





```

1 0001 0 %TITLE 'X.25 Protocol Module Parsing'
2 0002 0 MODULE NCPSTAMPR (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 X.25 protocol
36 0036 1     module parameters.
37 0037 1
38 0038 1 ENVIRONMENT:   VAX/VMS Operating System
39 0039 1
40 0040 1 AUTHOR:       Tim Halvorsen, July 1981
41 0041 1
42 0042 1 MODIFIED BY:
43 0043 1
44 0044 1     V03-006 RPG0006      Bob Grosso      04-Nov-1982
45 0045 1     Change 'MAXIMUM CIRCUIT seconds' prompt to
46 0046 1     'MAXIMUM CIRUIT count'.
47 0047 1
48 0048 1     V03-004 RPG0004      Bob Grosso      15-Sep-1982
49 0049 1     Flag when qualifier has been parsed so that ALL check
50 0050 1     in NCPVRBACT will work properly.
51 0051 1
52 0052 1     V03-004 TMH0004      Tim Halvorsen   19-Aug-1982
53 0053 1     Make CLEAR X-P GROUP FRED DTE 123 work.
54 0054 1
55 0055 1     V03-003 RPG0003      Bob Grosso      02-Aug-1982
56 0056 1     Add X-P MAX CIRCUITS.
57 0057 1     Add X-P MULTI-NETWORK SUPPORT.

```

```

: 58      0058 1  |  Add new ALL prompts for CLEAR DTE and CLEAR GROUP.
: 59      0059 1  |
: 60      0060 1  |  V03-002 RPG0002      Bob Grosso      28-Jun-1982
: 61      0061 1  |  Change prompting of CLEAR X-P GROUP ABC to only
: 62      0062 1  |  prompt for ALL, and CLEAR X-P DTE 1234 to prompt
: 63      0063 1  |  for ALL instead of LINE.
: 64      0064 1  |  Change parameter type of CHN from SAD to RNGL.
: 65      0065 1  |
: 66      0066 1  |  V001   TMH0001      Tim Halvorsen      05-Apr-1982
: 67      0067 1  |  Fix prompting.
: 68      0068 1  |  --

