


```
DDDDDDDD  TTTTTTTTTT  SSSSSSSS  MM  MM  AAAAAA  IIIIII  NN  NN
DDDDDDDD  TTTTTTTTTT  SSSSSSSS  MM  MM  AAAAAA  IIIIII  NN  NN
DD  DD  DD  TT  SS  SSSSSSSS  MMMM  MMMM  AA  AA  II  II  NN  NN
DD  DD  DD  TT  SS  SSSSSSSS  MMMM  MMMM  AA  AA  II  II  NN  NN
DD  DD  DD  TT  SS  SSSSSSSS  MM  MM  AA  AA  II  II  NNNN  NN
DD  DD  DD  TT  SS  SSSSSSSS  MM  MM  AA  AA  II  II  NNNN  NN
DD  DD  DD  TT  SS  SSSSSSSS  MM  MM  AA  AA  II  II  NN  NN
DD  DD  DD  TT  SS  SSSSSSSS  MM  MM  AA  AA  II  II  NN  NN
DD  DD  DD  TT  SS  SSSSSSSS  MM  MM  AA  AA  II  II  NN  NN
DD  DD  DD  TT  SS  SSSSSSSS  MM  MM  AA  AA  II  II  NN  NN
DDDDDDDD  TT  SSSSSSSS  MM  MM  AA  AA  IIIIII  NN  NN
DDDDDDDD  TT  SSSSSSSS  MM  MM  AA  AA  IIIIII  NN  NN
                                     ....
                                     ....
                                     ....
                                     ....
```

```
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
LLLLLLLLLL  IIIIII  SSSSSSSS
LLLLLLLLLL  IIIIII  SSSSSSSS
```


