

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 LL      000000 GGGGGGGG
NN      NN      CCCCCCCC PPPPPPPP SSSSSSSS TTTTTTTTTT AAAAAA LL      000000 GGGGGGGG
NN      NN      CC      PP      PP SS      TT      AA      AA LL      00      00 GG
NN      NN      CC      PP      PP SS      TT      AA      AA LL      00      00 GG
NNNN    NN      CC      PP      PP SS      TT      AA      AA LL      00      00 GG
NNNN    NN      CC      PP      PP SS      TT      AA      AA LL      00      00 GG
NN      NN      CC      PPPPPPPP SSSSSS      TT      AA      AA LL      00      00 GG
NN      NN      CC      PPPPPPPP SSSSSS      TT      AA      AA LL      00      00 GG
NN      NN      CC      PP      SS      TT      AAAAAAAAAA LL      00      00 GG
NN      NN      CC      PP      SS      TT      AAAAAAAAAA LL      00      00 GG
NN      NN      CC      PP      SS      TT      AA      AA LL      00      00 GG
NN      NN      CC      PP      SS      TT      AA      AA LL      00      00 GG
NN      NN      CCCCCCCC PP      SSSSSSSS      TT      AA      AA LLLLLLLLLL
NN      NN      CCCCCCCC PP      SSSSSSSS      TT      AA      AA LLLLLLLLLL

LL      I I I I I SSSSSSSS
LL      I I I I I SSSSSSSS
LL      I I      SS
LL      I I      SS
LL      I I      SS
LL      I I      SS
LL      I I      SSSSSS
LL      I I      SSSSSS
LL      I I      SS
LL      I I      SS
LL      I I      SS
LL      I I      SS
LL      I I      SS
LLLLLLLLLLLL I I I I I SSSSSSSS
LLLLLLLLLLLL I I I I I SSSSSSSS

```

```
0001 0 %TITLE 'Logging Parameter Parse States and Data'
0002 0 MODULE NCPSTALOG (IDENT = 'V04-000', LIST(NOOBJECT)) =
0003 1 BEGIN
0004 1
0005 1
0006 1 *****
0007 1 *
0008 1 *   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0009 1 *   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0010 1 *   ALL RIGHTS RESERVED.
0011 1 *
0012 1 *   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0013 1 *   ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0014 1 *   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0015 1 *   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0016 1 *   OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0017 1 *   TRANSFERRED.
0018 1 *
0019 1 *   THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0020 1 *   AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0021 1 *   CORPORATION.
0022 1 *
0023 1 *   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0024 1 *   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0025 1 *
0026 1 *
0027 1 *****
0028 1
0029 1
0030 1 ++
0031 1 FACILITY:      Network Control Program (NCP)
0032 1
0033 1 ABSTRACT:
0034 1
0035 1       States and data for the parsing of NCP logging parameters
0036 1
0037 1 ENVIRONMENT:   VAX/VMS Operating System
0038 1
0039 1 AUTHOR:       Darrell Duffy   , CREATION DATE: 10-September-79
0040 1
0041 1 MODIFIED BY:
0042 1
0043 1       V03-002 RPG0002      Bob Grosso      26-Feb-1983
0044 1       Rename ESMO to MODPRM.
0045 1
0046 1       V03-001 RPG0001      Bob Grosso      23-Feb-1983
0047 1       Add LOG MONITOR MODULE (X25-PROT,X25-SERVER,X29-SERVER).
0048 1
0049 1       V002      TMH0002      Tim Halvorsen  11-Nov-1981
0050 1       Allow a circuit source as well as line and node.
0051 1
0052 1       V001      TMH0001      Tim Halvorsen  22-Jun-1981
0053 1       Change BUILD_SDB reference to use full entity type field.
0054 1 --
```



```

: 56 0055 1 %SBTTL 'Definitions'
: 57 0056 1
: 58 0057 1
: 59 0058 1 INCLUDE FILES:
: 60 0059 1
: 61 0060 1
: 62 0061 1 LIBRARY 'LIB$:NMALIBRY';
: 63 0062 1 LIBRARY 'LIB$:NCPLIBRY';
: 64 0063 1 LIBRARY 'SYSS$LIBRARY:TPAMAC';
: 65 0064 1
: 66 0065 1
: 67 0066 1 EXTERNAL REFERENCES:
: 68 0067 1
: 69 0068 1
: 70 0069 1 ACT_DFN ! Action routine externals

```