```

```
70 0069 1 %SBTTL 'Definitions'
71 0070 1
72 0071 1
73 0072 1 !! INCLUDE FILES:
74 0073 1 !!
75 0074 1
76 0075 1     LIBRARY 'LIBS:NMALIBRY';
77 0076 1     LIBRARY 'LIBS:NCPLIBRY';
78 0077 1     LIBRARY 'SYSSLIBRARY:TPAMAC';
79 0078 1
80 0079 1 !!
81 0080 1 !! EXTERNAL REFERENCES:
82 0081 1 !!
83 0082 1     ACT_DFN           ! Action routine externals
84 0083 1 EXTERNAL
85 0084 1     NCP$GL_QUALPRS;
86 0085 1
87 0086 1 !!
88 0087 1 !! OWN STORAGE:
89 0088 1 !!
90 0089 1
91 0090 1 OWN
92 0091 1     GROUP_SPECIFIED;   ! Flag to indicate if GROUP specified or not
93 0092 1
94 0093 1 !!
95 0094 1 !! LITERALS
96 0095 1 !!
97 0096 1
98 0097 1 LITERAL
99 0098 1     QUALPRESENT = 1;    ! Qualifier was present on command line
```

```
101 0099 1 %SBTTL 'Set Parameter blocks'
102 0100 1
103 0101 1
104 0102 1 SET X25-PROTOCOL Parameter Blocks
105 0103 1
106 0104 1
107 P 0105 1 BUILD_PCL
108 P P 0106 1
109 P P 0107 1 (MPR, ! Module X25-PROTOCOL
110 P P 0108 1
111 P P 0109 1 DTE, TKN, PCXP_DTE, ;
112 P P 0110 1 GRP, TKN, PCXP_GRP, ;
113 P P 0111 1
114 P P 0112 1 STA, NUMB, PCXP_STA, ;
115 P P 0113 1 CTM, NUMW, PCXP_CTM, ;
116 P P 0114 1 NET, TKN, PCXP_NET, ;
117 P P 0115 1 LIN, TKN, PCXP_LIN, ;
118 P P 0116 1 CHN, RNGL, PCXP_CHN, ;
119 P P 0117 1 MCH, NUMW, PCXP_MCH, ;
120 P P 0118 1 DBL, NUMW, PCXP_DBL, ;
121 P P 0119 1 DWI, NUMB, PCXP_DWI, ;
122 P P 0120 1 MBL, NUMW, PCXP_MBL, ;
123 P P 0121 1 MWI, NUMB, PCXP_MWI, ;
124 P P 0122 1 MCL, NUMB, PCXP_MCL, ;
125 P P 0123 1 MRS, NUMB, PCXP_MRS, ;
126 P P 0124 1 MST, NUMB, PCXP_MST, ;
127 P P 0125 1 CAT, NUMB, PCXP_CAT, ;
128 P P 0126 1 CLT, NUMB, PCXP_CLT, ;
129 P P 0127 1 RST, NUMB, PCXP_RST, ;
130 P P 0128 1 STT, NUMB, PCXP_STT, ;
131 P P 0129 1 GDT, TKN, PCXP_GDT, ;
132 P P 0130 1 GNM, NUMW, PCXP_GNM, ;
133 P P 0131 1 GTY, NUMB, PCXP_GTY, ;
134 P P 0132 1
135 P P 0133 1 MNS, NUMB, PCXP_MNS, ;
136 P P 0134 1 MCI, NUMW, PCXP_MCI, ;
137 P P 0135 1
138 P P 0136 1 . END, . .
139 P 0137 1 )
140 0138 1
141 0139 1
142 P 0140 1 BUILD_PBK
143 P P 0141 1
144 P P 0142 1 (MPR, ! Module X25-PROTOCOL
145 P P 0143 1
146 P 0144 1 STAON, LITB, NMASC_XPRST_ON, MPR_STA,
147 P 0145 1 STAOFF, LITB, NMASC_XPRST_OFF, MPR_STA,
148 P 0146 1 STASHU, LITB, NMASC_XPRST_SHU, MPR_STA,
149 P P 0147 1 CTM, NUMW, . .
150 P P 0148 1 DTE, TKN, . .
151 P P 0149 1 KDT, LITB, NMASC_ENT_KNO, MPR_DTE, ! KNOWN DTES
152 P 0150 1 GRP, TKN, . .
153 P 0151 1 KGR, LITB, NMASC_ENT_KNO, MPR_GRP, ! KNOWN GROUPS
154 P 0152 1 NET, TKN, . .
155 P 0153 1 LIN, TKN, . .
156 P 0154 1 CHN, RNGL, . .
157 P 0155 1 MCH, NUMW, . .
```

```
158 P 0156 1 DBL, NUMW, . . .
159 P 0157 1 DWI, NUMB, . . .
160 P 0158 1 MBL, NUMW, . . .
161 P 0159 1 MWI, NUMB, . . .
162 P 0160 1 MCL, NUMB, . . .
163 P 0161 1 MRS, NUMB, . . .
164 P 0162 1 MST, NUMB, . . .
165 P 0163 1 CAT, NUMB, . . .
166 P 0164 1 CLT, NUMB, . . .
167 P 0165 1 RST, NUMB, . . .
168 P 0166 1 STT, NUMB, . . .
169 P 0167 1 GDT, TKN, . . .
170 P 0168 1 GNM, NUMW, . . .
171 P 0169 1 GTYBIL, LITB, NMASC_XPRTY_BIL, MPR_GTY,
172 P 0170 1
173 P 0171 1 MNSENA, LITB, NMASC_XPRMN_ENA, MPR_MNS,
174 P 0172 1 MNSDIS, LITB, NMASC_XPRMN_DIS, MPR_MNS,
175 P 0173 1 MCI, NUMW, . . .
176 P 0174 1
177 0175 1 )
178 0176 1
179 0177 1 BIND PDBSG_MPR_ENT = ! Protocol entity name
180 0178 1 UPLIT_BYTE(0, %ASCIC 'X25-PROTOCOL');
181 0179 1
182 P 0180 1 BUILD_SDB
183 P 0181 1
184 0182 1 (MPR, NMASC_ENT_MOD, MPR_ENT, MPR)
```

```

186 0183 1 XSBTTL 'Clear Parameter blocks'
187 0184 1
188 0185 1
189 0186 1
190 0187 1
191 0188 1
192 P 0189 1 BUILD_PCL
193 P 0190 1
194 P 0191 1 (CPR, ! Module X25-PROTOCOL
195 P 0192 1
196 P 0193 1 DTE, TKN, PCXP_DTE,
197 P 0194 1 GRP, TKN, PCXP_GRP,
198 P 0195 1
199 P 0196 1 STA, LITB, PCXP_STA,
200 P 0197 1 CTM, LITB, PCXP_CTM,
201 P 0198 1 NET, LITB, PCXP_NET,
202 P 0199 1 LIN, LITB, PCXP_LIN,
203 P 0200 1 CHN, LITB, PCXP_CHN,
204 P 0201 1 MCH, LITB, PCXP_MCH,
205 P 0202 1 DBL, LITB, PCXP_DBL,
206 P 0203 1 DWI, LITB, PCXP_DWI,
207 P 0204 1 MBL, LITB, PCXP_MBL,
208 P 0205 1 MWI, LITB, PCXP_MWI,
209 P 0206 1 MCL, LITB, PCXP_MCL,
210 P 0207 1 MRS, LITB, PCXP_MRS,
211 P 0208 1 MST, LITB, PCXP_MST,
212 P 0209 1 CAT, LITB, PCXP_CAT,
213 P 0210 1 CLT, LITB, PCXP_CLT,
214 P 0211 1 RST, LITB, PCXP_RST,
215 P 0212 1 STT, LITB, PCXP_STT,
216 P 0213 1 GDT, TKN, PCXP_GDT,
217 P 0214 1 GNM, LITB, PCXP_GNM,
218 P 0215 1 GTY, LITB, PCXP_GTY,
219 P 0216 1
220 P 0217 1 MNS, LITB, PCXP_MNS,
221 P 0218 1 MCI, LITB, PCXP_MCI,
222 P 0219 1
223 P 0220 1 . END, . .
224 P 0221 1 )
225 0222 1
226 P 0223 1 BUILD_PBK
227 P 0224 1
228 P 0225 1
229 P 0226 1 (CPR, ! Module X25-PROTOCOL
230 P 0227 1
231 P 0228 1 ALL, LITB, 0, VRB_ALL,
232 P 0229 1
233 P 0230 1 STA, LITB, 0,
234 P 0231 1 CTM, LITB, 0,
235 P 0232 1 DTE, TKN,
236 P 0233 1 KDT, LITB, NMASC_ENT_KNO, CPR_DTE, ! DTE is a qualifier
237 P 0234 1 GRP, TKN, ! KNOWN DTES
238 P 0235 1 KGR, LITB, NMASC_ENT_KNO, CPR_GRP, ! Group is a qualifier
239 P 0236 1 NET, LITB, 0, ! KNOWN GROUPS
240 P 0237 1 LIN, LITB, 0,
241 P 0238 1 CHN, LITB, 0,
242 P 0239 1 MCH, LITB, 0,

```



```
.....: 243 P 0240 1 DBL, LITB, 0. .  
.....: 244 P P 0241 1 DWI, LITB, 0. .  
.....: 245 P P P 0242 1 MBL, LITB, 0. .  
.....: 246 P P P P 0243 1 MWI, LITB, 0. .  
.....: 247 P P P P P 0244 1 MCL, LITB, 0. .  
.....: 248 P P P P P P 0245 1 MRS, LITB, 0. .  
.....: 249 P P P P P P P 0246 1 MST, LITB, 0. .  
.....: 250 P P P P P P P P 0247 1 CAT, LITB, 0. .  
.....: 251 P P P P P P P P P 0248 1 CLT, LITB, 0. .  
.....: 252 P P P P P P P P P P 0249 1 RST, LITB, 0. .  
.....: 253 P P P P P P P P P P P 0250 1 STT, LITB, 0. .  
.....: 254 P P P P P P P P P P P P 0251 1 GDT, TKN, . .  
.....: 255 P P P P P P P P P P P P P 0252 1 GNM, LITB, 0. .  
.....: 256 P P P P P P P P P P P P P P 0253 1 GTY, LITB, 0. .  
.....: 257 P P P P P P P P P P P P P P P 0254 1  
.....: 258 P P P P P P P P P P P P P P P P 0255 1 MNS, LITB, 0. .  
.....: 259 P P P P P P P P P P P P P P P P P 0256 1 MCI, LITB, 0. .  
.....: 260 P 0257 1 )  
.....: 261 0258 1 )  
.....: 262 0259 1 )  
.....: 263 P 0260 1 BUILD_SDB  
.....: 264 P 0261 1  
.....: 265 0262 1 (CPR, NMASC_ENT_MOD, MPR_ENT, CPR)
```