(2)	70	COMMAND LANGUAGE SYNTAX
(3)	166	DECLARATIONS
(4)	191	TST\$START_DTS - MAINLINE

```
0000 1 .TITLE TSTSDTSM - DTS MAINLINE
0000 2 .IDENT 'V04-000'
0000 3
0000 4
0000 5 *****
0000 6
0000 7 *
0000 8 * COPYRIGHT (c) 1978, 1980, 1982, 1984 BY
0000 9 * DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.
0000 10 * ALL RIGHTS RESERVED.
0000 11 *
0000 12 * THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED
0000 13 * ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE
0000 14 * INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER
0000 15 * COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY
0000 16 * OTHER PERSON. NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY
0000 17 * TRANSFERRED.
0000 18 *
0000 19 * THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE
0000 20 * AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT
0000 21 * CORPORATION.
0000 22 *
0000 23 * DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS
0000 24 * SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.
0000 25 *
0000 26 *****
0000 27
0000 28
0000 29 ++
0000 30 FACILITY: DTS/DTR DECNET TEST PACKAGE
0000 31
0000 32 ABSTRACT:
0000 33
0000 34 DTS/DTR ARE COOPERATING TEST PROGRAMS THAT EXERCISE NSP LEVEL
0000 35 FUNCTIONS ON A DECNET/VAX-11 NODE. DTS (SENDER) INITIATES A TEST
0000 36 SEQUENCE WHILE DTR (RECEIVER) IS THE PASSIVE PARTNER THAT PERFORMS
0000 37 THE REQUESTED FUNCTION. FIVE BASIC TESTS ARE IMPLEMENTED:
0000 38
0000 39 1. THE CONNECT TEST EXERCISES THE CONNECT ACCEPT/REJECT LOGIC
0000 40 WITH/WITHOUT OPTIONAL DATA.
0000 41
0000 42 2. THE DATA TEST TRANSMITS DATA MESSAGES AND OPTIONALLY COMPUTES
0000 43 THROUGHPUT STATISTICS. VARIOUS DATA VALIDATION CHECKS
0000 44 AND FLOW CONTROL MECHANISMS MAY BE ENABLED.
0000 45
0000 46 3. THE DISCONNECT TEST EXERCISES THE DISCONNECT SYNCHRONOUS/ABORT
0000 47 LOGIC WITH/WITHOUT OPTIONAL DATA.
0000 48
0000 49 4. THE INTERRUPT TEST TRANSMITS INTERRUPT MESSAGES AND OPTIONALLY
0000 50 COMPUTES THROUGHPUT STATISTICS. VARIOUS DATA VALIDATION
0000 51 CHECKS AND FLOW CONTROL MAY BE ENABLED.
0000 52
0000 53 5. THE MISCELLANEOUS TEST EXERCISES DECNET/VAX-11 SPECIFIC
0000 54 FUNCTIONS AND VALIDATES ERROR LOGIC.
0000 55
0000 56 ENVIRONMENT: DTS RUNS IN USER MODE AND REQUIRES NETWORK PRIVILEGE.
0000 57
```



```
0000 58 : AUTHOR: JAMES A. KRYCKA,      CREATION DATE: 11-AUG-77
0000 59 :
0000 60 : MODIFICATIONS:
0000 61 :
0000 62 :      V03-001 JAK0001      J A Krycka      27-JUN-1983
0000 63 :      Make call to LIB$ASN_WTH_MBX position independent.
0000 64 :
0000 65 :      X0.1-8 DJD0001      Darrell Duffy      8-December-1979
0000 66 :      Change to use LIB$ASN_WTH_MBX
0000 67 :
0000 68 :--
```

```
0000 70      .SBTTL  COMMAND LANGUAGE SYNTAX
0000 71
0000 72      :++
0000 73      :
0000 74      : THE DTS COMMAND (FOR VAX/VMS) HAS THE FOLLOWING FORMAT:
0000 75      :
0000 76      : $ DTSC[/qualifier,...]  parameter[/qualifier,...]
0000 77      :
0000 78      : WHERE THE COMMAND AND COMMAND QUALIFIERS ARE:
0000 79      :
0000 80      : $ DTSC[/NODENAME=xxxxxx][ /SPEED=nnnnnnnn][ /NOSTATISTICS][ /NOPRINT]
0000 81      :                                     [ /STATISTICS ][ /PRINT ]
0000 82      :
0000 83      :     [ /NODISPLAY ] ...
0000 84      :     [ /DISPLAY=nn ]
0000 85      :
0000 86      : AND WHERE THE PARAMETER AND PARAMETER QUALIFIERS ARE ONE OF THE FOLLOWING:
0000 87      :
0000 88      : ... CONNECT[/TYPE=REJECT][ /NORETURN      ]
0000 89      :                   [ /TYPE=ACCEPT][ /RETURN=STANDARD]
0000 90      :                   [ /RETURN=RECEIVED]
0000 91      :
0000 92      : ... DATA[/TYPE=SINK      ][ /SIZE=nnnnn ][ /HOURS=nnn ][ /SQUEUE=n ][ RQUEUE=n ]
0000 93      :                   [ /TYPE=SEQUENCE ][ /MINUTES=nnnn ]
0000 94      :                   [ /TYPE=PATTERN ][ /SECONDS=nnnnnn ]
0000 95      :                   [ /TYPE=ECHO ]
0000 96      :
0000 97      :                   [ /NOFLOW ][ /NONAK ][ /NOBACK ]
0000 98      :                   [ /FLOW=MESSAGE ][ /NAK=nnn ][ /BACK=nnn ]
0000 99      :                   [ /FLOW=SEGMENT ]
0000 100      :
0000 101      : ... DISCONNECT[/TYPE=SYNCHRONOUS][ /NORETURN      ]
0000 102      :                   [ /TYPE=ABORT ][ /RETURN=STANDARD]
0000 103      :                   [ /RETURN=RECEIVED]
0000 104      :
0000 105      : ... INTERRUPT[/TYPE=SINK      ][ /SIZE=nn ][ /HOURS=nnn ][ /SQUEUE=n ][ RQUEUE=n ]
0000 106      :                   [ /TYPE=SEQUENCE ][ /MINUTES=nnnn ]
0000 107      :                   [ /TYPE=PATTERN ][ /SECONDS=nnnnnn ]
0000 108      :                   [ /TYPE=ECHO ]
0000 109      :
0000 110      : ... MISCELLANEOUS[/TYPE=NAME ]
0000 111      :                   [ /TYPE=OBJECT ]
0000 112      :                   [ /TYPE=LOGICAL ]
0000 113      :
0000 114      :
0000 115      : COMMAND QUALIFIER DEFAULTS:
0000 116      :
0000 117      : NODE = NULL STRING; I.E., THE LOCAL NODE
0000 118      : SPEED = 1000000 BAUD (USED FOR STATISTICS)
0000 119      : STATISTICS
0000 120      : NOPRINT
0000 121      : NODISPLAY
0000 122      :
0000 123      : CONNECT TEST PARAMETER QUALIFIER DEFAULTS:
0000 124      :
0000 125      : TYPE = ACCEPT
0000 126      : NORETURN
```



```
0000 127 :  
0000 128 : DATA TEST PARAMETER QUALIFIER DEFAULTS:  
0000 129 :  
0000 130 : TYPE = SINK  
0000 131 : SIZE = 128 (BYTES)  
0000 132 : SECONDS = 30  
0000 133 : SQUEUE = 1  
0000 134 : RQUEUE = 1  
0000 135 : NOFLOW  
0000 136 : NONAK  
0000 137 : NOBACK  
0000 138 :  
0000 139 : DISCONNECT TEST PARAMETER QUALIFIER DEFAULTS:  
0000 140 :  
0000 141 : TYPE = ABORT  
0000 142 : NORETURN  
0000 143 :  
0000 144 : INTERRUPT TEST PARAMETER QUALIFIER DEFAULTS:  
0000 145 :  
0000 146 : TYPE = SINK  
0000 147 : SIZE = 16 (BYTES)  
0000 148 : SECONDS = 30  
0000 149 : SQUEUE = 1  
0000 150 : RQUEUE = 1  
0000 151 :  
0000 152 : MISCELLANEOUS TEST PARAMETER QUALIFIER DEFAULTS:  
0000 153 :  
0000 154 : TYPE = NAME  
0000 155 :  
0000 156 : NOTES:  
0000 157 :  
0000 158 : 1. A COLON MAY BE USED IN PLACE OF THE EQUAL_SIGN TO DELIMIT A  
0000 159 : QUALIFIER AND ITS VALUE.  
0000 160 :  
0000 161 : 2. ONLY THE FIRST FOUR CHARACTERS OF A PARAMETER, QUALIFIER, AND  
0000 162 : A NON-NUMERIC QUALIFIER VALUE ARE SIGNIFICANT.  
0000 163 :  
0000 164 :--
```

```
0000 166      .SBTTL  DECLARATIONS
0000 167
0000 168 :
0000 169 : INCLUDE FILES:
0000 170 :
0000 171      $DTSDEF      ;DTS ERROR MESSAGES
0000 172      $DEVDEF     ;DEFINE RMS DEVICE CHARS
0000 173      FLGDEF      ; DEFINE COMMAND PARSE FLAGS
0000 174      CMDDEF      ; DEFINE COMMAND LANGUAGE SYMBOLS
0000 175      VLDDEF      ; DEFINE VALID QUALIFIER FLAGS
0000 176      $RABDEF     ; DEFINE RAB OFFSETS
0000 177      .IIF NE K_LIST_MEB, .LIST MEB ; DEFINED IN DTPREFIX.MAR
