```
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPP
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPPP
UUU         UUU  EEEEEEEEEEEEEEE  TTTTTTTTTTTTTTT  PPPPPPPPPPPPP
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        PPPPPPPPPPPPP
UUU         UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPPP
UUU         UUU  EEEEEEEEEEEE           TTT        PPPPPPPPPPPPP
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
```

```
RRRRRRRR    MM      MM    SSSSSSSS  TTTTTTTTTT  EEEEEEEEEE    SSSSSSSS  TTTTTTTTTT      11
RRRRRRRR    MM      MM    SSSSSSSS  TTTTTTTTTT  EEEEEEEEEE    SSSSSSSS  TTTTTTTTTT      11
RR      RR  MMMM  MMMM  SS              TT      EE          SS              TT        1111
RR      RR  MMMM  MMMM  SS              TT      EE          SS              TT        1111
RR      RR  MM  MM  MM  SS              TT      EE          SS              TT          11
RR      RR  MM  MM  MM  SS              TT      EE          SS              TT          11
RRRRRRRR    MM      MM    SSSSSS        TT      EEEEEEEE      SSSSSS        TT          11
RRRRRRRR    MM      MM    SSSSSS        TT      EEEEEEEE      SSSSSS        TT          11
RR  RR      MM      MM          SS      TT      EE                  SS    TT          11
RR  RR      MM      MM          SS      TT      EE                  SS    TT          11
RR    RR    MM      MM          SS      TT      EE                  SS    TT          11      ....
RR    RR    MM      MM          SS      TT      EE                  SS    TT          11      ....
RR      RR  MM      MM  SSSSSSSS        TT      EEEEEEEEEE  SSSSSSSS      TT      111111    ....
RR      RR  MM      MM  SSSSSSSS        TT      EEEEEEEEEE  SSSSSSSS      TT      111111    ....

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     64          $BEGIN  RMSTEST1,009,__RMSTEST,<GENERAL RMS TEST PROGRAM>,<GBL,LONG>
0000     65
0000     66 ;
0000     67
0000     68          .ENABL  DBG
0000     69
0000     70 ;
0000     71 ;  this program tests the sequential file org, for a disk device,
0000     72 ;  with vfc record format.
0000     73 ;
0000     74 ;
0000     75 ;  test 1a:
0000     76 ;
0000     77 ;       create known test file of 1000 records, where each record has
0000     78 ;       a 4-byte fixed control field giving the record # and a variable
0000     79 ;       length portion of n mod 100 bytes of the ascii character n mod 10
0000     80 ;       (where 'n' is the record number).
0000     81 ;
0000     82
0000     83          $RMSDEF
0000     84          .NLIST  MEB
0000     85
0000     86 ;
0000     87 ;  macros:
0000     88 ;
0000     89
0000     90          .MACRO  BUFF NAM,SIZE
0000     91 NAM'BUF::
0000     92          .BLKB   SIZE
0000     93          NAM'BSZ==SIZE
0000     94          .ENDM   BUFF
0000     95
0000     96 ;
0000     97 ;
0000     98 ;
0000     99
0000    100          .MACRO  TYPE STRING, ?L
0000    101          STORE   <STRING>
0000    102          BLBC    VERBOSITY,L
0000    103          MOVL    #$$.TMPX,CMDORAB+RAB$L_RBF
0000    104          MOVW    #$$.TMPX1,CMDORAB+RAB$W_RSZ
0000    105          $PUT    RAB=CMDORAB,ERR=REPORT_ERROR
0000    106          BSBW    ERR
0000    107 L:
0000    108          .ENDM   TYPE
0000    109
0000    110 ;
0000    111
0000    112          .MACRO  WTTYPE STRING
0000    113          $WAIT   RAB=CMDORAB
0000    114          TYPE    <STRING>
0000    115          .ENDM   WTTYPE
0000    116
0000    117          .MACRO  WFIELD STRING
0000    118          $WAIT   RAB=CMDORAB
0000    119          FIELD   <STRING>
0000    120          .ENDM
```

```
0000    121
0000    122 ;
0000    123
```

```
0000    125          .MACRO  STORE STRING,PRE
0000    126          .SAVE
0000    127          .PSECT  __$RMSNAM
0000    128          $$.TMPX=.
0000    129          PRE                                     ; store any carriage control info
0000    130          .ASCII  %STRING%
0000    131          $$.TMPX1=.-$$.TMPX
0000    132          .RESTORE
0000    133          .ENDM   STORE
0000    134
0000    135 ;
0000    136
0000    137          .MACRO  BEGIN TSTNAM
0000    138          STORE   <TSTNAM>
0000    139          MOVL    #$$.TMPX,BEG_DESCR+4    ; addr
0000    140          MOVL    #$$.TMPX1,BEG_DESCR     ; len
0000    141          BSBW    BEGPUT
0000    142          .ENDM   BEGIN
0000    143          .MACRO  FINISH TSTNAM
0000    144          STORE   <TSTNAM>
0000    145          MOVL    #$$.TMPX,FIN_DESCR+4    ; addr
0000    146          MOVL    #$$.TMPX1,FIN_DESCR     ; len
0000    147          BSBW    FINPUT
0000    148          .ENDM   FINISH
0000    149          .MACRO  FIELD FLDNAM
0000    150          STORE   <FLDNAM>
0000    151          MOVL    #$$.TMPX,FLD_DESCR+4    ; addr
0000    152          MOVL    #$$.TMPX1,FLD_DESCR     ; len
0000    153          BSBW    FLDPUT
0000    154          .ENDM   FIELD
0000    155          .MACRO  MBPT, ?L
0000    156          BLBC    VERBOSITY,L
0000    157          BPT
0000    158 L:
0000    159          .ENDM   MBPT
0000    160
0000    161 ;
0000    162
```

```
                                00000000   164          .PSECT   RMSTEST,GBL,LONG
                                    0000   165          .ALIGN  LONG
                                    0000   166 T1START::
                                    0000   167 WTRAB:: $RAB
                           00000364 0044   168 RFATBL: .BLKQ   100
                                    0364   169 T1FAB:: $FAB    FAC=PUT,FNM=<TST$DISK:T1FILE.DAT;1>,org=seq,rfm=vfc,-
                                    0364   170          RAT=CR,FSZ=4,MRS=100,NAM=NAMBLK,FOP=<SUP,CTG>,-
                                    0364   171          ALQ=48,DEQ=12,SHR=<PUT,GET,UPI>
                                    03B4   172 T1RAB:: $RAB    FAB=T1FAB,UBF=CPYBUF,USZ=CPYBSZ,RBF=CPYBUF,MBC=4,MBF=2,-
                                    03B4   173          ROP=<WBH>,RHB=RECCNT,KBF=RECCNT
                                    03F8   174 RECCNT::
                   00000000 00000000 03F8   175          .LONG   0,0
                                    0400   176
                                    0400   177 ;
                                    0400   178
                   00000408'00000025' 0400   179 T1STR:  .LONG   T1L,T1S
44 52 4F 43 45 52 20 2E 4C 55 34 21 0408   180 T1S:    .ASCII  '!4UL. RECORDS WRITTEN.  RFA = !XL,!XW'
20 20 2E 4E 45 54 54 49 52 57 20 53 0414
58 21 2C 4C 58 21 20 3D 20 41 46 52 0420
                                 57 042C
                           00000025 042D   181          T1L=.-T1S
                                    042D   182
                                    042D   183 ;
                                    042D   184
                   00000435'0000005F' 042D   185 T2STR:  .LONG   T2L,T2S
20 3D 20 23 20 44 52 4F 43 45 52 20 0435   186 T2S:    .ASCII  ' RECORD # = !4UL, RHB = !4UL!/'
20 3D 20 42 48 52 20 2C 4C 55 34 21 0441
                  2F 21 4C 55 34 21 044D
4C 55 33 21 20 3D 20 45 5A 49 53 20 0453   187          .ASCII  ' SIZE = !3UL, RSZ = !UW!/'
21 57 55 21 20 3D 20 5A 53 52 20 2C 045F
                                 2F 046B
3D 20 41 46 52 20 44 45 56 41 53 20 046C   188          .ASCII  ' SAVED RFA = !XL,!XW, FILE RFA = !XL,!XW'
49 46 20 2C 57 58 21 2C 4C 58 21 20 0478
4C 58 21 20 3D 20 41 46 52 20 45 4C 0484
                        57 58 21 2C 0490
                           0000005F 0494   189          T2L=.-T2S
                                    0494   190
                                    0494   191 ;
                                    0494   192
                   0000049C'0000000F' 0494   193 T3STR:  .LONG   T3L,T3S
46 41 21 27 3D 20 44 52 4F 43 45 52 049C   194 T3S:    .ASCII  "RECORD ='!AF'!/"
                  2F 21 27 04A8
                           0000000F 04AB   195          T3L=.-T3S
                                 01 04AB   196 RHBSW:  .BYTE   1                        ; switch for modifying rhb contents
```

```
                                04AC   198 RMT$TEST_1A::
                           0FFC 04AC   199         .WORD   ^M<R2,R3,R4,R5,R6,R7,R8,R9,R10,R11>
                                04AE   200         BEGIN   <SEQUENTIAL TESTS>
        FB36 CF   00000002'EF   B0 04C3   201         MOVW    CMDORAB+RAB$W_ISI,WTRAB+RAB$W_ISI; copy output isi for
                                04CC   202
                                04CC   203 ;
                                04CC   204 ;DIFFERENT RAB $WAIT TEST
                                04CC   205 ;
                                04CC   206
        FE93 CF   00100000 8F   CA 04CC   207         BICL2   #FAB$M_CTG,T1FAB+FAB$L_FOP; don't want ctg set
             00000004'EF    01   C8 04D5   208         BISL2   #RAB$M_ASY,CMDORAB+RAB$L_ROP
                 FF17 CF    01   D0 04DC   209         MOVL    #1,RECCNT
                 58   FB5F CF   DE 04E1   210         MOVAL   RFATBL,R8
                                04E6   211         $CREATE FAB=T1FAB,ERR=REPORT_ERROR
                       FB06'    30 04F7   212         BSBW    ERR
                 5B    FEB6 CF   DE 04FA   213         MOVAL   T1RAB,R11
                                04FF   214         $CONNECT        RAB=R11,ERR=REPORT_ERROR
                       FAEF'    30 050E   215         BSBW    ERR
                                0511   216
                                0511   217 ;
                                0511   218
    52  FEDE CF   00000064 8F   7B 0511   219 NXTREC: EDIV    #100,RECCNT,R2,R6       ; compute record length
                           56       051B
        57  52  FED7 CF    0A   7B 051C   220         EDIV    #10,RECCNT,R2,R7        ; compute character for record
                     57    30   80 0523   221         ADDB    #48,R7                  ;  make it ascii
                   22 AB    56   B0 0526   222         MOVW    R6,RAB$W_RSZ(R11)       ; tell rms the record size
            56  57   6E    00   2C 052A   223         MOVC5   #0,(SP),R7,R6,CPYBUF    ; fill record with character
                 00000000'EF         052F
                                0534   224         $PUT    RAB=R11,ERR=REPORT_ERROR
                       FABA'    30 0543   225         BSBW    ERR
                     30   57    D1 0546   226         CMPL    R7,#48                  ; is this 10th record?
                          0B    12 0549   227         BNEQ    T1CNT                   ; branch if not
                   88   10 AB   7D 054B   228         MOVQ    RAB$W_RFA(R11),(R8)+    ; save rfa in table
                          56    D5 054F   229         TSTL    R6                      ; is this 100th record?
                          03    12 0551   230         BNEQ    T1CNT
                        093F    30 0553   231         BSBW    TYPRFA                  ; type out the rfa
                                0556   232
                                0556   233 ;
                                0556   234
FFB1 FE9A CF    01    03E8 8F   3D 0556   235 T1CNT:  ACBW    #1000,#1,RECCNT,NXTREC
                                0560   236         $CLOSE  FAB=T1FAB,ERR=REPORT_ERROR
                       FA8C'    30 0571   237         BSBW    ERR
                     FE3E CF    B4 0574   238         CLRW    T1RAB+RAB$W_ISI         ; allow rab to be re-used
                                0578   239         $WAIT   CMDORAB
                                0585   240         TYPE    <FINISHED TEST 1A - CREATED TEST FILE>
```

```
                                05B4    242
                                05B4    243 ;
                                05B4    244 ;  test 1b
                                05B4    245 ;
                                05B4    246 ;       re-read file created in test 1a and try random and sequential
                                05B4    247 ;       access via $get and $find.
                                05B4    248 ;
                                05B4    249
                                05B4    250         $WAIT   RAB=CMDORAB
                                05C1    251         TYPE    <BEGINNING TEST 1B - $GET AND $FIND>
      FD6F CF    01000000 8F   C8 05F0    252         BISL2   #FAB$M_NAM,T1FAB+FAB$L_FOP
           FD7C CF    02      88 05F9    253         BISB2   #FAB$M_GET,T1FAB+FAB$B_FAC
                                05FE    254         $OPEN   FAB=T1FAB,ERR=REPORT_ERROR
                     F9EE'      30 060F    255         BSBW    ERR
                                0612    256
                                0612    257 ;
                                0612    258 ;  \eventually include code here to verify file attributes.\
                                0612    259 ;
                                0612    260
        04 AB    00010000 8F   D0 0612    261         MOVL    #RAB$M_LOC,RAB$L_ROP(R11); locate mode
                                061A    262         $CONNECT        RAB=T1RAB,ERR=REPORT_ERROR
                     F9D2'      30 062B    263         BSBW    ERR
             1E AB     02      90 062E    264         MOVB    #RAB$C_RFA,RAB$B_RAC(R11); rfa access
           59    03E8 8F      3C 0632    265         MOVZWL  #1000,R9                ; starting record number
                       5A      D4 0637    266         CLRL    R10                     ;  as a quadword for ediv
                       57      D4 0639    267         CLRL    R7                      ; flag for 1st pass
                                063B    268
                                063B    269 ;
                                063B    270
                                063B    271 T1BLOOP1:
                       58   08 C2 063B    272         SUBL    #8,R8                   ; move to previous rfa
                          06C5 30 063E    273         BSBW    GETANDCHK
      FFF2 59    FFF6 8F    0A 3D 0641    274         ACBW    #10,#-10,R9,T1BLOOP1
                       59   0A D0 0649    275         MOVL    #10,R9                  ; start with record # 10
                                064C    276
                                064C    277 ;
                                064C    278
                                064C    279 T1BLOOP2:
                          06B7 30 064C    280         BSBW    GETANDCHK
                       58   08 C0 064F    281         ADDL    #8,R8                   ; move to next rfa
      FFF2 59    0A    03DE 8F   3D 0652    282         ACBW    #990,#10,R9,T1BLOOP2
                                065A    283
                                065A    284 ;
                                065A    285
                    43 57    00 E2 065A    286         BBSS    #0,R7,T1BDONE           ; branch if 2nd pass
                                065E    287         $WAIT   CMDORAB
                                066B    288         TYPE    <PASS 1 O.K.>
                    59   03DE 8F   3C 069A    289         MOVZWL  #990,R9                 ; start with rec # 990
                           9A   11 069F    290         BRB     T1BLOOP1
                                06A1    291 T1BDONE:
                                06A1    292         $WAIT   CMDORAB
                                06AE    293         TYPE    <PASS 2 O.K.>
                                06DD    294         $CLOSE  FAB=T1FAB,ERR=REPORT_ERROR
                     F90F'      30 06EE    295         BSBW    ERR
```

```
                        06F1    297
                        06F1    298 ;
                        06F1    299 ;  test 1c
                        06F1    300 ;
                        06F1    301 ;  use update to modify records created in test 1a
                        06F1    302 ;
                        06F1    303
                        06F1    304          WTTYPE  <START TEST 1C  -  $UPDATE>
          FC48 CF   08   88     072D    305          BISB2   #FAB$M_UPD,T1FAB+FAB$B_FAC
                        0732    306          $OPEN   FAB=T1FAB,ERR=REPORT_ERROR
                F8BA'   30      0743    307          BSBW    ERR
              02 AB     B4      0746    308          CLRW    RAB$W_ISI(R11)
                        0749    309          $CONNECT        RAB=R11,ERR=REPORT_ERROR
                F8A5'   30      0758    310          BSBW    ERR
            1E AB   02  90      075B    311          MOVB    #RAB$C_RFA,RAB$B_RAC(R11)
              59    0A  D0      075F    312          MOVL    #10,R9                  ; start at record 10
          58  F8DE CF  DE      0762    313          MOVAL   RFATBL,R8
                57  D4          0767    314          CLRL    R7                      ; get single record
                        0769    315
                        0769    316 ;
                        0769    317
                        0769    318 T1BLOOP4:
                059A    30      0769    319          BSBW    GETANDCHK
              FC88 CF   D6      076C    320          INCL    RECCNT                  ; modify rhb
    FD34 CF   FD37 CF   8E      0770    321          MNEGB   RHBSW,RHBSW             ; toggle rhb modify flag
                03      19      0777    322          BLSS    10$
              2C AB     D4      0779    323          CLRL    RAB$L_RHB(R11)          ; don't modify rhb (default)
            55  28 AB   D0      077C    324 10$:     MOVL    RAB$L_RBF(R11),R5       ; get record addr
            54  22 AB   3C      0780    325          MOVZWL  RAB$W_RSZ(R11),R4       ; and len
                85      96      0784    326 20$:     INCB    (R5)+                   ; modify record contents
              FB 54     F5      0786    327          SOBGTR  R4,20$
                        0789    328          $UPDATE RAB=R11,ERR=REPORT_ERROR
                F865'   30      0798    329          BSBW    ERR
        2C AB   FC59 CF   DE    079B    330          MOVAL   RECCNT,RAB$L_RHB(R11)   ; restore rhb addr
                58  08  C0      07A1    331          ADDL    #8,R8                   ; bump to next rfa entry
    FFBD 59     0A  03E8 8F   3D    07A4    332          ACBW    #1000,#10,R9,T1BLOOP4   ; modify every 10th record
                        07AC    333
                        07AC    334 ;
                        07AC    335 ;  now reread the modified file and check that every 10th (and only
                        07AC    336 ;  every 10th) record has been correctly modified
                        07AC    337 ;
                        07AC    338
          58    F894 CF   DE    07AC    339          MOVAL   RFATBL,R8
                59  D4          07B1    340          CLRL    R9                      ; reset record #
            1E AB   00  90      07B3    341          MOVB    #RAB$C_SEQ,RAB$B_RAC(R11)
                        07B7    342
                        07B7    343 ;
                        07B7    344 ;       $disconnect     r11
                        07B7    345 ;       bsbw    err
                        07B7    346 ;       $connect        r11             ;do an effective rewind
                        07B7    347 ;       bsbw    err
                        07B7    348 ;
                        07B7    349
                        07B7    350          $REWIND RAB=R11,ERR=REPORT_ERROR; so do a rewind
                F837'   30      07C6    351          BSBW    ERR
                        07C9    352
                        07C9    353 ;
```

```
                          07C9    354
                          07C9    355 T1BLOOP5:
                    57    D6    07C9    356          INCL    R7                      ; set switch for no rfa compare
                    09    DD    07CB    357          PUSHL   #9                      ; # gets of unmodified recs
                    59    D6    07CD    358 10$:     INCL    R9                      ; bump rec #
                                07CF    359          $GET    RAB=R11,ERR=REPORT_ERROR
                 F81F'    30    07DE    360          BSBW    ERR
                 057A     30    07E1    361          BSBW    CHKREC
                E6 6E     F5    07E4    362          SOBGTR  (SP),10$
                   8E     D5    07E7    363          TSTL    (SP)+
                   59     D6    07E9    364          INCL    R9                      ; bump rec #
                                07EB    365          $GET    RAB=R11,ERR=REPORT_ERROR
                 F803'    30    07FA    366          BSBW    ERR
                   57     D4    07FD    367          CLRL    R7                      ; specify rfa to be checked
FCA5 CF    FCA8 CF     8E    07FF    368          MNEGB   RHBSW,RHBSW             ; toggle rhb modified switch
                   04     19    0806    369          BLSS    15$
           FBEC CF     D6    0808    370          INCL    RECCNT
50    FBE7 CF     59    C3    080C    371 15$:     SUBL3   R9,RECCNT,R0            ; rhb contents = rec # + 1?
                   50     D7    0812    372          DECL    R0
                   03     13    0814    373          BEQL    20$                     ; branch if yes
                 0590     30    0816    374          BSBW    BADRHB                  ; report error
                 0549     30    0819    375 20$:     BSBW    CHKRC1                  ; check record len, rfa, and contents
              58   08     C0    081C    376          ADDL    #8,R8                   ; bump rfa table addr
        03E8  8F   59     B1    081F    377          CMPW    R9,#1000                ; done?
                   A3     12    0824    378          BNEQ    T1BLOOP5                ; branch if not
                                0826    379          $CLOSE  FAB=T1FAB,ERR=REPORT_ERROR
                 F7C6'    30    0837    380          BSBW    ERR
```

```
                                 083A   382
                                 083A   383 ;
                                 083A   384 ;  test 2  -  random i/o test for sequential file org
                                 083A   385 ;
                                 083A   386 ;  test 2a
                                 083A   387 ;
                                 083A   388 ;  create a known test file of 1000 records of fixed length = 49. bytes,
                                 083A   389 ;  1st longword has the record # (n) followed by 45 bytes of the ascii
                                 083A   390 ;  character (n mod 42) + 48.
                                 083A   391 ;
                                 083A   392
                                 083A   393          WTTYPE  <START TEST 2A  -  RANDOM SEQ I/O>
                                 0876   394 T2SETUP:
                      02 AB   B4 0876   395          CLRW    RAB$W_ISI(R11)
              28 AB   00000000'EF DE 0879   396          MOVAL   CPYBUF,RAB$L_RBF(R11)
                      22 AB   31 9B 0881   397          MOVZBW  #49,RAB$W_RSZ(R11)
                   5A   FADB CF DE 0885   398          MOVAL   T1FAB,R10
                      10 AA   30 9A 088A   399          MOVZBL  #48,FAB$L_ALQ(R10)
              04 AA   01000000 8F CA 088E   400          BICL2   #FAB$M_NAM,FAB$L_FOP(R10)
                         16 AA   94 0896   401          CLRB    FAB$B_FAC(R10)          ; check for put default
                      1F AA   01 90 0899   402          MOVB    #FAB$C_FIX,FAB$B_RFM(R10)
                      36 AA   31 B0 089D   403          MOVW    #49,FAB$W_MRS(R10)      ; rec len
                                 08A1   404          $RAB_STORE      RAB=(R11),ROP=<LOC,UIF>
                                 08A9   405          $CREATE FAB=R10,ERR=REPORT_ERROR
                            F745' 30 08B8   406          BSBW    ERR
            FAA8 CF   00010631 8F D1 08BB   407          CMPL    #RMS$_SUPERSEDE,T1FAB+FAB$L_STS
                         15 13 08C4   408          BEQL    SUPOK
                                 08C6   409          FIELD   <STATUS WORD IS NOT SUPERSEDE, THEREFORE IT>
                                 08DB   410 SUPOK:   $CONNECT        RAB=R11,ERR=REPORT_ERROR
                        F713' 30 08EA   411          BSBW    ERR
                                 08ED   412
                                 08ED   413 ;
                                 08ED   414 ;  pre-extend file on 1st pass, put sequentially on second pass
                                 08ED   415 ;
                                 08ED   416
                   1F 1E AA   03 E0 08ED   417          BBS     #FAB$V_BLK,FAB$B_RAT(R10),10$; branch if pass 2
               FAFF CF   03E8 8F 3C 08F2   418          MOVZWL  #1000,RECCNT
                      1E AB   01 90 08F9   419          MOVB    #RAB$C_KEY,RAB$B_RAC(R11)
                                 08FD   420          $PUT    RAB=R11,ERR=REPORT_ERROR
                            F6F1' 30 090C   421          BSBW    ERR
                               04 11 090F   422          BRB     20$
                      1E AB   00 90 0911   423 10$:     MOVB    #RAB$C_SEQ,RAB$B_RAC(R11)
                   FADE CF   01 3C 0915   424 20$:     MOVZWL  #1,RECCNT
                                 091A   425
                                 091A   426 ;
                                 091A   427
         57   52   FAD9 CF   2A 7B 091A   428 NXTRC2: EDIV    #42,RECCNT,R2,R7        ; compute char for record
                         57   30 80 0921   429          ADDB2   #48,R7                  ; make it ascii
          00000000'EF   FAD0 CF   D0 0924   430          MOVL    RECCNT,CPYBUF           ; insert rec #
              2D   57   6E   00 2C 092D   431          MOVC5   #0,(SP),R7,#45,CPYBUF+4 ; fill rec with char
                   00000004'EF      0932
                                 0937   432          $PUT    RAB=R11,ERR=REPORT_ERROR
                            F6B7' 30 0946   433          BSBW    ERR
                            05A1 30 0949   434          BSBW    CHKRFA
                                 094C   435
                                 094C   436 ;
                                 094C   437 ;  print message every 100 records
```

```
                                094C    438 ;
                                094C    439
   52    FAA3 CF    00000064 8F   7B  094C    440          EDIV    #100,RECCNT,R2,R7
                                  57  0956
                                  57  D5  0957    441          TSTL    R7
                                  03  12  0959    442          BNEQ    T2CNT
                               0537   30  095B    443          BSBW    TYPRFA
                                      095E    444
                                      095E    445 ;
                                      095E    446
   FFB2 FA92 CF    01    03E8 8F   3D  095E    447 T2CNT:   ACBW    #1000,#1,RECCNT,NXTRC2
                                      0968    448          $CLOSE  FAB=R10,ERR=REPORT_ERROR
                                F686' 30  0977    449          BSBW    ERR
                              02 AB   B4  097A    450          CLRW    RAB$W_ISI(R11)
                                      097D    451          WTTYPE  <END STEP 2A  -  START STEP 2B>
```

```
                                09B9    453
                                09B9    454 ;
                                09B9    455 ;  test 2b
                                09B9    456 ;
                                09B9    457 ;  reread file created in step 2a and try random and sequential
                                09B9    458 ;  access via $get
                                09B9    459 ;
                                09B9    460
          16 AA    09    90     09B9    461          MOVB    #FAB$M_PUT!FAB$M_UPD,FAB$B_FAC(R10); upd implies get access
    04 AA   01000000 8F    C8  09BD    462          BISL2   #FAB$M_NAM,FAB$L_FOP(R10)
                                09C5    463          $OPEN   FAB=R10,ERR=REPORT_ERROR
                    F629'  30   09D4    464          BSBW    ERR
                                09D7    465
                                09D7    466 ;
                                09D7    467 ;       movl    #rab$m_loc,rab$l_rop(r11)
                                09D7    468 ;
                                09D7    469
                                09D7    470          $RAB_STORE      RAB=(R11),ROP=<LOC,UIF>
          1E AB    01    90     09DF    471          MOVB    #RAB$C_KEY,RAB$B_RAC(R11)
                                09E3    472          $CONNECT        RAB=R11,ERR=REPORT_ERROR
                    F60B'  30   09F2    473          BSBW    ERR
    F9FC CF   03E8 8F    3C     09F5    474          MOVZWL  #1000,RECCNT
                                09FC    475
                                09FC    476 ;
                                09FC    477 ;  get all records in reverse order
                                09FC    478 ;
                                09FC    479
                     054D   30  09FC    480 10$:     BSBW    GTCHK2
FFF3 F9F1 CF   FFFF 8F    02    3D  09FF    481          ACBW    #2,#-1,RECCNT,10$
                                0A09    482
                                0A09    483 ;
                                0A09    484 ;  now get them all forward
                                0A09    485 ;
                                0A09    486
                      0540  30  0A09    487 20$:     BSBW    GTCHK2
FFF3 F9E4 CF   01     03E8 8F   3D  0A0C    488          ACBW    #1000,#1,RECCNT,20$
                                0A16    489
                                0A16    490 ;
                                0A16    491 ;  now get every 10th record in reverse order followed by the
                                0A16    492 ;  next 10 in sequential order
                                0A16    493 ;
                                0A16    494
          F9DB CF   03DE 8F     B0  0A16    495          MOVW    #990,RECCNT
                      052C  30  0A1D    496 30$:     BSBW    GTCHK2
                1E AB    00    90  0A20    497          MOVB    #RAB$C_SEQ,RAB$B_RAC(R11)
                   58    0A     D0  0A24    498          MOVL    #10,R8
                   F9CD  CF     D6  0A27    499 25$:     INCL    RECCNT
                      051E  30    0A2B    500          BSBW    GTCHK2
                   F6 58    F5     0A2E    501          SOBGTR  R8,25$
                1E AB    01    90  0A31    502          MOVB    #RAB$C_KEY,RAB$B_RAC(R11)
FFDE F9BB CF   FFEC 8F    01    3D  0A35    503          ACBW    #1,#-20,RECCNT,30$
                                0A3F    504
                                0A3F    505 ;
                                0A3F    506
                                0A3F    507          WTTYPE  <PASS 1 O.K.>
```

```
                                0A7B   509
                                0A7B   510 ;
                                0A7B   511 ;  do 10 random puts, changing record contents
                                0A7B   512 ;
                                0A7B   513
                     56   30    90 0A7B   514          MOVB    #^A/0/,R6                       ; updating character
             F974 CF    5B 8F    9A 0A7E   515          MOVZBL  #91,RECCNT
   00000000'EF    F970 CF       D0 0A84   516 40$:     MOVL    RECCNT,CPYBUF
         2D   56   6E    00      2C 0A8D   517          MOVC5   #0,(SP),R6,#45,CPYBUF+4 ; change the record
                00000004'EF             0A92
                                0A97   518          $PUT    RAB=R11,ERR=REPORT_ERROR
                      F557'     30 0AA6   519          BSBW    ERR
                       0441     30 0AA9   520          BSBW    CHKRFA
CC F942 CF    0065 8F    03E8 8F    3D 0AAC   521          ACBW    #1000,#101,RECCNT,40$
                                FF 0AB6
                       04EF     30 0AB8   522          BSBW    CHKMOD                          ; go verify changes
                                0ABB   523
                                0ABB   524 ;
                                0ABB   525 ;  do 10 random get/update pairs
                                0ABB   526 ;
                                0ABB   527
                     56   31    90 0ABB   528          MOVB    #^A/1/,R6                       ; updating character
             F934 CF    5B 8F    9A 0ABE   529          MOVZBL  #91,RECCNT                      ; starting rec #
                                0AC4   530 50$:     $GET    RAB=R11,ERR=REPORT_ERROR
                      F52A'     30 0AD3   531          BSBW    ERR
                58   00000004'EF  DE 0AD6   532          MOVAL   CPYBUF+4,R8                     ; get addr of record char
                         59   2D  D0 0ADD   533          MOVL    #45,R9
                              88  96 0AE0   534 55$:     INCB    (R8)+                           ; bump contents
                           FB 59  F5 0AE2   535          SOBGTR  R9,55$
                                0AE5   536          $UPDATE RAB=R11,ERR=REPORT_ERROR
                      F509'     30 0AF4   537          BSBW    ERR
                       03F3     30 0AF7   538          BSBW    CHKRFA
BE F8F4 CF    0065 8F    03E8 8F    3D 0AFA   539          ACBW    #1000,#101,RECCNT,50$
                                FF 0B04
                       04A1     30 0B06   540          BSBW    CHKMOD                          ; go verify changes
                                0B09   541
                                0B09   542 ;
                                0B09   543
                                0B09   544          WTTYPE  <PASS 2 O.K.>
                                0B45   545          $CLOSE  FAB=R10,ERR=REPORT_ERROR
                      F4A9'     30 0B54   546          BSBW    ERR
                3F 1E AA    03    E2 0B57   547          BBSS    #FAB$V_BLK,FAB$B_RAT(R10),DONE
                                0B5C   548          WTTYPE  <DUPLICATE TEST WITH RECORDS NOT CROSSING BLOCK BOUNDARIES>
                       FCDB     31 0B98   549          BRW     T2SETUP
                                0B9B   550
                                0B9B   551 ;
                                0B9B   552
                16 AA    01    90 0B9B   553 DONE:    MOVB    #FAB$M_PUT,FAB$B_FAC(R10)
                1E AB    00    90 0B9F   554          MOVB    #RAB$C_SEQ,RAB$B_RAC(R11)
                                0BA3   555          $WAIT   CMDORAB
   00000004'EF    00000400 8F    D0 0BB0   556          MOVL    #RAB$M_WBH,RAB$L_ROP+CMDORAB
       F7A4 CF    01000000 8F    CA 0BBB   557          BICL2   #FAB$M_NAM,T1FAB+FAB$L_FOP
                                0BC4   558
                                0BC4   559
                                0BC4   560 ;
                                0BC4   561 ;BUT FIRST -- TEST 3
                                0BC4   562 ;
```

```
                                 0BC4    563 ;       test truncate
                                 0BC4    564 ;
                                 0BC4    565
                                 0BC4    566         $FAB_STORE      FAB=R10,SHR=#0,FAC=<PUT,GET,TRN>; set up for trn
                                 0BCB    567         $OPEN   FAB=R10,ERR=REPORT_ERROR
                      F423'  30  0BDA    568         BSBW    ERR
                                 0BDD    569         $CONNECT        RAB=R11,ERR=REPORT_ERROR
                      F411'  30  0BEC    570         BSBW    ERR
                                 0BEF    571         $GET    RAB=R11,ERR=REPORT_ERROR; get 1st record, setting up cp
                      F3FF'  30  0BFE    572         BSBW    ERR
                                 0C01    573         $TRUNCATE       RAB=R11,ERR=REPORT_ERROR; truncate all
                      F3ED'  30  0C10    574         BSBW    ERR
                                 0C13    575         $GET    RAB=R11                 ; should get eof
          0001827A 8F   50     D1  0C1C    576         CMPL    R0,#RMS$_EOF
                          06     13  0C23    577         BEQL    TROK
                    5A    5B     D0  0C25    578         MOVL    R11,R10
                      F3D5'  30  0C28    579         BSBW    EOFPUT                  ; too bad
                                 0C2B    580 TROK:   TYPE    <TRUNCATE SUCCEEDED!>
                                 0C5A    581         $DISCONNECT     RAB=R11,ERR=REPORT_ERROR; clean up
                      F394'  30  0C69    582         BSBW    ERR
                                 0C6C    583         $CLOSE  FAB=R10,ERR=REPORT_ERROR
                      F382'  30  0C7B    584         BSBW    ERR
                                 0C7E    585         $FAB_STORE      FAB=R10,SHR=<PUT,GET,UPI>,FAC=PUT
             1F AA    03     90  0C87    586         MOVB    #FAB$C_VFC,FAB$B_RFM(R10)
          36 AA    64 8F     9B  0C8B    587         MOVZBW  #100,FAB$W_MRS(R10)
             1E AA    02     90  0C90    588         MOVB    #FAB$M_CR,FAB$B_RAT(R10)
             10 AA    30     D0  0C94    589         MOVL    #48,FAB$L_ALQ(R10)
             3F AA    04     90  0C98    590         MOVB    #4,FAB$B_FSZ(R10)
       28 AB    00000000'EF   DE  0C9C    591         MOVAL   CPYBUF,RAB$L_RBF(R11)
                                 0CA4    592
                                 0CA4    593 ;
                                 0CA4    594 ;       try to erase the file!
                                 0CA4    595 ;
                                 0CA4    596
                                 0CA4    597         $ERASE  FAB=T1FAB,ERR=REPORT_ERROR
                      F348'  30  0CB5    598         BSBW    ERR
                                 0CB8    599         TYPE    <T1FILE.DAT HAS BEEN ERASED>
    F678 CF    00100000 8F   C8  0CE7    600         BISL2   #FAB$M_CTG,FAB$L_FOP+T1FAB; for 'f' test
                                 0CF0    601         FINISH  <SEQUENTIAL TESTS>
                           04    0D05    602         RET
```

```
                        0D06   604
                        0D06   605 ;
                        0D06   606 ;  subroutine to read in a record
                        0D06   607 ;
                        0D06   608
                        0D06   609 GETANDCHK:
10 AB   68    06   28   0D06   610          MOVC3   #6,(R8),RAB$W_RFA(R11)  ; rfa to rab
          15 57    E8   0D0B   611          BLBS    R7,PASS2                ; branch if pass 2
                        0D0E   612          $GET    RAB=R11,ERR=REPORT_ERROR; get via rfa
             F2E0'  30   0D1D   613          BSBW    ERR
               3C   10   0D20   614          BSBB    CHKREC
                    05   0D22   615          RSB
                        0D23   616 PASS2:
      1E AB    02   90   0D23   617          MOVB    #RAB$C_RFA,RAB$B_RAC(R11)
                        0D27   618          $FIND   RAB=R11,ERR=REPORT_ERROR
             F2C7'  30   0D36   619          BSBW    ERR
      1E AB    00   90   0D39   620          MOVB    #RAB$C_SEQ,RAB$B_RAC(R11); switch back to sequential
                0A   DD   0D3D   621          PUSHL   #10                     ; loop count
                        0D3F   622 T1BLOOP3:
                        0D3F   623          $GET    RAB=R11,ERR=REPORT_ERROR
             F2AF'  30   0D4E   624          BSBW    ERR
               0B   10   0D51   625          BSBB    CHKREC
               59   D6   0D53   626          INCL    R9                      ; bump record count
          E7 6E     F5   0D55   627          SOBGTR  (SP),T1BLOOP3
       59    0A     C2   0D58   628          SUBL2   #10,R9                  ; restore record count
               8E   D5   0D5B   629          TSTL    (SP)+                   ; clean up stack
                    05   0D5D   630          RSB
```

```
                                    0D5E   632
                                    0D5E   633 ;
                                    0D5E   634 ;  subroutine to check that record read is really the right record
                                    0D5E   635 ;
                                    0D5E   636
                                    0D5E   637 CHKREC:
                F695 CF    59       D1  0D5E   638          CMPL    R9,RECCNT               ; fixed header = record #?
                           44       12  0D63   639          BNEQ    BADRHB
56   52    59    00000064 8F       7B  0D65   640 CHKRC1:  EDIV    #100,R9,R2,R6           ; compute record length
                  22 AB    56       B1  0D6E   641          CMPW    R6,RAB$W_RSZ(R11)       ; = rms record len?
                           4C       12  0D72   642          BNEQ    BADRSZ
     55   52    F67F CF    0A       7B  0D74   643          EDIV    #10,RECCNT,R2,R5        ; compute character
                     55     30       80  0D7B   644          ADDB    #48,R5                  ; make ascii
 6E   00    55    28 BB     56       2D  0D7E   645          CMPC5   R6,@RAB$L_RBF(R11),R5,#0,(SP); match the record?
                           50       12  0D85   646          BNEQ    BADRBF
                        07 57       E8  0D87   647          BLBS    R7,10$                  ; branch if pass 2
                10 AB   68 06       29  0D8A   648          CMPC3   #6,(R8),RAB$W_RFA(R11)  ; rfa the same?
                           01       12  0D8F   649          BNEQ    BADRFA
                                    05  0D91   650 10$:     RSB
```

```
                                0D92   652
                                0D92   653 ;
                                0D92   654 ;  handle errors
                                0D92   655 ;
                                0D92   656
                                0D92   657 BADRFA: FIELD   <RFA>
                         46  11 0DA7   658         BRB     ERROR
                                0DA9   659 BADRHB: FIELD   <RHB>
                         2F  11 0DBE   660         BRB     ERROR
                                0DC0   661 BADRSZ: FIELD   <RSZ>
                         18  11 0DD5   662         BRB     ERROR
                                0DD7   663 BADRBF: FIELD   <RBF>
                         01  11 0DEC   664         BRB     ERROR
                                0DEE   665
                                0DEE   666 ;
                                0DEE   667
                            05  0DEE   668 DONT:   RSB
                                0DEF   669 ERROR:
          F8 00000000'EF        E9  0DEF   670         BLBC    VERBOSITY,DONT
                                0DF6   671         $WAIT   RAB=CMDORAB
      00000004'EF       01      CA  0E03   672         BICL2   #RAB$M_ASY,RAB$L_ROP+CMDORAB
                                0E0A   673         $FAO_S  T2STR,CMDORAB+RAB$W_RSZ,FAOBUF,-
                                0E0A   674         R9,RECCNT,R6,RAB$W_RSZ(R11),-
                                0E0A   675         (R8),4(R8),RAB$W_RFA(R11),RAB$W_RFA+4(R11)
                     FF 50      E9  0E37   676         BLBC    R0,.
00000028'EF   00000000'EF       9E  0E3A   677         MOVAB   CMDBUF,CMDORAB+RAB$L_RBF
                                0E45   678         $PUT    RAB=CMDORAB,ERR=REPORT_ERROR
                      F1A5'     30  0E58   679         BSBW    ERR
                   50   22 AB   3C  0E5B   680         MOVZWL  RAB$W_RSZ(R11),R0
                                0E5F   681         $FAO_S  T3STR,CMDORAB+RAB$W_RSZ,FAOBUF,-
                                0E5F   682         R0,RAB$L_RBF(R11)
                     FF 50      E9  0E7B   683         BLBC    R0,.
                                0E7E   684         $PUT    RAB=CMDORAB,ERR=REPORT_ERROR
                      F16C'     30  0E91   685         BSBW    ERR
                                05  0E94   686         RSB
                                0E95   687
                                0E95   688 ;
                                0E95   689 ;  output a message
                                0E95   690 ;
                                0E95   691
          50 00000000'EF        E9  0E95   692 TYPRFA: BLBC    VERBOSITY,NO
                                0E9C   693         $WAIT   RAB=WTRAB                       ; wait on different rab for i/o to complete
                                0EA7   694         $FAO_S  T1STR,CMDORAB+RAB$W_RSZ,FAOBUF,RECCNT,RAB$W_RFA(R11),-
                                0EA7   695         RAB$W_RFA+4(R11)
                     FF 50      E9  0EC8   696         BLBC    R0,.
00000028'EF   00000000'EF       9E  0ECB   697         MOVAB   CMDBUF,CMDORAB+RAB$L_RBF
                                0ED6   698         $PUT    RAB=CMDORAB,ERR=REPORT_ERROR
                      F114'     30  0EE9   699         BSBW    ERR
                                05  0EEC   700 NO:     RSB
```

```
                                    0EED   702
                                    0EED   703 ;
                                    0EED   704 ;  subroutines for test 2
                                    0EED   705 ;
                                    0EED   706 ;  subroutine to check rfa correctness
                                    0EED   707 ;
                                    0EED   708
        50   F506 CF    01    C3    0EED   709 CHKRFA: SUBL3   #1,RECCNT,R0              ; get rec # - 1
           10 1E AA    03    E0    0EF3   710         BBS     #FAB$V_BLK,FAB$B_RAT(R10),10$
        51    00    32    50    7A 0EF8   711         EMUL    R0,#50,#0,R1               ; get byte addr
52      51    51    00000200 8F    7B 0EFD   712         EDIV    #512,R1,R1,R2              ; compute vbn-1 & offset
                              0A    11 0F06   713         BRB     20$
                                    0F08   714
                                    0F08   715 ;
                                    0F08   716 ;  alternate calc for records not crossing blk boundaries
                                    0F08   717 ;
                                    0F08   718
                              51    D4 0F08   719 10$:    CLRL    R1                         ; zero extend record #
        52    51    50    0A    7B 0F0A   720         EDIV    #10,R0,R1,R2               ; compute vbn-1 & rec # in blk
                    52    32    A4 0F0F   721         MULW2   #50,R2                     ; compute offset
                              51    D6 0F12   722 20$:    INCL    R1                         ; vbn
                 10 AB    51    D1 0F14   723         CMPL    R1,RAB$W_RFA(R11)
                              07    12 0F18   724         BNEQ    ERRRFA
                 14 AB    52    B1 0F1A   725         CMPW    R2,RAB$W_RFA+4(R11)
                              01    12 0F1E   726         BNEQ    ERRRFA
                              05    0F20   727         RSB
                                    0F21   728 ERRRFA: WFIELD  <RFA>
                                    0F43   729         MBPT
                              05    0F4B   730         RSB
```

```
                                 0F4C   732
                                 0F4C   733 ;
                                 0F4C   734 ;  subroutine to get and check a record for test 2
                                 0F4C   735 ;
                                 0F4C   736
    57   50   F4A7 CF   2A   7B  0F4C   737 GTCHK2: EDIV    #42,RECCNT,R0,R7        ; compute char
                   57   30   80  0F53   738         ADDB    #48,R7
                                 0F56   739 GTCHK2A:
                                 0F56   740         $GET    RAB=R11,ERR=REPORT_ERROR
                        F098'    30  0F65   741         BSBW    ERR
                        FF82     30  0F68   742         BSBW    CHKRFA
                     52   28 AB  D0  0F6B   743         MOVL    RAB$L_RBF(R11),R2       ; get rec addr
                   F484 CF   82   D1  0F6F   744         CMPL    (R2)+,RECCNT            ; reccnt o.k.?
                             09   12  0F74   745         BNEQ    ERRREC
    6E   00   57   62   2D   2D  0F76   746         CMPC5   #45,(R2),R7,#0,(SP)     ; match?
                             01   12  0F7C   747         BNEQ    ERRREC
                             05   RSB  0F7E   748
                                 0F7F   749 ERRREC: WFIELD  <RECORD CONTENTS>
                                 0FA1   750         MBPT
                             05  0FA9   751         RSB
                                 0FAA   752
                                 0FAA   753 ;
                                 0FAA   754 ;  subroutine to verify that updated records were changed but not
                                 0FAA   755 ;  the preceeding or following records
                                 0FAA   756 ;
                                 0FAA   757
                                 0FAA   758 CHKMOD:
          F448 CF   5A 8F   9A   0FAA   759         MOVZBL  #90,RECCNT              ; starting rec #
                      FF99   30  0FB0   760 CHKNXT: BSBW    GTCHK2
                   F441 CF   D6  0FB3   761         INCL    RECCNT
                   57   56   D0  0FB7   762         MOVL    R6,R7                   ; check character
                      FF99   30  0FBA   763         BSBW    GTCHK2A
                   F437 CF   D6  0FBD   764         INCL    RECCNT
    03E9 8F        F433 CF   B1  0FC1   765         CMPW    RECCNT,#1001            ; all done?
                             0C   13  0FC8   766         BEQL    10$
                      FF7F   30  0FCA   767         BSBW    GTCHK2
    F424 CF        0063 8F   A0  0FCD   768         ADDW    #99,RECCNT
                             DA   11  0FD4   769         BRB     CHKNXT
                                 0FD6   770 10$:     $GET    R11
     0001827A 8F        50   D1  0FDF   771         CMPL    R0,#RMS$_EOF
                             01   12  0FE6   772         BNEQ    20$
                             05   RSB  0FE8   773
                                 0FE9   774 20$:
                                 0FE9   775         $WAIT   RAB=CMDORAB
                             5A   DD  0FF6   776         PUSHL   R10                     ; save it
                   5A   5B   D0  0FF8   777         MOVL    R11,R10                 ; bad structure!!!