! GROUP DTE is a qualifier

```

267 0263 1 %SBTTL 'Prompt strings'
268 0264 1
269 0265 1
270 0266 1 Build prompt strings
271 0267 1
272 0268 1
273 0269 1 BIND
274 0270 1
275 P 0271 1 PROMPT_STRINGS
276 P P 0272 1 (MPR,
277 P P 0273 1
278 P P 0274 1 DAT, ' (DTE number, GROUP name, or KNOWN): ',
279 P P 0275 1 KWN, ' (DTES, GROUPS): ',
280 P P 0276 1
281 P 0277 1 CAT, 'Call timer (1-255 seconds): ',
282 P 0278 1 CLT, 'Clear timer (1-255 seconds): ',
283 P 0279 1 DBL, 'Default data (1-65535 bytes): ',
284 P P 0280 1 DWI, 'Default window (1-255 blocks): ',
285 P P 0281 1 MBL, 'Maximum data (1-65535 bytes): ',
286 P 0282 1 MCL, 'Maximum clears (1-255): ',
287 P 0283 1 MRS, 'Maximum resets (1-255): ',
288 P 0284 1 MST, 'Maximum restarts (1-255): ',
289 P 0285 1 MWI, 'Maximum window (1-255 blocks): ',
290 P 0286 1 NET, 'Network name (1-16 characters): ',
291 P 0287 1 RST, 'Reset timer (1-255 seconds): ',
292 P P 0288 1 STT, 'Restart timer (1-255 seconds): ',
293 P 0289 1
294 P 0290 1 DTE, 'DTE number (1-16 digits): ',
295 P 0291 1 CHN, 'Channel list (delimited by commas): ',
296 P 0292 1 CTM, 'Counter timer (1-65535 seconds): ',
297 P 0293 1 LIN, 'Line name (dev-c-u.t): ',
298 P 0294 1 STA, 'State (ON, OFF, SHUT): ',
299 P 0295 1
300 P 0296 1 GRP, 'Group name (1-16 characters): ',
301 P 0297 1 GDT, 'Group DTE number (1-16 digits): ',
302 P 0298 1 GNM, 'Group number (0-9999): ',
303 P 0299 1 GTY, 'Group type (BILATERAL): ',
304 P 0300 1
305 P 0301 1 MNS, 'Multi-network support (ENABLE, DISABLE): ',
306 P 0302 1 MCI, 'Maximum circuits (1-65535 count): ',
307 P 0303 1 ),
308 0304 1
309 P 0305 1 PROMPT_STRINGS
310 P 0306 1 (CPR,
311 P 0307 1
312 P 0308 1 DAT, ' (DTE number, GROUP name, or KNOWN): ',
313 P 0309 1 KWN, ' (DTES, GROUPS): ',
314 P P 0310 1
315 P 0311 1 ALL, 'All X.25 protocol parameters (Y, N): ',
316 P 0312 1 ALL2, 'All X.25 protocol group parameters (Y, N): ',
317 P 0313 1 ALL3, 'All X.25 protocol DTE parameters (Y, N): ',
318 P 0314 1 CAT, 'Call timer (Y, N): ',
319 P 0315 1 CLT, 'Clear timer (Y, N): ',
320 P 0316 1 MCL, 'Maximum clears (Y, N): ',
321 P 0317 1 MRS, 'Maximum resets (Y, N): ',
322 P 0318 1 MST, 'Maximum restarts (Y, N): ',
323 P 0319 1 MWI, 'Maximum window (Y, N): ',

```

```

: 324 P 0320 1 RST, 'Reset timer (Y, N): '
: 325 P 0321 1 STT, 'Restart timer (Y, N): '
: 326 P 0322 1
: 327 P 0323 1 DTE, 'DTE number (Y, N): '
: 328 P 0324 1 CTM, 'Counter timer (Y, N): '
: 329 P 0325 1 LIN, 'Line name (Y, N): '
: 330 P 0326 1
: 331 P 0327 1 GRP, 'Group name (Y, N): '
: 332 P 0328 1 GDT, 'Group DTE number (Y, N): '
: 333 P 0329 1 GTY, 'Group type (Y, N): '
: 334 P 0330 1
: 335 P 0331 1 MNS, 'Multi-network support (Y, N): '
: 336 P 0332 1 MCI, 'Maximum circuits (Y, N): '
: 337 P 0333 1
: 338 0334 1 );
```

```
340 0335 1 %SBTTL 'Declare entry points to TPARSE tables'  
341 0336 1  
342 0337 1  
343 0338 1  
344 0339 1  
345 0340 1  
346 0341 1 $ VIT_STATE (NCP$G_STTBL_MPR, NCP$G_KYTBL_MPR);  
347 0342 1  
348 0343 1 FORWARD  
349 0344 1 ST_MPRDTE: VECTOR [0], ; Set X25-Protocol DTE  
350 0345 1 ST_MPRGRP: VECTOR [0], ; Set X25-Protocol GROUP  
351 0346 1 ST_CPRDTE: VECTOR [0], ; Clear X25-Protocol DTE  
352 0347 1 ST_CPRGRP: VECTOR [0], ; Clear X25-Protocol GROUP  
353 0348 1 ST_CPR: VECTOR [0], ; Clear X25-Protocol  
354 0349 1  
355 0350 1 GLOBAL BIND  
356 0351 1 NCP$G_STTBL_MPRDTE = ST_MPRDTE,  
357 0352 1 NCP$G_KYTBL_MPRDTE = NCP$G_KYTBL_MPR,  
358 0353 1 NCP$G_STTBL_MPRGRP = ST_MPRGRP,  
359 0354 1 NCP$G_KYTBL_MPRGRP = NCP$G_KYTBL_MPR,  
360 0355 1 NCP$G_STTBL_CPR = ST_CPR,  
361 0356 1 NCP$G_KYTBL_CPR = NCP$G_KYTBL_MPR,  
362 0357 1 NCP$G_STTBL_CPRDTE = ST_CPRDTE,  
363 0358 1 NCP$G_KYTBL_CPRDTE = NCP$G_KYTBL_MPR,  
364 0359 1 NCP$G_STTBL_CPRGRP = ST_CPRGRP,  
365 0360 1 NCP$G_KYTBL_CPRGRP = NCP$G_KYTBL_MPR;
```

```
367 0361 1 %SBTTL 'SET X25-PROTOCOL Module Parameters'
368 0362 1
369 0363 1
370 0364 1 SET/DEFINE MODULE X25-PROTOCOL parameter states
371 0365 1
372 0366 1
373 P 0367 1 $STATE (ST_MPR,
374 P 0368 1 ((ST_MPR_INIT))
375 0369 1 );
376 0370 1
377 P 0371 1 $STATE (
378 P 0372 1 (TPAS_EOS, ST_MPR_PMT_DAT),
379 P 0373 1 (TPAS_LAMBDA, ST_MPR_DAT)
380 0374 1 );
381 0375 1
382 0376 1
383 0377 1 Dispatch to here from NCPSTAVRB if SET DTE parsed
384 0378 1
385 P 0379 1 $STATE (ST_MPRDTE,
386 P 0380 1 ((ST_MPR_INIT), ST_MPR_DAT_DTE)
387 0381 1 );
388 0382 1
389 0383 1
390 0384 1 Dispatch to here from NCPSTAVRB if SET GROUP parsed
391 0385 1
392 P 0386 1 $STATE (ST_MPRGRP,
393 P 0387 1 ((ST_MPR_INIT), ST_MPR_DAT_GRP)
394 0388 1 );
395 0389 1
396 0390 1
397 P 0391 1 $STATE (ST_MPR_INIT,
398 P 0392 1 (TPAS_LAMBDA, TPAS_EXIT, ACT$CLRLONG, . . ! Mark GROUP not specified yet
399 0393 1 GROUP_SPECIFIED));
400 0394 1
401 0395 1
402 0396 1 Determine which protocol sub-database we are talking about
403 0397 1 (due to grouping restrictions, the command must not mix
404 0398 1 sub-database parameters). The sub-database may be either
405 0399 1 protocol, closed user group, or DTE parameters.
406 0400 1
407 0401 1
408 P 0402 1 $STATE (ST_MPR_PMT_DAT,
409 0403 1 (TPAS_LAMBDA, ACT$PRMPT, . . . PMT$G_MPR_DAT));
410 0404 1
411 P 0405 1 $STATE (ST_MPR_DAT, ! Determine whether GROUP or DTE parameters
412 P 0406 1 ('GROUP', ST_MPR_DAT_GRP),
413 P 0407 1 ('DTE', ST_MPR_DAT_DTE),
414 P 0408 1 ('KNOWN', ST_MPR_DAT_KWN),
415 0409 1 (TPAS_LAMBDA));
416 0410 1
417 0411 1
418 0412 1 Prompt for protocol parameters
419 0413 1
420 0414 1
421 P 0415 1 $STATE (
422 P 0416 1 (TPAS_EOS),
423 0417 1 (TPAS_LAMBDA, ST_MPR_PRC));
```

```

: 424      0418 1
: 425      P 0419 1      PROMPT_STATES
: 426      P 0420 1      (MPR,
: 427      P 0421 1      CAT, CLT, DBL, DWI, MBL, MCL, MRS, MST, MWI, NET, RST, MNS,
: 428      0422 1      STT)
: 429      0423 1
: 430      P 0424 1 $STATE (
: 431      0425 1      (TPAS_LAMBDA, ST_MPR_DOIT));
: 432      0426 1
: 433      0427 1      |
: 434      0428 1      |      Prompt for closed user group parameters
: 435      0429 1      |
: 436      0430 1
: 437      P 0431 1 $STATE (ST_MPR_DAT_GRP,
: 438      0432 1      ((SE_GRP_NAME),, ACT$SAVPRM, QUALPRESENT, NCP$GL_QUALPRS, PBK$G_MPR_GRP));
: 439      0433 1
: 440      P 0434 1 $STATE (ST_MPR_PMT_GRP,
: 441      0435 1      (TPAS_LAMBDA,,, TRUE, GROUP_SPECIFIED));      ! Mark GROUP specified
: 442      0436 1
: 443      P 0437 1 $STATE (
: 444      P 0438 1      (TPAS_EOS),
: 445      0439 1      (TPAS_LAMBDA,ST_MPR_PRC));
: 446      0440 1
: 447      P 0441 1      PROMPT_STATES
: 448      P 0442 1      (MPR,
: 449      0443 1      GDT, GNM, GTY)
: 450      0444 1
: 451      P 0445 1 $STATE (
: 452      0446 1      (TPAS_LAMBDA, ST_MPR_DOIT));
: 453      0447 1
: 454      0448 1      |
: 455      0449 1      |      Prompt for DTE parameters
: 456      0450 1      |
: 457      0451 1
: 458      P 0452 1 $STATE (ST_MPR_DAT_DTE,
: 459      0453 1      ((SE_DTE_NUMBER),, ACT$SAVPRM, QUALPRESENT, NCP$GL_QUALPRS, PBK$G_MPR_DTE));
: 460      0454 1
: 461      P 0455 1 $STATE (ST_MPR_PMT_DTE,
: 462      P 0456 1      (TPAS_EOS),
: 463      0457 1      (TPAS_LAMBDA,ST_MPR_PRC));
: 464      0458 1
: 465      P 0459 1      PROMPT_STATES
: 466      P 0460 1      (MPR,
: 467      0461 1      CHN, CTM, LIN, STA, MCI)
: 468      0462 1
: 469      P 0463 1 $STATE (ST_MPR_DOIT,
: 470      P 0464 1      (TPAS_EOS, TPAS_EXIT, ACT$VRB_UTILITY, , , SDB$G_MPR),
: 471      0465 1      );
: 472      0466 1
: 473      0467 1      |
: 474      0468 1      |      Dispatch on KNOWN keyword during prompting
: 475      0469 1      |
: 476      0470 1
: 477      P 0471 1 $STATE (ST_MPR_DAT_KWN,
: 478      0472 1      (TPAS_LAMBDA));
: 479      0473 1
: 480      P 0474 1      COMMAND_PROMPT

```

NCPSTAMPR  
V04-000

X.25 Protocol Module Parsing  
SET X25-PROTOCOL Module Parameters

L 11  
16-Sep-1984 00:54:06  
14-Sep-1984 12:48:29

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

Page 13  
(7)

```
: 481      P 0475 1      (MPR, KWN, NCP$ INVKEY,  
: 482      P 0476 1  
: 483      P 0477 1      ('DTES', ST_MPR_PMT_DTE, ACT$SAVPRM, PBK$G_MPR_KDT)  
: 484      P 0478 1      ('GROUP$', ST_MPR_PMT_GRP, ACT$SAVPRM, PBK$G_MPR_KGR)  
: 485      0479 1      )
```

N  
V

```

: 487      0480 1
: 488      0481 1
: 489      0482 1
: 490      0483 1
: 491      0484 1
: 492      P 0485 1 $STATE (ST_MPR_PRC,
: 493      P 0486 1
: 494      P 0487 1 ((SE_ALL), ST_MPR_DOIT),
: 495      P 0488 1
: 496      P 0489 1 DISPATCH_STATES
: 497      P 0490 1 (MPR,
: 498      P 0491 1
: 499      P 0492 1 CAT, 'CALL',
: 500      P 0493 1 CLT, 'CLEAR',
: 501      P 0494 1 CHN, 'CHANNELS',
: 502      P 0495 1 CTM, 'COUNTER',
: 503      P 0496 1 DEF, 'DEFAULT',
: 504      P 0497 1 DTEX, 'DTE',
: 505      P 0498 1 GRP, 'GROUP',
: 506      P 0499 1 KWN, 'KNOWN',
: 507      P 0500 1 LIN, 'LINE',
: 508      P 0501 1 MAX, 'MAXIMUM',
: 509      P 0502 1 MUL, 'MULTI',
: 510      P 0503 1 NET, 'NETWORK',
: 511      P 0504 1 GNM, 'NUMBER',
: 512      P 0505 1 RST, 'RESET',
: 513      P 0506 1 STT, 'RESTART',
: 514      P 0507 1 STA, 'STATE',
: 515      P 0508 1 GTY, 'TYPE',
: 516      P 0509 1
: 517      P 0510 1 )
: 518      P 0511 1 ;(TPAS_EOS, SI_MPR_DOIT)
: 519      0512 1 );

```



```
521 0513 1  
522 0514 1  
523 0515 1  
524 0516 1  
525 0517 1  
526 P 0518 1 $STATE (ST_MPR_PRC_KWN,  
527 0519 1 ((SE_MPR_KWN), ST_MPR_PRC));  
528 0520 1  
529 P 0521 1 $STATE (SE_MPR_KWN,  
530 P 0522 1  
531 P 0523 1 KEYWORD_STATE  
532 P 0524 1 (MPR,  
533 P 0525 1  
534 P 0526 1 KDT, 'DTES',  
535 P 0527 1 KGR, 'GROUPS',  
536 P 0528 1  
537 0529 1 ));  
538 0530 1  
539 0531 1  
540 0532 1  
541 0533 1  
542 0534 1  
543 P 0535 1 $STATE (ST_MPR_PRC_MAX,  
544 P 0536 1  
545 P 0537 1 DISPATCH_STATES  
546 P 0538 1 (MPR,  
547 P 0539 1  
548 P 0540 1 MBL, 'DATA',  
549 P 0541 1 MCI, 'CIRCUITS',  
550 P 0542 1 MCL, 'CLEARS',  
551 P 0543 1 MRS, 'RESETS',  
552 P 0544 1 MST, 'RESTARTS',  
553 P 0545 1 MWI, 'WINDOW',  
554 P 0546 1 )  
555 P 0547 1 );  
556 0548 1  
557 0549 1  
558 0550 1  
559 0551 1  
560 0552 1  
561 0553 1  
562 P 0554 1 $STATE (ST_MPR_PRC_DEF,  
563 P 0555 1  
564 P 0556 1 DISPATCH_STATES  
565 P 0557 1 (MPR,  
566 P 0558 1  
567 P 0559 1 DEL, 'DATA',  
568 P 0560 1 DWI, 'WINDOW',  
569 P 0561 1 )  
570 P 0562 1 );  
571 0563 1  
572 0564 1  
573 0565 1  
574 0566 1  
575 0567 1  
576 0568 1  
577 P 0569 1 $STATE (ST_MPR_PRC_MUL,
```

Dispatch on KNOWN keyword

Dispatch on MAXIMUM keyword

Dispatch on DEFAULT keyword

MULTI keyword dispatch

```
: 578 P 0570 1 ('-'),  
: 579 P 0571 1 (TPAS_LAMBDA, ST_MPR_PRC_MNS),  
: 580 P 0572 1 );  
: 581 P 0573 1 $STATE (  
: 582 P 0574 1 ('NETWORK', ST_MPR_PRC_MNS),  
: 583 P 0575 1 (TPAS_LAMBDA, ST_MPR_PRC_MNS),  
: 584 0576 1 );  
: 585 0577 1  
: 586 0578 1  
: 587 0579 1 | Dispatch on DTE keyword. Decide if this is a DTE database qualifier  
: 588 0580 1 | or a DTE parameter qualifier by GROUP (GDT). In both cases, the  
: 589 0581 1 | keyword is the same, so we must decide the NICE code by context.  
: 590 0582 1 |  
: 591 0583 1 |  
: 592 P 0584 1 $STATE (ST_MPR_PRC_DTEX,  
: 593 P 0585 1 (TPAS_LAMBDA, ST_MPR_PRC_GDT, ACT$TESTLONG,,, GROUP_SPECIFIED),  
: 594 0586 1 (TPAS_LAMBDA, ST_MPR_PRC_DTE));
```

```
596 0587 1
597 0588 1
598 0589 1
599 0590 1
600 0591 1
601 P 0592 1
602 P 0593 1
603 P 0594 1
604 P 0595 1
605 P 0596 1
606 P 0597 1
607 P 0598 1
608 P 0599 1
609 P 0600 1
610 P 0601 1
611 P 0602 1
612 P 0603 1
613 P 0604 1
614 P 0605 1
615 P 0606 1
616 P 0607 1
617 P 0608 1
618 P 0609 1
619 P 0610 1
620 P 0611 1
621 P 0612 1
622 P 0613 1
623 P 0614 1
624 P 0615 1
625 P 0616 1
626 P 0617 1
627 P 0618 1
628 P 0619 1
629 0620 1

Process states - call subexpressions to store value

PROCESS_STATES
(MPR,
STA,
CTM, 'TIMER',
DTE,
GRP,
NET,
LIN,
CHN,
DBL,
DWI,
MBL,
MWI,
MCL,
MRS,
MST,
CAT, 'TIMER',
CLT, 'TIMER',
RST, 'TIMER',
STT, 'TIMER',
GDT,
GNM,
GTY,
MNS, 'SUPPORT',
MCI,
)

! From DEF
! From DEF
! From MAX
! From MAX
! From MAX
! From MAX
! From MAX
```

```
631 0621 1
632 0622 1
633 0623 1
634 0624 1
635 0625 1
636 P 0626 1
637 P 0627 1
638 P 0628 1
639 P 0629 1
640 P 0630 1
641 P 0631 1
642 P 0632 1
643 P 0633 1
644 P 0634 1
645 P 0635 1
646 P 0636 1
647 P 0637 1
648 P 0638 1
649 P 0639 1
650 P 0640 1
651 P 0641 1
652 P 0642 1
653 P 0643 1
654 P 0644 1
655 P 0645 1
656 P 0646 1
657 P 0647 1
658 P 0648 1
659 0649 1
```