0000 178 :
0000 179 : MACROS:
0000 180 :
0000 181 :     NONE
0000 182 :
0000 183 : EQUATED SYMBOLS:
0000 184 :
0000 185 :     NONE
0000 186 :
0000 187 : OWN STORAGE:
0000 188 :
0000 189 :     NONE
```



```
0000 191 .SBTTL TST$START_DTS - MAINLINE
0000 192 .PSECT TST$CODE NOWRT
0000 193 S:: ; SYMBOL FOR DEBUGGING PURPOSES
0000 194
0000 195
0000 196 :++
0000 197 : FUNCTIONAL DESCRIPTION:
0000 198 :
0000 199 : NONE
0000 200 : CALLING SEQUENCE:
0000 201 :
0000 202 : $ RUN DTS
0000 203 :
0000 204 : INPUT PARAMETERS:
0000 205 :
0000 206 : NONE
0000 207 :
0000 208 : IMPLICIT INPUTS:
0000 209 :
0000 210 : NONE
0000 211 :
0000 212 : OUTPUT PARAMETERS:
0000 213 :
0000 214 : NONE
0000 215 :
0000 216 : IMPLICIT OUTPUTS:
0000 217 :
0000 218 : NONE
0000 219 :
0000 220 : COMPLETION CODES:
0000 221 :
0000 222 : NONE
0000 223 :
0000 224 : SIDE EFFECTS:
0000 225 :
0000 226 : NONE
0000 227 :
0000 228 :--
0000 229
0000 230 .ENTRY TST$START_DTS,^M<> ; ENTRY POINT FROM EXECUTIVE
0002 231
0002 232 :
0002 233 : OPEN THE PRINT FILE
0002 234 :
0002 235
0002 236 $OPEN FAB=W^TST$PRTFAB ; OPEN THE FILE
000D 237 CHECK_RMS ; CHECK COMPLETION CODE
0010 238 $CONNECT RAB=W^TST$PRTRAB ; ESTABLISH A RECORD STREAM
001B 239 CHECK_RMS ; CHECK COMPLETION CODE
001E 240
001E 241 :
001E 242 : OUTPUT INITIALIZATION MESSAGE TO PRINT FILE
001E 243 :
001E 244
001E 245 $FAO_S CTRSTR=W^TST$GQ_INIT- ; FORMAT MESSAGE
001E 246 OUTLEN=W^TST$GQ_PRTLEN- ;
001E 247 OUTBUF=W^TST$GQ_PRTBUF- ;
```

```
001E 248 P1=#TST$GT_DTS- ; INSERT DTS ID
001E 249 P2=#TST$GT_VERSION- ; INSERT DTS VERSION NUMBER
001E 250 P3=#0 ; INSERT DATE AND TIME
003F 251 CHECK_SS ; CHECK STATUS CODE
FFBB' 30 0042 252 BSBW TST$PRINT_FAO ; PRINT MESSAGE
0045 253
0045 254
0045 255 : CREATE A TEMPORARY MAILBOX THAT WILL BE USED AS THE ASSOCIATED LINK
0045 256 : CHANNEL MAILBOX AND ASSIGN A CHANNEL TO IT.
0045 257 :
0045 258
0045 259 :
0045 260 : CREATE A CONTROL/INFORMATION PATH TO NETACP IN PREPARATION FOR
0045 261 : NON-TRANSPARENT NETWORK I/O. ALSO ASSOCIATE A MAILBOX WITH THE
0045 262 : CHANNEL.
0045 263 :
0045 264 :
0045 265 :
0045 266 : These things are all done by the LIB$ASN_WTH_MBX routine
0045 267 :
0045 268
00000000'8F DD 0045 269 PUSHL #TST$K_MAILQUOTA ; Setup mailbox quota
52 5E DO 004B 270 MOVL SP, R2 ; Save its address
00000000'8F DD 004E 271 PUSHL #TST$K_MAILBUF ; Mailbox message size
51 5E DO 0054 272 MOVL SP, R1 ; its address too
0000'CF 3F 0057 273 PUSHAW W^TST$GW_MAILCHAN ; Address of mailbox channel
0000'CF 3F 005B 274 PUSHAW W^TST$GW_LINKCHAN ; Address for link channel
06 BB 005F 275 PUSHR #M<R1, R2> ; Those addresses we saved
0000'CF 7F 0061 276 PUSHAQ W^TST$GQ_LINKNAME ; The name for the device (_NET)
00000000'GF 05 FB 0065 277 CALLS #5,G^LIB$ASN_WTH_MBX ; Assign the channels and create mbx
5E 08 CO 006C 278 ADDL2 #8, SP ; Dump the quota and message size
006F 279 CHECK_SS ; Check the system service status code
0072 280
0072 281
0072 282 :
0072 283 : OPEN THE COMMAND FILE
0072 284 :
0072 285
0072 286 $OPEN FAB=W^TST$CMDTAB ; OPEN THE FILE
007D 287 CHECK_RMS ; CHECK COMPLETION CODE
0080 288 $CONNECT RAB=W^TST$CMDRAB ; ESTABLISH A RECORD STREAM
008B 289 CHECK_RMS ; CHECK COMPLETION CODE
008E 290
008E 291 :
008E 292 : APPLY COMMAND QUALIFIER DEFAULTS (NOT PARAMETER QUALIFIER DEFAULTS).
008E 293 : THESE QUALIFIERS ARE "STICKY" IN THE SENSE THAT ONCE MODIFIED BY A