                      F002'    30  0FFB   778         BSBW    EOFPUT
                   0400 8F     BA  0FFE   779         POPR    #^M<R10>
                                 1002   780         MBPT
                             05  100A   781         RSB
                                 100B   782
                                 100B   783         .END
```

```
$$.PSECT_EP                     = 00000000
$$.TAB                          = 000003B4 R   D   01
$$.TABEND                       = 000003F8 R   D   01
$$.TMP                          = 00000043     D
$$.TMP1                         = 00000001     D
$$.TMP2                         = 000000EF
$$.TMPX                         = 00000187 R   D   04
$$.TMPX1                        = 0000000F     D
$$RMSTEST                       = 0000001E
$$RMS_PBUGCHK                   = 00000010
$$RMS_TBUGCHK                   = 00000008
$$RMS_UMODE                     = 00000004
$$T2                            = 00000006     D
..AFLG                          = 00000000     D
..FLG                           = 00000002     D
..MOD                           = 00000001     D
..N                             = 00000003     D
..TYP                           = 00000003     D
.LEN                            = 00000001     D
BADRBF                            00000DD7 R   D   01
BADRFA                            00000D92 R   D   01
BADRHB                            00000DA9 R   D   01
BADRSZ                            00000DC0 R   D   01
BEGPUT                            ********     X   01
BEG_DESCR                         ********     X   01
CHKMOD                            00000FAA R   D   01
CHKNXT                            00000FB0 R   D   01
CHKRC1                            00000D65 R   D   01
CHKREC                            00000D5E R   D   01
CHKRFA                            00000EED R   D   01
CMDBUF                            ********     X   01
CMDORAB                           ********     X   01
CPYBSZ                            ********     X   01
CPYBUF                            ********     X   01
DONE                              00000B9B R   D   01
DONT                              00000DEE R   D   01
EOFPUT                            ********     X   01
ERR                               ********     X   01
ERROR                             00000DEF R   D   01
ERRREC                            00000F7F R   D   01
ERRRFA                            00000F21 R   D   01
FAB$B_FAC                       = 00000016     D
FAB$B_FNS                       = 00000034     D
FAB$B_FSZ                       = 0000003F     D
FAB$B_RAT                       = 000000?E     D
FAB$B_RFM                       = 0000001F     D
FAB$B_SHR                       = 00000017     D
FAB$C_BID                       = 00000003     D
FAB$C_BLN                       = 00000050     D
FAB$C_FIX                       = 00000001     D
FAB$C_SEQ                       = 00000000     D
FAB$C_VFC                       = 00000003     D
FAB$L_ALQ                       = 00000010     D
FAB$L_FNA                       = 0000002C     D
FAB$L_FOP                       = 00000004     D
FAB$L_STS                       = 00000008     D
FAB$M_CR                        = 00000002     D
FAB$M_CTG                       = 00100000     D
FAB$M_GET                       = 00000002     D
FAB$M_NAM                       = 01000000     D
FAB$M_PUT                       = 00000001     D
FAB$M_UPD                       = 00000008     D
FAB$V_BLK                       = 00000003     D
FAB$V_CHAN_MODE                 = 00000002     D
FAB$V_CR                        = 00000001     D
FAB$V_CTG                       = 00000014     D
FAB$V_FILE_MODE                 = 00000004     D
FAB$V_GET                       = 00000001     D
FAB$V_LNM_MODE                  = 00000000     D
FAB$V_PUT                       = 00000000     D
FAB$V_SUP                       = 00000002     D
FAB$V_TRN                       = 00000004     D
FAB$V_UPI                       = 00000006     D
FAB$W_GBC                       = 00000048     D
FAB$W_MRS                       = 00000036     D
FAOBUF                            ********     X   01
FINPUT                            ********     X   01
FIN_DESCR                         ********     X   01
FLDPUT                            ********     X   01
FLD_DESCR                         ********     X   01
GETANDCHK                         00000D06 R   D   01
GTCHK2                            00000F4C R   D   01
GTCHK2A                           00000F56 R   D   01
NAMBLK                            ********     X   01
NO                                00000EEC R   D   01
NXTRC2                            0000091A R   D   01
NXTREC                            00000511 R   D   01
PASS2                             00000D23 R   D   01
RAB$B_RAC                       = 0000001E     D
RAB$C_BID                       = 00000001     D
RAB$C_BLN                       = 00000044     D
RAB$C_KEY                       = 00000001     D
RAB$C_RFA                       = 00000002     D
RAB$C_SEQ                       = 00000000     D
RAB$L_CTX                       = 00000018     D
RAB$L_RBF                       = 00000028     D
RAB$L_RHB                       = 0000002C     D
RAB$L_ROP                       = 00000004     D
RAB$M_ASY                       = 00000001     D
RAB$M_LOC                       = 00010000     D
RAB$M_WBH                       = 00000400     D
RAB$V_LOC                       = 00000010     D
RAB$V_UIF                       = 00000004     D
RAB$V_WBH                       = 0000000A     D
RAB$W_ISI                       = 00000002     D
RAB$W_RFA                       = 00000010     D
RAB$W_RSZ                       = 00000022     D
RECCNT                            000003F8 RG  D   01
REPORT_ERROR                      ********     X   01
RFATBL                            00000044 R   D   01
RHBSW                             000004AB R   D   01
RMS$_EOF                        = 0001827A     D
RMS$_SUPERSEDE                  = 00010631     D
RMT$TEST_1A                       000004AC RG  D   01
```

```
SUPOK                           000008DB R   D   01
SYS$CLOSE                       ********  GX   01
SYS$CONNECT                     ********  GX   01
SYS$CREATE                      ********  GX   01
SYS$DISCONNECT                  ********  GX   01
SYS$ERASE                       ********  GX   01
SYS$FAO                         ********   X   01
SYS$FIND                        ********  GX   01
SYS$GET                         ********  GX   01
SYS$OPEN                        ********  GX   01
SYS$PUT                         ********  GX   01
SYS$REWIND                      ********  GX   01
SYS$TRUNCATE                    ********  GX   01
SYS$UPDATE                      ********  GX   01
SYS$WAIT                        ********  GX   01
T1BDONE                         000006A1 R   D   01
T1BLOOP1                        0000063B R   D   01
T1BLOOP2                        0000064C R   D   01
T1BLOOP3                        00000D3F R   D   01
T1BLOOP4                        00000769 R   D   01
T1BLOOP5                        000007C9 R   D   01
T1CNT                           00000556 R   D   01
T1FAB                           00000364 RG  D   01
T1L                           = 00000025     D
T1RAB                           000003B4 RG  D   01
T1S                             00000408 R   D   01
T1START                         00000000 RG  D   01
T1STR                           00000400 R   D   01
T2CNT                           0000095E R   D   01
T2L                           = 0000005F     D
T2S                             00000435 R   D   01
T2SETUP                         00000876 R   D   01
T2STR                           0000042D R   D   01
T3L                           = 0000000F     D
T3S                             0000049C R   D   01
T3STR                           00000494 R   D   01
TROK                            00000C2B R   D   01
TYPRFA                          00000E95 R   D   01
VERBOSITY                       ********    X   01
WTRAB                           00000000 RG  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	0000100B (4107.)	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
$RMSNAM	00000015 (21.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE
__$RMSNAM	00000196 (406.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD WRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	38	00:00:00.07	00:00:00.55
Command processing	133	00:00:00.54	00:00:02.18
Pass 1	376	00:00:17.01	00:00:35.60
Symbol table sort	0	00:00:00.74	00:00:01.42
Pass 2	138	00:00:03.96	00:00:06.21
Symbol table output	18	00:00:00.13	00:00:00.17
Psect synopsis output	2	00:00:00.03	00:00:00.12
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	707	00:00:22.49	00:00:46.25

The working set limit was 1650 pages.
86298 bytes (169 pages) of virtual memory were used to buffer the intermediate code.
There were 30 pages of symbol table space allocated to hold 618 non-local and 34 local symbols.
783 source lines were read in Pass 1, producing 58 object records in Pass 2.
69 pages of virtual memory were used to define 50 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SYS.OBJ]LIB.MLB;1	0
_255DUA28:[SYSLIB]STARLET.MLB;2	36
TOTALS (all libraries)	36

1074 GETS were required to define 36 macros.

There were no errors, warnings or information messages.

```
MACRO/LIS=LIS$:RMSTEST1/OBJ=OBJ$:RMSTEST1 MSRC$:RMSTEST1/UPDATE=(ENH$:RMSTEST1)+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

SAT55F17
LIS