Sub\_expressions

SUB\_EXPRESSIONS  
(MPR,

CTM, TPAS\_DECIMAL,  
DTE, (SE\_DTE\_NUMBER),  
GRP, (SE\_GRP\_NAME),  
NET, (SE\_NET\_NAME),  
LIN, (SE\_LINE\_ID),  
CHN, (SE\_RNG\_LIST),  
DBL, TPAS\_DECIMAL,  
DWI, TPAS\_DECIMAL,  
MBL, TPAS\_DECIMAL,  
MWI, TPAS\_DECIMAL,  
MCL, TPAS\_DECIMAL,  
MRS, TPAS\_DECIMAL,  
MST, TPAS\_DECIMAL,  
CAT, TPAS\_DECIMAL,  
CLT, TPAS\_DECIMAL,  
RST, TPAS\_DECIMAL,  
STT, TPAS\_DECIMAL,  
GDT, (SE\_DTE\_NUMBER),  
GNM, TPAS\_DECIMAL,  
MCI, TPAS\_DECIMAL,  
)

```

: 661      0650 1  |
: 662      0651 1  |
: 663      0652 1  |
: 664      0653 1  |
: 665      P 0654 1  | $STATE (ST_MPR_STA,
: 666      P 0655 1  |
: 667      P 0656 1  | KEYWORD_STATE
: 668      P 0657 1  | (MPR,
: 669      P 0658 1  |
: 670      P 0659 1  | STAOFF, 'OFF',
: 671      P 0660 1  | STAON, 'ON',
: 672      P 0661 1  | STASHU, 'SHUT',
: 673      P 0662 1  |
: 674      0663 1  | ));
: 675      0664 1  |
: 676      0665 1  |
: 677      0666 1  |
: 678      0667 1  |
: 679      0668 1  |
: 680      P 0669 1  | $STATE (ST_MPR_GTY,
: 681      P 0670 1  |
: 682      P 0671 1  | KEYWORD_STATE
: 683      P 0672 1  | (MPR,
: 684      P 0673 1  |
: 685      P 0674 1  | GTYBIL, 'BILATERAL',
: 686      P 0675 1  |
: 687      0676 1  | ));
: 688      0677 1  |
: 689      0678 1  |
: 690      0679 1  |
: 691      0680 1  |
: 692      0681 1  |
: 693      P 0682 1  | $STATE (ST_MPR_MNS,
: 694      P 0683 1  |
: 695      P 0684 1  | KEYWORD_STATE
: 696      P 0685 1  | (MPR,
: 697      P 0686 1  |
: 698      P 0687 1  | MNSENA, 'ENABLED',
: 699      P 0688 1  | MNSDIS, 'DISABLED',
: 700      P 0689 1  |
: 701      0690 1  | ));
```

```

703 0691 1 %SBTTL 'CLEAR X25-PROTOCOL Module Parameters'
704 0692 1
705 0693 1
706 0694 1
707 0695 1
708 0696 1
709 P 0697 1 $STATE (ST_CPR
710 P 0698 1 ((ST_CPR_INIT))
711 0699 1
712 0700 1
713 P 0701 1 $STATE (
714 P 0702 1 (TPAS_EOS, ST_CPR_PMT_DAT),
715 P 0703 1 (TPAS_LAMBDA, ST_CPR_DAT)
716 0704 1
717 0705 1
718 0706 1
719 0707 1
720 0708 1
721 P 0709 1 $STATE (ST_CPRDTE,
722 P 0710 1 ((ST_CPR_INIT), ST_CPR_DAT_DTE)
723 0711 1
724 0712 1
725 0713 1
726 0714 1
727 0715 1
728 P 0716 1 $STATE (ST_CPRGRP,
729 P 0717 1 ((ST_CPR_INIT), ST_CPR_DAT_GRP)
730 0718 1
731 0719 1
732 0720 1
733 P 0721 1 $STATE (ST_CPR_INIT,
734 P 0722 1 (TPAS_LAMBDA, TPAS_EXIT, ACT$CLRLONG,,, ! Mark GROUP not specified yet
735 0723 1 GROUP_SPECIFIED));
736 0724 1
737 0725 1
738 0726 1
739 0727 1
740 0728 1
741 0729 1
742 0730 1
743 0731 1
744 P 0732 1 $STATE (ST_CPR_PMT_DAT,
745 0733 1 (TPAS_LAMBDA,, ACT$PRMPT,,, PMT$G_CPR_DAT));
746 0734 1
747 P 0735 1 $STATE (ST_CPR_DAT, ! Determine whether GROUP or DTE parameters
748 P 0736 1 ('GROUP', ST_CPR_DAT_GRP),
749 P 0737 1 ('DTE', ST_CPR_DAT_DTE),
750 P 0738 1 ('KNOWN', ST_CPR_DAT_KWN),
751 0739 1 (TPAS_LAMBDA));
752 0740 1
753 0741 1
754 0742 1
755 0743 1
756 0744 1
757 P 0745 1 $STATE (
758 P 0746 1 (TPAS_EOS),
759 0747 1 (TPAS_LAMBDA,ST_CPR_PRC));