```
72 0070 1 %SBTTL 'Parameter blocks'
73 0071 1
74 0072 1
75 0073 1 BIND DATA:
76 0074 1
77 0075 1
78 0076 1
79 0077 1 Parameter Blocks
80 0078 1
81 0079 1
82 P 0080 1 BUILD_PCL
83 P 0081 1
84 P 0082 1 (LOG,
85 P 0083 1
86 P 0084 1 STA, NUMB, PCLO_STA,
87 P 0085 1 NAM, TKN, PCLO_LNA,
88 P 0086 1 SNO, NADR, PCLO_SIN,
89 P 0087 1 EVL, ESET, PCLO_EVE, VRB_EVE, ! Event logging type stored
90 P 0088 1 ! in PDB$G_VRB_EVE
91 P 0089 1
92 P 0090 1 , END, ,
93 0091 1 )
94 0092 1
95 P 0093 1 BUILD_PBK
96 P 0094 1
97 P 0095 1 (LOG,
98 P 0096 1
99 P 0097 1 EVL, LITB, 0,
100 P 0098 1 KNE, LITB, 0, LOG_EVL,
101 P 0099 1 LIN, ESLI, , VRB_EVE,
102 P 0100 1
103 P 0101 1 ! ACT$SAVPRM will specially handle type MODPRM to store the correct module string in PDB$G_VRB
104 P 0102 1 MPR, MODPRM, NCP$C_ENT_MODPRO, VRB_EVE,
105 P 0103 1 MSE, MODPRM, NCP$C_ENT_MODSER, VRB_EVE,
106 P 0104 1 M9S, MODPRM, NCP$C_ENT_MOD29S, VRB_EVE,
107 P 0105 1
108 P 0106 1 NAM, TKN, , VRB_EVE,
109 P 0107 1 NOD, ESN0, , VRB_EVE,
110 P 0108 1 CIR, ESCI, , VRB_EVE,
111 P 0109 1 SEX, LITL, 0, LOG_SNO,
112 P 0110 1 SNO, NADR, ,
113 P 0111 1 STAON, LITB, NMASC_STATE_ON, LOG_STA,
114 P 0112 1 STAOFF, LITB, NMASC_STATE_OFF, LOG_STA,
115 P 0113 1 STAHL, LITB, NMASC_STATE_HOL, LOG_STA,
116 P 0114 1
117 0115 1 )
118 0116 1
119 P 0117 1 BUILD_SDB
120 P 0118 1
121 0119 1 (LOG, NMASC_ENT_LOG, VRB_ENT, LOG)
```



```
123 0120 1 %SBTTL 'Prompt strings'
124 0121 1
125 0122 1
126 0123 1 Build prompt strings
127 0124 1
128 0125 1
129 0126 1 BIND
130 0127 1
131 P 0128 1 PROMPT_STRINGS
132 P 0129 1 (LOG,
133 P 0130 1
134 P 0131 1 EVK, 'Event list (cls.typ,typ-typ, KNOWN): ',
135 P 0132 1 NAM, 'Sink name (filename.typ): ',
136 P 0133 1 MOD, 'Module (X25-PROTOCOL, X25-SERVER, X29-SERVER): ',
137 P 0134 1 SNK, 'Sink node (NODE node-id, EXECUTOR): ',
138 P 0135 1 SRC,
139 L 0136 1 %STRING('Source (NODE node-id, LINE line-id,',CRLF,
140 P 0137 1 'CIRCUIT circ-id, MODULE module-name): '),
141 P 0138 1 STA, 'Sink state (ON, OFF, HOLD): ',
142 P 0139 1
143 0140 2 )
144 0141 1 ;
```

