

UUU	UUU	EEEEEEEEEEEEEEEE	TTTTTTTTTTTTTTTT	PPPPPPPPPPPP	
UUU	UUU	EEEEEEEEEEEEEEEE	TTTTTTTTTTTTTTTT	PPPPPPPPPPPP	
UUU	UUU	EEEEEEEEEEEEEEEE	TTTTTTTTTTTTTTTT	PPPPPPPPPPPP	
UUU	UUU	EEE	TTT	PPP	PPP
UUU	UUU	EEE	TTT	PPP	PPP
UUU	UUU	EEE	TTT	PPP	PPP
UUU	UUU	EEE	TTT	PPP	PPP
UUU	UUU	EEE	TTT	PPP	PPP
UUU	UUU	EEE	TTT	PPP	PPP
UUU	UUU	EEE	TTT	PPP	PPP
UUU	UUU	EEEEEEEEEEEEEEEE	TTT	PPPPPPPPPPPP	
UUU	UUU	EEEEEEEEEEEEEEEE	TTT	PPPPPPPPPPPP	
UUU	UUU	EEEEEEEEEEEEEEEE	TTT	PPPPPPPPPPPP	
UUU	UUU	EEE	TTT	PPP	
UUU	UUU	EEE	TTT	PPP	
UUU	UUU	EEE	TTT	PPP	
UUU	UUU	EEE	TTT	PPP	
UUU	UUU	EEE	TTT	PPP	
UUU	UUU	EEE	TTT	PPP	
UUU	UUU	EEE	TTT	PPP	
UUUUUUUUUUUUUUUU	UUUUUUUUUUUUUUUU	EEEEEEEEEEEEEEEE	TTT	PPP	
UUUUUUUUUUUUUUUU	UUUUUUUUUUUUUUUU	EEEEEEEEEEEEEEEE	TTT	PPP	
UUUUUUUUUUUUUUUU	UUUUUUUUUUUUUUUU	EEEEEEEEEEEEEEEE	TTT	PPP	

```

RRRRRRRR  MM      MM      SSSSSSSS  TTTTTTTTTT  EEEEEEEEE  SSSSSSSS  TTTTTTTTTT  000000
RRRRRRRR  MM      MM      SSSSSSSS  TTTTTTTTTT  EEEEEEEEE  SSSSSSSS  TTTTTTTTTT  000000
RR      RR  MMMM  MMMM  SS          TT          EE          SS          TT          00      00
RR      RR  MMMM  MMMM  SS          TT          EE          SS          TT          00      00
RR      RR  MM   MM   MM   SS          TT          EE          SS          TT          00      0000
RR      RR  MM   MM   MM   SS          TT          EE          SS          TT          00      0000
RRRRRRRR  MM      MM      SSSSSS    TT          EE          SS          TT          00      00 00
RRRRRRRR  MM      MM      SSSSSS    TT          EE          SS          TT          00      00 00
RR  RR    MM      MM      SS          TT          EE          SS          TT          0000   00
RR  RR    MM      MM      SS          TT          EE          SS          TT          0000   00
RR  RR    MM      MM      SS          TT          EE          SS          TT          00      00
RR  RR    MM      MM      SSSSSSSS  TT          EEEEEEEEE  SSSSSSSS  TT          00      00
RR  RR    MM      MM      SSSSSSSS  TT          EEEEEEEEE  SSSSSSSS  TT          000000
RR  RR    MM      MM      SSSSSSSS  TT          EEEEEEEEE  SSSSSSSS  TT          000000

```

```

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

```

```

0000 73          $BEGIN RMSTEST,028,_,_RMSTEST,<USER ROUTINE TO CALL AND TEST RMS>,<LONG,GBL>
0000 74          .ENABL  DBG
0000 75
0000 76          :
0000 77          .NLIST  MEB
0000 78          :
0000 79          :   MACROS:
0000 80          :
0000 81
0000 82          .MACRO  BUFF  NAM,SIZE
0000 83  NAM'BUF::
0000 84          .BLKB   SIZE
0000 85          NAM'BSZ==SIZE
0000 86          .ENDM   BUFF
0000 87
0000 88          :
0000 89
0000 90          .MACRO  PROMPT STRING,NULL,?L
0000 91          STORE  <STRING>,<.BYTE 13,10>
0000 92          MOVL   $$$TMPX,CMDIRAB+RAB$L_PBF
0000 93          MOVB   $$$TMPX1,CMDIRAB+RAB$B_PSZ
0000 94          $GET   RAB=CMDIRAB
0000 95          BLBS   RO,L
0000 96          BSBW   EOFDET
0000 97  L:
0000 98          .IF    NB NULL
0000 99          TSTW   CMDIRAB+RAB$W_RSZ
0000 100         BEQL   NULL
0000 101         .ENDC
0000 102         .ENDM   PROMPT
0000 103
0000 104         :
0000 105         :
0000 106
0000 107         .MACRO  TYPE  STRING,?L
0000 108         STORE  <STRING>
0000 109         BLBC   VERBOSITY,L
0000 110         MOVL   $$$TMPX,CMDORAB+RAB$L_RBF
0000 111         MOVW   $$$TMPX1,CMDORAB+RAB$W_RSZ
0000 112         $PUT   RAB=CMDORAB,ERR=REPORT_ERROR
0000 113         BSBW   ERR
0000 114  L:
0000 115         .ENDM   TYPE
0000 116
0000 117         :
0000 118
0000 119         .MACRO  STORE STRING,PRE
0000 120         .SAVE
0000 121         .PSECT  _$_RMSNAM
0000 122         $$$TMPX=.
0000 123         PRE
0000 124         .ASCII  %STRING%           ; store any carriage
0000 125         $$$TMPX1=.-$$$TMPX       ; control info
0000 126         .RESTORE
0000 127         .ENDM   STORE

```

```

0000 129      .MACRO BEGIN TSTNAM
0000 130      STORE <TSTNAM>
0000 131      MOVL  $$$TMPX1,BEG_DESCR+4      ; addr
0000 132      MOVL  $$$TMPX1,BEG_DESCR      ; len
0000 133      BSBW  BEGPUT
0000 134      .ENDM BEGIN
0000 135      .MACRO FINISH TSTNAM
0000 136      STORE <TSTNAM>
0000 137      MOVL  $$$TMPX1,FIN_DESCR+4      ; addr
0000 138      MOVL  $$$TMPX1,FIN_DESCR      ; len
0000 139      BSBW  FINPUT
0000 140      .ENDM FINISH
0000 141      .MACRO FIELD FLDNAM
0000 142      STORE <FLDNAM>
0000 143      MOVL  $$$TMPX1,FLD_DESCR+4      ; addr
0000 144      MOVL  $$$TMPX1,FLD_DESCR      ; len
0000 145      BSBW  FLDPUT
0000 146      .ENDM FIELD
0000 147      .MACRO MBPT ?L      ; BPT, maybe?
0000 148      BLBC  VERBOSITY,L
0000 149      BPT
0000 150      L:
0000 151      .ENDM MBPT
0000 152
0000 153      .MACRO SFNM STRING
0000 154      STORE <STRING>
0000 155      MOVB  $$$TMPX1,STICKY_FAB2+FAB$B_FNS
0000 156      MOVL  $$$TMPX1,STICKY_FAB2+FAB$L_FNA
0000 157      .ENDM SFNM
0000 158
0000 159      .MACRO ALWAYS STRING
0000 160      STORE <STRING>
0000 161      MOVL  $$$TMPX1,CMDORAB+RAB$L_RBF
0000 162      MOVW  $$$TMPX1,CMDORAB+RAB$W_RSZ
0000 163      $PUT  RAB=CMDORAB,ERR=REPORT_ERROR
0000 164      BSBW  ERR
0000 165      .ENDM ALWAYS
0000 166
0000005D 0000 167      RBRACK=93
0000003B 0000 168      SEMI=59

```

```

00000000 170      .PSECT  __RMSTEST,GBL,LONG
0000 171
0000 172 ;
0000 173
0000 174 RELZERO:
0000 175 .IF DF RMSDEBUG
0000 176 CMDIFAB::
0000 177 $FAB FAC=<GET>,FNM=TST$INPUT:,MRS=132,RAT=CR
0000 178 CMDOFAB::
0000 179 $FAB FAC=PUT,FNM=TST$OUTPUT:,MRS=132,RAT=CR,RFM=VAR
0000 180 .IFF
0000 181 CMDIFAB::
0000 182 $FAB FAC=<GET>,FNM=SYSS$INPUT:,MRS=132,RAT=CR
0050 183 CMDOFAB::
0050 184 $FAB FAC=PUT,FNM=SYSS$OUTPUT:,MRS=132,RAT=CR,RFM=VAR
00A0 185 .ENDC
00A0 186 CMDIRAB::
00A0 187 $RAB UBF=CMDBUF,USZ=CMDBSZ,FAB=CMDIFAB,ROP=PMT
00E4 188 CMDORAB::
00E4 189 $RAB UBF=CMDBUF,USZ=CMDBSZ,FAB=CMDOFAB
0128 190
0128 191 ;
0128 192
0128 193 INFAB:: $FAB FAC=GET,FNA=PNSBUF,DNA=DNSBUF,NAM=NAMBLK
0178 194 INRAB:: $RAB UBF=CPYBUF,USZ=CPYBSZ,FAB=INFAB,RHB=RHBBUF,-
0178 195 MBC=4,MBF=-2
01BC 196 C TFAB::
01BC 197 $FAB FAC=PUT,FNA=PNSBUF,DNA=DNSBUF,NAM=NAMBLK
020C 198 O. TRAB::
020C 199 $RAB UBF=CPYBUF,USZ=CPYBSZ,RBF=CPYBUF,FAB=OUTFAB,ROP=EOF,-
020C 200 RHB=RHBBUF,MBC=8,MBF=1
0250 201 NAMBLK::
0250 202 $NAM ESA=CMDBUF,ESS=CMDBSZ,rsa=RESBUF,RSS=RESBSZ
0280 203 WCDNAM::
0280 204 $NAM ESA=WCDBUF,ESS=WCDBSZ,rsa=RWCBUF,RSS=RWCBSZ
0310 205
0310 206 ;
0310 207
0310 208 STICKY_FAB2:
0310 209 $FAB NAM=STICKY_NAMBLK,FOP=<OFP,CBT,SUP>,-
0310 210 ALQ=4,FAC=PUT,DNM=<TST$DISK:>
0360 211 STICKY_NAMBLK:
0360 212 $NAM RLF=RELATED_NAM,rsa=RESBUF,RSS=RESBSZ,ESA=EXPBUF,ESS=EXPBSZ
03C0 213 RELATED_NAM:
03C0 214 $NAM rsa=REL_FILE ; must set RSL at RUN-TIME
0420 215 REL_FILE:
54 41 44 2E 53 54 0420 216 .ASCII /TS.DAT/
00000006 0426 217 REL_FILE SIZE=.-REL_FILE
0426 218 .ALIGN WORD
0426 219 SAVE_FID:
0000042C 0426 220 .BLKB 6 ; to save FID
042C 221 .ALIGN LONG
042C 222 OLDFAB::
042C 223 $FAB FNM=<TS.DAT>,DNM=<TST$DISK:>,NAM=OLDNAM
047C 224 NEWFAB::
047C 225 $FAB FNM=<TST$DISK:RENAMED.TST;32>,nam=newnam
04CC 226 OLDNAM::

```

```

04CC 227          $NAM   ESA=OLD_ESBUF,ESS=NAM$C_MAXRSS,-
04CC 228          RSA=OLD_RSBUF,RSS=NAM$C_MAXRSS
052C 229 NEWNAM::
052C 230          $NAM   ESA=NEW_ESBUF,ESS=NAM$C_MAXRSS,-
052C 231          RSA=NEW_RSBUF,RSS=NAM$C_MAXRSS
058C 232 OLD_ESBUF:
0000068B 058C 233          .BLKB  NAM$C_MAXRSS
068B 234 OLD_RSBUF:
0000078A 068B 235          .BLKB  NAM$C_MAXRSS
078A 236 NEW_ESBUF:
00000889 078A 237          .BLKB  NAM$C_MAXRSS
0889 238 NEW_RSBUF:
00000988 0889 239          .BLKB  NAM$C_MAXRSS
0988 240          .ALIGN  LONG
0988 241
0988 242 ;
0988 243
0988 244          BUFF   WCD,80
09D8 245          BUFF   RWC,80
0A28 246          BUFF   EXP,64
0A68 247          BUFF   RES,64
0AA8 248 RHBBUF::
00000AB8 0AA8 249          .BLKB  16
00000000 0AB8 250 DIRLEN: .LONG  0
00000AE4 0ABC 251 DIRBUF: .BLKB  40
0AE4 252 DIRDESC:
00000028 0AE4 253          .LONG  40
00000ABC' 0AE8 254          .LONG  DIRBUF
3A 54 53 45 54 53 4D 52 0AEC 255 ECHOBFF: .ASCII /RMSTEST:/ ; must immediately precede
0AF4 256          BUFF   CMD,132 ; CMDBUF
0B78 257 PNSBUF::
00000BC8 0B78 258          .BLKB  80
0BC8 259          BUFF   DNS,80
0C18 260          BUFF   CPY,512
00000E20 0E18 261          .BLKW  4
0000 0E20 262 DNSSIZ: .WORD  0 ; size of DEFAULT NAME STRING
00000000 0E22 263 NAM:: .LONG  0 ; addr of NAMBLK for TST$SHOWNAM
00 0E26 264 CPYFLG: .BYTE  0 ; zero for COPY, 1 for APPEND
00 0E27 265 WCFLG: .BYTE  0 ; 1 if wild card spec
00 0E28 266 ORGFLG: .BYTE  0 ; 1 if choice of org
FF 0E29 267 VSW: .BYTE -1 ; verify switch
00 0E2A 268 CMD: .BYTE  0 ; character of current command
0E2B 269          .ALIGN  LONG
00000C18'00000200 0E2C 270 CPYBUFDSC:
0E2C 271          .LONG  CPYBSZ,CPYBUF

```

```

OE34 273
OE34 274 :
OE34 275 :   FAO data
OE34 276 :
OE34 277
00000E44'0000003F' OE34 278 FAOSTR: .LONG   FSTR1LEN,FSTR1
OE3C 279 FAOBUF::
00000AF4'00000084 OE3C 280
4F 49 54 41 52 45 50 4F 20 53 4D 52 OE44 281 FSTR1:  .ASCII  'RMS OPERATION ERROR!/  RMSTEST REL. PC = !XL'
20 20 20 2F 21 52 4F 52 52 45 20 4E OE50
2E 4C 45 52 20 54 53 45 54 53 4D 52 OE5C
3D 20 2E 43 50 20 2E 53 42 41 20 20 OE68
2F 21 4C 58 21 20 OE71 282 .ASCII  ' ABS. PC. = !XL!/'
OE7D
49 41 46 20 4E 57 4F 44 20 4E 55 52 OE83 283 TR1LEN=-FSTR1
49 54 4E 4F 43 20 2D 20 45 52 55 4C OE83 284 DMSG:  .ASCII  'RUN DOWN FAILURE - CONTINUING'
OE8F
47 4E 49 55 4E OE9B
0000001D OEAO 285 RDMSGLEN=-RDMSG
OEAO 286
OEAO 287 :
OEAO 288 :   Descriptors for ASCII inserts into messages
OEAO 289 :
OEAO 290
OFAO 291 .ALIGN  LONG
OEAO 292 BEG_DESCR::
00000000 00000000 OEAO 293 .LONG  0,0
OEAB 294 FIN_DESCR::
00000000 00000000 OEAB 295 .LONG  0,0
OEBO 296 FLD_DESCR::
00000000 00000000 OEBO 297 .LONG  0,0
OEBC 298 KRF_DESC:
00000000 00000000 OEBC 299 .LONG  0,0 ; KEY OF REFERENCE desc
OECC 300 INTEGER:
00000000 OECC 301 .LONG  0 ; return from int. input conv
OECD 302 ILLSERDEC:
00000ED7'00000007' OECD 303 .LONG  ILLSERL,ILLSER
OECC 304 EOF_DSC:
00000ED4'00000003' OECC 305 .LONG  EXPEOF_LEN,EXPEOF
46 4F 45 OECD 306 EXPEOF:  .ASCII  /EOF/
00000003 OED7 307 EXPEOF_LEN=-EXPEOF
OED7 308
OED7 309 ;
OED7 310
6C 61 67 65 6C 6C 69 OED7 311 ILLSER:  .ASCII  /illegal/
00000007 OEDE 312 ILLSERL=-ILLSER

```

```

OEDE 314
OEDE 315 :
OEDE 316 :   Setups
OEDE 317 :
OEDE 318 :
0000 OEDE 319 START:: .WORD 0
      OEEO 320 $OPEN FAB=CMDIFAB
      FF 50 E9 OEEO 321 BLBC RO,, ; trap on error
      OEEO 322 $CONNECT RAB=CMDIRAB
      FF 50 E9 OEEO 323 BLBC RO,, ; trap on error
      OEEO 324 $OPEN FAB=CMDOFAB
      FF 50 E9 OEEO 325 BLBC RO,, ; trap on error
      OEEO 326 $CONNECT RAB=CMDORAB
      FF 50 E9 OEEO 327 BLBC RO,, ; trap on error
      FF03 CF F334 CF DE OEEO 328 MOVAL NAMBLK,NAM ; in case of error
      FODE' 30 OEEO 329 BSBW TRAN ; set up verbosity flag
      OEEO 330
      OEEO 331 :
      OEEO 332 :   Get a command and execute it
      OEEO 333 :
      OEEO 334 :
      OEEO 335 CMD_LOOP:
      F1BD CF 01 CA OEEO 336 CLRFB ORGFLG
      OEEO 337 BICL2 #RAB$M_ASY,CMDORAB+RAB$L_ROP ; insure cmd output sync.
      OEEO 338 PROMPT RMSTEST: ,NULL=BR_COPY ; get command
      OEEO 339 ECHOIT: BSBW ECHO ; echo cmd line if non-interactive
      50 FB9D CF 90 OEEO 340 MOVFB CMDBUF,RO ; pick up input character
      21 50 91 OEEO 341 CMPB RO,#^A/!/ ; comment?
      01 F161 CF B1 OEEO 342 BEQL CMD_LOOP ; branch if yes
      42 12 OEEO 343 CMPW CMDIRAB+RAB$W_RSZ,#1 ; single byte input?
      FEC1 CF 50 90 OEEO 344 BNEQ QMARK
      F253 CF D4 OEEO 345 MOVFB RO,CMD ; store cmd for possible action
      OEEO 346 CLRL FAB$L_FOP+OUTFAB ; turn off any stale options
      OEEO 347
      OEEO 348 :
      OEEO 349 :   Decide where to go
      OEEO 350 :
      OEEO 351 :
      OEEO 352 CASE TYPE=B, SRC=RO, LIMIT=#^A/A/, -
      OEEO 353 DISPLIST=<TST$APPEND, TST$BLOCK, TST$COPY, TST$DEBUG, QMARK, TST$F, -
      OEEO 354 TST$GENERAL, HELP, TST$SHR, QMARK, TST$STICKY, TST$RELATIVE, -
      OEEO 355 TST$MTAPE, TST$NET, TST$ORG, TST$PRINT, QMARK, TST$RUNDWN, TST$SETDFLT, -
      OEEO 356 QMARK, TST$SETDDIR, SETVSW, QMARK, TST$INDEX, QMARK, QMARK>
      OEEO 357
      OEEO 358 QMARK: ALWAYS <?> ; type always
      OEEO 359
      OEEO 360 CMD_LOOP BR:
      FF57 31 OEEO 361 BRW CMD_LOOP
      OEEO 362
      OEEO 363 BR_COPY:
      FB23 CF 43 BF 90 OEEO 364 MOVFB #^A/C/, CMDBUF ; default to 'C' command
      FOED CF B6 OEEO 365 INCW CMDIRAB+RAB$W_RSZ ; say 1 char input
      FF78 31 OEEO 366 BRW ECHOIT ; go process
      OEEO 367
      OEEO 368 :
      OEEO 369
      OEEO 370 SETVSW:

```


RMSTEST
028

USER ROUTINE TO CALL AND TEST RMS ; M 2

16-SEP-1984 01:44:23
5-SEP-1984 04:21:33

VAX/VMS Macro V04-00
[UETP.SRC]RMSTEST0.MAR;1

Page 7
(7)

RM:
021

C7 00000000'EF	E9	OFD8	371
FE43 CF FE46 CF	8E	OFDF	372
EO	11	OFE6	373

BLBC	VERBOSITY,QMARK
MNEGB	VSW,VSW
BRB	CMD_LOOP_BR

; toggle the verify switch

```

375 OFE8
376 OFE8
377 OFE8 : COPY routine - copy input file to output file
378 OFE8 :
379 OFE8
380 OFE8 TST$COPY:
381 OFE8 CLR B W C F L G
382 OFE8 CLR B C P Y F L G
383 OFE8 MOVA L N A M B L K , N A M
384 OFE8 COPY_PROMPT:
385 OFE8 MOVB #^A/C/,CMD ; no matter how we got here
386 OFE8 ; (A,O,C,NULL)
387 OFE8 ; signal this is a COPY
388 OFE8 PROMPT <INPUT FILE:>,NULL=CMD LOOP BR
389 OFE8 MOVC3 C M D I R A B + R A B $ W _ R S Z , C M D B U F , P N S B U F ; save PRIMARY NAME STRING
390 OFE8 MOVW C M D I R A B + R A B $ W _ R S Z , - ; and its length
391 OFE8 I N F A B + F A B $ B _ F N S
392 OFE8 MOVB D N S S I Z , I N F A B + F A B $ B _ D N S ; set DEFAULT STRING len
393 OFE8 MOVB #FAB$M_CR,INFAB+FAB$B_RAT ; get implicit LF if TRM
394 OFE8 $PARSE FAB=INFAB,ERR=REPORT_ERROR ; parse, checking for WC
395 OFE8 BSBW ERR
396 OFE8 BBS #NAM$V_WILDCARD,NAM$L_FNB+NAMBLK,WC
397 OFE8 BRW NOWC
398 OFE8 WC:
399 OFE8 MOVB #1,WCFLG ; it is a wildcard spec
400 OFE8 MOVZBL NAMBLK+NAM$B_ESL,RO
401 OFE8 MOVC3 RO,CMDBUF,WCDBUF ; save parsed string
402 OFE8 ; len must be a word, at least
403 OFE8 MOVC3 #NAM$C_BLN,NAMBLK,WCNAM ; copy the whole NAMBLK
404 OFE8 $NAM_STORE NAM=WCNAM,-
405 OFE8 ESA=WCDBUF,-
406 OFE8 ESS=WCDBSZ,-
407 OFE8 RSA=RWCBUF,-
408 OFE8 ESS=RWCBUF
409 OFE8 MOVL RO,INFAB+FAB$L_NAM ; use WC NAMBLK for now on
410 OFE8 MOVL RO,NAM ; set up for TST$SHOWNAM
411 OFE8 $SEARCH INFAB ; find 1st one
412 OFE8 CML RO,#RMS$_FNF ; are there any?
413 OFE8 BNEQ GOT_ONE ; guess so
414 OFE8 ALWAYS <NO SUCH FILES>
415 OFE8 BRW CLEAN ; clean up and start over
416 OFE8 GOT_ONE:
417 OFE8 BLBS RO,10$
418 OFE8 MOVAL INFAB,R10
419 OFE8 BSBW REPORT_ERR
420 OFE8 BSBW ERR
421 OFE8 10$: BISL #FAB$M_NAM,FAB$L_FOP+INFAB ; now can open by FID
422 OFE8 NOWC: $O^EN FAB=INFAB ; branch on success
423 OFE8 BLBS RO,5$
424 OFE8 MOVAL INFAB,R10
425 OFE8 BSBW REPORT_ERR
426 OFE8 BSBW ERR
427 OFE8 5$: BSBW TST$SHOWNAM
428 OFE8 CLRL INTEGER ; init return
429 OFE8 BLBC ORGFLG,CON ; skip if not org test
430 OFE8 BITB #FAB$C_IDX,INFAB+FAB$B_ORG ; is input file index file?
431 OFE8 BEQL CON ; eql no

```

```

1115 432 PROMPT <INPUT FILE KEY OF REFERENCE:>,NULL=CON
FD77 CF EF84 CF B0 113A 433 MOVW CMDIRAB+RAB$W_RSZ,KRF_DESC ; set up descriptor
FD74 CF F9AF CF 9E 1141 434 MOVAB CMDBUF,KRF_DESC+4
          FD74 CF 9F 1148 435 PUSHAB INTEGER
          FD68 CF 9F 114C 436 PUSHAB KRF_DESC
00000000'GF 02 FB 1150 437 CALLS #2,G^FOR$CNV IN I
F04F CF FD65 CF 90 1157 438 MOVB INTEGER,INRAB+RAB$B_KRF
115E 439 CON:
115E 440 $CONNECT RAB=INRAB,ERR=REPORT_ERROR
          0471 30 116F 441 BSBW ERR
          1172 442
          1172 443 ;
          1172 444
          1172 445 PROMPT <OUTPUT FILE:>,NULL=10$
          03 11 1197 446 BRB 20$
          0225 31 1199 447 10$: BRW CLSINF
F9D2 CF F951 CF EF22 CF 28 119C 448 20$: MOVVC3 CMDIRAB+RAB$W_RSZ,CMDBUF,PNSBUF
F043 CF EF18 CF B0 11A6 449 MOVW CMDIRAB+RAB$W_RSZ,OUTFAB+FAB$B_FNS
F03D CF FC6F CF 90 11AD 450 MOVB DNSSIZ,OUTFAB+FAB$B_DNS
F01F CF EF8E CF 90 11B4 451 MOVB INFAB+FAB$B_RAT,OUTFAB+FAB$B_RAT ; same record attributes
F019 CF EF88 CF 90 11BB 452 MOVB INFAB+FAB$B_RFM,OUTFAB+FAB$B_RFM
          11C2 453
          11C2 454 ;
          11C2 455 ; Create specific setups
          11C2 456 ;
          11C2 457
          37 FC62 CF E9 11C2 458 BLBC ORGFLG,SAME_ORG ; if plain copy, then copy org
          11C7 459 PROMPT <OUTPUT FILE ORG:>,- ; else ask for it,
          11C7 460 NULL=SEQ_ORG ; default is SEQ
          50 F904 CF 90 11EC 461 MOVB CMDBUF,RO
          52 8F 50 91 11F1 462 CMPB RO,#'A/R/'
          10 12 11F5 463 BNEQ SEQ_ORG ; if not 'R', then SEQ
          EFDD CF 10 90 11F7 464 MOVB #FAB$C_REL,OUTFAB+FAB$B_ORG ; make it RELATIVE
          OE 11 11FC 465 BRB GO
          EF43 CF 90 11FE 466 SAME_ORG: MOVB INFAB+FAB$B_ORG,- ; copy org
          EFD4 CF 1202 468 OUTFAB+FAB$B_ORG
          05 11 1205 469 BRB GO
          EFCD CF 00 90 1207 470 SEQ_ORG: MOVB #FAB$C_SEQ,OUTFAB+FAB$B_ORG ; make it SEQ
          50 EF17 CF 01000000 8F CB 120C 472 GO:
          EFA5 CF 50 CB 1216 473 BICL3 #FAB$M_NAM,INFAB+FAB$B_FOP,RO
          EFDD CF EF3F CF B0 121B 474 BISL2 RO,OUTFAB+FAB$B_FOP ; copy ctg bit
          EFCB CF EF3A CF D0 1222 475 MOVW INFAB+FAB$W_MRS,OUTFAB+FAB$W_MRS
          EFCB CF EF3A CF 90 1229 476 MOVL INFAB+FAB$B_MRN,OUTFAB+FAB$B_MRN
          EF95 CF EF04 CF D0 1230 477 MOVB INFAB+FAB$B_FSZ,OUTFAB+FAB$B_FSZ
          EF9D CF B4 1237 478 MOVL INFAB+FAB$B_ALQ,OUTFAB+FAB$B_ALQ
          2B FBE7 CF E8 1238 479 CLRW OUTFAB+FAB$B_BLS
          1240 480 BLBS CPYFLG,APPEND
          1251 481 BEGIN <COPY>
          037E 30 1262 482 $CREATE FAB=OUTFAB,ERR=REPORT_ERROR
          EFA7 CF D4 1265 483 BSBW ERR
          74 11 1269 484 CLRL RAB$B_ROP+OUTRAB
          1268 485 BRB CPYNAM
          1268 486
          1268 487 ;
          1268 488

```

```

EF3B CF 02000000 8F C8 126B 489 APPEND: BEGIN <APPEND>
127C 490 BLSL2 #FABSM_CIF,FABSL_FOP+OUTFAB
1285 491 $CREATE FAB=OUTFAB,ERR=REPORT_ERROR
034A 30 1296 492 BSBW ERR
EF6E CF 00000100 8F D0 1299 493 MOVL #RABSM_EOF,RABSL_R0P+OUTRAB ; connect to EOF
EF15 CF 02000000 8F CA 12A2 494 BICL2 #FABSM_CIF,FABSL_FOP+OUTFAB
00000000'8F EF15 CF D1 12AB 495 CMPL FABSL_STS+OUTFAB,#RMS$_CREATED ; was file created?
29 12 12B4 496 BNEQ CPYNAM ; branch if not
12B6 497 TYPE <OUTPUT FILE CREATED!>
FB3C CF EF6D CF DE 12DF 498 CPYNAM: MOVAL NAMBLK,NAM ; for TST$SIJWNAM
0A9A 30 12E6 499 BSBW TST$SHOWNAM
02E6 30 12E9 500 $CONNECT RAB=OUTRAB,ERR=REPORT_ERROR
12FA 501 BSBW ERR
12FD 502
12FD 503 ;
12FD 504
12FD 505 COPY_LOOP:
12FD 506 $GET INRAB
00000000'8F 50 D1 1308 507 CMPL R0,#RMS$_EOF
2B 13 130F 508 BEQL COPY_DONE
0B 50 E8 1311 509 BLBS R0,10$
SA EE60 CF DE 1314 510 MOVAL INRAB,R10
0262 30 1319 511 BSBW REPORT_ERR
02C4 30 131C 512 BSBW ERR
EF08 CF EE77 CF B0 131F 513 10$: MOVW INRAB+RAB$_RSZ,OUTRAB+RAB$_RSZ
1326 514 $PUT RAB=OUTRAB,ERR=REPORT_ERROR
02A9 30 1337 515 BSBW ERR
C1 11 133A 516 BRB COPY_LOOP
133C 517
133C 518 ;
133C 519
133C 520 COPY_DONE:
133C 521 TYPE <END OF INPUT FILE>
54 FAE CF E9 1365 522 BLBC WCFLG,DONE ; if no WC, then really done
00C1 30 136A 523 BSBW CLS_IN_FL
FAAE CF EF3F CF DE 136D 524 MOVAL WCDNAM,NAM ; use WC NAMBLK for duration
1374 525 $SEARCH INFAB ; look for next file
00000000'8F 50 D1 137F 526 CMPL R0,#RMS$_NMF
42 13 1386 527 BEQL NO_MORE_FILES
0B 50 E8 1388 528 BLBS R0,10$
SA ED99 CF DE 138B 529 MOVAL INFAB,R10
01EB 30 1390 530 BSBW REPORT_ERR
024D 30 1393 531 BSBW ERR
09D9 30 1396 532 10$: $OPEN FAB=INFAB,ERR=REPORT_ERROR ; by FID, as set before
FF3F 31 13A7 533 BSBW TST$SHOWNAM
0096 30 13AA 534 $CONNECT RAB=INRAB,ERR=REPORT_ERROR
006A 30 13BB 535 BRW COPY_LOOP
00B9 30 13BE 536 DONE: BSBW CLS_OUT_FL
FB58 31 13C1 537 CLSINF: BSBW CLS_IN_FL
13C4 538 BSBW REPORT_END
13C7 539 BRW CMD_LOOP
13CA 540
13CA 541 NO_MORE_FILES:
0061 30 13F3 542 TYPE <NO MORE FILES>
0087 30 13F6 543 BSBW CLS_OUT_FL ; CLOSE output file
ED2A CF 01000000 8F CA 13F9 544 BSBW REPORT_END
545 CLEAN: BICL #FABSM_NAM,FABSL_FOP+INFAB ; no more open by FID

```



```

157E 644 ; REPORT_ERR - Entry point to enter report in-line, as opposed to by AST
157E 645 ; save R0, then push STV and R0 to set up for CALL to REPORT
157E 646 ;
157E 647 ;
50 DD 157E 648 PUSHL R0 ; error code in R0
OC AA DD 1580 649 ; not in structure
50 DD 1580 650 PUSHL FAB$STV(R10)
50 DD 1583 651 PUSHL R0
14 11 1585 652 $WAIT CMDORAB
1590 653 BRB R
1592 654 REPORT::
1592 655 ;
1592 656 ;
1592 657 ; REPORT - Subroutine to print out error
1592 658 ; R10 is the structure at fault
1592 659 ;
1592 660 ;
08 AA DD 1592 661 ASSUME FAB$STV EQ RAB$STV ; could be either structure
PUSHL FAB$STV(R10) ; save status code
$WAIT CMDORAB
OC AA DD 15A0 664 PUSHL FAB$STV(R10) ; set up for LIB$PUT_MESG
08 AA DD 15A3 665 PUSHL FAB$STV(R10)
00 DD 15A6 666 R: PUSHL #0 ; no. of args for FAO
03E90003'8F DD 15A8 667 PUSHL #SHR$ RMSERROR+<RMSTEST@16>+3 ; "RMS ERROR" message
00000000'GF 04 FB 15AE 668 CALLS #4,G^CIB$SIGNAL
15B5 669 ; if FNF is the error,
15B5 670 ; print filename
00000000'8F 6E D1 15B5 671 CML (SP),#RMS$_FNF ; error is on the top of the STACK
22 12 15BC 672 BNEQ 10$
SB 28 AA D0 15BE 673 MOVL FAB$NAM(R10),R11 ; R10 is the guilty FAB,
15C2 674 ; get NAMBLK addr
15C2 675 BEQL 10$ ; if no NAMBLK, exit
08 AB 9B 15C4 676 MOVZBW NAMS$_ESL(R11),- ; length of FILENAM
EB3C CF 15C7 677 RABS$_RSZ+CMDORAB
14 13 15CA 678 BEQL 10$ ; if there is no name, exit
OC AB D0 15CC 679 MOVL NAMS$_ESA(R11),- ; addr of FILENAM
EB3A CF 15CF 680 RABS$_RBF+CMDORAB
FF 50 E9 15D2 681 $PUT CMDORAB ; print it
01 BA 15E0 682 BLBC R0, ; trap if error
05 OS 15E2 683 10$: POPR #^M<R0> ; restore R0
15E2 684 RSB
15E3 685 ;
15E3 686 ;
15E3 687 ; This is a routine to ask the interactive user if it is desired to continue.
15E3 688 ; The ABSOLUTE and RELATIVE PC'S are displayed and then 'CONTINUE' is asked.
15E3 689 ;
15E3 690 ;
01 50 E9 15E3 691 ERR:: BLBC R0,ERR1
05 OS 15E6 692 RSB ; return if no error
15E7 693 ; (secondary entry point in
15E7 694 ERR1: ; case R0 might have been OK)
00000000'8F 50 D1 15E7 695 CML R0,#SS$_ILLSER ; if illegal service, no
1D 12 15EE 696 BNEQ 10$ ; msg output yet
52 6E 14 C3 15F0 697 SUBL3 #20,(SP),R2 ; so output one
50 DD 15F4 698 PUSHL R0 ; better save R0
52 DD 15F6 700 PUSHL R2 ; PC of error

```

```

      F8CB CF   DF 15F8 701      PUSHAL ILLSERDEC          ; descriptor
      02 DD 15FC 702      PUSHL  #2                ; number of args
03E90002'8F DD 15FE 703      PUSHL  #SHR$ SYSERROR+<RMSTEST*65536>+2 ; mesg ID
00000000'GF 04 FB 1604 704      CALLS  #4,G^CIB$SIGNAL
      01 BA 160B 705      POPR   #^M<R0>          ; restore R0 after mesg
      160D 706
7E 7E 6E 14 C3 160D 707      10$:  SUBL3  #20,(SP),-(SP)
6E 6E 00000000'8F C3 1611 708      SUBL3  #RELZERO,(SP),-(SP)
      F81F CF   7F 1619 709      PUSHAQ FAOBUF
      EAES CF   3F 161D 710      PUSHAW CMDORAB+RAB$W_RSZ
      F80F CF   7F 1621 711      PUSHAQ FAOSTR
00000000'GF 05 FB 1625 712      CALLS  #5,G^SYSS$FAO
      FF 50 E9 162C 713      BLBC  R0, .
EAD6 CF F4C1 CF 9E 162F 714      MOVAB  CMDBUF,CMDORAB+RAB$L_RBF
      FF 50 E9 1636 715      $PUT  CMDORAB
      2E 00000000'EF E9 1641 716      BLBC  R0, .
      59 8F F480 CF 91 1644 717      BLBC  VERBOSITY,POPPC
      01 12 1648 718      PROMPT <CONTINUE?>,NULL=POPPC
      05 1670 719      CMPB  CMDBUF,#^A/Y/
      43 8F F7AB CF 91 1676 720      BNEQ  POPPC
      01 05 1678 721      RSB
      2D 12 1679 722      POPR  #^M<R0>          ; continue
EABA CF 01000000 8F CA 167B 723      POPPC: CMPB  CMD,#^A/C/    ; remove RETURN PC
      EAA7 CF EBAA CF DE 1681 724      BNEQ  CLBR1           ; is it a copy command?
      F772 CF EBA3 CF DE 1683 725      $CLOSE FAB=INFAB     ; if so, clean up, ignoring errors
      F86F 31 168E 726      >CLOSE FAB=OUTFAB
      EAA7 CF EBAA CF DE 1699 727      BICL  #FAB$M_NAM,FAB$L_FOP+INFAB
      F772 CF EBA3 CF DE 16A2 728      MOVAL NAMBLK,INFAB+FAB$L_NAM
      F86F 31 16A9 729      MOVAL NAMBLK,NAM
      16B0 730 CLBR1: BRW  CMD_LOOP          ; make it completely restartable

```

0D
49
0A
20
20
4C
20
45
20
43
0A
4C
45
45
20
49
46
4F
4C
4F
4D
4E
46
54
46
20
49


```

16B3 732
16B3 733 :
16B3 734 : 'U' command - Set default user file directory
16B3 735 :
16B3 736 :
16B3 737 TST$SETDDIR:
16B3 738 PROMPT <DEFAULT UFD:>,NULL=CLBR1
F496 CF F415 CF E9E6 CF 28 16D8 739 MOV C3 CMDIRAB+RAB$W_RSZ,CMDBUF,PNSBUF
52 E9DC CF 3C 16E2 740 MOVZWL CMDIRAB+RAB$W_RSZ,R2 ; copy NAME STRING
EA70 CF 52 B0 16E7 741 MOVW R2,INFAB+FAB$B_FNS ; and length
EA6D CF B4 16EC 742 CLRW INFAB+FAB$B_DNS
F400 CF DF 16F0 743 PUSHAL CMDBUF ; build descriptor for new
52 DD 16F4 744 PUSHL R2 ; default dir (addr & length)
F3EA CF DF 16F6 745 PUSHAL DIRDESC ; push args for SETDDIR
16FA 746 ; (output buff)
F3BA CF DF 16FA 747 PUSHAL DIRLEN ; output length
08 AE DF 16FE 748 PUSHAL 8(SP) ; addr of descriptor
00000000'EF 03 FB 1701 749 .IF NDF RMSDEBUG
1701 750 CALLS #3,SYS$SETDDIR ; set default directory name
1708 751 .IFF
1708 752 CALLS #3,RMD$SETDDIR
1708 753 .ENDC
06 BA 1708 754 POPR #^M<R1,R2> ; clean up STACK
08 50 E8 170A 755 BLBS R0,CLBR2
SA EA17 CF DE 170D 756 MOVAL INFAB,R10
FE69 30 1712 757 BSBW REPORT_ERR
FECB 30 1715 758 BSBW ERK
F807 31 1718 759 CLBR2: BRW CMD_LOOP ; get next command
171B 760
171B 761 :
171B 762
171B 763 BADUFD: ALWAYS <BAD DIRECTORY SPEC!>
DB 11 173B 764 BRB CLBR2
173D 765 BADPARSE:
173D 766 FIELD <PARSED DIRECTORY SPEC>
174E 767 MBPT
CO 11 1756 768 BRB CLBR2

```


RMSTEST
028

USER ROUTINE TO CALL AND TEST RMS^J 3

16-SEP-1984 01:44:23
5-SEP-1984 04:21:33

VAX/VMS Macro V04-00
[UETP.SRC]RMSTESTO.MAR;1

Page 17
(14)

FF 50 E9 185F 827
 1862 828
 05 1882 829

BLBC RO
ALWAYS <COMMAND FAB REOPENED.>
RSB

RM
Sy
FA
FA
FA
FA
FI
FI
FL
FL
FO
FS
FS
GO
GO
HE
HL
HL
IL
IL
IL
IN
IN
IN
IN
KR
LI
MO
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NA
NE
NE
NE
NE
NE
NO
NO
OK
OK
OL
OL
OL
OL
OR
OU


```

F116 CF 0040 8F 5D 8F 3A 1949 888
          18 13 1949 889 MORE: LOCC #RBRACK,#RESBSZ,RESBUF ; try to find right bracket
          50 EAC8 CF DE 1952 890 BEQL 10$ ; bad string
          51 D6 1954 891 MOVAL REL_FILE,RO
          52 06 D0 1959 892 INCL R1 ; pass bracket by
          81 80 91 195B 893 MOVL #REL_FILE_SIZE,R2
          09 12 195E 894 5$: CMPB (R0)+,(R1)+ ; comp NAM and TYP
          F8 52 F5 1961 895 BNEQ 10$
          3B 61 91 1963 896 SOBGTR R2,5$
          01 12 1966 897 CMPB (R1),#SEMI ; end of RES too?
          05 1968 898 BNEQ 10$ ; worked
          196C 900 10$: FIELD <RESULTANT STRING>
          197D 901 MBPT
          F59A 31 1985 902 BRW CMD_LOOP
          1988 903
          1988 904 :
          1988 905
          1988 906 NEXT: TYPE <STICKY CREATE WORKED>
          19AF 907
          19AF 908 :
          19AF 909 : CREATED it alright - Now enter it checking for lower version
          19AF 910 : and SEARCH for it
          19AF 911 :
          19AF 912
          E9B7 CF 02 82 19AF 913 SUBB2 #2,STICKY_NAMBLK+NAM$B_ESL ; get rid of VERSION in EXP STR
          19B4 914 ; just for good measure zero
          19B4 915 ; FNA
          E984 CF D4 19B4 916 CLRL STICKY_FAB2+FAB$B_FNA ; no PRIMARY NAM STRING
          E988 CF 94 19B8 917 CLRFB STICKY_FAB2+FAB$B_FNS ; therefore clear length
          19BC 918 $ENTER FAB=STICKY_FAB2,- ; enter takes EXP STR as input
          19BC 919 ERR=REPORT_ERROR
          FC15 30 19CB 920 BSBW ERR
          13 E9C1 CF OE E0 19CE 921 BBS #NAM$V_LOWER,STICKY_NAMBLK+NAM$B_FNB,SOK
          49 11 19D4 922 FIELD <LOWER_VERSION BIT IN FNB>
          19E5 923 BRB REM_SRCH
          50 E975 CF DE 19E7 924 SOK: MOVAL STICKY_NAMBLK,RO
          OC AO F078 CF DE 19EC 925 MOVAL RESBUF,NAM$B_ESA(RO)
          OB AO 03 AO 90 19F2 926 MOVB NAM$B_RSL(RO),NAM$B_ESL(RO)
          19F7 927 $SEARCH FAB=STICKY_FAB2,ERR=REPORT_ERROR
          FBDA 30 1A06 928 BSBW ERR
          1A09 929 TYPE <ENTER WITH LOWER VERSION AND SEARCH OK >
          1A30 930
          1A30 931 :
          1A30 932 :
          1A30 933 : Now remove that newest entry and SEARCH for it, hoping the SEARCH fails
          1A30 934 :
          1A30 935 :
          1A30 936 REM_SRCH:
          50 E92C CF DE 1A30 937 MOVAL STICKY_NAMBLK,RO
          E9EB CF 24 AO D0 1A35 938 MOVL NAM$W_FID(RO),SAVE_FID
          E9E9 CF 28 AO B0 1A3B 939 MOVW NAM$W_FID+4(RO),SAVE_FID+4 ; save FID for later
          1A41 940 $NAM_STORE NAM=STICKY_NAMBLK,- ; zero FID and CONTEXT
          1A41 941 FID=#0,WCC=#0
          1A4F 942 $REMOVE FAB=STICKY_FAB2,- ; remove the DIR. entry
          1A4F 943 ERR=REPORT_ERROR
          FB82 30 1A5E 944 BSBW ERR

```

```

50  E8FB CF  DE 1A61 945      MOVAL STICKY_NAMBLK,RO
OC  A0  EFFE CF  DE 1A66 946      MOVAL RESBUF,NAM$L_ESA(RO)      ; move RESULTANT STR ADDR
                                DE 1A6C 947      ; to ESA
OB  A0  03 A0  90 1A6C 948      MOVB  NAM$B_RSL(RO),NAM$B_ESL(RO)
E8E9 CF  EFAB CF  DE 1A71 949      $SEARCH STICKY_FAB2      ; look for it
00000000'8F 50  D1 1A7C 950      MOVAL EXPBUF,NAM$L_ESA+STICKY_NAMBLK
                                D1 1A83 951      CMPL  RO,#RMS$_NMF      ; better not be there
                                40 13 1A8A 952      BEQL  OK
00000000'8F 50  D1 1A8C 953      CMPL  RO,#RMS$_NORMAL      ; but it was
                                29 12 1A93 954      BNEQ  CHK
                                35 11 1A95 955      TYPE  <SYS$REMOVE DIDN'T DO ITS JOB>
                                OB 50  E8 1ABC 956      BRB   OUT
SA  E84B CF  DE 1ABE 957      BLBS  RO,10$      CHK:
                                FAB5 30 1AC1 958      MOVAL STICKY_FAB2,R10
                                FB17 30 1AC6 959      BSBW  REPORT_ERR
                                30 1AC9 960      BSBW  ERR
                                1ACC 961      10$:
                                1ACC 962      OK:
                                1ACC 963      TYPE  <REMOVE AND SEARCH OK>
                                1AF3 964      OUT:
                                1AF3 965      $NAM_STORE  NAM=STICKY_NAMBLK,-      ; restore FID
                                1AF3 966      FID=SAVE_FID
                                1B04 967
                                1B04 968      ;
                                1B04 969      ; Now test $RENAME function by renaming TS.DAT to RENAMED.TST;32 and back.
                                1B04 970      ;
                                1B04 971      ;
                                1B04 972      $r 1_STORE  NAM=OLDNAM,ESL=#0,RSL=#0
                                1B0F 973      $NAM_STORE  NAM=NEWNAM,ESL=#0,RSL=#0
                                1B1A 974      $RENAME OLDFAB=OLDFAB,NEWFAB=NEWFAB,ERR=REPORT_ERROR
                                FAB1 30 1B2F 975      BSBW  ERR
                                F2CC CF 95 1B32 976      TYPE  <FILE RENAMED!>
                                62 19 1B59 977      TSTB  VSW
                                F295 CF  E942 CF  DE 1B5D 978      BLSS  RERENM      ; omit printout if not wanted
                                01F3 30 1B5F 979      TYPE  <OLD NAME:>
                                F264 CF  E971 CF  DE 1B86 980      MOVAL  OLDNAM,NAM
                                01C2 30 1B8D 981      BSBW  TST$SHOWNAM
                                FA0A 30 1B90 982      TYPE  <NEW NAME:>
                                30 1BBE 983      MOVAL  NEWNAM,NAM
                                1BC1 984      BSBW  TST$SHOWNAM
                                1BC1 985      RERENM: $RENAME OLDFAB=NEWFAB,-      ; re-rename file
                                1BD6 986      NEWFAB=OLDFAB,ERR=REPORT_ERROR
                                1BD9 987      BSBW  ERR
                                1C00 988      TYPE  <ORIGINAL FILENAME RESTORED>
                                1C00 989
                                1C00 990      ;
                                1C00 991      ; Now OPEN the original file by FILE ID and DELETE it on CLOSE by FID
                                1C00 992      ;
                                1C00 993      ;
                                01000000 8F  C8 1C00 994      BISL  #FABS$M_NAM,-      ; OPEN by FID
                                E70B CF  1C06 995      FAB$L_FOP+STICKY_FAB2
                                00300000 8F  CA 1C09 996      BICL  #FABS$M_CBT!FABS$M_CTG,-      ; to do CBT test
                                E702 CF  1C0F 997      FAB$L_FOP+STICKY_FAB2
                                1C12 998      $OPEN  FAB=STICKY_FAB2,-      ; OPEN orig. file
                                1C12 999      ERR=REPORT_ERROR
                                11 E6EB CF  F9BF 30 1C21 1000      BSBW  ERR
                                15  E0 1C24 1001      BBS  #FABS$V_CBT,FAB$L_FOP+STICKY_FAB2,CBTOK

```



```

1CB8 1019
1CB8 1020 :
1CB8 1021 : General RMS tests
1CB8 1022 :
1CB8 1023 : TEST 1
1CB8 1024 :
1CB8 1025 : SEQUENTIAL FILE ORG, DISK, VFC RFM
1CB8 1026 :
1CB8 1027 :
1CB8 1028 TST$GENERAL:
00000000'EF 00 FB 1CB8 1029 CALLS #0,RMT$TEST_1A
F260 31 1CBF 1030 BRW CMD_LOOP
1CC2 1031
1CC2 1032
1CC2 1033
1CC2 1034 :
1CC2 1035 : SHARED FILE Tests
1CC2 1036 :
1CC2 1037 :
1CC2 1038 :
1CC2 1039 TST$SHR:
00000000'EF 00 FB 1CC2 1040 CALLS #0,SHRSTRT
F256 31 1CC9 1041 BRW CMD_LOOP
1CCC 1042
1CCC 1043
1CCC 1044
1CCC 1045 :
1CCC 1046 : BLOCK I/O Tests
1CCC 1047 :
1CCC 1048 :
1CCC 1049 : TEST 2
1CCC 1050 :
1CCC 1051 :
1CCC 1052 TST$BLOCK:
00000000'EF 00 FB 1CCC 1053 CALLS #0,TST$BLOCKIO
F24C 31 1CD3 1054 BRW CMD_LOOP
1CD6 1055
1CD6 1056
1CD6 1057
1CD6 1058 :
1CD6 1059 : RELATIVE Tests
1CD6 1060 :
1CD6 1061 : TEST 3
1CD6 1062 :
1CD6 1063 :
1CD6 1064 TST$RELATIVE:
00000000'EF 00 FB 1CD6 1065 CALLS #0,RMT$TEST_3A
F242 31 1CDD 1066 BRW CMD_LOOP
1CE0 1067
1CE0 1068 :
1CE0 1069 : INDEXED Tests
1CE0 1070 :
1CE0 1071 : RMSTEST8
1CE0 1072 :
1CE0 1073 :
1CE0 1074 TST$INDEX:
00000000'EF 00 FB 1CE0 1075 CALLS #0,RMT$INDEX

```



```

F238 31 1CE7 1076 BRW CMD_LOOP
1CEA 1077
1CEA 1078
1CEA 1079
1CEA 1080
1CEA 1081
1CEA 1082
1CEA 1083
1CEA 1084 : FLUSH and FHC XAB Tests
1CEA 1085 :
1CEA 1086 : TEST 4
1CEA 1087 :
1CEA 1088 :
1CEA 1089 TST$F:
00000000'EF 00 FB 1CEA 1090 CALLS #C,RMT$TEST_4A
F22E 31 1CF1 1091 BRW CMD_LOOP
1CF4 1092
1CF4 1093
1CF4 1094
1CF4 1095 :
1CF4 1096 : MAGTAPE Tests
1CF4 1097 :
1CF4 1098 : TEST 6
1CF4 1099 :
1CF4 1100
1CF4 1101 TST$MTAPE:
00000000'EF 00 FB 1CF4 1102 CALLS #0,TST$MTA
F224 31 1CFB 1103 BRW CMD_LOOP
1CFE 1104
1CFE 1105
1CFE 1106 :
1CFE 1107 : NETWORK Tests
1CFE 1108 :
1CFE 1109 :
1CFE 1110
1CFE 1111 TST$NET:
1CFE 1112
1CFE 1113 :
1CFE 1114 : COMMENT CALL until TST$NETWORK is implemented
1CFE 1115 : to prevent an undefined symbol at link-time
1CFE 1116 : CALLS #0,TST$NETWORK
1CFE 1117 :
1CFE 1118
F221 31 1CFE 1119 ABORTP: BRW CMD_LOOP
1D01 1120
1D01 1121 :
1D01 1122 : PRINT or SUBMIT OUTPUT File
1D01 1123 :
1D01 1124 :
1D01 1125
1D01 1126 TST$PRINT:
50 8F EDCA CF 91 1D01 1127 PROMPT <PRINT (P) OR SUBMIT COMMAND FILE (S)?>,NULL=ABORTP
0B 12 1D26 1128 CMPB CMDBUF,#^A/P/
E489 CF 00002000 8F C8 1D2C 1129 BNEQ 10$
11 11 1D2E 1130 BISL2 #FAB$M_SPL,FAB$S_L_FOP+OUTFAB
53 8F EDB7 CF 91 1D37 1131 BRB 20$
1D39 1132 10$: CMPB CMDBUF,#^A/S/

```

E476 CF	00004000	BD	12	1D3F	1133	BNEQ	ABORTP		
		BF	C8	1D41	1134	BISL2	#FABSM SCF,FABSL_FOP+OUTFAB		
				1D4A	1135	PROMPT	<DELETE FILE AFTER QUEUING?>,NULL=30\$		
	59	BF	ED81	CF	91	1D6F	1136	CMPB	CMDBUF,#^A/Y/
					12	1D75	1137	BNEQ	30\$
E440 CF	00008000	BF	C8	1D77	1138	BISL2	#FABSM DLT,FABSL_FOP+OUTFAB		
		F265	31	1D80	1139	BRW	TST\$COPY		

```

1D83 1141
1D83 1142 :
1D83 1143 : Subroutine to PRINT the EXPANDED and RESULTANT NAME STRINGS from
1D83 1144 : the NAM BLOCK -- The address of the NAM BLOCK is found in NAM
1D83 1145 :
1D83 1146 :
1D83 1147 TST$SHOWNAM:
      FOA2 CF 95 1D83 1148 TSTB VSW ; long typeout wanted?
      73 19 1D87 1149 BLSS RTN ; branch if yes
54 F095 CF D0 1D89 1150 MOVL NAM,R4 ; get addr of NAMBLK in R4
      46 10 1D8E 1151 TYPE <EXPANDED NAME STRING:>
      1D85 1152 BSBB TYPEXP ; type EXPANDED NAME STRING
E322 CF 03 A4 98 1D87 1153 TYPE <RESULTANT NAME STRING:>
E322 CF 04 A4 D0 1DDE 1154 MOVZBW NAM$B_RSL(R4),CMDORAB+RAB$W_RSZ
      F7E7 30 1DE4 1155 MOVL NAM$L_RSA(R4),CMDORAB+RAB$L_RBF
      05 1DFC 1156 $PUT RAB=CMDORAB,ERR=REPORT_ERROR
      1DF9 1157 BSBW ERR
      1DFC 1158 RTN: RSB
      1DFD 1159
      1DFD 1160 :
      1DFD 1161 : PRINT EXPANDED NAME STRING
      1DFD 1162 : R4 must point to a NAMBLK
      1DFD 1163 :
      1DFD 1164 :
E303 CF 0B A4 98 1DFD 1165 TYPEXP: MOVZBW NAM$B_ESL(R4),CMDORAB+RAB$W_RSZ
E303 CF 0C A4 D0 1E03 1166 MOVL NAM$L_ESA(R4),CMDORAB+RAB$L_RBF
      F7C8 30 1E09 1167 $PUT RAB=CMDORAB,ERR=REPORT_ERROR
      05 1E18 1168 BSBW ERR
      1E1B 1169 RSB
      1E1C 1170
      1E1C 1171 :
      1E1C 1172 : Routines to issue messages
      1E1C 1173 :
      1E1C 1174 :
      1E1C 1175 :
000003E9 1E1C 1176 RMSTEST == 1001 ; FACILITY CODE for RMSTEST
      1E1C 1177
      1E1C 1178 :
      1E1C 1179 :
      1E1C 1180
      1E1C 1181 BEGPUT::
      00 DD 1E1C 1182 PUSHL #0 ; get current TIME
      F07E CF DF 1E1E 1183 PUSHAL BEG_DESCR ; string descr
      02 DD 1E22 1184 PUSHL #2 ; # of FAO args
03E90001'8F DD 1E24 1185 PUSHL #SHR$ BEGINT+<RMSTEST*65536>+1 ; MESSAGE CODE
00000000'GF 04 FB 1E2A 1186 CALLS #4,G^CIB$SIGNAL
      05 1E31 1187 RSB
      1E32 1188 FINPUT::
      00 DD 1E32 1189 PUSHL #0 ; current TIME
      F070 CF DF 1E34 1190 PUSHAL FIN_DESCR ; string descr
      02 DD 1E38 1191 PUSHL #2 ; # of FAO args
03E90001'8F DD 1E3A 1192 PUSHL #SHR$ ENDEDT+<RMSTEST*65536>+1 ; MESSAGE ID
00000000'GF 04 FB 1E40 1193 CALLS #4,G^CIB$SIGNAL
      05 1E47 1194 RSB
      1E48 1195 FLDPUT::
52 6E 15 C3 1E48 1196 SUBL3 #21,(SP),R2 ; try to find ABS. PC of ERROR
      50 DD 1E4C 1197 PUSHL R0 ; save status code, if nec.

```

```

      52 DD 1E4E 1198      PUSHL R2          ; PC
      F05C CF 1E50 1199      PUSHAL FLD_DESCR    ; string descr
      02 DD 1E54 1200      PUSHL #2          ; # of FAO args
      03E90002'8F DD 1E56 1201      PUSHL #SHR$ BADFIELD+<RMSTEST*65536>+2 ; MESSAGE ID
      00000000'GF 04 FB 1E5C 1202      CALLS #4,G^CIB$SIGNAL
      01 BA 1E63 1203      POPR #^M<R0>          ; restore R0
      05 OS 1E65 1204      RSB
      1E66 1205
      1E66 1206
      1E66 1207 : Subroutine to issue 'EXPECTED EOF NOT RECEIVED' msg, and display actual
      1E66 1208 : error R10 points to the structure at fault, for REPORT_ERR
      1E66 1209 :
      1E66 1210 :
      1E66 1211 EOFPUT::
      52 6E 05 C3 1E66 1212      SUBL3 #5,(SP),R2
      F05C CF 50 DD 1E6A 1213      PUSHL R0          ; save STATUS, if nec.
      01 DF 1E6C 1214      PUSHAL EOF_DSC      ; 'EOF' arg for FAO
      03E90003'8F 01 DD 1E70 1215      PUSHL #1          ; # of args, for FAO
      00000000'GF 03 FB 1E72 1216      PUSHL #SHR$ UNXPCTSTS+<RMSTEST@16>+3 ; error code
      01 BA 1E7F 1218      POPR #^M<R0>          ; restore STATUS
      F6FA 30 1E81 1219      BSBW REPORT_ERR    ; find out what error was
      F760 31 1E84 1220      BRW ERR1          ; continue?,even if R0=SUC
      1E87 1221
      1E87 1222 :
      1E87 1223 : Routine to echo the command if command output not to a terminal
      1E87 1224 :
      1E87 1225 :
      2B 00000000'EF E9 1E87 1226      ECHO: BLBC VERBOSITY,10$ ; omit if low verbosity
      00000000'8F E0 1E8E 1227      BBS #DEV$V TRM,- ; or if terminal output
      21 E1F9 CF 1E94 1228      CMDOFAB+FAB$ DEV,10$
      E225 CF 08 A1 1E98 1229      ADDW3 #8,CMDIRAB+RAB$W RSZ,- ; add size of PROMPT to len
      E266 CF 1E9D 1230      CMDORAB+RAB$W RSZ
      E265 CF EC48 CF DE 1EA0 1231      MOVAL ECHOF,CMDORAB+RAB$ RBF ; address of output line
      F72A 30 1EA7 1232      $PUT RAB=CMDORAB,ERR=REPORT_ERROR
      05 OS 1EB6 1233      BSBW ERR
      1EB9 1234 10$: RSB
  
