```
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSSSSSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSSSSSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSSSSSS
RRR         RRR  MMMMMM   MMMMMM  SSS
RRR         RRR  MMMMMM   MMMMMM  SSS
RRR         RRR  MMMMMM   MMMMMM  SSS
RRR         RRR  MMM   MMM   MMM  SSS
RRR         RRR  MMM   MMM   MMM  SSS
RRR         RRR  MMM   MMM   MMM  SSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSS
RRRRRRRRRRRR     MMM         MMM     SSSSSSSSS
RRR   RRR        MMM         MMM              SSS
RRR   RRR        MMM         MMM              SSS
RRR   RRR        MMM         MMM              SSS
RRR      RRR     MMM         MMM              SSS
RRR      RRR     MMM         MMM              SSS
RRR      RRR     MMM         MMM              SSS
RRR         RRR  MMM         MMM  SSSSSSSSSSSSS
RRR         RRR  MMM         MMM  SSSSSSSSSSSSS
RRR         RRR  MMM         MMM  SSSSSSSSSSSSS
```

```
**FILE**ID**RM2OPEN                                   L  6
```

```
RRRRRRRR    MM      MM    222222      000000    PPPPPPPP    EEEEEEEEEE  NN      NN
RRRRRRRR    MM      MM    222222      000000    PPPPPPPP    EEEEEEEEEE  NN      NN
RR      RR  MMMM  MMMM  22        22  00      00  PP      PP  EE          NN      NN
RR      RR  MMMM  MMMM  22        22  00      00  PP      PP  EE          NN      NN
RR      RR  MM  MM  MM          22  00      00  PP      PP  EE          NNNN    NN
RR      RR  MM  MM  MM          22  00      00  PP      PP  EE          NNNN    NN
RRRRRRRR    MM      MM        22    00      00  PPPPPPPP    EEEEEEEE    NN  NN  NN
RRRRRRRR    MM      MM        22    00      00  PPPPPPPP    EEEEEEEE    NN  NN  NN
RR  RR      MM      MM      22      00      00  PP          EE          NN    NNNN
RR  RR      MM      MM      22      00      00  PP          EE          NN    NNNN
RR    RR    MM      MM    22        00      00  PP          EE          NN      NN      ....
RR    RR    MM      MM    22        00      00  PP          EE          NN      NN      ....
RR      RR  MM      MM  2222222222    000000    PP          EEEEEEEEEE  NN      NN      ....
RR      RR  MM      MM  2222222222    000000    PP          EEEEEEEEEE  NN      NN      ....


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

RM2
V04

```
0000            $BEGIN  RM2OPEN,000,RM$RMS2,<RELATIVE SPECIFIC OPEN>
0000
0000    ;
0000    ;****************************************************************************
0000    ;*                                                                          *
0000    ;*   COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                *
0000    ;*   DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                 *
0000    ;*   ALL RIGHTS RESERVED.                                                   *
0000    ;*                                                                          *
0000    ;*   THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
0000    ;*   ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
0000    ;*   INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER  *
0000    ;*   COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY  *
0000    ;*   OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY  *
0000    ;*   TRANSFERRED.                                                           *
0000    ;*                                                                          *
0000    ;*   THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
0000    ;*   AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
0000    ;*   CORPORATION.                                                           *
0000    ;*                                                                          *
0000    ;*   DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
0000    ;*   SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                *
0000    ;*                                                                          *
0000    ;*                                                                          *
0000    ;****************************************************************************
0000    ;
```

```
0000    ;++
0000    ;
0000    ; Facility: RMS32
0000    ;
0000    ; Abstract:
0000    ;       this module provides the organization-specific
0000    ;       open processing for relative files.
0000    ;
0000    ; Environment:
0000    ;               star processor running starlet exec.
0000    ;
0000    ; Author: L F Laverdure,        creation date: 10-OCT-1977
0000    ;
0000    ; Modified By:
0000    ;
0000    ;       V03-007 RAS0284         Ron Schaefer            30-Mar-1984
0000    ;               Fix STV value on error paths for RMS$_RPL and RMS$_WPL errors.
0000    ;
0000    ;       V03-006 DAS0001         David Solomon           25-Mar-1984
0000    ;               Fix broken branches to RM$COPRTN.
0000    ;
0000    ;       V03-005 RAS0265         Ron Schaefer             9-Mar-1984
0000    ;               Bump IFB$W_AVLCL to count the BDB & buffer we allocate.
0000    ;
0000    ;       V03-004 KBT0475         Keith B. Thompson       26-Jan-1983
0000    ;               Remove check for ppf open.
0000    ;
0000    ;       V03-003 KBT0463         Keith B. Thompson       13-Jan-1983
0000    ;               Allocate a bdb and buffer to read in prologue
0000    ;
0000    ;       V03-002 KBT0130         Keith B. Thompson       20-Aug-1982
0000    ;               Reorganize psect
0000    ;
0000    ;       V03-001 KBT0112         Keith B. Thompson       6-Aug-1982
0000    ;               Remove ref to upd_sifb_eof
0000    ;
0000    ;       V02-013 CDS0002         C Saether                5-Feb-1982
0000    ;               Back out V02-012.  GBC is now a record attribute.
0000    ;
0000    ;       V02-012 CDS0001         C Saether                3-Jan-1982
0000    ;               Return GBC field from prologue to FAB.
0000    ;               Fix comment fields.
0000    ;
0000    ;       V02-011 REFORMAT        Keith B. Thompson       29-Jul-1980
0000    ;
0000    ;--
0000    ;
0000
```

```
0000            .SBTTL  DECLARATIONS
0000
0000    ;
0000    ; Include Files:
0000    ;
0000
0000    ;
0000    ; Macros:
0000    ;
0000
0000            $BDBDEF
0000            $FABDEF
0000            $IFBDEF
0000            $PLGDEF
0000            $RMSDEF
0000
0000    ;
0000    ; Equated Symbols:
0000    ;
0000
0000    ;
0000    ; Own Storage:
0000    ;
0000
```

```
                 0000          .SBTTL  RM$OPEN2 - PROCESS RELATIVE FILE PROLOG
                 0000
                 0000  ;++
                 0000  ;
                 0000  ;  RM$OPEN2     -
                 0000  ;
                 0000  ;       this routine performs the file open functions that are
                 0000  ;       specific to the relative file organization, including:
                 0000  ;
                 0000  ;       1  -  verify inter-process record locking not specified
                 0000  ;             since not yet implemented
                 0000  ;       2  -  reading in the prolog and setting the ebk,dvbn,
                 0000  ;             and mrn ifab fields based upon its contents.
                 0000  ;       3  -  setting the mrn fab field.
                 0000  ;
                 0000  ; Calling sequence:
                 0000  ;
                 0000  ;       entered via case branch from RMS$OPEN. returns by
                 0000  ;       jumping to RM$COPRTN.
                 0000  ;
                 0000  ; Input Parameters:
                 0000  ;
                 0000  ;       R11     impure area address
                 0000  ;       R9      ifab address
                 0000  ;       R8      fab address
                 0000  ;
                 0000  ; Implicit Inputs:
                 0000  ;
                 0000  ;       the contents of the ifab
                 0000  ;
                 0000  ; Output Parameters:
                 0000  ;
                 0000  ;       R0                status code
                 0000  ;       R10               ifab addr
                 0000  ;       R1-R5,AP          destroyed
                 0000  ;
                 0000  ; Implicit Outputs:
                 0000  ;
                 0000  ;       various fields in the ifab and fab are initialized.
                 0000  ;
                 0000  ; Completion Codes:
                 0000  ;
                 0000  ;       standard rms, in particular suc,plg,shr,rpl, and ver.
                 0000  ;
                 0000  ; Side Effects:
                 0000  ;
                 0000  ;       may wait quite some time for prolog to become
                 0000  ;       free initially. leaves prolog locked.
                 0000  ;
                 0000  ;--
                 0000
                 0000  RM$OPEN2::
5E A9      95    0000          TSTB    IFB$B_BKS(R9)           ; make sure bks non-zero
   5D      13    0003          BEQL    ERRIFA                  ; if yes, is error
   10      93    0005          BITB    #FAB$C_REL,-            ; really relative?
50 A9            0007                  IFB$B_RFMORG(R9)
   51      13    0009          BEQL    EXIT                    ; aha - a bogus seq file posing
```

```
                   000B                                          ; as relative for sharing
                   000B  ;
                   000B  ; if bio access, then prolog read is not required.
                   000B  ;
                   000B
          05  E0   000B          BBS     #IFB$V_BIO,-            ; leave successfully
    49 22 A9       000D                  IFB$B_FAC(R9),SEXIT
                   0010
                   0010  ;
                   0010  ; read and process prolog
                   0010  ;
                   0010
     5A   59  D0   0010          MOVL    R9,R10                  ; set ifab addr
55   0200 8F  3C   0013          MOVZWL  #512,R5                 ; ask for one block to read prologue
       FFE5'  30   0018          BSBW    RM$ALDBUF               ; allocate bdb and buffer
       3E 50  E9   001B          BLBC    R0,EXIT                 ; get out on error
     0084 C9  B6   001E          INCW    IFB$W_AVLCL(R9)         ; count BDB & buffer
                   0022          $CACHE  VBN=#1,-                ; read the prolog
                   0022                  SIZE=#512,-             ; (R5=buffer addr)
                   0022                  FLAGS=LOCK,-
                   0022                  ERR=ERRRPL
       FFCA'  30   0033          BSBW    RM$CHKSUM               ; validate its checksum
       23 50  E9   0036          BLBC    R0,EXIT                 ; get out on error
       74 A5  B1   0039          CMPW    PLG$W_VER_NO(R5),-      ; supported version?
          01       003C                  #PLG$C_VER_NO
          52  12   003D          BNEQ    ERRPLV                  ; branch if not
                   003F
                   003F  ;
                   003F  ; set up ifab values
                   C03F  ;
                   003F
       70 A5  D0   003F          MOVL    PLG$L_EOF(R5),-         ; copy eof vbn
       74 A9       0042                  IFB$L_EBK(R9)
       68 A5  3C   0044          MOVZWL  PLG$W_DVBN(R5),-        ; copy vbn of first data bucket
     00B0 C9       0047                  IFB$L_DVBN(R9)
       6C A5  D0   004A          MOVL    PLG$L_MRN(R5),-         ; copy max. record number
     00AC C9       004D                  IFB$L_MRN(R9)
       5C A9  B4   0050          CLRW    IFB$W_FFB(R9)           ; set blk offset=0
                   0053
                   0053  ;
                   0053  ; set mrn, gbc in fab
                   0053  ;
                   0053
     00AC C9  D0   0053  SET:    MOVL    IFB$L_MRN(R9),-         ; set mrn
       38 A8       0057                  FAB$L_MRN(R8)
                   0059  SEXIT:  RMSSUC                          ; show success
 00000000'EF  17   005C  EXIT:   JMP     RM$COPRTN               ; & rejoin common open code
                   0062                                          ; note: the bdb will
                   0062                                          ; be released there
```

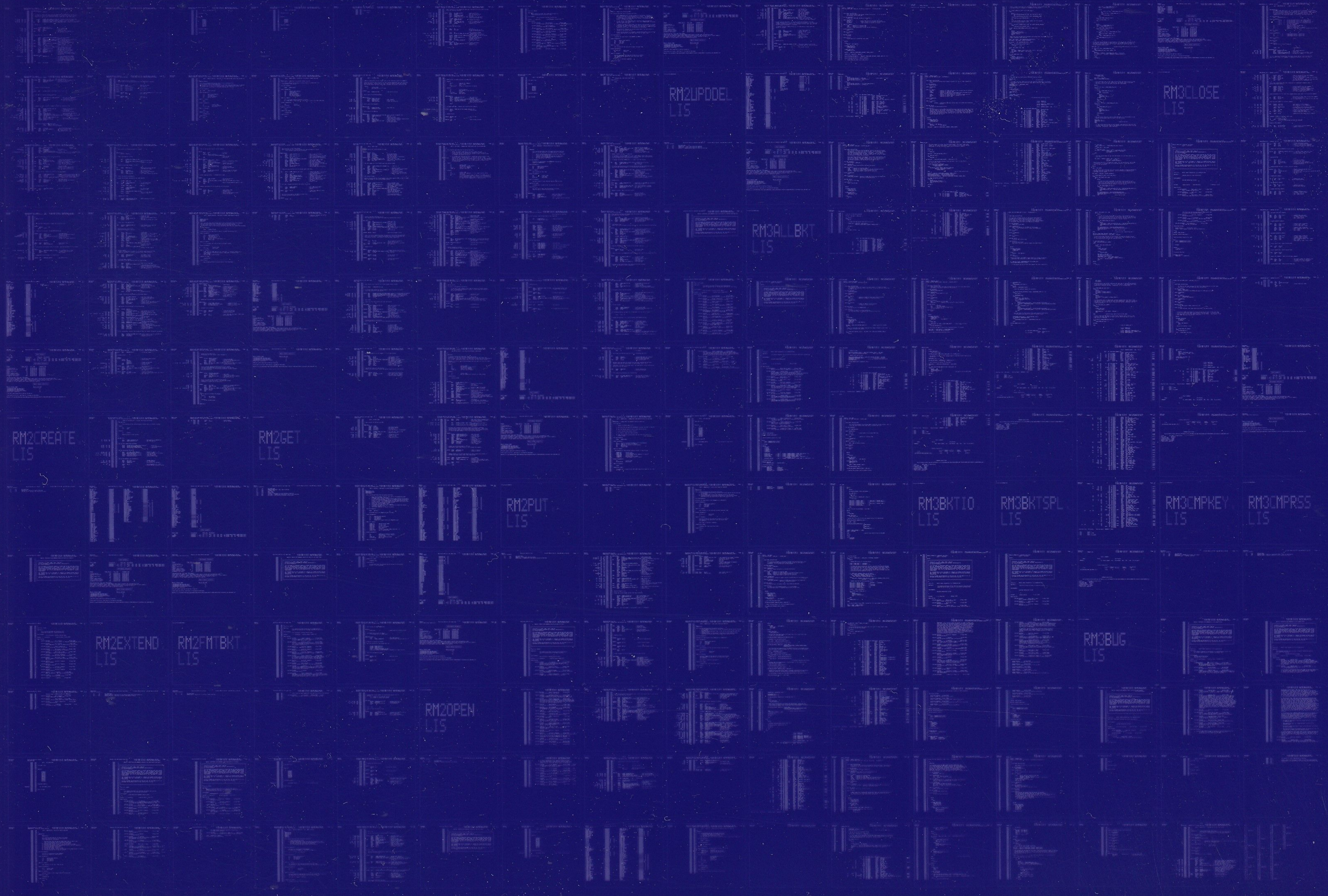
```
                                  0062   209
                                  0062   210 ;
                                  0062   211 ; handle errors
                                  0062   212 ;
                                  0062   213
                                  0062   214 ERRIFA:
      0C A8   0001841C 8F    D0   0062   215         MOVL    #RMS$_BKS,FAB$L_STV(R8) ; set secondary error info
                                  006A   216         RMSERR  IFA                     ; illegal file attributes
                    1A       11   006F   217         BRB     ERRXIT
                                  0071   218
                                  0071   219 ERRORG:
                                  0071   220         RMSERR  ORG                     ; trying to open a ppf
                    13       11   0076   221         BRB     ERRXIT
                                  0078   222
                                  0078   223 ERRRPL:
                 0C A8       D5   0078   224         TSTL    FAL ;L_STV(R8)          ; do we have an stv?
                    09       12   007B   225         BNEQ    10$                     ; okay use it
  0C A8    50   00001000 8F  C9   007D   226         BISL3   #^X1000,R0,FAB$L_STV(R8); else set the RMS error there
                                  0086   227 10$:    RMSERR  RPL                     ; prolog read error
                                  008B   228
                                  008B   229 ;
                                  008B   230 ; (stv has ss error code)
                                  008B   231 ;
                                  008B   232
              00000000'EF    17   008B   233 ERRXIT: JMP     RM$COPRTN               ; go clean up
                                  0091   234
                                  0091   235
                                  0091   236 ERRPLV:
                                  0091   237         RMSERR  PLV                     ; unsupported prolog version
                    F3       11   0096   238         BRB     ERRXIT
                                  0098   239
                                  0098   240         .END
```

```
$$.PSECT_EP                    = 00000000
$$.TMP                         = 00000001
$$RMSTEST                      = 0000001A
$$RMS_PBUGCHK                  = 00000010
$$RMS_TBUGCHK                  = 00000008
$$RMS_UMODE                    = 00000004
CSH$M_LOCK                     = 00000001
CSH$M_NOBUFFER                 = 00000008
ERRIFA                           00000062 R      01
ERRORG                           00000071 R      01
ERRPLV                           00000091 R      01
ERRRPL                           00000078 R      01
ERRXIT                           0000008B R      01
EXIT                             0000005C R      01
FAB$C_REL                      = 00000010
FAB$L_MRN                      = 00000038
FAB$L_STV                      = 0000000C
IFB$B_BKS                      = 0000005E
IFB$B_FAC                      = 00000022
IFB$B_RFMORG                   = 00000050
IFB$L_DVBN                     = 000000B0
IFB$L_EBK                      = 00000074
IFB$L_MRN                      = 000000AC
IFB$V_BIO                      = 00000005
IFB$W_AVLCL                    = 00000084
IFB$W_FFB                      = 0000005C
PLG$C_VER_NO                   = 00000001
PLG$L_EOF                      = 00000070
PLG$L_MRN                      = 0000006C
PLG$W_DVBN                     = 00000068
PLG$W_VER_NO                   = 00000074
RM$ALDBUF                        ********    X   01
RM$CACHE                         ********    X   01
RM$CHKSUM                        ********    X   01
RM$COPRTN                        ********    X   01
RM$OPEN2                         00000000 RG     01
RMS$_BKS                       = 0001841C
RMS$_IFA                       = 0001C124
RMS$_ORG                       = 0001860C
RMS$_PLV                       = 0001872C
RMS$_RPL                       = 0001C104
SET                              00000053 R      01
SEXIT                            00000059 R      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
RM$RMS2                          00000098  (  152.)  01 (  1.)    PIC   USR   CON   REL   GBL NOSHR   EXE   RD   NOWRT NOVEC BYTE
$ABS$                            00000000  (    0.)  02 (  2.)  NOPIC   USR   CON   ABS   LCL NOSHR   EXE   RD     WRT NOVEC BYTE
```

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

Phase                    Page faults    CPU Time       Elapsed Time
-----                    -----------    --------       ------------
Initialization                    32    00:00:00.08    00:00:00.59
Command processing               131    00:00:00.78    00:00:03.23
Pass 1                           241    00:00:06.47    00:00:18.76
Symbol table sort                  0    00:00:00.72    00:00:00.78
Pass 2                            55    00:00:01.27    00:00:03.42
Symbol table output                5    00:00:00.07    00:00:00.14
Psect synopsis output              2    00:00:00.03    00:00:00.24
Cross-reference output             0    00:00:00.00    00:00:00.00
Assembler run totals             468    00:00:09.44    00:00:27.17

The working set limit was 1350 pages.
34621 bytes (68 pages) of virtual memory were used to buffer the intermediate code.
There were 40 pages of symbol table space allocated to hold 664 non-local and 1 local symbols.
240 source lines were read in Pass 1, producing 13 object records in Pass 2.
20 pages of virtual memory were used to define 19 macros.

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

Macro library name                             Macros defined
------------------                             --------------
_$255$DUA28:[RMS.OBJ]RMS.MLB;1                           12
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                            0
_$255$DUA28:[SYSLIB]STARLET.MLB,2                         3
TOTALS (all libraries)                                   15

794 GETS were required to define 15 macros.

There were no errors, warnings or information messages.

MACRO/LIS=LIS$:RM2OPEN/OBJ=OBJ$:RM2OPEN MSRC$:RM2OPEN/UPDATE=(ENH$:RM2OPEN)+EXECML$/LIB+LIB$:RMS/LIB
```

0323 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



RM2UPDDEL
LIS

RM3CLOSE
LIS

RM3ALLBKT
LIS

RM2CREATE
LIS

RM2GET
LIS

RM2PUT
LIS

RM3BKTIO
LIS

RM3BKTSPL
LIS

RM3CMPKEY
LIS

RM3CMPRSS
LIS

RM2EXTEND
LIS

RM2FMTBKT
LIS

RM3BUG
LIS

RM2OPEN
LIS