```

```

760 0748 1
761 P 0749 1 QUERY_STATES
762 P 0750 1 (CPR,
763 0751 1 ALL, CAT, CLT, MCL, MRS, 1ST, RST, STT, MNS)
764 0752 1
765 P 0753 1 $STATE (
766 0754 1 (TPAS_LAMBDA, ST_CPR_DOIT));
767 0755 1
768 0756 1
769 0757 1 Prompt for closed user group parameters
770 0758 1
771 0759 1
772 P 0760 1 $STATE (ST_CPR_DAT_GRP,
773 0761 1 ((SE_GRP_NAME),, ACT$SAVPRM,,, PBK$G_CPR_GRP));
774 0762 1
775 P 0763 1 $STATE (ST_CPR_PMT_GRP,
776 0764 1 (TPAS_LAMBDA,,, TRUE, GROUP_SPECIFIED)); ! Mark GROUP specified
777 0765 1
778 P 0766 1 $STATE (
779 P 0767 1 (TPAS_EOS),
780 0768 1 (TPAS_LAMBDA,ST_CPR_PRC));
781 0769 1
782 P 0770 1 QUERY_STATES_S
783 P 0771 1 (CPR,
784 P 0772 1 ALL, ALL2,
785 0773 1 GDT, GDT)
786 0774 1
787 P 0775 1 $STATE (
788 0776 1 (TPAS_LAMBDA, ST_CPR_DOIT));
789 0777 1
790 0778 1
791 0779 1 Prompt for DTE parameters
792 0780 1
793 0781 1
794 P 0782 1 $STATE (ST_CPR_DAT_DTE,
795 0783 1 ((SE_DTE_NUMBER),, ACT$SAVPRM,,, PBK$G_CPR_DTE));
796 0784 1
797 P 0785 1 $STATE (ST_CPR_PMT_DTE,
798 P 0786 1 (TPAS_EOS),
799 0787 1 (TPAS_LAMBDA,ST_CPR_PRC));
800 0788 1
801 P 0789 1 QUERY_STATES_S
802 P 0790 1 (CPR,
803 P 0791 1 ALL, ALL3,
804 0792 1 CTM, CTM)
805 0793 1
806 P 0794 1 $STATE (ST_CPR_DOIT,
807 P 0795 1 (TPAS_EOS, TPAS_EXIT, ACT$VRB_UTILITY, , , SDB$G_CPR),
808 0796 1 );
809 0797 1
810 0798 1
811 0799 1 Dispatch on KNOWN keyword during prompting
812 0800 1
813 0801 1
814 P 0802 1 $STATE (ST_CPR_DAT_KWN,
815 0803 1 (TPAS_LAMBDA));
816 0804 1

```

```
: 817      P 0805 1      COMMAND PROMPT
: 818      P 0806 1      (CPR, KQN, NCP$ INVKEY,
: 819      P 0807 1
: 820      P 0808 1      ('DTES', ST_CPR_PMT_DTE, ACT$SAVPRM,,, PBK$G_CPR_KDT),
: 821      P 0809 1      ('GROUPS', ST_CPR_PMT_GRP, ACT$SAVPRM,,, PBK$G_CPR_KGR)
: 822      0810 1      )
```



```
.. 824      0811  1
.. 825      0812  1
.. 826      0813  1      Dispatch States
.. 827      0814  1
.. 828      0815  1
.. 829      P 0816  1  $STATE (ST_CPR_PRC,
.. 830      P 0817  1
.. 831      P 0818  1      DISPATCH_STATES
.. 832      P 0819  1      (CPR,
.. 833      P 0820  1
.. 834      P 0821  1      ALL, 'ALL',
.. 835      P 0822  1      CAT, 'CALL',
.. 836      P 0823  1      CLT, 'CLEAR',
.. 837      P 0824  1      CHN, 'CHANNELS',
.. 838      P 0825  1      CTM, 'COUNTER',
.. 839      P 0826  1      DEF, 'DEFAULT',
.. 840      P 0827  1      DTEX, 'DTE',
.. 841      P 0828  1      GRP, 'GROUP',
.. 842      P 0829  1      KWN, 'KNOWN',
.. 843      P 0830  1      LIN, 'LINE',
.. 844      P 0831  1      MAX, 'MAXIMUM',
.. 845      P 0832  1      MUL, 'MULTI',
.. 846      P 0833  1      NET, 'NETWORK',
.. 847      P 0834  1      GNM, 'NUMBER',
.. 848      P 0835  1      RST, 'RESET',
.. 849      P 0836  1      STT, 'RESTART',
.. 850      P 0837  1      STA, 'STATE',
.. 851      P 0838  1      GTY, 'TYPE',
.. 852      P 0839  1      )
.. 853      P 0840  1      ,(TPAS_EOS, ST_CPR_DOIT)
.. 854      0841  1      );
```

```
856 0842 1  
857 0843 1  
858 0844 1      Dispatch on KNOWN keyword  
859 0845 1  
860 0846 1  
861 P 0847 1 $STATE (ST_CPR_PRC_KWN,  
862 0848 1      ((SE_CPR_KWN), ST_CPR_PRC));  
863 0849 1  
864 P 0850 1 $STATE (SE_CPR_KWN,  
865 P 0851 1  
866 P 0852 1      KEYWORD_STATE  
867 P 0853 1      (CPR,  
868 P 0854 1  
869 P 0855 1      KDT, 'DTES',  
870 P 0856 1      KGR, 'GROUPS',  
871 P 0857 1  
872 0858 1      ));  
873 0859 1  
874 0860 1  
875 0861 1      Dispatch on MAXIMUM keyword  
876 0862 1  
877 0863 1  
878 P 0864 1 $STATE (ST_CPR_PRC_MAX,  
879 P 0865 1  
880 P 0866 1      DISPATCH_STATES  
881 P 0867 1      (CPR,  
882 P 0868 1  
883 P 0869 1      MBL, 'DATA',  
884 P 0870 1      MCI, 'CIRCUITS',  
885 P 0871 1      MCL, 'CLEARS',  
886 P 0872 1      MRS, 'RESETS',  
887 P 0873 1      MST, 'RESTARTS',  
888 P 0874 1      MWI, 'WINDOW',  
889 P 0875 1  
890 P 0876 1      )  
891 0877 1      );  
892 0878 1  
893 0879 1  
894 0880 1      Dispatch on DEFAULT keyword  
895 0881 1  
896 0882 1  
897 P 0883 1 $STATE (ST_CPR_PRC_DEF,  
898 P 0884 1  
899 P 0885 1      DISPATCH_STATES  
900 P 0886 1      (CPR,  
901 P 0887 1  
902 P 0888 1      DBL, 'DATA',  
903 P 0889 1      DWI, 'WINDOW',  
904 P 0890 1  
905 P 0891 1      )  
906 0892 1      );  
907 0893 1  
908 0894 1  
909 0895 1      MULTI keyword dispatch  
910 0896 1  
911 0897 1  
912 P 0898 1 $STATE (ST_CPR_PRC_MUL,
```

```
: 913 P 0899 1 ('-'),
: 914 P 0900 1 (TPAS_LAMBDA, ST_CPR_PRC_MNS),
: 915 P 0901 1 );
: 916 P 0902 1 $STATE (
: 917 P 0903 1 ('NETWORK', ST_CPR_PRC_MNS),
: 918 P 0904 1 (TPAS_LAMBDA, ST_CPR_PRC_MNS),
: 919 0905 1 );
: 920 0906 1
: 921 0907 1
: 922 0908 1 |
: 923 0909 1 | Dispatch on DTE keyword. Decide if this is a DTE database qualifier
: 924 0910 1 | or a DTE parameter qualifier by GROUP (GDT). In both cases, the
: 925 0911 1 | keyword is the same, so we must decide the NICE code by context.
: 926 0912 1 |
: 927 P 0913 1 $STATE (ST_CPR_PRC_DTEX,
: 928 P 0914 1 (TPAS_LAMBDA, ST_CPR_PRC_GDT, ACT$TESTLONG,,, GROUP_SPECIFIED),
: 929 0915 1 (TPAS_LAMBDA, ST_CPR_PRC_DTE));
```

```
.....: 931      0916      1
.....: 932      0917      1
.....: 933      0918      1
.....: 934      0919      1
.....: 935      0920      1
.....: 936      P 0921      1
.....: 937      P 0922      1
.....: 938      P 0923      1
.....: 939      P 0924      1
.....: 940      P 0925      1
.....: 941      P 0926      1
.....: 942      P 0927      1
.....: 943      P 0928      1
.....: 944      P 0929      1
.....: 945      P 0930      1
.....: 946      P 0931      1
.....: 947      P 0932      1
.....: 948      P 0933      1
.....: 949      P 0934      1
.....: 950      P 0935      1
.....: 951      P 0936      1
.....: 952      P 0937      1
.....: 953      P 0938      1
.....: 954      P 0939      1
.....: 955      P 0940      1
.....: 956      P 0941      1
.....: 957      P 0942      1
.....: 958      P 0943      1
.....: 959      P 0944      1
.....: 960      P 0945      1
.....: 961      P 0946      1
.....: 962      P 0947      1
.....: 963      P 0948      1
.....: 964      0949      1

Process states - call subexpressions to store parameter ID's

PROCESS_STATES
(CPR,
ALL, .
STA, .
CTM, 'TIMER',
DTE, .
GRP, .
NET, .
LIN, .
CHN, .
DBL, .
DWI, .
MBL, .
MWI, .
MCL, .
MRS, .
MST, .
CAT, 'TIMER',
CLT, 'TIMER',
RST, 'TIMER',
STT, 'TIMER',
GDT, .
GNM, .
GTY, .
MNS, 'SUPPORT',
MCI, .
)

! From DEF
! From DEF
! From MAX
! From MAX
! From MAX
! From MAX
! From MAX
! From MAX
```

```

: 966      0950  1
: 967      0951  1
: 968      0952  1
: 969      0953  1
: 970      0954  1
: 971      P 0955  1
: 972      P 0956  1
: 973      P 0957  1
: 974      P 0958  1
: 975      P 0959  1
: 976      P 0960  1
: 977      P 0961  1
: 978      P 0962  1
: 979      P 0963  1
: 980      P 0964  1
: 981      P 0965  1
: 982      P 0966  1
: 983      P 0967  1
: 984      P 0968  1
: 985      P 0969  1
: 986      P 0970  1
: 987      P 0971  1
: 988      P 0972  1
: 989      P 0973  1
: 990      P 0974  1
: 991      P 0975  1
: 992      P 0976  1
: 993      P 0977  1
: 994      P 0978  1
: 995      P 0979  1
: 996      P 0980  1
: 997      P 0981  1
: 998      P 0982  1
: 999      0983  1