```
: 146      0142 1 %SBTTL 'State Table for Logging Parameters'
: 147      0143 1
: 148      0144 1 $INIT_STATE (NCP$G_STTBL_LOG, NCP$G_KYTBL_LOG);
: 149      0145 1
: 150      0146 1 !
: 151      0147 1 ! SET/DEFINE LOGGING Parameter States
: 152      0148 1 !
: 153      0149 1 !
: 154      P 0150 1 $STATE (ST_LOG,
: 155      P 0151 1 (TPAS_LAMBDA, , ACT$SAVPRM, , , PBK$G_EVE_ESET)
: 156      0152 1 );
: 157      0153 1
: 158      P 0154 1 $STATE (,
: 159      P 0155 1 ( (SE_ALL), ST_LOG DOIT), ! All parameter
: 160      P 0156 1 (TPAS_EOS, , ACT$PMT_ON), ! Prompt if no keywords
: 161      P 0157 1 (TPAS_LAMBDA, ST_LOG_PRC, ACT$PMT_OFF) ! Process keywords
: 162      0158 1 );
: 163      0159 1
: 164      P 0160 1 PROMPT_STATES ! Prompt for these parameters
: 165      P 0161 1 (LOG,
: 166      P 0162 1 SRC, EVK, SNK, NAM, STA
: 167      0163 1 )
: 168      0164 1
: 169      P 0165 1 $STATE (ST_LOG DOIT,
: 170      P 0166 1 (TPAS_EOS, TPAS_EXIT, ACT$VRB_UTILITY, , , SDB$G_LOG),
: 171      0167 1 );
```



```
173 0168 1 %SBTTL 'Substate Dispatching'
174 0169 1
175 0170 1
176 0171 1 Substate dispatching
177 0172 1
178 0173 1
179 P 0174 1 $STATE (ST_LOG_PRC,
180 P 0175 1 DISPATCH_STATES
181 P 0176 1 (LOG,
182 P 0177 1
183 P 0178 1 CIR, 'CIRCUIT',
184 P 0179 1 EVL, 'EVENTS',
185 P 0180 1 KNE, 'KNOWN',
186 P 0181 1 LIN, 'LINE',
187 P 0182 1 NAM, 'NAME',
188 P 0183 1 NOD, 'NODE',
189 P 0184 1 SNK, 'SINK',
190 P 0185 1 STA, 'STATE',
191 P 0186 1
192 P 0187 1 ),
193 P 0188 1
194 P 0189 1 ('MODULE', ST_LOG_MOD),
195 P 0190 1 ('X25', ST_LOG_X25),
196 P 0191 1 ('X29', ST_LOG_X29),
197 P 0192 1 (TPAS_EOS, ST_LOG_DOIT)
198 P 0193 1 );
199 0194 1
200 0195 1 %SBTTL 'Process States'
201 0196 1
202 0197 1
203 0198 1 Process states
204 0199 1
205 0200 1
206 0201 1
207 P 0202 1 PROCESS_STATES
208 P 0203 1 (LOG,
209 P 0204 1
210 P 0205 1 CIR, ,
211 P 0206 1 EVL, ,
212 P 0207 1 KNE, 'EVENTS',
213 P 0208 1 LIN, ,
214 P 0209 1 NAM, ,
215 P 0210 1 NOD, ,
216 P 0211 1 SEX, ,
217 P 0212 1 SNK, ,
218 P 0213 1 STA, ,
219 P 0214 1
220 0215 1 )
```



