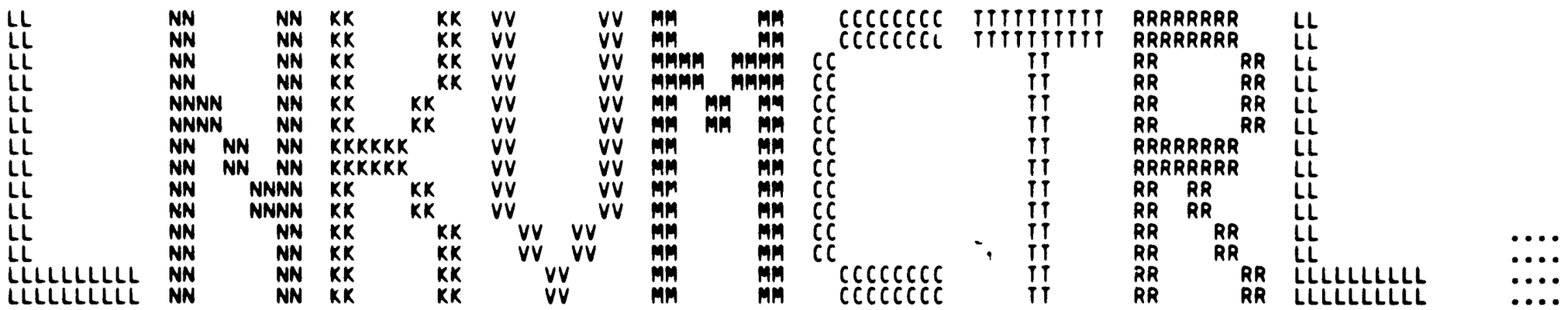
```
LLL                 IIIIIIIII     NNN         NNN  KKK         KKK  EEEEEEEEEEEEEEE   RRRRRRRRRRR
LLL                 IIIIIIIII     NNN         NNN  KKK         KKK  EEEEEEEEEEEEEEE   RRRRRRRRRRR
LLL                 IIIIIIIII     NNN         NNN  KKK         KKK  EEEEEEEEEEEEEEE   RRRRRRRRRRR
LLL                    III        NNN         NNN  KKK         KKK  EEE               RRR         RRR
LLL                    III        NNN         NNN  KKK         KKK  EEE               RRR         RRR
LLL                    III        NNN         NNN  KKK         KKK  EEE               RRR         RRR
LLL                    III        NNNNNN      NNN  KKK       KKK    EEE               RRR         RRR
LLL                    III        NNNNNN      NNN  KKK       KKK    EEE               RRR         RRR
LLL                    III        NNNNNN      NNN  KKK       KKK    EEE               RRR         RRR
LLL                    III        NNN   NNN   NNN  KKKKKKKKK        EEEEEEEEEEE       RRRRRRRRRRR
LLL                    III        NNN   NNN   NNN  KKKKKKKKK        EEEEEEEEEEE       RRRRRRRRRRR
LLL                    III        NNN   NNN   NNN  KKKKKKKKK        EEEEEEEEEEE       RRRRRRRRRRR
LLL                    III        NNN      NNNNNN  KKK       KKK    EEE               RRR   RRR
LLL                    III        NNN      NNNNNN  KKK       KKK    EEE               RRR   RRR
LLL                    III        NNN      NNNNNN  KKK       KKK    EEE               RRR   RRR
LLL                    III        NNN         NNN  KKK         KKK  EEE               RRR      RRR
LLL                    III        NNN         NNN  KKK         KKK  EEE               RRR      RRR
LLL                    III        NNN         NNN  KKK         KKK  EEE               RRR      RRR
LLLLLLLLLLLLLLLL    IIIIIIIII     NNN         NNN  KKK         KKK  EEEEEEEEEEEEEEE   RRR         RRR
LLLLLLLLLLLLLLLL    IIIIIIIII     NNN         NNN  KKK         KKK  EEEEEEEEEEEEEEE   RRR         RRR
LLLLLLLLLLLLLLLL    IIIIIIIII     NNN         NNN  KKK         KKK  EEEEEEEEEEEEEEE   RRR         RRR
```

FILEID**LNKVMCTRL



```
;     1          0001  0 MODULE LNK_VMCTRL (      ! CONTROL FOR ALLOCATION OF VIRTUAL MEMORY
;     2          0002  0                 IDENT = 'V04-000',
;     3          0003  0                 ADDRESSING_MODE(EXTERNAL=GENERAL,
;     4          0004  0                                 NONEXTERNAL=LONG_RELATIVE)
;     5          0005  0                 ) =
;     6          0006  1 BEGIN
;     7          0007  1
;     8          0008  1 !*****************************************************************************
;     9          0009  1 !*                                                                           *
;    10          0010  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    11          0011  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    12          0012  1 !*  ALL RIGHTS RESERVED.                                                      *
;    13          0013  1 !*                                                                           *
;    14          0014  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED     *
;    15          0015  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
;    16          0016  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER     *
;    17          0017  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY     *
;    18          0018  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY     *
;    19          0019  1 !*  TRANSFERRED.                                                              *
;    20          0020  1 !*                                                                           *
;    21          0021  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE     *
;    22          0022  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT     *
;    23          0023  1 !*  CORPORATION.                                                              *
;    24          0024  1 !*                                                                           *
;    25          0025  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS     *
;    26          0026  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                   *
;    27          0027  1 !*                                                                           *
;    28          0028  1 !*                                                                           *
;    29          0029  1 !*****************************************************************************
;    30          0030  1 !
;    31          0031  1 !++
;    32          0032  1 ! FACILITY:     LINKER
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT:     THIS MODULE CONTAINS THE ROUTINES TO ALLOCATE VIRTUAL MEMORY
;    35          0035  1 !                       AT END OF PASS 1.
;    36          0036  1 !
;    37          0037  1 !
;    38          0038  1 ! ENVIRONMENT:  VMS NATIVE MODE
;    39          0039  1 !
;    40          0040  1 ! AUTHOR:       T.J. PORTER, CREATION DATE: 15-JUN-78
;    41          0041  1 !
;    42          0042  1 ! MODIFIED BY:
;    43          0043  1 !
;    44          0044  1 !       V03-001 BLS0007         Benn Schreiber,         3-Jun-1980
;    45          0045  1 !               Convert to MDL data structures.
;    46          0046  1 !--
```

```
;    48          0047  1 !
;    49          0048  1 !++
;    50          0049  1 !       FUNCTIONAL DESCRIPTION
;    51          0050  1 !
;    52          0051  1 LIBRARY
;    53          0052  1         'STARLETL32';
;    54          0053  1 REQUIRE
;    55          0054  1         'PREFIX';
;    56          0169  1 LIBRARY
;    57          0170  1         'DATBAS';
;    58          0171  1 !
;    59          0172  1 EXTERNAL ROUTINE
;    60          0173  1         LNK$ALLOBLK : NOVALUE,          ! DYNAMIC MEMORY ALLOCATOR
;    61          0174  1         LNK$DEALBLK : NOVALUE;          ! AND DEALLOCATOR
;    62          0175  1 !
;    63          0176  1 EXTERNAL
;    64          0177  1         LNK$GL_FVMLST;                          ! LISTHEAD OF FREE V'R MEM DESCRIPTORS
;    65          0178  1 !
;    66          0179  1 GLOBAL
;    67          0180  1         LNK$GL_MINVA : INITIAL(MAX_ADDRESS),    ! LOWEST ADDRESS ALLOCATED
;    68          0181  1         LNK$GL_MAXVA : INITIAL(0);              ! HIGHEST ADDRESS ALLOCATED
```

```
;    70          0182  1 GLOBAL ROUTINE LNK$ALLOVIRMEM(ADDRESS,PAGES) =
;    71          0183  2 BEGIN
;    72          0184  2 !
;    73          0185  2 !++
;    74          0186  2 !       THIS ROUTINE IS CALLED TO ATTEMPT ALLOCATION OF A SPECIFIC
;    75          0187  2 !       PIECE OF VIRTUAL MEMORY. THE ADDRESS AND PAGE COUNT REQUIRED
;    76          0188  2 !       ARE THE INPUT ARGUMENTS. THE ROUTINE RETURNS TRUE OR FALSE
;    77          0189  2 !       DEPENDING ON WHETHER THE SPECIFIED MEMORY IS AVAILABLE OR
;    78          0190  2 !       ALREADY ALLOCATED.
;    79          0191  2 !
;    80          0192  2 !       THE AVALIABLE VIRTUAL MEMORY IS CONTROLLED BY A SINGLY LINKED
;    81          0193  2 !       LIST OF BLOCKS WITH FORMAT:
;    82          0194  2 !                       ------------------
;    83          0195  2 !                       ! NEXT DESCRIPTOR !
;    84          0196  2 !                       ------------------
;    85          0197  2 !                       !  FREE  ADDRESS  !
;    86          0198  2 !                       ------------------
;    87          0199  2 !                       !  # FREE  BYTES  !
;    88          0200  2 !                       ------------------
;    89          0201  2 !--
;    90          0202  2 !
;    91          0203  2 LOCAL
;    92          0204  2         BYTES,                                  ! BYTE COUNT REQUESTED
;    93          0205  2         FREBLK : REF BLOCK[,BYTE],              ! FREE BLOCK POINTER
;    94          0206  2         PREVBLK : REF BLOCK[,BYTE];             ! PREVIOUS FREE BLOCK IN LIST
;    95          0207  2 !
;    96          0208  2 IF (BYTES = .PAGES * 512) EQL 0                 ! IF ZERO SIZE
;    97          0209  2 THEN RETURN TRUE;                               ! RETURN SUCCESS
;    98          0210  2 !
;    99          0211  2 PREVBLK = LNK$GL_FVMLST;                        ! SET AT LISTHEAD
;   100          0212  2 WHILE (FREBLK = .PREVBLK[FVM$L_NXTFVM]) NEQ 0   ! MOVE TO NEXT BLOCK DESCRIPTOR IN LIST
;   101          0213  2 DO IF .ADDRESS LSSU .FREBLK[FVM$L_ADDRESS]       ! LOOP TILL WE GET TO ONE FOR REQUESTED MEMORY
;   102          0214  3    OR .ADDRESS GEQU (.FREBLK[FVM$L_ADDRESS] + .FREBLK[FVM$L_BYTES])
;   103          0215  2    THEN PREVBLK = .FREBLK                       ! ...
;   104          0216  3    ELSE BEGIN
;   105          0217  4         IF (.ADDRESS + .BYTES) GTRU (.FREBLK[FVM$L_ADDRESS] +   ! DOES DESIRED BLOCK FIT
;   106          0218  4                                 .FREBLK[FVM$L_BYTES])           ! IN THE HOLE
;   107          0219  3         THEN EXITLOOP;                          ! QUIT IF NOT
;   108          0220  3         IF .ADDRESS EQL .FREBLK[FVM$L_ADDRESS]  ! IF AT FRONT OF HOLE
;   109          0221  4         THEN IF (FREBLK[FVM$L_BYTES] = .FREBLK[FVM$L_BYTES] -   ! REDUCE THE SIZE OF REMAINDER
;   110          0222  3                 .BYTES) EQL 0                   ! AND IF GOING TO ZERO
;   111          0223  4              THEN BEGIN
;   112          0224  4                   PREVBLK[FVM$L_NXTFVM] = .FREBLK[FVM$L_NXTFVM];  ! TAKE DESCRIPTOR OFF LIST
;   113          0225  4                   LNK$DEALBLK(FVM$C_SIZE,.FREBLK);                ! AND DEALLOCATE IT
;   114          0226  4              END
;   115          0227  3              ELSE FREBLK[FVM$L_ADDRESS] = .FREBLK[FVM$L_ADDRESS] + .BYTES    ! SET ADDRESS OF REMAINDER
;   116          0228  4         ELSE BEGIN                                                           ! NOT AT THE FRONT
;   117          0229  4              IF (.FREBLK[FVM$L_ADDRESS] + .FREBLK[FVM$L_BYTES]) EQL  ! CHECK IF IT IS ON END
;   118          0230  5                 (.ADDRESS + .BYTES)                                  ! OF THE BLOCK
;   119          0231  4              THEN FREBLK[FVM$L_BYTES] = .FREBLK[FVM$L_BYTES] - .BYTES! IF SO JUST REDUCE SIZE
;   120          0232  5              ELSE BEGIN
;   121          0233  5                   LOCAL NXTBLK : REF BLOCK[,BYTE];                ! OTHERWISE NEED TO SPLIT THE BLOCK
;   122          0234  5                   LNK$ALLOBLK(FVM$C_SIZE,NXTBLK);                 ! ALLOCATE ANOTHER DESCRIPTOR
;   123          0235  5                   NXTBLK[FVM$L_NXTFVM] = .FREBLK[FVM$L_NXTFVM];   ! WHICH POINTS ONWARD
;   124          0236  5                   FREBLK[FVM$L_NXTFVM] = .NXTBLK;                 ! AND PONTED TO BY CURRENT
;   125          0237  5                   NXTBLK[FVM$L_ADDRESS] = .ADDRESS + .BYTES;      ! DESCRIBES THE REMNANT AT END
;   126          0238  5                   NXTBLK[FVM$L_BYTES] = .FREBLK[FVM$L_BYTES] -    !
```

```
;  127          0239  6                                   (.NXTBLK[FVM$L_ADDRESS] -
;  128          0240  5                                   .FREBLK[FVM$L_ADDRESS]);
;  129          0241  5                       FREBLK[FVM$L_BYTES] = .ADDRESS - .FREBLK[FVM$L_ADDRESS];        ! AND SET CURRENT FR
;  130          0242  4                   END;
;  131          0243  3               END;
;  132          0244  3           IF (.ADDRESS + .BYTES -1) GTRU .LNK$GL_MAXVA            ! MAXIMIZE THE ADDRESS
;  133          0245  3           THEN LNK$GL_MAXVA = .ADDRESS + .BYTES - 1;              ! SPACE ALLOCATED
;  134          0246  3           IF .ADDRESS LSSU .LNK$GL_MINVA                          ! AND RESET MINIMUM IF NEXESSARY
;  135          0247  3           THEN LNK$GL_MINVA = .ADDRESS;
;  136          0248  3           RETURN TRUE;
;  137          0249  2           END;
;  138          0250  2 RETURN FALSE;
;  139          0251  1 END;                        ! OF LNK$ALLOVIRMEM


                                                    .TITLE  LNK_VMCTRL
                                                    .IDENT  \V04-000\

                                                    .PSECT  $GLOBAL$,NOEXE,2

                                      C0000000  00000 LNK$GL_MINVA::
                                                    .LONG   -1073741824                                     ;
                                      00000000  00004 LNK$GL_MAXVA::
                                                    .LONG   0                                               ;

                                                    .EXTRN  LNK$ALLOBLK, LNK$DEALBLK
                                                    .EXTRN  LNK$GL_FVMLST

                                                    .PSECT  $CODE$,NOWRT,2

                                           01FC 00000           .ENTRY  LNK$ALLOVIRMEM, Save R2,R3,R4,R5,R6,R7,R8  ; 0182
                                 58 00000000'  EF   9E 00002    MOVAB   LNK$GL_MAXVA, R8                          ;
                                 5E            04   C2 00009    SUBL2   #4, SP                                     ;
                          56     08     AC      09   78 0000C    ASHL    #9, PAGES, BYTES                           ; 0208
                                               03   12 00011    BNEQ    1$                                         ;
                                            00A4   31 00013    BRW     11$                                        ;
                                 57 00000000G  00   9E 00016 1$: MOVAB   LNK$GL_FVMLST, PREVBLK                    ; 0211
                                 54        04   AC   D0 0001D    MOVL    ADDRESS, R4                                ; 0213
                                 52             67   D0 00021 2$: MOVL    (PREVBLK), FREBLK                          ; 0212
                                               03   12 00024    BNEQ    3$                                         ;
                                            0095   31 00026    BRW     12$                                        ;
                                        04     A2   54   D1 00029 3$: CMPL    R4, 4(FREBLK)                          ; 0213
                                               0B   1F 0002D    BLSSU   4$                                         ;
                          51   04     A2   08     A2   C1 0002F    ADDL3   8(FREBLK), 4(FREBLK), R1                ; 0214
                                 51            54   D1 00035    CMPL    R4, R1                                     ;
                                               05   1F 00038    BLSSU   5$                                         ;
                                 57            52   D0 0003A 4$: MOVL    FREBLK, PREVBLK                            ; 0215
                                               E2   11 0003D    BRB     2$                                         ;
                          55                   54   C1 0003F 5$: ADDL3   BYTES, R4, R5                          ; 0217
                                 53        08   A2   9E 00043    MOVAB   8(FREBLK), R3                              ; 0218
                          51   04     A2        63   C1 00047    ADDL3   (R3), 4(FREBLK), R1                        ;
                                 51            55   D1 0004C    CMPL    R5, R1                                     ; 0217
                                               6D   1A 0004F    BGTRU   12$                                        ;
                                        04     A2   54   D1 00051    CMPL    R4, 4(FREBLK)                          ; 0220
                                               1B   12 00055    BNEQ    7$                                         ;
                                 63            56   C2 00057    SUBL2   BYTES, (R3)                                ; 0222
                                               10   12 0005A    BNEQ    6$                                         ;
```

```
                                67              62  D0 0005C          MOVL    (FREBLK), (PREVBLK)                     ; 0224
                                                52  DD 0005F          PUSHL   FREBLK                                  ; 0225
                                                0C  DD 00061          PUSHL   #12                                     ;
                     00000000G  00              02  FB 00063          CALLS   #2, LNK$DEALBLK                         ;
                                                38  11 00066          BRB     9$                                      ; 0221
                           04   A2              56  C0 00068  6$:     ADDL2   BYTES, 4(FREBLK)                        ; 0227
                                                32  11 0006D          BRB     9$                                      ; 0221
                                55              51  D1 0006F  7$:     CMPL    R1, R5                                  ; 0230
                                                05  12 00072          BNEQ    8$                                      ;
                                63              56  C2 00074          SUBL2   BYTES, (R3)                             ; 0231
                                                28  11 00077          BRB     9$                                      ;
                                                5E  DD 00079  8$:     PUSHL   SP                                      ; 0234
                                                0C  DD 0007B          PUSHL   #12                                     ;
                     00000000G  00              02  FB 0007D          CALLS   #2, LNK$ALLOBLK                         ;
                                50              6E  D0 00084          MOVL    NXTBLK, R0                              ; 0235
                                60              62  D0 00087          MOVL    (FREBLK), (R0)                          ;
                                62              50  D0 0008A          MOVL    R0, (FREBLK)                            ; 0236
                           04   A0              55  D0 0008D          MOVL    R5, 4(R0)                               ; 0237
                51         04   A2        04    A0  C3 00091          SUBL3   4(R0), 4(FREBLK), R1                    ; 0240
          08    A0              51              63  C1 00097          ADDL3   (R3), R1, 8(R0)                         ; 0239
                63              54        04    A2  C3 0009C          SUBL3   4(FREBLK), R4, (R3)                     ; 0241
                                50        FF    A5  9E 0009F  9$:     MOVAB   -1(R5), R0                              ; 0244
                                68              50  D1 000A3          CMPL    R0, LNK$GL_MAXVA                        ;
                                                03  1B 000A6          BLEQU   10$                                     ;
                                68              50  D0 000A8          MOVL    R0, LNK$GL_MAXVA                        ; 0245
                           FC   A8              54  D1 000AB  10$:     CMPL    R4, LNK$GL_MINVA                        ; 0246
                                                04  1E 000AF          BGEQU   11$                                     ;
                           FC   A8              54  D0 000B1          MOVL    R4, LNK$GL_MINVA                        ; 0247
                                50              01  D0 000B5  11$:     MOVL    #1, R0                                  ; 0248
                                                    04 000B8          RET                                             ;
                                                50  D4 000B9  12$:     CLRL    R0                                      ; 0250
                                                    04 000BB          RET                                             ; 0251

; Routine Size:  193 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  141          0252  1 GLOBAL ROUTINE LNK$FINDVIRMEM(RETADR,PAGES,LOWESTVA) =
;  142          0253  2 BEGIN
;  143          0254  2 !
;  144          0255  2 !++
;  145          0256  2 !       THIS ROUTINE IS SIMILAR TO LNK$ALLOVIRMEM EXCEPT THAT ANY FREE
;  146          0257  2 !       VIRTUAL MEMORY LARGE ENOUGH IS ALLOCATED, STARTING AT LOWEST
;  147          0258  2 !       ADDRESS END.
;  148          0259  2 !
;  149          0260  2 !       IF LOWESTVA IS SUPPLIED IT SPECIFIES THAT THE VIRTUAL MEMORY
;  150          0261  2 !       ALLOCATED MUST BE AT A HIGHER ADDRESS THAN LOWESTVA
;  151          0262  2 !
;  152          0263  2 !               RETADR = ADDRESS OF CELL TO RECEIVE THE ADDRESS OF
;  153          0264  2 !                        THE FIRST BYTE ALLOCATED.
;  154          0265  2 !
;  155          0266  2 !--
;  156          0267  2 MAP
;  157          0268  2     RETADR : REF VECTOR[,LONG];
;  158          0269  2
;  159          0270  2 BUILTIN
;  160          0271  2     NULLPARAMETER;
;  161          0272  2
;  162          0273  2 LOCAL
;  163          0274  2     BYTES,
;  164          0275  2     FREBLK : REF BLOCK[,BYTE],
;  165          0276  2     PREVBLK : REF BLOCK[,BYTE],
;  166          0277  2     SPECIALADR;
;  167          0278  2
;  168          0279  2 IF (BYTES = .PAGES * 512) EQL 0
;  169          0280  3 THEN BEGIN
;  170          0281  3     RETADR[0] = 0;                                  ! IF REQUESTED SIZE IS ZERO
;  171          0282  3     RETURN TRUE                                     ! SUCCESS
;  172          0283  2     END;
;  173          0284  2
;  174          0285  2 SPECIALADR = NOT NULLPARAMETER(3)                   ! SET FLAG IF TO ALLOCATE ABOVE SPECIFIC ADDRESS
;  175          0286  2                 AND (.LOWESTVA NEQ 0);
;  176          0287  2
;  177          0288  2 PREVBLK = LNK$GL_FVMLST;                            ! SET TO START OF LIST
;  178          0289  2 WHILE (FREBLK = .PREVBLK[FVM$L_NXTFVM]) NEQ 0       ! GO DOWN LIST LOOKING FOR LARGE ENOUGH PIECE OF FRE
;  179          0290  3 DO IF (IF .SPECIALADR
;  180          0291  5       THEN ((.FREBLK[FVM$L_ADDRESS] LSSU .LOWESTVA)
;  181          0292  6             OR ((.FREBLK[FVM$L_ADDRESS]+.FREBLK[FVM$L_BYTES]
;  182          0293  6                                 LSSU .LOWESTVA)
;  183          0294  6                 OR (.FREBLK[FVM$L_ADDRESS]+.FREBLK[FVM$L_BYTES]
;  184          0295  4                                 LSSU .LOWESTVA+.BYTES)))
;  185          0296  3       ELSE (.BYTES GTRU .FREBLK[FVM$L_BYTES]))
;  186          0297  2    THEN PREVBLK = .FREBLK
;  187          0298  3    ELSE BEGIN
;  188          0299  3         RETADR[0] = .FREBLK[FVM$L_ADDRESS];        ! RETURN ITS VIRTUAL ADDRESS
;  189          0300  4         IF (FREBLK[FVM$L_BYTES] = .FREBLK[FVM$L_BYTES] -    ! REDUCE ITS SIZE
;  190          0301  3                             .BYTES) EQL 0           ! AND IF COMPLETELY CONSUMED
;  191          0302  4         THEN BEGIN
;  192          0303  4             PREVBLK[FVM$L_NXTFVM] = .FREBLK[FVM$L_NXTFVM];  ! REMOVE DESCRIPTOR FROM LIST
;  193          0304  4             LNK$DEALBLK(FVM$C_SIZE,.FREBLK);        ! AND DEALLOCATE THE DESCRIPTOR
;  194          0305  4             END
;  195          0306  3         ELSE FREBLK[FVM$L_ADDRESS] = .FREBLK[FVM$L_ADDRESS] +   ! OTHERWISE JUST ADJUST THE
;  196          0307  3                                 .BYTES;             ! VIRTUAL ADDRESS REMAINING
;  197          0308  3         IF (.RETADR[0] + .BYTES - 1) GTRU .LNK$GL_MAXVA      ! MAXIMIZE THE ADDRESS
```

```
;  198     0309  3            THEN LNK$GL_MAXVA = ..RETADR + .BYTES - 1;          ! SPACE ALLOCATED
;  199     0310  3        IF .RETADR[0] LSSU .LNK$GL_MINVA                        ! AND RESET MINIMUM IF NEXESSARY
;  200     0311  3            THEN LNK$GL_MINVA = .RETADR[0];
;  201     0312  3        RETURN TRUE                                             ! RETURN SUCCESS
;  202     0313  2        END;
;  203     0314  2 RETURN FALSE                                                   ! NO MEMOFY FOUND
;  204     0315  1 END;                                                           ! END OF LNK$FINDVIRMEM


                                 007C 00000         .ENTRY  LNK$FINDVIRMEM, Save R2,R3,R4,R5,R6     ; 0252
                          56 00000000'   EF  9E 00002         MOVAB   LNK$GL_MAXVA, R6
                53        08   AC        09  78 00009         ASHL    #9, PAGES, BYTES                ; 0279
                                          06  12 0000E         BNEQ    1$
                                    04    BC  D4 00010         CLRL    @RETADR                         ; 0281
                                  0093    31  00013         BRW     12$                             ; 0282
                                03        6C  91 00016 1$:     CMPB    (AP), #3                        ; 0285
                                          05  1E 00019         BGEQU   2$
                                50        01  D0 0001B         MOVL    #1, R0
                                          09  11 0001E         BRB     3$
                                          50  D4 00020 2$:     CLRL    R0
                                    0C    AC  D5 00022         TSTL    12(AP)
                                          02  12 00025         BNEQ    3$
                                          50  D6 00027         INCL    R0
                                          51  D4 00029 3$:     CLRL    R1                              ; 0286
                                    0C    AC  D5 0002B         TSTL    LOWESTVA
                                          02  13 0002E         BEQL    4$
                                          51  D6 00030         INCL    R1
                55                51        50  CB 00032 4$:     BICL3   R0, R1, SPECIALADR
                                  54 00000000G   00  9E 00036         MOVAB   LNK$GL_FVMLST, PREVBLK  ; 0288
                                  52             64  D0 0003D 5$:     MOVL    (PREVBLK), FREBLK       ; 0289
                                                 6B  13 00040         BEQL    13$
                                  1F             55  E9 00042         BLBC    SPECIALADR, 6$          ; 0291
                          0C      AC       04    A2  D1 00045         CMPL    4(FREBLK), LOWESTVA
                                                 1E  1F 0004A         BLSSU   7$
                51        04      A2       08    A2  C1 0004C         ADDL3   8(FREBLK), 4(FREBLK), R1 ; 0292
                          0C      AC             51  D1 00052         CMPL    R1, LOWESTVA            ; 0293
                                                 12  1F 00056         BLSSU   7$
                50                53       0C    AC  C1 00058         ADDL3   LOWESTVA, BYTES, R0     ; 0295
                                  50             51  D1 0005D         CMPL    R1, R0
                                                 0D  1E 00060         BGEQU   8$
                                                 06  11 00062         BRB     7$
                          08      A2             53  D1 00064 6$:     CMPL    BYTES, 8(FREBLK)        ; 0296
                                                 05  1B 00068         BLEQU   8$
                                  54             52  D0 0006A 7$:     MOVL    FREBLK, PREVBLK         ; 0297
                                                 CE  11 0006D         BRB     5$
                          04      BC       04    A2  D0 0006F 8$:     MOVL    4(FREBLK), @RETADR      ; 0299
                          08      A2             53  C2 00074         SUBL2   BYTES, 8(FREBLK)        ; 0301
                                                 10  12 00078         BNEQ    9$
                                  64             62  D0 0007A         MOVL    (FREBLK), (PREVBLK)     ; 0303
                                                 52  DD 0007D         PUSHL   FREBLK                  ; 0304
                                                 0C  DD 0007F         PUSHL   #12
                   00000000G      00             02  FB 00081         CALLS   #2, LNK$DEALBLK
                                                 04  11 00088         BRB     10$                     ; 0300
                          04      A2             53  C0 0008A 9$:     ADDL2   BYTES, 4(FREBLK)        ; 0307
```

```
                                50       04   BC   D0 0008E 10$:    MOVL    @RETADR, R0                          ; 0308
                                51     FF A340   9E 00092          MOVAB   -1(BYTE$)[R0], R1                    ;
                                66          51   D1 00097          CMPL    R1, LNK$GL_MAXVA                     ;
                                            03   1B 0009A          BLEQU   11$                                  ;
                                66          51   D0 0009C          MOVL    R1, LNK$GL_MAXVA                     ; 0309
                             FC   A6        50   D1 0009F 11$:     CMPL    R0, LNK$GL_MINVA                     ; 0310
                                            04   1E 000A3          BGEQU   12$                                  ;
                             FC   A6        50   D0 000A5          MOVL    R0, LNK$GL_MINVA                     ; 0311
                                50          01   D0 000A9 12$:     MOVL    #1, R0                               ; 0312
                                                 04 000AC          RET                                          ;
                                            50   D4 000AD 13$:     CLRL    R0                                   ; 0314
                                                 04 000AF          RET                                          ; 0315

; Routine Size:  176 bytes,    Routine Base:  $CODE$ + 00C1

;  205          0316  0 END ELUDOM



;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $GLOBAL$                       8  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                       369  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)



;                                 Library Statistics
;
;                                          -------- Symbols --------      Pages       Processing
;       File                               Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]STARLET.L32;1        9776         6         0         581       00:01.0
;  _$255$DUA28:[LINKER.OBJ]DATBAS.L32;1      538         4         0          28       00:00.5



;                                     COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:LNKVMCTRL/OBJ=OBJ$:LNKVMCTRL MSRC$:LNKVMCTRL/UPDATE=(ENH$:LNKVMCTRL)

; Size:          369 code + 8 data bytes
; Run Time:        00:09.7
; Elapsed Time:    00:29.0
; Lines/CPU Min:    1962
; Lexemes/CPU-Min: 17180
; Memory Used:  94 pages
```

; Compilation Complete

_$2

Vir
Sta
Ima
Ima
Ima
Num
Num
Num
Num
Num
Num
Num
Ima
Map
Est

Per

Tot
Usi
Tot

Num

85
A t
LIN

LNKUMCTRL
LIS

RDBSHR
LIS

SYSACLSRV
LIS

RDBDISP
LIS

LOADSS

SECURESHR
MAP

FINDHELD
LIS