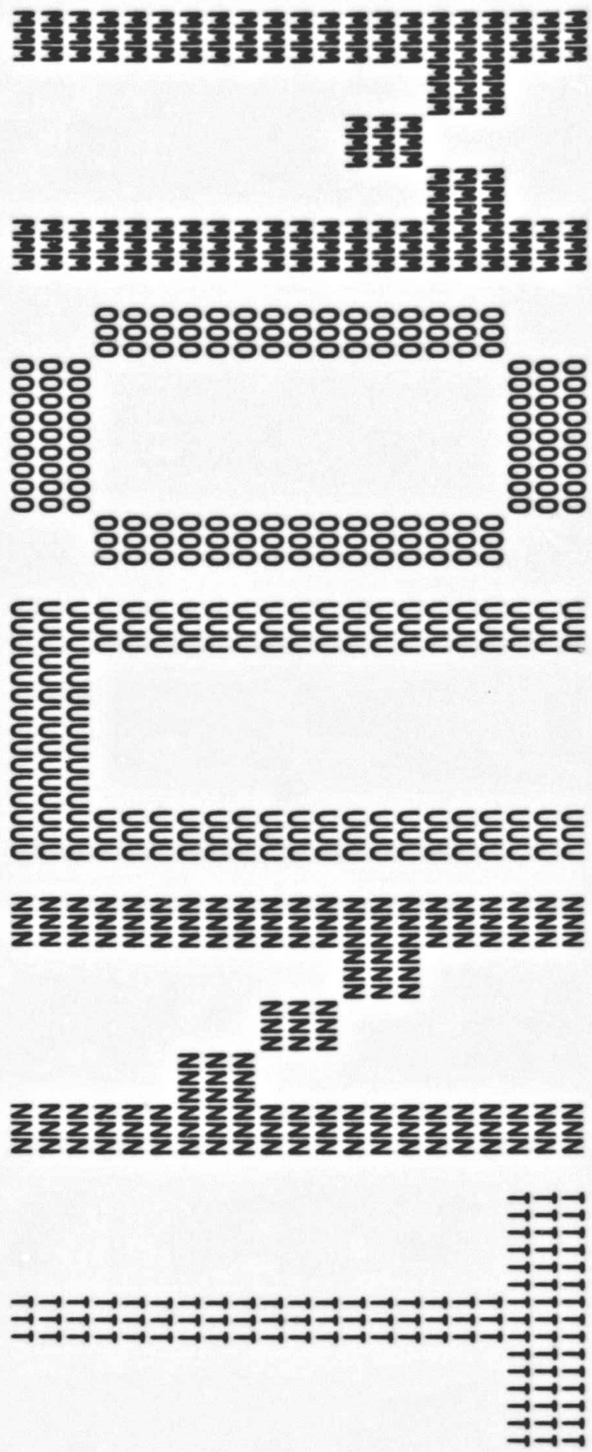

FILEID**INIFC2

```
IIIIII    NN      NN    IIIIII    FFFFFFFFFF    CCCCCCCC    222222
IIIIII    NN      NN    IIIIII    FFFFFFFFFF    CCCCCCCC    222222
  II      NN      NN      II      FF          CC          22      22
  II      NN      NN      II      FF          CC          22      22
  II      NNNN    NN      II      FF          CC                  22
  II      NNNN    NN      II      FF          CC                  22
  II      NN  NN  NN      II      FFFFFFFF    CC                22
  II      NN  NN  NN      II      FFFFFFFF    CC                22
  II      NN    NNNN      II      FF          CC              22
  II      NN    NNNN      II      FF          CC              22
  II      NN      NN      II      FF          CC            22              ....
  II      NN      NN      II      FF          CC            22              ....
IIIIII    NN      NN    IIIIII    FF            CCCCCCCC  2222222222      ....
IIIIII    NN      NN    IIIIII    FF            CCCCCCCC  2222222222      ....

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
```

```
;     1          0001  0 MODULE INIFC2 (
;     2          0002  0                 LANGUAGE (BLISS32),
;     3          0003  0                 IDENT = 'V04-000'
;     4          0004  0                 ) =
;     5          0005  1 BEGIN
;     6          0006  1
;     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 !
;    33          0033  1 ! FACILITY:  F11ACP Structure Level 2
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:
;    36          0036  1 !
;    37          0037  1 !       These routines create and initialize a file control block
;    38          0038  1 !       from the given file header.
;    39          0039  1 !
;    40          0040  1 ! ENVIRONMENT:
;    41          0041  1 !
;    42          0042  1 !       STARLET operating system, including privileged system services
;    43          0043  1 !       and internal exec routines. These routines must be called in
;    44          0044  1 !       kernel mode.
;    45          0045  1 !
;    46          0046  1 !
;    47          0047  1 !--
;    48          0048  1 !
;    49          0049  1 !
;    50          0050  1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  14-Dec-1976  16:48
;    51          0051  1 !
;    52          0052  1 ! MODIFIED BY:
;    53          0053  1 !
;    54          0054  1 !       V03-006 ACG0440         Andrew C. Goldstein,    23-Jul-1984  13:32
;    55          0055  1 !               Add ref count and classification valid flag to ORB in the FCB
;    56          0056  1 !
;    57          0057  1 !       V03-005 CDS0003         Christian D. Saether     1-May-1984
```

```
;    58          0058  1 !                   Set up FCB$L_LOCKBASIS.
;    59          0059  1 !
;    60          0060  1 !       V03-004 LMP0221         L. Mark Pilant,         27-Apr-1984  11:17
;    61          0061  1 !               Add support for an ORB within an FCB.
;    62          0062  1 !
;    63          0063  1 !       V03-003 ACG0408         Andrew C. Goldstein,    22-Mar-1984  21:59
;    64          0064  1 !               Make APPLY_RVN and DEFAULT_RVN macros
;    65          0065  1 !
;    66          0066  1 !       V03-002 CDS0002         Christian D. Saether    18-Jan-1984
;    67          0067  1 !               Modify interface to APPLY_RVN.
;    68          0068  1 !
;    69          0069  1 !       V03-001 CDS0001         Christian D. Saether     8-Dec-1983
;    70          0070  1 !               Use MOUDEF instead of FCPDEF.
;    71          0071  1 !
;    72          0072  1 !       V02-003 ACG0241         Andrew C. Goldstein,    11-Dec-1981  22:53
;    73          0073  1 !               Make updating of FCB common code, add handling
;    74          0074  1 !               of directory bit
;    75          0075  1 !
;    76          0076  1 !       V02-002 ACG0167         Andrew C. Goldstein,    16-Apr-1980  19:26
;    77          0077  1 !               Previous revision history moved to F11B.REV
;    78          0078  1 !**
;    79          0079  1
;    80          0080  1
;    81          0081  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    82          0082  1 REQUIRE 'SRC$:MOUDEF.B32';
;    83          0614  1
;    84          0615  1
;    85          0616  1 FORWARD ROUTINE
;    86          0617  1         INIT_FCB2       : NOVALUE,      ! initialize FCB
;    87          0618  1         FILL_FCB        : NOVALUE;      ! fill in basic FCB fields
```

```
;    89          0619  1 GLOBAL ROUTINE INIT_FCB2 (FCB, HEADER) : NOVALUE =
;    90          0620  1
;    91          0621  1 !++
;    92          0622  1 !
;    93          0623  1 ! FUNCTIONAL DESCRIPTION:
;    94          0624  1 !
;    95          0625  1 !       This routine initializes the given FCB according to the given
;    96          0626  1 !       file header.
;    97          0627  1 !
;    98          0628  1 ! CALLING SEQUENCE:
;    99          0629  1 !       INIT_FCB (ARG1, ARG2)
;   100          0630  1 !
;   101          0631  1 ! INPUT PARAMETERS:
;   102          0632  1 !       ARG1: FCB address
;   103          0633  1 !       ARG2: header address
;   104          0634  1 !
;   105          0635  1 ! IMPLICIT INPUTS:
;   106          0636  1 !       HEADER_LBN contains LBN of header block
;   107          0637  1 !
;   108          0638  1 ! OUTPUT PARAMETERS:
;   109          0639  1 !       NONE
;   110          0640  1 !
;   111          0641  1 ! IMPLICIT OUTPUTS:
;   112          0642  1 !       NONE
;   113          0643  1 !
;   114          0644  1 ! ROUTINE VALUE:
;   115          0645  1 !       NONE
;   116          0646  1 !
;   117          0647  1 ! SIDE EFFECTS:
;   118          0648  1 !       FCB initialized
;   119          0649  1 !
;   120          0650  1 !--
;   121          0651  1
;   122          0652  2 BEGIN
;   123          0653  2
;   124          0654  2 MAP
;   125          0655  2         FCB             : REF BBLOCK,   ! FCB argument
;   126          0656  2         HEADER          : REF BBLOCK;   ! file header arg
;   127          0657  2
;   128          0658  2 LINKAGE
;   129          0659  2         L_MAP_POINTER   = JSB :
;   130          0660  2                           GLOBAL (COUNT = 6, LBN = 7, MAP_POINTER = 8);
;   131          0661  2
;   132          0662  2 GLOBAL REGISTER
;   133          0663  2         COUNT           = 6,            ! retrieval pointer count
;   134          0664  2         LBN             = 7,            ! retrieval pointer LBN
;   135          0665  2         MAP_POINTER     = 8;            ! pointer to scan map area
;   136          0666  2
;   137          0667  2 EXTERNAL ROUTINE
;   138          0668  2         GET_MAP_POINTER : L_MAP_POINTER; ! get value of file map pointer
;   139          0669  2
;   140          0670  2
;   141          0671  2 ! Scan the map area. Count up the file size from the retrieval pointers.
;   142          0672  2 !
;   143          0673  2
;   144          0674  2 MAP_POINTER = .HEADER + .HEADER[FH2$B_MPOFFSET]*2;
;   145          0675  2 FCB[FCB$L_FILESIZE] = 0;
```

```
;  146      0676  2
;  147      0677  2 UNTIL .MAP_POINTER GEQA .HEADER + (.HEADER[FH2$B_MPOFFSET] + .HEADER[FH2$B_MAP_INUSE]) * 2
;  148      0678  2 DO
;  149      0679  3     BEGIN
;  150      0680  3     GET_MAP_POINTER ();
;  151      0681  3     FCB[FCB$L_FILESIZE] = .FCB[FCB$L_FILESIZE] + .COUNT;
;  152      0682  2     END;
;  153      0683  2
;  154      0684  2 ! Now set up the rest of the fields.
;  155      0685  2 !
;  156      0686  2
;  157      0687  2 FILL_FCB (.FCB, .HEADER);
;  158      0688  2
;  159      0689  1 END;                                        ! end of routine INIT_FCB


                                                        .TITLE  INIFC2
                                                        .IDENT  \V04-000\

                                                        .EXTRN  GET_MAP_POINTER

                                                        .PSECT  $CODE$,NOWRT,2

                               OFFC 00000               .ENTRY  INIT_FCB2, Save R2,R3,R4,R5,R6,R7,R8,R9,-        ; 0619
                                                                R10,R11                                          ;
              52      04   AC   7D 00002                MOVQ    FCB, R2                                          ; 0675
              54      01   A3   9A 00006                MOVZBL  1(R3), R4                                        ; 0674
              58         6344   3E 0000A                MOVAW   (R3)[R4], MAP_POINTER                            ;
                      38   A2   D4 0000E                CLRL    56(R2)                                           ; 0675
              50      3A   A3   9A 00011  1$:           MOVZBL  58(R3), R0                                       ; 0677
              50           54   C0 00015                ADDL2   R4, R0                                           ;
              50         6340   3E 00018                MOVAW   (R3)[R0], R0                                     ;
              50           58   D1 0001C                CMPL    MAP_POINTER, R0                                  ;
                           09   1E 0001F                BGEQU   2$                                               ;
                        0000G   30 00021                BSBW    GET_MAP_POINTER                                  ; 0680
        38    A2           56   C0 00024                ADDL2   COUNT, 56(R2)                                    ; 0681
                           E7   11 00028                BRB     1$                                               ; 0677
                           0C   BB 0002A  2$:           PUSHR   #^M<R2,R3>                                       ; 0687
     0000V    CF           02   FB 0002C                CALLS   #2, FILL_FCB                                     ;
                                04 00031                RET                                                      ; 0689

; Routine Size:  50 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  161     0690  1 GLOBAL ROUTINE FILL_FCB (FCB, HEADER) : NOVALUE =
;  162     0691  1
;  163     0692  1 !++
;  164     0693  1 !
;  165     0694  1 ! FUNCTIONAL DESCRIPTION:
;  166     0695  1 !
;  167     0696  1 !       This routine updates the file attributes of the specified FCB,
;  168     0697  1 !       if any, with the file attributes of the given header. The file size
;  169     0698  1 !       is preserved.
;  170     0699  1 !
;  171     0700  1 !
;  172     0701  1 ! CALLING SEQUENCE:
;  173     0702  1 !       FILL_FCB (ARG1, ARG2)
;  174     0703  1 !
;  175     0704  1 ! INPUT PARAMETERS:
;  176     0705  1 !       ARG1: address of FCB
;  177     0706  1 !       ARG2: address of file header
;  178     0707  1 !
;  179     0708  1 ! IMPLICIT INPUTS:
;  180     0709  1 !       NONE
;  181     0710  1 !
;  182     0711  1 ! OUTPUT PARAMETERS:
;  183     0712  1 !       NONE
;  184     0713  1 !
;  185     0714  1 ! IMPLICIT OUTPUTS:
;  186     0715  1 !       NONE
;  187     0716  1 !
;  188     0717  1 ! ROUTINE VALUE:
;  189     0718  1 !       NONE
;  190     0719  1 !
;  191     0720  1 ! SIDE EFFECTS:
;  192     0721  1 !       FCB is updated if it exists
;  193     0722  1 !
;  194     0723  1 !--
;  195     0724  1
;  196     0725  2 BEGIN
;  197     0726  2
;  198     0727  2 BUILTIN
;  199     0728  2         REMQUE,
;  200     0729  2         ROT;
;  201     0730  2
;  202     0731  2 LOCAL
;  203     0732  2         FCB_ORB         : REF BBLOCK,   ! Address of the ORB within the FCB
;  204     0733  2         DUMMY;                          ! dummy target for REMQUE
;  205     0734  2
;  206     0735  2 MAP
;  207     0736  2         FCB             : REF BBLOCK,   ! FCB arg
;  208     0737  2         HEADER          : REF BBLOCK;   ! file header arg
;  209     0738  2
;  210     0739  2 LINKAGE
;  211     0740  2         L_MAP_POINTER   = JSB :
;  212     0741  2                           GLOBAL (COUNT = 6, LBN = 7, MAP_POINTER = 8);
;  213     0742  2
;  214     0743  2 GLOBAL REGISTER
;  215     0744  2         COUNT           = 6,            ! retrieval pointer count
;  216     0745  2         LBN             = 7,            ! retrieval pointer LBN
;  217     0746  2         MAP_POINTER     = 8;            ! pointer to scan map area
```

```
;  218          0747  2
;  219          0748  2 EXTERNAL
;  220          0749  2         CURRENT_RVN,                    ! RVN of current volume
;  221          0750  2         CURRENT_VCB     : REF BBLOCK,   ! VCB of current volume
;  222          0751  2         HEADER_LBN;                     ! LBN of file header
;  223          0752  2
;  224          0753  2 EXTERNAL
;  225          0754  2         EXE$GL_DYNAMIC_FLAGS    : BITVECTOR ADDRESSING_MODE (ABSOLUTE);
;  226          0755  2                                         ! Dynamic SYSGEN flags
;  227          0756  2
;  228          0757  2 EXTERNAL LITERAL
;  229          0758  2         EXE$V_CLASS_PROT;               ! Set if doing non-discretionary checks
;  230          0759  2
;  231          0760  2 EXTERNAL ROUTINE
;  232          0761  2         GET_MAP_POINTER : L_MAP_POINTER; ! get value of file map pointer
;  233          0762  2
;  234          0763  2 ! Note the address of the ORB within the FCB.
;  235          0764  2
;  236          0765  2 FCB_ORB = FCB[FCB$R_ORB];
;  237          0766  2
;  238          0767  2 ! Get the known constants and the simple stuff from the file header
;  239          0768  2 ! (i.e., header LBN, file ID, starting VBN, file owner and file protection).
;  240          0769  2 !
;  241          0770  2
;  242          0771  2 FCB[FCB$L_HDLBN]         = .HEADER_LBN;
;  243          0772  2 FCB[FCB$W_FID_NUM]       = .HEADER[FH2$W_FID_NUM];
;  244          0773  2 FCB[FCB$W_FID_SEQ]       = .HEADER[FH2$W_FID_SEQ];
;  245          0774  2 FCB[FCB$W_FID_RVN]       = .HEADER[FH2$W_FID_RVN];
;  246          0775  2 FCB[FCB$W_SEGN]          = .HEADER[FH2$W_SEG_NUM];
;  247          0776  2
;  248          0777  2 CH$FILL (0, ORB$C_LENGTH, .FCB_ORB);
;  249          0778  2 FCB_ORB[ORB$L_OWNER]     = .HEADER[FH2$L_FILEOWNER];
;  250          0779  2 FCB_ORB[ORB$V_PROT_16]   = 1;
;  251          0780  2 FCB_ORB[ORB$W_PROT]      = .HEADER[FH2$W_FILEPROT];
;  252          0781  2 FCB_ORB[ORB$V_MODE_VECTOR] = 1;
;  253          0782  2 FCB_ORB[ORB$L_MODE_PROTL] = NOT .HEADER[FH2$B_ACC_MODE];
;  254          0783  2 FCB_ORB[ORB$L_MODE_PROTH] = -1;
;  255          0784  2
;  256          0785  2 $ASSUME (FH2$S_CLASS_PROT EQL ORB$S_MIN_CLASS);
;  257          0786  2 $ASSUME (FH2$S_CLASS_PROT EQL ORB$S_MAX_CLASS);
;  258          0787  2
;  259          0788  2 IF .EXE$GL_DYNAMIC_FLAGS[EXE$V_CLASS_PROT]
;  260          0789  2 AND .HEADER[FH2$B_IDOFFSET]*2 GTRU FH2$C_LENGTH
;  261          0790  2 THEN
;  262          0791  3     BEGIN
;  263          0792  3     CH$MOVE (FH2$S_CLASS_PROT, HEADER[FH2$R_CLASS_PROT], FCB_ORB[ORB$R_MIN_CLASS]);
;  264          0793  3     CH$MOVE (FH2$S_CLASS_PROT, HEADER[FH2$R_CLASS_PROT], FCB_ORB[ORB$R_MAX_CLASS]);
;  265          0794  3     FCB_ORB[ORB$V_CLASS_PROT] = 1;
;  266          0795  2     END;
;  267          0796  2
;  268          0797  2 FCB[FCB$W_VERSIONS]      = .BBLOCK [HEADER[FH2$W_RECATTR], FAT$W_VERSIONS];
;  269          0798  2 APPLY_RVN (FCB[FCB$W_FID_RVN], .CURRENT_RVN);
;  270          0799  2 (FCB[FCB$L_LOCKBASIS])<0,16> = .FCB[FCB$W_FID_NUM];
;  271          0800  2 (FCB[FCB$L_LOCKBASIS])<16,8> = .FCB[FCB$B_FID_NMX];
;  272          0801  2 (FCB[FCB$L_LOCKBASIS])<24,8> = .FCB[FCB$B_FID_RVN];
;  273          0802  2
;  274          0803  2 IF .HEADER[FH2$V_SPOOL] THEN FCB[FCB$V_SPOOL] = 1;
```

```
;  275          0804  2
;  276          0805  2 FCB[FCB$L_EFBLK]          = ROT (.BBLOCK[HEADER[FH2$W_RECATTR], FAT$L_EFBLK], 16);
;  277          0806  2 IF  .FCB[FCB$L_EFBLK] NEQ 0
;  278          0807  2 AND .BBLOCK[HEADER[FH2$W_RECATTR], FAT$W_FFBYTE] EQL 0
;  279          0808  2 THEN FCB[FCB$L_EFBLK] = .FCB[FCB$L_EFBLK] - 1;
;  280          0809  2
;  281          0810  2 IF .FCB[FCB$L_EFBLK] GTR .FCB[FCB$L_FILESIZE]
;  282          0811  2 THEN FCB[FCB$L_EFBLK] = .FCB[FCB$L_FILESIZE];
;  283          0812  2
;  284          0813  2 ! If the directory bit is being turned off, credit one to the volume's
;  285          0814  2 ! directory LRU.
;  286          0815  2 !
;  287          0816  2
;  288          0817  2 IF .FCB[FCB$V_DIR]
;  289          0818  2 AND NOT .HEADER[FH2$V_DIRECTORY]
;  290          0819  2 THEN
;  291          0820  3     BEGIN
;  292          0821  3     FCB[FCB$V_DIR] = 0;
;  293          0822  3     CURRENT_VCB[VCB$B_LRU_LIM] = .CURRENT_VCB[VCB$B_LRU_LIM] + 1;
;  294          0823  3     IF .FCB[FCB$W_ACNT] EQL 0
;  295          0824  3     THEN REMQUE (.FCB, DUMMY);
;  296          0825  2     END;
;  297          0826  2
;  298          0827  2 ! Now scan the map area. Get the starting LBN if the file is contiguous.
;  299          0828  2 !
;  300          0829  2
;  301          0830  2 MAP_POINTER = .HEADER + .HEADER[FH2$B_MPOFFSET]*2;
;  302          0831  2 GET_MAP_POINTER ();
;  303          0832  2 FCB[FCB$L_STLBN] = 0;
;  304          0833  2 IF .HEADER[FH2$V_CONTIG]
;  305          0834  2 THEN FCB[FCB$L_STLBN] = .LBN;
;  306          0835  2
;  307          0836  1 END;                                    ! end of routine FILL_FCB


                                                .EXTRN  CURRENT_RVN, CURRENT_VCB
                                                .EXTRN  HEADER_LBN, EXE$GL_DYNAMIC_FLAGS
                                                .EXTRN  EXE$V_CLASS_PROT

                                    0FFC 00000  .ENTRY  FILL_FCB, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,-; 0690
                                                        R11                                          ;
                          57    04  AC  7D 00002  MOVQ   FCB, R7                                     ; 0765
                          56    58  A7  9E 00006  MOVAB  88(R7), FCB_ORB                            ;
                    34    A7  0000G CF  D0 0000A  MOVL   HEADER_LBN, 52(R7)                         ; 0771
                    24    A7    08  A8  D0 00010  MOVL   8(R8), 36(R7)                             ; 0772
                          59    28  A7  9E 00015  MOVAB  40(R7), R9                                ; 0774
                          69    0C  A8  B0 00019  MOVW   12(R8), (R9)                             ;
                    2A    A7    04  A8  B0 0001D  MOVW   4(R8), 42(R7)                            ; 0775
0058  8F          00        6E        00  2C 00022  MOVC5  #0, (SP), #0, #88, (FCB_ORB)           ; 0777
                                      66    00029                                                  ;
                          66    3C  A8  D0 0002A  MOVL   60(R8), (FCB_ORB)                        ; 0778
                    0B    A6        01  88 0002E  BISB2  #1, 11(FCB_ORB)                          ; 0779
                    18    A6    40  A8  B0 00032  MOVW   64(R8), 24(FCB_ORB)                      ; 0780
                    0B    A6        04  88 00037  BISB2  #4, 11(FCB_ORB)                          ; 0781
                    10    A6    3B  A8  9A 0003B  MOVZBL 59(R8), 16(FCB_ORB)                      ; 0782
                    10    A6    10  A6  D2 00040  MCOML  16(FCB_ORB), 16(FCB_ORB)                 ;
```

```
                    14   A6                  01  CE 00045          MNEGL   #1, 20(FCB_ORB)                                          ; 0783
             1F 00000000G 9F 00000000G       8F  E1 00049          BBC     #EXE$V_CLASS_PROT, @#EXE$GL_DYNAMIC_FLAGS, -             ; 0788
                                                                           1$
                    50                       68  9A 00055          MOVZBL  (R8), R0                                                 ; 0789
                    50                       02  C4 00058          MULL2   #2, R0
                50000050   8F                50  D1 0005B          CMPL    R0, #80
                                             10  1B 00062          BLEQU   1$
          30    A6    58   A8                14  28 00064          MOVC3   #20, 88(R8), 48(FCB_ORB)                                 ; 0792
          44    A6    58   A8                14  28 0006A          MOVC3   #20, 88(R8), 68(FCB_ORB)                                 ; 0793
                      0B   A6                10  88 00070          BISB2   #16, 11(FCB_ORB)                                         ; 0794
                      40   A7          32    A8  B0 00074 1$:      MOVW    50(R8), 64(R7)                                           ; 0797
                                             69  95 00079          TSTB    (R9)                                                     ; 0798
                                             05  12 0007B          BNEQ    2$
                      69         0000G       CF  90 0007D          MOVB    CURRENT_RVN, (R9)
                      01                     69  91 00082 2$:      CMPB    (R9), #1
                                             08  12 00085          BNEQ    3$
                           0000G             CF  D5 00087          TSTL    CURRENT_RVN
                                             02  12 0008B          BNEQ    3$
                                             69  94 0008D          CLRB    (R9)
                      4C   A7          24    A7  B0 0008F 3$:      MOVW    36(R7), 76(R7)                                           ; 0799
                      4E   A7          29    A7  90 00094          MOVB    41(R7), 78(R7)                                           ; 0800
                      4F   A7                69  90 00099          MOVB    (R9), 79(R7)                                             ; 0801
                04    35   A8                04  E1 0009D          BBC     #4, 53(R8), 4$                                           ; 0803
                      22   A7                10  88 000A2          BISB2   #16, 34(R7)
                           50          3C    A7  9E 000A6 4$:      MOVAB   60(R7), R0                                               ; 0805
                60    1C   A8                10  9C 000AA          ROTL    #16, 28(R8), (R0)
                                             07  13 000AF          BEQL    5$                                                       ; 0806
                                       20    A8  B5 000B1          TSTW    32(R8)                                                   ; 0807
                                             02  12 000B4          BNEQ    5$
                                             60  D7 000B6          DECL    (R0)                                                     ; 0808
                      38   A7                60  D1 000B8 5$:      CMPL    (R0), 56(R7)                                             ; 0810
                                             04  15 000BC          BLEQ    6$
                           60          38    A7  D0 000BE          MOVL    56(R7), (R0)                                             ; 0811
                           19          22    A7  E9 000C2 6$:      BLBC    34(R7), 7$                                               ; 0817
                14         35   A8                05  E0 000C6          BBS     #5, 53(R8), 7$                                           ; 0818
                           22   A7                01  8A 000CB          BICB2   #1, 34(R7)                                               ; 0821
                           50         0000G       CF  D0 000CF          MOVL    CURRENT_VCB, R0                                          ; 0822
                                      49    A0    96  D4 000D4          INCB    73(R0)
                                      1A    A7    B5  D7 000D7          TSTW    26(R7)                                                   ; 0823
                                             03  12 000DA          BNEQ    7$
                           50                67  0F 000DC          REMQUE  (R7), DUMMY                                              ; 0824
                           52          08    AC  D0 000DF 7$:      MOVL    HEADER, R2                                               ; 0830
                           50          01    A2  9A 000E3          MOVZBL  1(R2), R0
                           58                6240  3E 000E7          MOVAW   (R2)[R0], MAP_POINTER
                                       0000G 30  000EB          BSBW    GET_MAP_POINTER                                          ; 0831
                           50          04    AC  D0 000EE          MOVL    FCB, R0                                                  ; 0832
                                       30    A0  D4 000F2          CLRL    48(R0)
                                       34    A2  95 000F5          TSTB    52(R2)                                                   ; 0833
                                             04  18 000F8          BGEQ    8$
                      30   A0                57  D0 000FA          MOVL    LBN, 48(R0)                                              ; 0834
                                                04 000FE 8$:      RET                                                              ; 0836

; Routine Size:  255 bytes,    Routine Base:  $CODE$ + 0032

;  308          0837  1
;  309          0838  1 END
```

```
;  310          0839  0 ELUDOM



;                                PSECT SUMMARY
;
;      Name                    Bytes                          Attributes
;
;  $CODE$                        305  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]LIB.L32;1          18619         58         0         1000        00:02.0



;                                COMMAND QUALIFIERS
;
;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:INIFC2/OBJ=OBJ$:INIFC2 MSRC$:INIFC2/UPDATE=(ENH$:INIFC2)

; Size:         305 code + 0 data bytes
; Run Time:         00:16.2
; Elapsed Time:     00:35.8
; Lines/CPU Min:     3101
; Lexemes/CPU-Min: 32365
; Memory Used:  166 pages
; Compilation Complete
```

CHKHM2
LIS

CHNUCB
LIS

GETUIC
LIS

ERASE
LIS

LEFTONE
LIS

MOUDK2
LIS

CHKHM1
LIS

CHKSM2
LIS

CLUSTRMNT
LIS

INIFC2
LIS

MOUDK1
LIS

MAKRVT
LIS

MAKLOG
LIS