008E 294 : COMMAND, THEY RETAIN THEIR NEW SETTING UNTIL MODIFIED AGAIN BY A COMMAND.
008E 295 : THIS IS IN CONTRAST TO PARAMETER QUALIFIERS WHICH ARE RESET TO THEIR
008E 296 : ORIGINAL DEFAULT VALUES BEFORE PARSING EACH COMMAND.
008E 297 :
008E 298
0000'CF 00 90 008E 299 MOVB #DFT_K_DISPLAY,W^TST$GB_DISPLAY ; [NO]DISPLAY
0000'CF 94 0093 300 CLR B W^TST$GT_NODENAME ; NODENAME IS NULL STRING
0000'CF 00 90 0097 301 MOVB #DFT_K_PRINT,W^TST$GB_PRINT ; [NO]PRINT
0000'CF 000F4240 8F DO 009C 302 MOVL #DFT_K_SPEED,W^TST$GL_SPEED ; LINE SPEED IN BAUD
0000'CF 01 90 00A5 303 MOVB #DFT_K_STAT,W^TST$GB_STAT ; [NO]STATISTICS
00AA 304
```



```
00AA 305 :  
00AA 306 : PROCESS THE NEXT COMMAND  
00AA 307 :  
00AA 308 :  
00AA 309 NEXT_COMMAND:  
00AA 310 CLRL R11 : CLEAR ALL COMMAND PARSE FLAGS  
00AC 311 CLRL W^TST$GL_VALID : CLEAR ALL VALID QUALIFIER FLAGS  
00B0 312 BLSL2 #VLD_M_DISPLAY- : DENOTE VALID QUALIFIERS:  
00B1 313 : VLD_M_NODENAME-  
00B1 314 : VLD_M_NODISPLAY-  
00B1 315 : VLD_M_NOPRINT-  
00B1 316 : VLD_M_NOSTAT-  
00B1 317 : VLD_M_PRINT-  
00B1 318 : VLD_M_SPEED-  
00B1 319 : VLD_M_STAT,-  
00B1 320 W^TST$GL_VALID :  
00B9 321 :  
00B9 322 :  
00B9 323 : INPUT NEXT LINE OF COMMAND  
00B9 324 :  
00B9 325 :  
00B9 326 READ_LINE:  
00B9 327 $GET RAB=W^TST$CMDRAB : READ NEXT LINE  
00C4 328 CHECK_RMS : CHECK COMPLETION CODE  
03 51 E8 00C7 329 BLBS R1,20$ : WAS IT END-OF-FILE OR TIME-OUT?  
017D 31 00CA 330 10$: BRW TERMINATE : YES, EXIT  
00CD 331 :  
00CD 332 :  
00CD 333 : PARSE THE COMMAND  
00CD 334 :  
00CD 335 :  
58 0028'CF D0 00CD 336 20$: MOVL W^TST$CMDRAB+RAB$RBF,R8 : GET ADDRESS OF COMMAND STRING  
59 0022'CF 3C 00D2 337 MOVZWL W^TST$CMDRAB+RAB$RWSZ,R9 : GET LENGTH OF COMMAND STRING  
00D7 338 :  
00D7 339 : ECHO THE COMMAND  
00D7 340 :  
00D7 341 BBS #DEV$V TRM,-  
00D9 342 W^TST$CMDFAB+FAB$R_DEV,-  
00DC 343 25$ : DONT ECHO IF TERMINAL DEVICE  
00DD 344 MOVW R9,W^TST$GW_PRTLEN  
00E2 345 MOVL R8,W^TST$PRTRAB+RAB$RBF  
00E7 346 BSBW TST$PRINT_FAO  
00EA 347 MOVAL W^TST$GB_PRTBUF,-  
00EE 348 W^TST$PRTRAB+RAB$RBF  
00F1 349 25$: ADDL2 R8,R9 : CALCULATE ADDRESS OF END OF  
00F1 350 : COMMAND BUFFER + 1  
00F4 351 : IS IT REQUEST TO EXIT?  
68 45 8F 91 00F4 352 CMPB #^A\E\,(R8) :  
00F8 353 BEQL 10$ : BRANCH IF YES  
00FA 354 CALLS #0,W^TST$PARSE : PARSE THE COMMAND LINE  
00FF 355 BBCC #FLG V PARSEERROR,R11,30$ : BRANCH IF PARSE WAS SUCCESSFUL  
0103 356 $FAO_S CTRSTR=W^TST$GQ_PARSE- :  
0103 357 OUTLEN=W^TST$GW_PRTLEN- :  
0103 358 OUTBUF=W^TST$GQ_PRTBUF- :  
0103 359 P1=#TST$GT_DTS : INSERT DTS ID  
011C 360 CHECK_SS : CHECK STATUS CODE  
FEDE' 30 011F 361 BSBW TST$PRINT_FAO : PRINT MESSAGE
```

```
91 5B 86 11 0122 362 BRB NEXT_COMMAND ; START AGAIN
01 01 E4 0124 363 30$: BBSC #FLG_V_MULTILINE,R11,- ; GET ANOTHER LINE IF REQUIRED
0128 364 READ_LINE
0128 365
0128 366
0128 367 : COMMAND PARSING IS COMPLETE. R10 = COMMAND PARAMETER (TESTTYPE).
0128 368
0128 369 : START TO BUILD THE TEST REQUEST STRING WHICH WILL BE SENT TO DTR IN
0128 370 : THE USERDATA FIELD OF THE NSP CONNECT INITIATE MESSAGE. THE REQUEST
0128 371 : STRING IS ASSEMBLED AS A COUNTED ASCII STRING.
0128 372
0128 373
01 AB 5B 0000'CF DE 0128 374 MOVAL W^TST$GT_USERDATA,R11 ; GET ADDRESS OF USERDATA STRING
5A 0000'CF 81 012D 375 ADDB3 W^TST$GB_PRINT,R10,1(R11) ; UPDATE DTS/R TESTTYPE FIELD
0134 376 ; WHICH IS DERIVED FROM BOTH
0134 377 ; THE COMMAND PARAMETER AND
0134 378 ; THE /[NO]PRINT QUALIFIER
0134 379
0134 380 :
0134 381 : DISPATCH TO APPROPRIATE ROUTINE FOR EXECUTING THE COMMAND
0134 382
0134 383
45'AF 9F 0134 384 PUSHAB B^TEST_COMPLETE ; PUT RETURN ADDRESS ON STACK
0137 385 ; SO THAT ROUTINES EXECUTED BY
0137 386 ; 'CASE' CAN EXIT VIA 'RSB'
0137 387 $CASEB SELECTOR=R10,DISPL=<- ; DISPATCH TO:
0137 388 TST$CONN_DTS- ; CONNECT TEST
0137 389 TST$DATA_DTS- ; DATA TEST
0137 390 TST$DISC_DTS- ; DISCONNECT TEST
0137 391 TST$INTE_DTS- ; INTERRUPT TEST
0137 392 TST$MISC_DTS- ; MISCELLANEOUS NSP TEST
0137 393 >
0145 394
0145 395 :
0145 396 : OUTPUT END-OF-TEST MESSAGE TO THE PRINT FILE
0145 397
0145 398
0000'CF 03 B0 0145 399 TEST_COMPLETE:
0000'CF 5C D0 014A 400 MOVW #3,W^TST$GT_DTSMSG ; ASSUME THERE IS AN FAO ARG
50 01 D1 014F 401 MOVL R0,W^TST$GL_DERROR ; SETUP ERROR CODE
0000'CF 05 12 0152 402 CMPL #1,R0
01 01 B0 0154 403 BNEQU 1$
0159 404 MOVW #1,W^TST$GT_DTSMSG
0159 405 1$:
0159 406 $PUTMSG_S MSGVEC=W^TST$GT_DTSMSG-
0159 407 FACNAM=W^TST$GQ_FACDESC ; CHECK STATUS CODE
016C 408 CHECK_SS
016F 409
016F 410 :
016F 411 : CALCULATE AND PRINT STATISTICS FOR THE DATA AND INTERRUPT TESTS
016F 412
016F 413
01 0000'CF 91 016F 414 CMPB W^TST$GB_STAT,#VAL_K_STAT-YES ; ARE STATISTICS DESIRED?
03 13 0174 415 BEQLU 10$ ; BRANCH IF YES
FF31 31 0176 416 5$: BRW NEXT_COMMAND ; START AGAIN
01 5A 91 0179 417 10$: CMPB R10,#VAL_K_TEST_DATA ; WAS THIS A DATA TEST?
OC 13 017C 418 BEQLU 20$ ; BRANCH IF YES
```



```
03  5A  91  017E  419      CMPB      R10,#VAL_K_TEST_INTE      ; WAS THIS AN INTERRUPT TEST?
      F3  12  0181  420      BNEQU     5$                      ; NO, START AGAIN
0000'CF 7D  0183  421      MOVQ      W^TST$GL_XMITINTE,-        ; YES, COPY TRANSMIT AND RECEIVE
0000'CF      0187  422      W^TST$GL_XMITDATA      ; INTE COUNTERS TO DATA COUNTERS
      018A  423
      018A  424
      018A  425      ; OUTPUT TEST PARAMETERS TO THE PRINT FILE
      018A  426
      018A  427
52  0000'CF 3C  018A  428 20$: MOVZWL  W^TST$GW_SIZE,R2          ; GET MESSAGE SIZE
      0000'CF C2  018F  429      SUBL2     W^TST$GL_CLOCK,-        ; GET TIME TEST RAN
      0000'CF      0193  430      W^TST$GL_SECONDS      ; BR IF SOMETIME HAS PASSED
      05  12  0196  431      BNEQU     21$                      ; MAKE TIME SMALLEST INCR
0000'CF 01  D0  0198  432      MOVL      #1,W^TST$GL_SECONDS
      019D  433 21$: SFAO_S   CTRSTR=W^TST$GQ_STAT1-        ;
      019D  434      OUTLEN=W^TST$GW_PRTLEN-        ;
      019D  435      OUTBUF=W^TST$GQ_PRTBUF-        ;
      019D  436      P1=W^TST$GL_SECONDS-        ; DURATION OF TEST IN SECONDS
      019D  437      P2=#TST$GT_NODENAME-        ; ADDRESS OF NODENAME STRING
      019D  438      P3=W^TST$GL_SPEED-        ; LINE SPEED (BAUD)
      019D  439      P4=R2                      ; MESSAGE SIZE IN BYTES
      01C0  441      CHECK_SS   TST$PRINT_FAO      ; CHECK STATUS CODE
FE3A' 30  01C3  442      BSBW      ; PRINT MESSAGE
      01C6  443
      01C6  444      ; CALCULATE DATA TRANSFER TOTALS
      01C6  445
      01C6  446
      01C6  447
51  0000'CF C1  01C6  448      ADDL3     W^TST$GL_XMITDATA,-        ; CALCULATE TOTAL NUMBER OF
      0000'CF      01CA  449      W^TST$GL_RECVDATA,R1      ; MESSAGES TRANSFERRED
52  51  C4  01CE  450      MULL2     R1,R2          ; CALCULATE TOTAL NUMBER OF
      01D1  451      ; BYTES TRANSFERRED