```
222 0216 1 %SBTTL 'Subexpressions'
223 0217 1
224 0218 1
225 0219 1 Sub_expressions
226 0220 1
227 0221 1
228 P 0222 1 SUB_EXPRESSIONS
229 P 0223 1 (LOG,
230 P 0224 1
231 P 0225 1 CIR, (SE_CIRC_ID),
232 P 0226 1 EVL, (SE_EVENT_LIST),
233 P 0227 1 KNE, (SE_EVENT_KNOWN),
234 P 0228 1 LIN, (SE_LINE_ID),
235 P 0229 1 NOD, (SE_NODE_ID),
236 P 0230 1 NAM, (SE_FILE_ID),
237 P 0231 1 SEX, TPAS_LAMBDA,
238 P 0232 1 SNO, (SE_NODE_ID),
239 P 0233 1
240 0234 1 )
241 0235 1
242 P 0236 1 $STATE (ST_LOG_EVK,
243 P 0237 1 ('KNOWN'),
244 P 0238 1 ((SE_EVENT_LIST), TPAS_EXIT, ACT$SAVPRM, , , PBK$G_LOG_EVL),
245 0239 1 );
246 0240 1
247 P 0241 1 $STATE (,
248 P 0242 1 ((SE_EVENT_KNOWN), TPAS_EXIT, ACT$SAVPRM, , , PBK$G_LOG_EVL),
249 0243 1 );
250 0244 1
251 0245 1
252 P 0246 1 $STATE (ST_LOG_SRC,
253 P 0247 1 ('LINE', ST_LOG_LIN),
254 P 0248 1 ('MODULE', ST_LOG_MOD),
255 P 0249 1 ('NODE', ST_LOG_NOD),
256 0250 1 ('CIRCUIT', ST_LOG_CIR));
257 0251 1
258 P 0252 1 $STATE (ST_LOG_SNK,
259 P 0253 1 ('EXECUTOR', ST_LOG_SEX),
260 P 0254 1 ('NODE', ST_LOG_SNO),
261 0255 1 );
262 0256 1
263 P 0257 1 $STATE (ST_LOG_STA,
264 P 0258 1
265 P 0259 1 KEYWORD_STATE
266 P 0260 1 (LOG,
267 P 0261 1
268 P 0262 1 STAOFF, 'OFF',
269 P 0263 1 STAON, 'ON',
270 P 0264 1 STAHL, 'HOLD',
271 P 0265 1
272 P 0266 1 )
273 P 0267 1
274 0268 1 );
```

```
: 276      0269 1 %SBTTL 'MODULE'
: 277      0270 1
: 278      0271 1
: 279      0272 1
: 280      0273 1
: 281      P 0274 1 $STATE (ST_LOG_MOD,
: 282      P 0275 1 ('X25', ST_LOG_X25),
: 283      P 0276 1 ('X29', ST_LOG_X29),
: 284      P 0277 1 (TPAS_EOS, ST_LOG_PMT_MOD),
: 285      0278 1 );
: 286      0279 1
: 287      P 0280 1 $STATE (ST_LOG_PMT_MOD,
: 288      0281 1 (TPAS_LAMBDΛ, , ACT$PRMPT, , , PMT$G_LOG_MOD));
: 289      0282 1
: 290      P 0283 1 $STATE (
: 291      P 0284 1 ('X25', ST_LOG_X25),
: 292      P 0285 1 ('X29', ST_LOG_X29),
: 293      P 0286 1 (TPAS_EOS, ST_LOG_PMT_MOD)
: 294      0287 1 );
: 295      0288 1
: 296      P 0289 1 $STATE (ST_LOG_X25,
: 297      0290 1 ('-T'));
: 298      P 0291 1 $STATE (
: 299      P 0292 1 ('PROTOCOL', ST_LOG_PRC, ACT$SAVPRM, , , PBK$G_LOG_MPR),
: 300      P 0293 1 ('SERVER', ST_LOG_PRC, ACT$SAVPRM, , , PBK$G_LOG_MSE),
: 301      0294 1 );
: 302      0295 1
: 303      P 0296 1 $STATE (ST_LOG_X29,
: 304      0297 1 ('-T'));
: 305      P 0298 1 $STATE (
: 306      P 0299 1 ('SERVER', ST_LOG_PRC, ACT$SAVPRM, , , PBK$G_LOG_M9S),
: 307      0300 1 );
```



```

: 309      0301 1 %SBTTL 'Define Subexpressions'
: 310      0302 1
: 311      0303 1
: 312      0304 1
: 313      0305 1
: 314      0306 1
: 315      0307 1
: 316      0308 1
: 317      0309 1
: 318      0310 1
: 319      0311 1
: 320      0312 1

          !
          ! Define subexpressions from library
          !
          SEM_ALL      ! All parameter
          SEM_FILE_ID  ! File id string
          SEM_EVENT_LIST ! Event List
          SEM_LINE_ID  ! Line id strings
          SEM_CIRC_ID  ! Circuit id strings
          SEM_NODE_ID  ! Node id strings
```

NCPSTALOG
V04-000

Logging Parameter Parse States and Data
Object Listing of Parse Table

J 3
16-Sep-1984 00:39:19
14-Sep-1984 12:48:24

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

Page 10
(10)

: 322
: 323
: 324
: 325

0313 1 %SBTTL 'Object Listing of Parse Table'
0314 1
0315 1 END
0316 0 ELUDOM

!End of module

0270

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY