```
                                                              K  1



UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  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  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP
UUU         UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPP
UUU         UUU  EEEEEEEEEEEE           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
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP
UUUUUUUUUUUUUUU  EEEEEEEEEEEEEEE        TTT        PPP
```

```
_$


Va
---
000
000
000
7F
7F
7F
7F
7F
7F
7F
7F
7F
```

FILEID**RMSTEST5   I 10

```
RRRRRRRR    MM      MM    SSSSSSSS  TTTTTTTTTT  EEEEEEEEEE    SSSSSSSS  TTTTTTTTTT  5555555555
RRRRRRRR    MM      MM    SSSSSSSS  TTTTTTTTTT  EEEEEEEEEE    SSSSSSSS  TTTTTTTTTT  5555555555
RR      RR  MMMM  MMMM  SS              TT      EE          SS              TT      55
RR      RR  MMMM  MMMM  SS              TT      EE          SS              TT      55
RR      RR  MM  MM  MM  SS              TT      EE          SS              TT      555555
RR      RR  MM  MM  MM  SS              TT      EE          SS              TT      555555
RRRRRRRR    MM      MM    SSSSSS        TT      EEEEEEEE      SSSSSS        TT            55
RRRRRRRR    MM      MM    SSSSSS        TT      EEEEEEEE      SSSSSS        TT            55
RR  RR      MM      MM          SS      TT      EE                  SS      TT              55
RR  RR      MM      MM          SS      TT      EE                  SS      TT              55
RR    RR    MM      MM          SS      TT      EE                  SS      TT      55      55      ....
RR    RR    MM      MM          SS      TT      EE                  SS      TT      55      55      ....
RR      RR  MM      MM  SSSSSSSS        TT      EEEEEEEEEE  SSSSSSSS        TT        555555        ....
RR      RR  MM      MM  SSSSSSSS        TT      EEEEEEEEEE  SSSSSSSS        TT        555555        ....


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              $BEGIN  RMSTEST5,009,__RMSTEST,<TEST RECORD LOCKING>,<GBL,LONG>
0000
0000      ;
0000
0000              .ENABL  DBG
0000
0000      ;
0000
0000              $RMSDEF
0000              .NLIST  MEB
0000
0000      ;
0000      ;  macros:
0000      ;
0000
0000              .MACRO  BUFF NAM,SIZE
0000      NAM'BUF::
0000              .BLKB   SIZE
0000              NAM'BSZ==SIZE
0000              .ENDM   BUFF
0000
0000              .MACRO  TYPE STRING, ?L
0000              STORE   <STRING>
0000              BLBC    VERBOSITY,L
0000              MOVL    #$$.TMPX,CMDORAB+RAB$L_RBF
0000              MOVW    #$$.TMPX1,CMDORAB+RAB$W_RSZ
0000              $PUT    RAB=CMDORAB,ERR=REPORT_ERROR
0000              BSBW    ERR
0000      L:
0000              .ENDM   TYPE
0000
0000
0000              .MACRO  WTTYPE STRING
0000              $WAIT   RAB=CMDORAB
0000              TYPE    <STRING>
0000              .ENDM   WTTYPE
0000
0000
0000              .MACRO  STORE STRING,PRE
0000              .SAVE
0000              .PSECT  __$RMSNAM
0000              $$.TMPX=.
0000              PRE                             ; store any carriage control info
0000              .ASCII  %STRING%
0000              $$.TMPX1=.-$$.TMPX
0000              .RESTORE
0000              .ENDM   STORE
```

```
0000
0000
0000            .MACRO  BEGIN TSTNAM
0000            STORE   <TSTNAM>
0000            MOVL    #$$.TMPX,BEG_DESCR+4      ; addr
0000            MOVL    #$$.TMPX1,BEG_DESCR       ; len
0000            BSBW    BEGPUT
0000            .ENDM   BEGIN
0000            .MACRO  FINISH TSTNAM
0000            STORE   <TSTNAM>
0000            MOVL    #$$.TMPX,FIN_DESCR+4      ; addr
0000            MOVL    #$$.TMPX1,FIN_DESCR       ; len
0000            BSBW    FINPUT
0000            .ENDM   FINISH
0000            .MACRO  FIELD FLDNAM
0000            STORE   <FLDNAM>
0000            MOVL    #$$.TMPX,FLD_DESCR+4      ; addr
0000            MOVL    #$$.TMPX1,FLD_DESCR       ; len
0000            BSBW    FLDPUT
0000            .ENDM   FIELD
0000            .MACRO  MBPT, ?L
0000            BLBC    VERBOSITY,L
0000            BPT
0000      L:
0000            .ENDM   MBPT
0000
0000
```

```
                 00000000   141             .PSECT    RMSTEST,GBL,LONG
                     0000   142             .ALIGN  LONG
                     0000   143 T5START::
                     0000   144 LOCK_FAB::
                     0000   145             $FAB    FAC=<GET,PUT,UPD>,DNM=<TST$DISK:.FIL;1>,shr=mse,nam=namblk
                     0050   146 RAB1:       $RAB    FAB=LOCK_FAB,UBF=LCKBUF,USZ=LCKBSZ,RBF=LCKBUF,-
                     0050   147                     RSZ=15,KBF=KEY
                     0094   148 RAB2:       $RAB    FAB=LOCK_FAB,UBF=LCK2BUF,USZ=LCKBSZ,-
                     0094   149                     RBF=LCK2BUF,RSZ=15,ROP=ULK,KBF=KEY
                     00D8   150             BUFF    LCK,200
                     01A0   151             BUFF    LCK2,200
        00000000     0268   152 KEY:        .LONG   0
```

```
                                026C   RMT$TEST_5A::
                                026C          BEGIN   <LOCKING TESTS>
                                0281          $OPEN   FAB=LOCK_FAB,ERR=REPORT_ERROR
               FD6B'   30       0292          BSBW    ERR
                                0295          $CONNECT        RAB=RAB1,ERR=REPORT_ERROR
               FD57'   30       02A6          BSBW    ERR
                                02A9          $CONNECT        RAB=RAB2,ERR=REPORT_ERROR
               FD43'   30       02BA          BSBW    ERR
       57    FD8F CF   DE       02BD          MOVAL   RAB1,R7                 ; r7 will be addr of rab1
       58    FDCE CF   DE       02C2          MOVAL   RAB2,R8                 ; r8 will be addr of rab2
                                02C7
                                02C7   ;
                                02C7   ; force lots of collisions
                                02C7   ; 1st stream 1 does a get (seq., so 1st record), locking it
                                02C7   ; the 2nd stream asks for the same record, but fails
                                02C7   ; the 1st stream releases it and the 2nd stream takes it
                                02C7   ; using manual locking the 2nd stream gets the 2nd record (locking 2 records)
                                02C7   ; the 1st stream asks for the 2nd record, finding it locked
                                02C7   ; until the 2nd stream frees both records
                                02C7   ; at which time the 1st stream takes the 2nd record
                                02C7   ;
                                02C7
                                02C7          $GET    RAB=R7,ERR=REPORT_ERROR ; get 1st rec., locking it
               FD27'   30       02D6          BSBW    ERR
                                02D9          $GET    RAB=R8                  ; this should fail
  000182AA 8F     50   D1       02E2          CMPL    R0,#RMS$_RLK
                  20   13       02E9          BEQL    OK_LCK
                                02EB          FIELD   <RMS RETURNED STATUS CODE>
       5A    FD90 CF   DE       0300          MOVAL   RAB2,R10
               FCF8'   30       0305          BSBW    REPORT_ERR
               FCF5'   30       0308          BSBW    ERR
                                030B   OK_LCK:
                                030B          $RELEASE        RAB=R7,ERR=REPORT_ERROR; release it
               FCE3'   30       031A          BSBW    ERR
                                031D
                                031D
                                031D          $GET    RAB=R8,ERR=REPORT_ERROR ; now get it
               FCD1'   30       032C          BSBW    ERR
                                032F          $GET    RAB=R8,ERR=REPORT_ERROR ; and another
               FCBF'   30       033E          BSBW    ERR
                                0341          $GET    RAB=R7                  ; this should fail
  000182AA 8F     50   D1       034A          CMPL    R0,#RMS$_RLK
                  20   13       0351          BEQL    OK_LCK1
                                0353          FIELD   <RMS RETURNED STATUS CODE>
       5A    FCE4 CF   DE       0368          MOVAL   RAB1,R10
               FC90'   30       036D          BSBW    REPORT_ERR
               FC8D'   30       0370          BSBW    ERR
                                0373   OK_LCK1:
                                0373          $FREE   RAB=R8,ERR=REPORT_ERROR ; free both records
               FC7B'   30       0382          BSBW    ERR
                                0385          $GET    RAB=R7,ERR=REPORT_ERROR
               FC69'   30       0394          BSBW    ERR                     ; should get it this time
         02     38 A7  D1       0397          CMPL    RAB$L_BKT(R7),#2        ; should have been 2nd
                  15   13       039B          BEQL    BKT_OK
                                039D          FIELD   <BKT IN RAB (RECORD NUMBER)>
                                03B2   BKT_OK:
                                  03B2
                                  03B2     ;
                                  03B2     ; test 2
                                  03B2     ; one stream now, using manual locking gets a record
                                  03B2     ; when it tries to get it again, it succeeds with the code, already locked
                                  03B2     ; shifting to automatic, it gets the 2nd record (actually locking them both)
                                  03B2     ; so when it tries to get the 1st again, it succeeds w/ the code, already locked
                                  03B2     ; but this released the 2nd record, so when it tries to release the 2nd, it fails
                                  03B2     ;
                                  03B2
                                  03B2                 $GET     RAB=R8,ERR=REPORT_ERROR
                     FC3C'   30   03C1                 BSBW     ERR                       ; get a record
                1E A8   01   90   03C4                 MOVB     #RAB$C_KEY,RAB$B_RAC(R8); control which rec.
         FE9A CF   38 A8     D0   03C8                 MOVL     RAB$L_BKT(R8),KEY         ; move rec. # into key
                                  03CE                 $GET     RAB=R8,ERR=REPORT_ERROR ; get same one
                     FC20'   30   03DD                 BSBW     ERR
        00018039 8F    50    D1   03E0                 CMPL     R0,#RMS$_OK_ALK           ; funny success
                       15    13   03E7                 BEQL     OK_ALK
                                  03E9                 FIELD    <RETURNED SUCCESS CODE>
     04 A8   00040000 8F     CA   03FE     OK_ALK:     BICL     #RAB$M_ULK,RAB$L_ROP(R8); shift to automatic
                 FE5E CF     D6   0406                 INCL     KEY                       ; get next record
                                  040A                 $GET     RAB=R8,ERR=REPORT_ERROR
                     FBE4'   30   0419                 BSBW     ERR                       ; get 2nd rec.,
                                  041C                                                    ; maintaining hold on 1st
        10 A7   10 A8        7D   041C                 MOVQ     RAB$W_RFA(R8),RAB$W_RFA(R7); save rfa for
                                  0421                                                    ; release of 2nd
                 FE43 CF     D7   0421                 DECL     KEY                       ; get 1st again, making sure it
                                  0425                                                    ; was still locked
                                  0425                 $GET     RAB=R8,ERR=REPORT_ERROR ; releasing 2nd one
                     FBC9'   30   0434                 BSBW     ERR
        00018039 8F    50    D1   0437                 CMPL     R0,#RMS$_OK_ALK
                       15    13   043E                 BEQL     OK_ALK1
                                  0440                 FIELD    <RETURNED SUCCESS CODE>
                                  0455     OK_ALK1:
        10 A8   10 A7        7D   0455                 MOVQ     RAB$W_RFA(R7),RAB$W_RFA(R8); restore rfa for release
                                  045A                 $RELEASE         RAB=R8            ; release 2nd one
        000181A0 8F    50    D1   0463                 CMPL     R0,#RMS$_RNL              ; r0 = rec. not locked?
                       20    13   046A                 BEQL     OK_RNL
                                  046C                 FIELD    <RMS RETURNED ERROR CODE>
              5A   FCCF CF   DE   0481                 MOVAL    RAB2,R10
                     FB77'   30   0486                 BSBW     REPORT_ERR                ; report error
                     FB74'   30   0489                 BSBW     ERR
                                  048C     OK_RNL:
     04 A8   00040000 8F     C8   048C                 BISL     #RAB$M_ULK,RAB$L_ROP(R8); back to manual
```

```
                                0494   256
                                0494   257 ;
                                0494   258 ; test 3
                                0494   259 ; with manual locking, get 1st record, locking it
                                0494   260 ; second user can't get it
                                0494   261 ; update it successfully, which will not unlock it
                                0494   262 ; so when updating it again, it will succeed because it's still manually locked
                                0494   263 ;
                                0494   264
         FDCF CF    01     D0   0494   265             MOVL     #1,KEY                     ; 1st record
                                0499   266             $GET     RAB=R8,ERR=REPORT_ERROR ; get w/ manual locking
                 FB55'     30   04A8   267             BSBW     ERR
           1E A7    01     90   04AB   268             MOVB     #RAB$C_KEY,RAB$B_RAC(R7)
                                04AF   269             $GET     RAB=R7                     ; try to get same record
      000182AA 8F   50     D1   04B8   270             CMPL     R0,#RMS$_RLK
                    20     13   04BF   271             BEQL     RLK_OK
                                04C1   272             FIELD    <RMS STATUS CODE>
        5A    FB76 CF      DE   04D6   273             MOVAL    RAB1,R10
                 FB22'     30   04DB   274             BSBW     REPORT_ERR
                 FB1F'     30   04DE   275             BSBW     ERR
                                04E1   276 RLK_OK:
   04 A8    00040000 8F    CA   04E1   277             BICL     #RAB$M_ULK,RAB$L_ROP(R8); switch to automatic
                                04E9   278             $UPDATE RAB=R8,ERR=REPORT_ERROR
                 FB05'     30   04F8   279             BSBW     ERR
                                04FB   280             $FIND    RAB=R8,ERR=REPORT_ERROR ; get a current record
                 FAF3'     30   050A   281             BSBW     ERR                     ; this should not fail
                                050D   282             $UPDATE RAB=R8,ERR=REPORT_ERROR ; this should succeed
                 FAE1'     30   051C   283             BSBW     ERR
                                051F   284             $RELEASE         RAB=R8,ERR=REPORT_ERROR; this should not fail
                 FACF'     30   052E   285             BSBW     ERR
                                0531   286 STS_OK:
```

```
                             0531   288
                             0531   289 ;
                             0531   290 ; test 4
                             0531   291 ; asynchronously do 2 gets on records in the same bucket
                             0531   292 ;
                             0531   293
         04 A7   01    D0    0531   294         MOVL    #RAB$M_ASY,RAB$L_ROP(R7)
         04 A8   01    D0    0535   295         MOVL    #RAB$M_ASY,RAB$L_ROP(R8)
       FD2A CF   01    D0    0539   296         MOVL    #1,KEY
                             053E   297         $GET    RAB=R7,ERR=REPORT_ERROR ; gets 1st record
          FD17 CF      D6    054D   298         INCL    KEY
                             0551   299         $GET    RAB=R8,ERR=REPORT_ERROR ; gets 2nd record
                             0560   300         $WAIT   RAB=R7
                             0569   301         $WAIT   RAB=R8                  ; wait them out
            04 A8      D4    0572   302         CLRL    RAB$L_ROP(R8)           ; rop in r7 will be set soon
                             0575   303
                             0575   304 ;
                             0575   305 ; that's it for test 4
                             0575   306 ;
                             0575   307
```

```
                                      0575   309
                                      0575   310 ;
                                      0575   311 ; test 5
                                      0575   312 ; do threee tests,  getting the same record with each rab
                                      0575   313 ; with no locking, supposedly.
                                      0575   314 ; 1st, set the nlk bit. then the rlk bit. and then let fac=get
                                      0575   315 ; all these should cause no locking
                                      0575   316 ;
                                      0575   317
       04 A7   00100000 8F      D0    0575   318          MOVL     #RAB$M_NLK,RAB$L_ROP(R7)
            FCE6 CF     01      D0    057D   319          MOVL     #1,KEY
                      012D      30    0582   320          BSBW     GET_TWO                    ; try to get the record twice
       04 A7   00080000 8F      D0    0585   321          MOVL     #RAB$M_RLK,RAB$L_ROP(R7)
       04 A8   00100000 8F      D0    058D   322          MOVL     #RAB$M_NLK,RAB$L_ROP(R8); can't try to lock record
                 FCCF CF        D6    0595   323          INCL     KEY
                      0116      30    0599   324          BSBW     GET_TWO
       00018021 8F    08 A8     D1    059C   325          CMPL     RAB$L_STS(R8),#RMS$_OK_RLK
                        15      13    05A4   326          BEQL     RLK_OK2
                                      05A6   327          FIELD    <RMS SUCCESS CODE>
                                      05BB   328 RLK_OK2:
       04 A7   00010000 8F      D0    05BB   329          MOVL     #RAB$M_LOC,RAB$L_ROP(R7); locate mode
       04 A8   00010000 8F      D0    05C3   330          MOVL     #RAB$M_LOC,RAB$L_ROP(R8); diito
            FA46 CF     02      90    05CB   331          MOVB     #FAB$M_GET,FAB$B_FAC+LOCK_FAB; only get access
                                      05D0   332
                                      05D0   333 ;
                                      05D0   334 ; have to reopen file w/ fac=get
                                      05D0   335 ;
                                      05D0   336
                                      05D0   337          $DISCONNECT      RAB=R7,ERR=REPORT_ERROR
                    FA1E'       30    05DF   338          BSBW     ERR
                                      05E2   339          $DISCONNECT      RAB=R8,ERR=REPORT_ERROR
                    FA0C'       30    05F1   340          BSBW     ERR
                                      05F4   341          $CLOSE   FAB=LOCK_FAB,ERR=REPORT_ERROR
                    F9F8'       30    0605   342          BSBW     ERR
                                      0608   343          $OPEN    FAB=LOCK_FAB,ERR=REPORT_ERROR
                    F9E4'       30    0619   344          BSBW     ERR
                                      061C   345          $CONNECT         RAB=R7,ERR=REPORT_ERROR
                    F9D2'       30    062B   346          BSBW     ERR
                                      062E   347          $CONNECT         RAB=R8,ERR=REPORT_ERROR
                    F9C0'       30    063D   348          BSBW     ERR
                 FC24 CF        D7    0640   349          DECL     KEY
                      006B      30    0644   350          BSBW     GET_TWO
                                      0647   351
                                      0647   352 ;
                                      0647   353 ; clean up
                                      0647   354 ;
                                      0647   355
              1E A7     00      90    0647   356          MOVB     #RAB$C_SEQ,RAB$B_RAC(R7)
              1E A8     00      90    064B   357          MOVB     #RAB$C_SEQ,RAB$B_RAC(R8)
                     04 A7      D4    064F   358          CLRL     RAB$L_ROP(R7)
       04 A8   00040000 8F      D0    0652   359          MOVL     #RAB$M_ULK,RAB$L_ROP(R8)
            F9B7 CF     01      C8    065A   360          BISL     #FAB$M_PUT,FAB$B_FAC+LOCK_FAB
            F9B2 CF     08      C8    065F   361          BISL     #FAB$M_UPD,FAB$B_FAC+LOCK_FAB
                                      0664   362
                                      0664   363 ;
                                      0664   364 ;        close up
                                      0664   365 ;
```

```
                  0664   366
                  0664   367          $DISCONNECT     RAB=R7,ERR=REPORT_ERROR
      F98A'   30  0673   368          BSBW    ERR
                  0676   369          $DISCONNECT     RAB=R8,ERR=REPORT_ERROR
      F978'   30  0685   370          BSBW    ERR
                  0688   371          $CLOSE  FAB=LOCK_FAB,ERR=REPORT_ERROR
      F964'   30  0699   372          BSBW    ERR
                  069C   373          FINISH  <LOCKING TESTS>
              05  06B1   374          RSB
```

```
                                  06B2     376 GET_TWO:
                                  06B2     377
                                  06B2     378 ;
                                  06B2     379 ; subroutine to get the same record w/ two rabs
                                  06B2     380 ; and then compare the results
                                  06B2     381 ;
                                  06B2     382
                                  06B2     383         $GET    RAB=R7,ERR=REPORT_ERROR
                        F93C'  30 06C1     384         BSBW    ERR
                                  06C4     385         $GET    RAB=R8,ERR=REPORT_ERROR
                        F92A'  30 06D3     386         BSBW    ERR
                 22 A8  22 A7  B1 06D6     387         CMPW    RAB$W_RSZ(R7),RAB$W_RSZ(R8)
                           15  13 06DB     388         BEQL    RSZ_OK
                                  06DD     389         FIELD   <RSZ>
                                  06F2     390 RSZ_OK:
         28 B8  28 B7  22 A7   29 06F2     391         CMPC3   RAB$W_RSZ(R7),@RAB$L_RBF(R7),@RAB$L_RBF(R8)
                           15  13 06F9     392         BEQL    REC_OK
                                  06FB     393         FIELD   <RECORDS>
                               05 0710     394 REC_OK: RSB
                                  0711     395         .END
```

```
$$.PSECT_EP                = 00000000              RAB$C_SEQ                = 00000000    D
$$.TAB                     = 00000094 R   D   01   RAB$L_BKT                = 00000038    D
$$.TABEND                  = 000000D8 R   D   01   RAB$L_CTX                = 00000018    D
$$.TMP                     = 00040000     D        RAB$L_RBF                = 00000028    D
$$.TMP1                    = 00000002     D        RAB$L_ROP                = 00000004    D
$$.TMP2                    = 00000058              RAB$L_STS                = 00000008    D
$$.TMPX                    = 000000C7 R   D   04   RAB$M_ASY                = 00000001    D
$$.TMPX1                   = 00000007     D        RAB$M_LOC                = 00010000    D
$$RMSTEST                  = 0000001E              RAB$M_NLK                = 00100000    D
$$RMS_PBUGCHK              = 00000010              RAB$M_RLK                = 00080000    D
$$RMS_TBUGCHK              = 00000008              RAB$M_ULK                = 00040000    D
$$RMS_UMODE                = 00000004              RAB$V_ULK                = 00000012    D
BEGPUT                       ********     X   01   RAB$W_RFA                = 00000010    D
BEG_DESCR                    ********     X   01   RAB$W_RSZ                = 00000022    D
BKT_OK                       000003B2 R   D   01   RAB1                       00000050 R  D  01
ERR                          ********     X   01   RAB2                       00000094 R  D  01
FAB$B_DNS                  = 00000035     D        REC_OK                     00000710 R  D  01
FAB$B_FAC                  = 00000016     D        REPORT_ERR                 ********   X  01
FAB$C_BID                  = 00000003     D        REPORT_ERROR               ********   X  01
FAB$C_BLN                  = 00000050     D        RLK_OK                     000004E1 R  D  01
FAB$C_SEQ                  = 00000000     D        RLK_OK2                    000005BB R  D  01
FAB$C_VAR                  = 00000002     D        RMS$_OK_ALK              = 00018039    D
FAB$L_ALQ                  = 00000010     D        RMS$_OK_RLK              = 00018021    D
FAB$L_DNA                  = 00000030     D        RMS$_RLK                 = 000182AA    D
FAB$L_FOP                  = 00000004     D        RMS$_RNL                 = 000181A0    D
FAB$M_GET                  = 00000002     D        RMT$TEST_5A                0000026C RG D  01
FAB$M_PUT                  = 00000001     D        RSZ_OK                     000006F2 R  D  01
FAB$M_UPD                  = 00000008     D        STS_OK                     00000531 R  D  01
FAB$V_CHAN_MODE            = 00000002     D        SYS$CLOSE                  ********  GX  01
FAB$V_FILE_MODE            = 00000004     D        SYS$CONNECT                ********  GX  01
FAB$V_GET                  = 00000001     D        SYS$DISCONNECT             ********  GX  01
FAB$V_LNM_MODE             = 00000000     D        SYS$FIND                   ********  GX  01
FAB$V_MSE                  = 00000004     D        SYS$FREE                   ********  GX  01
FAB$V_PUT                  = 00000000     D        SYS$GET                    ********  GX  01
FAB$V_UPD                  = 00000003     D        SYS$OPEN                   ********  GX  01
FAB$W_GBC                  = 00000048     D        SYS$RELEASE                ********  GX  01
FINPUT                       ********     X   01   SYS$UPDATE                 ********  GX  01
FIN_DESCR                    ********     X   01   SYS$WAIT                   ********  GX  01
FLDPUT                       ********     X   01   T5START                    00000000 RG D  01
FLD_DESCR                    ********     X   01
GET_TWO                      000006B2 R   D   01
KEY                          00000268 R   D   01
LCK2BSZ                    = 000000C8   G D
LCK2BUF                      000001A0 RG  D   01
LCKBSZ                     = 000000C8   G D
LCKBUF                       000000D8 RG  D   01
LOCK_FAB                     00000000 RG  D   01
NAMBLK                       ********     X   01
OK_ALK                       000003FE R   D   01
OK_ALK1                      00000455 R   D   01
OK_LCK                       0000030B R   D   01
OK_LCK1                      00000373 R   D   01
OK_RNL                       0000048C R   D   01
RAB$B_RAC                  = 0000001E     D
RAB$C_BID                  = 00000001     D
RAB$C_BLN                  = 00000044     D
RAB$C_KEY                  = 00000001     D
```

```
                                             +-----------------+
                                             ! 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                       00000711  ( 1809.)  01 (   1.)  NOPIC    USR    CON    REL    GBL NOSHR   EXE   RD     WRT NOVEC LONG
$ABS$                           00000000  (    0.)  02 (   2.)  NOPIC    USR    CON    ABS    LCL NOSHR   EXE   RD     WRT NOVEC BYTE
$RMSNAM                         0000000F  (   15.)  03 (   3.)  NOPIC    USR    CON    REL    LCL NOSHR   EXE   RD     WRT NOVEC BYTE
__$RMSNAM                       000000CE  (  206.)  04 (   4.)  NOPIC    USR    CON    REL    LCL NOSHR   EXE   RD     WRT NOVEC BYTE

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

Phase                    Page faults    CPU Time       Elapsed Time
-----                    -----------    --------       ------------
Initialization                   38    00:00:00.09    00:00:00.55
Command processing              133    00:00:00.62    00:00:02.22
Pass 1                          268    00:00:09.14    00:00:21.78
Symbol table sort                 0    00:00:00.56    00:00:01.13
Pass 2                           85    00:00:02.24    00:00:05.24
Symbol table output              12    00:00:00.09    00:00:00.21
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            541    00:00:12.78    00:00:31.17

The working set limit was 1350 pages.
49361 bytes (97 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 573 non-local and 0 local symbols.
395 source lines were read in Pass 1, producing 49 object records in Pass 2.
40 pages of virtual memory were used to define 34 macros.

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

Macro library name                         Macros defined
------------------                         --------------
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                     0
_$255$DUA28:[SYSLIB]STARLET.MLB;2                 21
TOTALS (all libraries)                            21

801 GETS were required to define 21 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:RMSTEST5/OBJ=OBJ$:RMSTEST5 MSRC$:RMSTEST5/UPDATE=(ENH$:RMSTEST5)+EXECML$/LIB
```

0409 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

RMSTEST4
LIS

RMSTEST7
LIS

RMSTEST8
LIS

RMSTESTR
LIS

RMSTEST0
LIS

RMSTEST2
LIS

RMSTEST3
LIS

RMSTEST5
LIS

RMSTEST6
LIS

RMSTESTT
LIS

RMSTEST1
LIS

SATSSF17
LIS