      P 0683 1
: 697      P 0684 1      MBK, 'BLOCKS',
: 698      P 0685 1      MBF, 'BUFFERS',
: 699      P 0686 1      MVR, 'VERSIONS',
: 700      P 0687 1
: 701      0688 1      ));
: 702      0689 1
: 703      0690 1      :
: 704      0691 1      :      Dispatch on CAPTURE keyword
: 705      0692 1
: 706      0693 1
: 707      P 0694 1  $STATE (ST_CTR_PRC_CAP,
: 708      P 0695 1
: 709      P 0696 1      DISPATCH_STATES
: 710      P 0697 1      (CTR,
: 711      P 0698 1
: 712      P 0699 1      CPL, 'LIMIT',
: 713      P 0700 1      CPS, 'SIZE',
: 714      P 0701 1
: 715      0702 1      ));

```

```
.. 717      0703  1
.. 718      0704  1
.. 719      0705  1
.. 720      0706  1
.. 721      0707  1
.. 722      P 0708  1
.. 723      P 0709  1
.. 724      P 0710  1
.. 725      P 0711  1
.. 726      P 0712  1
.. 727      P 0713  1
.. 728      P 0714  1
.. 729      P 0715  1
.. 730      P 0716  1
.. 731      P 0717  1
.. 732      P 0718  1
.. 733      P 0719  1
.. 734      P 0720  1
.. 735      P 0721  1
.. 736      P 0722  1
.. 737      0723  1

          )

          Process states

          PROCESS_STATES
          (CTR,
          ALL, .
          BSZ, 'SIZE',
          CPL, 'LIMIT',
          CPS, 'SIZE',
          FNM,
          MBK, 'BLOCKS',
          MBF, 'BUFFERS',
          MVR, 'VERSIONS',
          STA, .
          TST, .
```

```
: 739      0724  1
: 740      0725  1
: 741      0726  1
: 742      0727  1
: 743      0728  1
: 744      P 0729  1
: 745      P 0730  1
: 746      P 0731  1
: 747      P 0732  1
: 748      P 0733  1
: 749      P 0734  1
: 750      P 0735  1
: 751      P 0736  1
: 752      P 0737  1
: 753      P 0738  1
: 754      P 0739  1
: 755      P 0740  1
: 756      P 0741  1
: 757      P 0742  1
: 758      P 0743  1
: 759      0744  1
```

Subexpression states

SUB EXPRESSIONS
(CTR,

ALL, TPAS_EOS,

BSZ, TPAS_LAMBDA,
CPL, TPAS_LAMBDA,
CPS, TPAS_LAMBDA,
FNM, TPAS_LAMBDA,
MBK, TPAS_LAMBDA,
MBF, TPAS_LAMBDA,
MVR, TPAS_LAMBDA,
STA, TPAS_LAMBDA,
TST, TPAS_LAMBDA,

)

```
: 761      0745 1 %SBTTL 'Define Subexpressions'  
: 762      0746 1  
: 763      0747 1 |  
: 764      0748 1 | Define Subexpressions from Library  
: 765      0749 1 |  
: 766      0750 1  
: 767      0751 1 SEM_ALL           | All parameter  
: 768      0752 1 SEM_FILE_ID        | File name  
: 769      0753 1 SEM_LINE_ID        | For tracepoint name  
: 770      0754 1 SEM_QUERY           | Query state subexpressions  
: 771      0755 1 SEM_TRCPNT_NAME     | Tracepoint name
```

NCPSTAMTR
V04-000

X.25 Trace Module Parsing
Define Subexpressions

C 1
16-Sep-1984 01:11:16
14-Sep-1984 12:48:31

VAX-11 Bliss-32 V4.0-742
[NCP.SRC]NCPSTAMTR.B32;1

Page 22
(18)

NC
VO

: 773
: 774

0756 1 END
0757 0 ELUDOM

:
:
:
:
:
:
:

0271

AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

NCPSTANOD
LIS

NCPSTASTA
LIS

NCPSTASHL
LIS

NCPSTAOBJ
LIS

NCPSTATRI
LIS

NCPSTAZER
LIS

NCPTERMIO
LIS

NCPTABLES
LIS

NCPURBACT
LIS