```

Sub\_expressions

SUB EXPRESSIONS  
(CPR,  
ALL, TPAS\_EOS,  
STA, TPAS\_LAMBDA,  
CTM, TPAS\_LAMBDA,  
DTE, (SE\_DTE\_NUMBER),  
GRP, (SE\_GRP\_NAME),  
NET, TPAS\_LAMBDA,  
LIN, TPAS\_LAMBDA,  
CHN, TPAS\_LAMBDA,  
DBL, TPAS\_LAMBDA,  
DWI, TPAS\_LAMBDA,  
MBL, TPAS\_LAMBDA,  
MWI, TPAS\_LAMBDA,  
MCL, TPAS\_LAMBDA,  
MRS, TPAS\_LAMBDA,  
MST, TPAS\_LAMBDA,  
CAT, TPAS\_LAMBDA,  
CLT, TPAS\_LAMBDA,  
RST, TPAS\_LAMBDA,  
STT, TPAS\_LAMBDA,  
GDT, (SE\_DTE\_NUMBER),  
GNM, TPAS\_LAMBDA,  
GTY, TPAS\_LAMBDA,  
MNS, TPAS\_LAMBDA,  
MCI, TPAS\_LAMBDA,  
)

```
: 1001      0984 1 %SBTTL 'Define Subexpressions'  
: 1002      0985 1  
: 1003      0986 1  
: 1004      0987 1  
: 1005      0988 1  
: 1006      0989 1  
: 1007      0990 1  
: 1008      0991 1  
: 1009      0992 1  
: 1010      0993 1  
: 1011      0994 1  
: 1012      0995 1  
: 1013      0996 1
```

Define Subexpressions from Library

```
SEM_ALL           ! All parameter  
SEM_LINE_ID      ! Device name string  
SEM_DTE_NUMBER   ! DTE call number string  
SEM_GRP_NAME     ! Group name  
SEM_NET_NAME     ! Network name  
SEM_RNG_LIST     ! Range lists  
SEM_QUERY        ! Query state subexpressions
```

NCPSTAMPR  
V04-000

X.25 Protocol Module Parsing  
Define Subexpressions

: 1015  
: 1016

0997 1 END  
0998 0 ELUDOM

B 13  
16-Sep-1984 00:54:06  
14-Sep-1984 12:48:29

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

Page 29  
(19)

NCP  
V04

.....