53  52  03  9C  01D1  452      ROTL      #3,R2,R3      ; CALCULATE TOTAL NUMBER OF
      01D5  453      ; BITS TRANSFERRED
      01D5  454
      01D5  455      ; CALCULATE DATA RATES WITH ROUNDED RESULTS STORED IN INTEGER FORMAT
      01D5  456
      01D5  457
50  0000'CF 4E  01D5  458      CVTLF     W^TST$GL_SECONDS,R0      ; PUT TIME IN FLOATING FORMAT
      01DA  459      ; ***** MSG/SEC --> {R4,R5}
59  51  4E  01DA  460      CVTLF     R1,R9          ; PUT #MSG IN FLOATING FORMAT
59  22  44  01DD  461      MULF2     #10,R9          ; CALCULATE MESSAGES PER SECOND
59  50  46  01E0  462      DIVF2     R0,R9          ; X 10
54  59  4B  01E3  463      CVTRFL    R9,R4          ; ROUND AND STORE AS INTEGER
      55  D4  01E6  464      CLRL      R5          ; DOUBLE PRECISION DIVIDE FOLLOWS
55  54  54  0A  7B  01E8  465      EDIV     #10,R4,R4,R5      ; SPLIT RESULT INTO X.Y VALUES
      01ED  466      ; ***** BYTES/SEC --> R6
59  52  4E  01ED  467      CVTLF     R2,R9          ; PUT #BYTES IN FLOATING FORMAT
59  50  46  01F0  468      DIVF2     R0,R9          ; CALCULATE BYTES PER SECOND
56  59  4B  01F3  469      CVTRFL    R9,R6          ; ROUND AND STORE AS INTEGER
      01F6  470      ; ***** BITS/SEC --> R7
59  53  4E  01F6  471      CVTLF     R3,R9          ; PUT #BITS IN FLOATING FORMAT
59  50  46  01F9  472      DIVF2     R0,R9          ; CALCULATE BITS PER SECOND
57  59  4B  01FC  473      CVTRFL    R9,R7          ; ROUND AND STORE AS INTEGER
      01FF  474      ; ***** LINE UTILIZATION --> {R8,R9}
50  0000'CF 4E  01FF  475      CVTLF     W^TST$GL_SPEED,R0      ; PUT SPEED IN FLOATING FORMAT
```

```
59 0000457A 8F 44 0204 476 MULF2 #1000,R9 ; CALCULATE PERCENT OF LINE
      59 50 46 020B 477 DIVF2 R0,R9 ; UTILIZATION X 10
      58 59 4A 020E 478 CVTFL R9,R8 ; ROUND AND STORE AS INTEGER
      59 59 D4 0211 479 CLRL R9 ; DOUBLE PRECISION DIVIDE FOLLOWS
59 58 58 0A 7B 0213 480 EDIV #10,R8,R8,R9 ; SPLIT RESULT INTO X.Y VALUES
      0218 481
      0218 482 ;
      0218 483 ; OUTPUT TEST STATISTICS TO THE PRINT FILE
      0218 484 ;
      0218 485 ;
      0218 486 $FAO_S CTRSTR=W^TST$GQ_STAT2- ;
      0218 487 OUTLEN=W^TST$GW_PRTLEN- ;
      0218 488 OUTBUF=W^TST$GQ_PRTBUF- ;
      0218 489 P1=W^TST$GL_XMITDATA- ; TOTAL # OF MESSAGES TRANSMITTED
      0218 490 P2=W^TST$GL_RECVDATA- ; TOTAL # OF MESSAGES RECEIVED
      0218 491 P3=R2- ; TOTAL # BYTES TRANSFERRED
      0218 492 P4=R4- ; MESSAGES PER SECOND
      0218 493 P5=R5- ;
      0218 494 P6=R6- ; BYTES PER SECOND
      0218 495 P7=R7- ; LINE THROUGHPUT (BAUD)
      0218 496 P8=R8- ; PERCENT OF LINE UTILIZATION
      0218 497 P9=R9 ;
      0241 498 CHECK_SS ; CHECK STATUS CODE
FDB9' 30 0244 499 BSBW TST$PRINT_FAO ; PRINT MESSAGE
FE60 31 0247 500 BRW NEXT_COMMAND ; START AGAIN
      024A 501
      024A 502 ;
      024A 503 ; OUTPUT TERMINATION MESSAGE TO THE PRINT FILE
      024A 504 ;
      024A 505 ;
      024A 506 TERMINATE: ;
      024A 507 $FAO_S CTRSTR=W^TST$GQ_TERM- ; END OF TEST
      024A 508 OUTLEN=W^TST$GW_PRTLEN- ; FORMAT MESSAGE
      024A 509 OUTBUF=W^TST$GQ_PRTBUF- ;
      024A 510 P1=#TST$GT_DTS- ;
      024A 511 P2=#0 ; INSERT DTS ID
      0265 512 CHECK_SS ; INSERT DATE AND TIME
FD95' 30 0268 513 BSBW TST$PRINT_FAO ; CHECK STATUS CODE
      0268 514 $EXIT_S ; PRINT MESSAGE
      0274 515 .END TST$START_DTS ; EXIT TO VMS
```


TSTSDTSMAN
Symbol table

- DTS MAINLINE

B 5

16-SEP-1984 01:24:51 VAX/VMS Macro V04-00
5-SEP-1984 00:22:28 [DTS DTR.SRC]DTSMAIN.MAR;1

Page 12
(4)

\$\$TMP1	=	00000001		
\$\$TMP2	=	000000CF		
\$\$COUNT	=	00000005		
\$\$T2	=	00000005		
DEVSV_TRM	=	00000002		
DFT_K_DISPLAY	=	00000000		
DFT_K_PRINT	=	00000000		
DFT_K_SPEED	=	000F4240		
DFT_K_STAT	=	00000001		
FABSL_DEV	*****		X	02
FLG_V_MULTILINE	=	00000001		
FLG_V_PARSERERROR	=	00000000		
K_LIST_MEB	=	00000000		
LIB\$ASN_WTH_MBX	*****		X	02
NEXT_COMMAND	000000AA		R	02
RABSC_RBF	=	00000028		
RABSW_RSZ	=	00000022		
READ_CINE	000000B9		R	02
S	00000000		RG	02
SIZ...	=	00000001		
SYSSCONNECT	*****		GX	02
SYSEXIT	*****		GX	02
SYSSFAO	*****		X	02
SYSSGET	*****		GX	02
SYSSOPEN	*****		GX	02
SYSSPUTMSG	*****		GX	02
TERMINATE	0000024A		R	02
TEST_COMPLETE	00000145		R	02
TST\$CHECK_RMS	*****		X	02
TST\$CHECK_SS	*****		X	02
TST\$CMDFAB	*****		X	02
TST\$CMDRAB	*****		X	02
TST\$CONN_DTS	*****		X	02
TST\$DATA_DTS	*****		X	02
TST\$DISC_DTS	*****		X	02
TST\$GB_DISPLAY	*****		X	02
TST\$GB_PRINT	*****		X	02
TST\$GB_PRTBUF	*****		X	02
TST\$GB_STAT	*****		X	02
TST\$GL_CLOCK	*****		X	02
TST\$GL_DERROR	*****		X	02
TST\$GL_RECVDATA	*****		X	02
TST\$GL_SECONDS	*****		X	02
TST\$GL_SPEED	*****		X	02
TST\$GL_VALID	*****		X	02
TST\$GL_XMITDATA	*****		X	02
TST\$GL_XMITINTE	*****		X	02
TST\$GQ_FACDESC	*****		X	02
TST\$GQ_INIT	*****		X	02
TST\$GQ_LINKNAME	*****		X	02
TST\$GQ_PARSE	*****		X	02
TST\$GQ_PRTBUF	*****		X	02
TST\$GQ_STAT1	*****		X	02
TST\$GQ_STAT2	*****		X	02
TST\$GQ_TERM	*****		X	02
TST\$GT_DTS	*****		X	02
TST\$GT_DTSMSG	*****		X	02

TST\$GT_NODENAME	*****	X	02
TST\$GT_USERDATA	*****	X	02
TST\$GT_VERSION	*****	X	02
TST\$GW_LINKCHAN	*****	X	02
TST\$GW_MAILCHAN	*****	X	02
TST\$GW_PRTLEN	*****	X	02
TST\$GW_SIZE	*****	X	02
TST\$INTE_DTS	*****	X	02
TST\$K_MAILBUF	*****	X	02
TST\$K_MAILQUOTA	*****	X	02
TST\$MISC_DTS	*****	X	02
TST\$PARSE	*****	X	02
TST\$PRINT_FAO	*****	X	02
TST\$PRTFAB	*****	X	02
TST\$PRTTAB	*****	X	02
TST\$START_DTS	00000000	RG	02
VAL_K_BACK_NO	=	00000000	
VAL_K_DISP_NO	=	00000000	
VAL_K_FLOW_MESS	=	00000002	
VAL_K_NAK_NO	=	00000000	
VAL_K_PRINT_NO	=	00000000	
VAL_K_RETU_NO	=	00000000	
VAL_K_STAT_YES	=	00000001	
VAL_K_TEST_DATA	=	00000001	
VAL_K_TEST_INTE	=	00000003	
VAL_K_TYPE_ABRT	=	00000001	
VAL_K_TYPE_ACCE	=	00000001	
VAL_K_TYPE_NAME	=	00000000	
VAL_K_TYPE_SINK	=	00000000	
VLD_M_DISPLAY	=	00000002	
VLD_M_NODENAME	=	00000080	
VLD_M_NODISPLAY	=	00000100	
VLD_M_NOPRINT	=	00000800	
VLD_M_NOSTAT	=	00002000	
VLD_M_PRINT	=	00004000	
VLD_M_SPEED	=	00080000	
VLD_M_STAT	=	00200000	

+-----+
! Psect synopsis !
+-----+

PSECT name	Allocation	PSECT No.	Attributes
ABS	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
\$ABS\$	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
TST\$CODE	00000274 (628.)	02 (2.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE

+-----+
! Performance indicators !
+-----+

Phase	Page faults	CPU Time	Elapsed Time
Initialization	32	00:00:00.07	00:00:00.71
Command processing	107	00:00:00.62	00:00:02.40
Pass 1	256	00:00:07.96	00:00:23.05
Symbol table sort	0	00:00:00.36	00:00:00.62
Pass 2	108	00:00:02.09	00:00:06.20
Symbol table output	12	00:00:00.06	00:00:00.15
Psect synopsis output	2	00:00:00.02	00:00:00.02
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	519	00:00:11.20	00:00:33.16

The working set limit was 1200 pages.

38462 bytes (76 pages) of virtual memory were used to buffer the intermediate code.

There were 20 pages of symbol table space allocated to hold 392 non-local and 10 local symbols.

577 source lines were read in Pass 1, producing 21 object records in Pass 2.

36 pages of virtual memory were used to define 30 macros.

+-----+
! Macro library statistics !
+-----+

Macro library name	Macros defined
\$255\$DUA28:[DTS\$DTR.OBJ]DTS\$DTR.MLB;1	8
\$255\$DUA28:[SYSLIB]STARLET.MLB;2	15
TOTALS (all libraries)	23

538 GETS were required to define 23 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:DTSMAIN/OBJ=OBJ\$:DTSMAIN MSRC\$:DTPREFIX/UPDATE=(ENH\$:DTPREFIX)+MSRC\$:DTSMAIN/UPDATE=(ENH\$:DTSMAIN)

0123 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