```

```

00000003 1EBA 1236 .MDELETE $FAB,$RAB,$NAM
00000001 1EBA 1237 .MDELETE $FAB_STORE,$NAM_STORE,$RAB_STORE
1EBA 1238
1EBA 1239 :
1EBA 1240 :
1EBA 1241 : 'H' Switch for HELP !!!!!
1EBA 1242 :
1EBA 1243 :
1EBA 1244 HELP:
03 00000000'EF E8 1EBA 1245 BLBS VERBOSITY,10$
FOE2 31 1EC1 1246 BRW QMARK
1EC4 1247 10$:
1EC4 1248 .SAVE
00000367 1249 .PSECT __$RMSNAM
0367 1250 HLPMSG:
0D 44 4E 45 50 50 41 20 2D 2D 20 41 0367 1251 .ASCII /A -- APPEND/<13><10>
0A 0373
49 20 4B 43 4F 4C 42 20 2D 2D 20 42 0374 1252 .ASCII %B -- BLOCK I/O%<13><10>
0A 0D 4F 2F 0380
0A 0D 59 50 4F 43 20 2D 2D 20 43 0384 1253 .ASCII /C -- COPY/<13><10>
20 48 53 55 4C 46 24 20 2D 2D 20 44 038F 1254 .ASCII /D -- DEBUG/<13><10>
20 42 41 58 20 43 48 46 20 44 4E 41 039B 1255 .ASCII /F -- $FLUSH AND FHC XAB TESTS/<13><10>
0A 0D 53 54 53 45 54 03A7
4C 41 52 45 4E 45 47 20 2D 2D 20 47 03B3
20 4C 41 49 54 4E 45 55 51 45 53 20 03B8
0A 0D 53 54 53 45 54 03D2
45 4D 20 53 49 48 54 20 2D 2D 20 48 03D9 1257 .ASCII /H -- THIS MESSAGE/<13><10>
0A 0D 45 47 41 53 53 03E5
20 59 4B 43 49 54 53 20 2D 2D 20 4B 03EC 1258 .ASCII /K -- STICKY CREATE,DIRECTORY TESTS/<13><10>
43 45 52 49 44 2C 45 54 41 45 52 43 03F8
0A 0D 53 54 53 45 54 20 59 52 4F 54 0404
4C 41 52 45 4E 45 47 20 2D 2D 20 4C 0410 1259 .ASCII /L -- GENERAL RELATIVE TESTS/<13><10>
45 54 20 45 56 49 54 41 4C 45 52 20 041C
0A 0D 53 54 53 0428
45 50 41 54 47 41 4D 20 2D 2D 20 4D 042D 1260 .ASCII /M -- MAGTAPE TESTS/<13><10>
0A 0D 53 54 53 45 54 20 0439
20 54 45 4E 43 45 44 20 2D 2D 20 4E 0441 1261 .ASCII /N -- DECNET TESTS/<13><10>
0A 0D 53 54 53 45 54 044D
49 57 20 59 50 4F 43 20 2D 2D 20 4F 0454 1262 .ASCII /O -- COPY WITH CHOICE OF FILE ORG FOR OUTPUT FILE/<13><10>
46 4F 20 45 43 49 4F 48 43 20 48 54 0460
4F 46 20 47 52 4F 20 45 4C 49 46 20 046C
4C 49 46 20 54 55 50 54 55 4F 20 52 0478
0A 0D 45 0484
4F 20 54 4E 49 52 50 20 2D 2D 20 50 0487 1263 .ASCII /P -- PRINT OR SUBMIT COMMAND FILE/<13><10>
4D 4F 43 20 54 49 4D 42 55 53 20 52 0493
0A 0D 45 4C 49 46 20 44 4E 41 4D 049F
4E 55 52 20 4F 2F 49 20 2D 2D 20 52 04AA 1264 .ASCII %R -- I/O RUNDOWN%<13><10>
0A 0D 4E 57 4F 44 04B6
46 45 44 20 54 45 53 20 2D 2D 20 53 04BC 1265 .ASCII /S -- SET DEFAULT NAME STRING/<13><10>
54 53 20 45 4D 41 4E 20 54 4C 55 41 04C8
0A 0D 47 4E 49 52 04D4
46 45 44 20 54 45 53 20 2D 2D 20 55 04DA 1266 .ASCII /U -- SET DEFAULT UFD/<13><10>
0A 0D 44 46 55 20 54 4C 55 41 04E6
20 59 46 49 52 45 56 20 2D 2D 20 56 04F0 1267 .ASCII /V -- VERIFY FILE SPECIFICATIONS/<13><10>
49 46 49 43 45 50 53 20 45 4C 49 46 04FC
0A 0D 53 4E 4F 49 54 41 43 0508

```

RMSTEST
028

USER ROUTINE TO CALL AND TEST RMS ; H 4

16-SEP-1984 01:44:23 VAX/VMS Macro V04-00
5-SEP-1984 04:21:33 [UETP.SRC]RMSTEST0.MAR;1

Page 28
(21)

RM
VO

```
44 45 58 45 44 4E 49 20 2D 2D 20 58 0511 1268
OA 0C 54 53 45 54 20 45 4C 49 46 20 051D
45 4C 20 52 45 48 54 4F 20 4C 4C 41 0529 1269
4E 49 20 45 52 41 20 53 52 45 54 54 0535
                                OA 0D 44 49 4C 41 56 0541
                                000001E1 0548 1270
                                00001EC4 1271
E23F CF 00000367'EF DE 1EC4 1272
E232 CF 01E1 8F B0 1ECD 1273
                                1ED4 1274
                                F6FD 30 1EE3 1275
                                F039 31 1EE6 1276
                                1EE9 1277
```

```
.ASCII /X -- INDEXED FILE TEST/<13><10>
.ASCII /ALL OTHER LETTERS ARE INVALID/<13><10>
```

```
HLPMSGSZ=-HLPMSG
.RESTORE
MOVAL HLPMSG,CMDORAB+RAB$L RBF
MOVW #HLPMSGSZ,CMDORAB+RAB$W RSZ
$PUT RAB=CMDORAB,ERR=REPORT_ERROR
BSBW ERR
BRW CMD_LOOP
.END START
```

RMSTEST
Symbol table

USER ROUTINE TO CALL AND TEST RMS ; 1 4

16-SEP-1984 01:44:23 VAX/VMS Macro V04-00
5-SEP-1984 04:21:33 [UETP.SRC]RMSTEST0.MAR;1

Page 29
(21)

RM
V0

\$\$PSECT_EP	= 00000000				DNSBSZ	= 00000050	G	D	
\$\$TAB	= 0000052C	R	D	01	DNSBUF	000008C8	RG	D	01
\$\$TABEND	= 0000058C	R	D	01	DNSSIZ	00000E20	R	D	01
\$\$TMP	= 00000000		D		DONE	0000138E	R	D	01
\$\$TMP1	= 00000002		D		ECHO	00001E87	R	D	01
\$\$TMP2	= 000000CF		D		ECHOBF	00000AEC	R	D	01
\$\$TMPX	= 00000351	R	D	04	ECHOIT	00000F50	R	D	01
\$\$TMPX1	= 00000016		D		EOFDET	0000155F	R	D	01
\$\$RMSTEST	= 0000001E		D		EOFPUT	00001E66	RG	D	01
\$\$RMS_PBUGCHK	= 00000010		D		EOF_DSC	00000ECC	R	D	01
\$\$RMS_TBUGCHK	= 00000008		D		ERR	000015E3	RG	D	01
\$\$RMS_UMODE	= 00000004		D		ERR1	000015E7	R	D	01
..AFLG	= 00000000		D		EXPBSZ	= 00000040	G	D	
..FLG	= 00000001		D		EXPBUF	00000A28	RG	D	01
..TMP	= 000000C0		D		EXPEOF	00000ED4	R	D	01
..TYP	= 000000CF		D		EXPEOF_LEN	= 00000003		D	
.LEN	= 00000001		D		FAB\$B_DNS	= 00000035		D	
ABORTP	00001CFE	R	D	01	FAB\$B_FNS	= 00000034		D	
APPEND	0000126B	R	D	01	FAB\$B_FSZ	= 0000003F		D	
BADPARSE	0000173D	R	D	01	FAB\$B_ORG	= 0000001D		D	
BADUFD	0000171B	R	D	01	FAB\$B_RAT	= 0000001E		D	
BEGPUT	00001E1C	RG	D	01	FAB\$B_RFM	= 0000001F		D	
BEG_DESCR	00000EA0	RG	D	01	FAB\$C_BID	= 00000003		D	
BR_COPY	00000FCB	R	D	01	FAB\$C_BLN	= 00000050		D	
CBTOK	00001C3B	R	D	01	FAB\$C_IDX	= 00000020		D	
CHK	00001ABE	R	D	01	FAB\$C_REL	= 00000010		D	
CLBR1	000016B0	R	D	01	FAB\$C_SEQ	= 00000000		D	
CLBR2	00001718	R	D	01	FAB\$C_VAR	= 00000002		D	
CLEAN	000013F9	R	D	01	FAB\$S_ALQ	= 00000010		D	
CLOSE_UP	00001513	R	D	01	FAB\$S_DEV	= 00000040		D	
CLSINF	000013C1	R	D	01	FAB\$S_DNA	= 00000030		D	
CLS_IN_FL	0000142E	R	D	01	FAB\$S_FNA	= 0000002C		D	
CLS_OUT_FL	00001457	R	D	01	FAB\$S_FOP	= 00000004		D	
CMD	00000E2A	R	D	01	FAB\$S_MRN	= 00000038		D	
CMDBSZ	= 00000084		G	D	FAB\$S_NAM	= 00000028		D	
CMDBUF	00000AF4	RG	D	01	FAB\$S_STS	= 00000008		D	
CMDIFAB	00000000	RG	D	01	FAB\$S_STV	= 0000000C		D	
CMDIRAB	000000A0	RG	D	01	FAB\$M_CBT	= 00200000		D	
CMDOFAB	00000050	RG	D	01	FAB\$M_CIF	= 02000000		D	
CMDORAB	000000E4	RG	D	01	FAB\$M_CR	= 00000002		D	
CMD_LOOP	00000F22	R	D	01	FAB\$M_CTG	= 00100000		D	
CMD_LOOP_BR	00000FC8	R	D	01	FAB\$M_DLT	= 00008000		D	
CON	0000115E	R	D	01	FAB\$M_NAM	= 01000000		D	
COPY_DONE	0000133C	R	D	01	FAB\$M_SCF	= 00004000		D	
COPY_LOOP	000012FD	R	D	01	FAB\$M_SPL	= 00002000		D	
COPY_PROMPT	00000FF7	R	D	01	FAB\$V_CBT	= 00000015		D	
CPYBSZ	= 00000200		G	D	FAB\$V_CHAN_MODE	= 00000002		D	
CPYBUF	00000C18	RG	D	01	FAB\$V_CR	= 00000001		D	
CPYBUFDSC	00000E2C	R	D	01	FAB\$V_CTG	= 00000014		D	
CPYEND	00001497	R	D	01	FAB\$V_FILE_MODE	= 00000004		D	
CPYFLG	00000E26	R	D	01	FAB\$V_GET	= 00000001		D	
CPYNAM	000012DF	R	D	01	FAB\$V_LNM_MODE	= 00000000		D	
CTGOK	00001C52	R	D	01	FAB\$V_OFF	= 0000001D		D	
DEVSV TRM	*****		X	01	FAB\$V_PUT	= 000000C0		D	
DIRBUF	00000ABC	R	D	01	FAB\$V_SUP	= 00000002		D	
DIRDESC	00000AE4	R	D	01	FAB\$W_BLS	= 0000003C		D	
DIRLEN	00000ABB	R	D	01	FAB\$W_GBC	= 00000048		D	

RMSTEST
Symbol table

USER ROUTINE TO CALL AND TEST RMS ; 4

16-SEP-1984 01:44:23 VAX/VMS Macro V04-00
5-SEP-1984 04:21:33 [UETP.SRC]RMSTEST0.MAR;1

FABSW_IFI	= 00000002	D			OUTFAB	000001BC	RG	D	01
FABSW_MRS	= 00000036	D			OUTRAB	0000020C	RG	D	01
FAOBUF	00000E3C	RG	D	01	PNSBUF	00000B78	RG	D	01
FAOSTR	00000E34	R	D	01	POPPC	00001679	R	D	01
FINPUT	00001E32	RG	D	01	QMARK	00000FA6	R	D	01
FIN_DESCR	00000EA8	RG	D	01	R	000015A6	R	D	01
FLDPUT	00001E48	RG	D	01	RAB\$B_KRF	= 00000035	D		
FLD_DESCR	00000EB0	RG	D	01	RAB\$B_PSZ	= 00000034	D		
FOR\$CNV_IN_1	*****	X		01	RAB\$B_RAC	= 0000001E	D		
FSTR1	00000E44	R	D	01	RAB\$C_BID	= 00000001	D		
FSTR1LEN	= 0000003F	D			RAB\$C_BLN	= 00000044	D		
GO	0000120C	R	D	01	RAB\$C_SEQ	= 00000000	D		
GOT ONE	000010D2	R	D	01	RAB\$C_CTX	= 00000018	D		
HELP	00001EBA	R	D	01	RAB\$C_PBF	= 00000030	D		
HLPMSG	00000367	R	D	04	RAB\$C_RBF	= 00000028	D		
HLPMSGSZ	= 000001E1	D			RAB\$C_ROP	= 00000004	D		
ILLSER	00000ED7	R	D	01	RAB\$C_STS	= 00000008	D		
ILLSERDEC	00000EC4	R	D	01	RAB\$M_ASY	= 00000001	D		
ILLSERL	= 00000007	D			RAB\$M_EOF	= 00000100	D		
INFAB	00000128	RG	D	01	RAB\$V_EOF	= 00000008	D		
INRAB	00000178	RG	D	01	RAB\$V_PMT	= 0000001E	D		
INTEGER	00000EC0	R	D	01	RAB\$W_ISI	= 00000002	D		
KRF_DESC	00000EB8	R	D	01	RAB\$W_RSZ	= 00000022	D		
LIB\$SIGNAL	*****	X		01	RBRACK	= 0000005D	D		
MORE	00001949	R	D	01	RD1	00001789	R	D	01
NAM	00000E22	RG	D	01	RD2	000017DE	R	D	01
NAM\$B_ESL	= 0000000B	D			RDMSG	00000E83	R	D	01
NAM\$B_ESS	= 0000000A	D			RDMSGLEN	= 0000001D	D		
NAM\$B_NOP	= 00000008	D			RELATED_NAM	000003C0	R	D	01
NAM\$B_RSL	= 00000003	D			RELZERO	00000000	R	D	01
NAM\$B_RSS	= 00000002	D			REL_FILE	00000420	R	D	01
NAM\$C_BID	= 00000002	D			REL_FILE_SIZE	= 00000006	D		
NAM\$C_BLN	= 00000060	D			REM_SRCH	00001A30	R	D	01
NAM\$C_MAXRSS	= 000000FF	D			REOPEN_CMD	0000181A	R	D	01
NAM\$C_ESA	= 0000000C	D			REPORT	00001592	RG	D	01
NAM\$C_FNB	= 00000034	D			REPORT_END	00001480	R	D	01
NAM\$C_RSA	= 00000004	D			REPORT_ERR	0000157E	RG	D	01
NAM\$C_WCC	= 00000030	D			REPORT_ERROR	00001574	RG	D	01
NAM\$V_LOWVER	= 0000000E	D			RERENM	00001BC1	R	D	01
NAM\$V_WILDCARD	= 00000008	D			RESBSZ	= 00000040	G	D	
NAM\$W_FID	= 00000024	D			RESBUF	00000A68	RG	D	01
NAMBLR	00000250	RG	D	01	RHBBUF	00000AA8	RG	D	01
NEWFAB	0000047C	RG	D	01	RM\$S_CREATED	*****	X		01
NEWNAM	0000052C	RG	D	01	RM\$S_EOF	*****	X		01
NEW_ESBUF	0000078A	R	D	01	RM\$S_FNF	*****	X		01
NEW_RSBUF	00000889	R	D	01	RM\$S_NMF	*****	X		01
NEXT	00001988	R	D	01	RM\$S_NORMAL	*****	X		01
NOWC	000010E9	R	D	01	RM\$S_SUPERSEDE	*****	X		01
NO_MORE_FILES_MORE_FILES					000013CA	R	D	01	RMSTEST
OK	00001ACC	R	D	01	RMT\$INDEX	*****	X		01
OKSUP	00001933	R	D	01	RMT\$TEST_1A	*****	X		01
OLDFAB	0000042C	RG	D	01	RMT\$TEST_3A	*****	X		01
OLDNAM	000004CC	RG	D	01	RMT\$TEST_4A	*****	X		01
OLD_ESBUF	0000058C	R	D	01	RTN	00001DFC	R	D	01
OLD_RSBUF	0000068B	R	D	01	RWCBSZ	= 00000050	G	D	
ORGFLG	00000E28	R	D	01	RWCBUF	000009D8	RG	D	01
OUT	00001AF3	R	D	01	SAME_ORG	000011FE	R	D	01

RMSTEST
Symbol table

USER ROUTINE TO CALL AND TEST RMS^K; 4

16-SEP-1984 01:44:23 VAX/VMS Macro V04-00
5-SEP-1984 04:21:33 [UETP.SRC]RMSTEST0.MAR;1

SAVE_FID	00000426	R	D	01
SEMI	= 0000003B		D	
SEQ_ORG	00001207	R	D	01
SETVSW	00000FD8	R	D	01
SHRS_BADFIELD	*****		X	01
SHRS_BEGINT	*****		X	01
SHRS_ENEDET	*****		X	01
SHRS_RMSERROR	*****		X	01
SHRS_SYSEERROR	*****		X	01
SHRS_UNXPCTSTS	*****		X	01
SHRSTR	*****		X	01
SOK	000019E7	R	D	01
SS\$ ILLSER	*****		X	01
START	00000EDE	RG	D	01
STICKY_FAB2	00000310	R	D	01
STICKY_NAMBLK	00000360	R	D	01
SYSSCLOSE	*****		GX	01
SYSSCONNECT	*****		GX	01
SYSSCREATE	*****		GX	01
SYSSDISCONNECT	*****		GX	01
SYSSENTER	*****		GX	01
SYSS\$EXIT	*****		GX	01
SYSS\$FAO	*****		X	01
SYSS\$GET	*****		GX	01
SYSS\$OPEN	*****		GX	01
SYSS\$PARSE	*****		GX	01
SYSS\$PUT	*****		GX	01
SYSS\$REMOVE	*****		GX	01
SYSS\$RENAME	*****		GX	01
SYSS\$RMSRUNDWN	*****		X	01
SYSS\$SEARCH	*****		GX	01
SYSS\$SETDDIR	*****		X	01
SYSS\$WAIT	*****		GX	01
TRAN	*****		X	01
TST\$APPEND	00001413	R	D	01
TST\$BLOCK	00001CCC	R	D	01
TST\$BLOCKIO	*****		X	01
TST\$COPY	00000FE8	R	D	01
TST\$DEBUG	000014A9	R	D	01
TST\$F	00001CEA	R	D	01
TST\$GENERAL	00001CB8	R	D	01
TST\$INDEX	00001CE0	R	D	01
TST\$MTA	*****		X	01
TST\$MTAPE	00001CF4	R	D	01
TST\$NET	00001CFE	R	D	01
TST\$ORG	00001426	R	D	01
TST\$PRINT	00001D01	R	D	01
TST\$RELATIVE	00001CD6	R	D	01
TST\$RUNDWN	00001758	R	D	01
TST\$SETDDIR	000016B3	R	D	01
TST\$SETDFLT	000014E0	R	D	01
TST\$SHOWNAM	00001D83	R	D	01
TST\$SHR	00001CC2	R	D	01
TST\$STICKY	00001883	R	D	01
TYPEXP	00001DFD	R	D	01
VERBOSITY	*****		X	01
VSW	00000E29	R	D	01

WC
WCDBSZ
WCDBUF
WCDNAM
WCFLG

= 0000105C	R	D	01
= 00000050	G	D	
00000988	RG	D	01
00000280	RG	D	01
00000E27	R	D	01

! Psect synopsis !

PSECT name	Allocation	PSECT No.	Attributes
. ABS	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
RMSTEST	00001EE9 (7913.)	01 (1.)	NOPIC USR CON REL GBL NOSHR EXE RD WRT NOVEC LONG
\$ABSS	00000000 (0.)	02 (2.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
\$RMSNAM	00000044 (68.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
__SRMSNAM	00000548 (1352.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

! Performance indicators !

Phase	Page faults	CPU Time	Elapsed Time
Initialization	37	00:00:00.09	00:00:00.53
Command processing	134	00:00:00.59	00:00:01.90
Pass 1	425	00:00:21.26	00:00:49.69
Symbol table sort	0	00:00:00.95	00:00:02.00
Pass 2	336	00:00:05.70	00:00:14.24
Symbol table output	39	00:00:00.26	00:00:00.73
Psect synopsis output	5	00:00:00.05	00:00:00.11
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	978	00:00:28.91	00:01:09.21

The working set limit was 900 pages.
109934 bytes (215 pages) of virtual memory were used to buffer the intermediate code.
There were 40 pages of symbol table space allocated to hold 672 non-local and 50 local symbols.
1277 source lines were read in Pass 1, producing 68 object records in Pass 2.
53 pages of virtual memory were used to define 47 macros.

! Macro library statistics !

Macro library name	Macros defined
\$_\$255\$DUA28:[SYS.OBJ]LIB.MLB;1	1
\$_\$255\$DUA28:[SYSLIB]STARLET.MLB;2	35
TOTALS (all libraries)	36

1012 GETS were required to define 36 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS\$:RMSTESTO/OBJ=OBJ\$:RMSTESTO MSRC\$:RMSTESTO/UPDATE=(ENH\$:RMSTESTO)+EXECMLS/LIB

