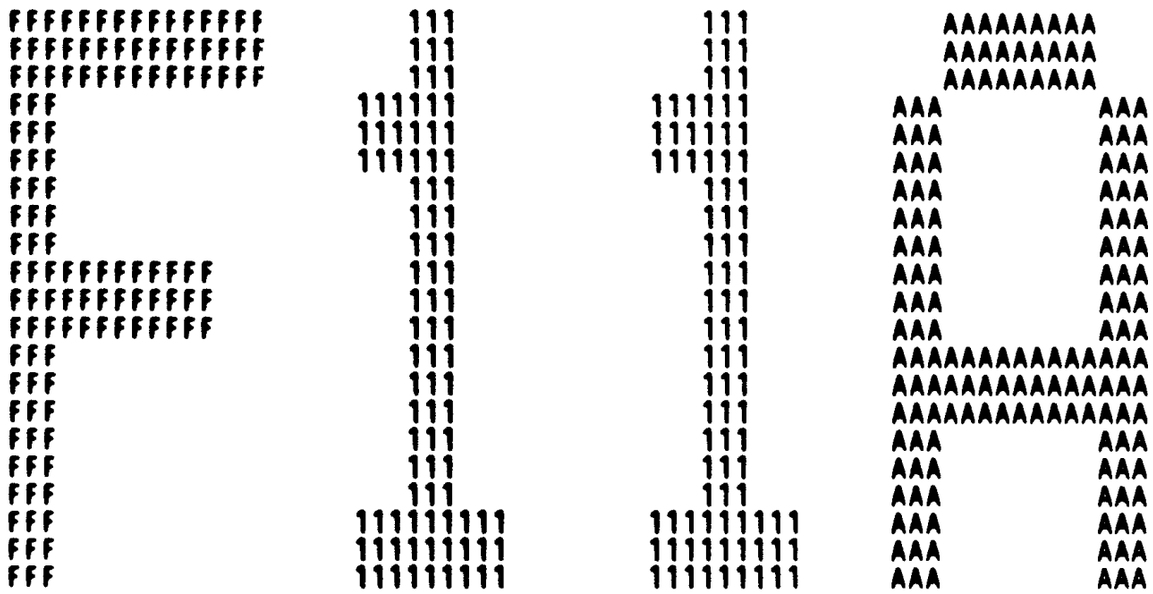
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFFFFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFFFFFFFFFFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
FFF
111
111
111
111111
111111
111111
111
111
111
111
111
111
111
111
111
111
111111111
111111111
111111111
111
111
111
111111
111111
111111
111
111
111
111
111
111
111
111
111
111
111111111
111111111
111111111
AAAAAAAAA
AAAAAAAAA
AAAAAAAAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
AAAAAAAAAAAAAAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA
AAA AAA

```
 IIIIII    NN       NN    IIIIII    FFFFFFFFFF    CCCCCCCC  PPPPPPPP
 IIIIII    NN       NN    IIIIII    FFFFFFFFFF    CCCCCCCC  PPPPPPPP
   II      NN       NN      II      FF            CC        PP      PP
   II      NN       NN      II      FF            CC        PP      PP
   II      NNNN     NN      II      FF            CC        PP      PP
   II      NNNN     NN      II      FF            CC        PP      PP
   II      NN  NN   NN      II      FFFFFFFF      CC        PPPPPPPP
   II      NN  NN   NN      II      FFFFFFFF      CC        PPPPPPPP
   II      NN    NNNN       II      FF            CC        PP
   II      NN    NNNN       II      FF            CC        PP
   II      NN       NN      II      FF            CC        PP              ....
   II      NN       NN      II      FF            CC        PP              ....
 IIIIII    NN       NN    IIIIII    FF              CCCCCCCC  PP              ....
 IIIIII    NN       NN    IIIIII    FF              CCCCCCCC  PP              ....

LL            IIIIII      SSSSSSS
LL            IIIIII     SSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSS
LL              II        SSSSS
LL              II             SS
LL              II             SS
LL              II             SS
LL              II             SS
LLLLLLLLLL    IIIIII    SSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSS
```

```
;  1     0001  0 MODULE INIFCP (
;  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 1
; 34     0034  1 !
; 35     0035  1 ! ABSTRACT:
; 36     0036  1 !
; 37     0037  1 !       This routine does the one time initialization for FCP.
; 38     0038  1 !
; 39     0039  1 ! ENVIRONMENT:
; 40     0040  1 !
; 41     0041  1 !       STARLET operating system, including privileged system services
; 42     0042  1 !       and internal exec routines. This routine must be called
; 43     0043  1 !       in kernel mode.
; 44     0044  1 !
; 45     0045  1 !--
; 46     0046  1 !
; 47     0047  1 !
; 48     0048  1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  20-Dec-1976  16:30
; 49     0049  1 !
; 50     0050  1 ! MODIFIED BY:
; 51     0051  1 !
; 52     0052  1 !       V03-001 ACG0346         Andrew C. Goldstein,    2-Aug-1983  15:17
; 53     0053  1 !               Remove creation of job controller reply mailbox
; 54     0054  1 !
; 55     0055  1 !       V02-001 ACG0245         Andrew C. Goldstein,    22-Dec-1981  20:35
; 56     0056  1 !               Add job controller mailbox
; 57     0057  1 !
```

```
;    58          0058  1 !       V02-000 ACG0167         Andrew C. Goldstein,     7-May-1980  18:50
;    59          0059  1 !               Previous revision history moved to F11A.REV
;    60          0060  1 !**
;    61          0061  1
;    62          0062  1
;    63          0063  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    64          0064  1 REQUIRE 'SRC$:FCPDEF.B32';
;    65          0379  1
;    66          0380  1 ! Dummy vectors to bracket the locked down code and data psects.
;    67          0381  1 !
;    68          0382  1
;    69          0383  1 PSECT   OWN     = $LOCKEDC0$ (NOWRITE, EXECUTE, ALIGN (9));
;    70          0384  1 OWN     L_CODE_START    : VECTOR [0];
;    71          0385  1
;    72          0386  1 PSECT   OWN     = $LOCKEDC9$ (NOWRITE, EXECUTE, ALIGN (2));
;    73          0387  1 OWN     L_CODE_END      : VECTOR [0];
;    74          0388  1
;    75          0389  1 PSECT   OWN     = $LOCKEDD0$;
;    76          0390  1 OWN     L_DATA_START    : VECTOR [0];
;    77          0391  1
;    78          0392  1 PSECT   OWN     = $LOCKEDD9$;
;    79          0393  1 OWN     L_DATA_END      : VECTOR [0];
;    80          0394  1
;    81          0395  1 PSECT   OWN     = $LOCKEDD1$;
```

```
;    83          0396  1 GLOBAL ROUTINE INIT_FCP : NOVALUE =
;    84          0397  1
;    85          0398  1 !++
;    86          0399  1 !
;    87          0400  1 ! FUNCTIONAL DESCRIPTION:
;    88          0401  1 !
;    89          0402  1 !       This routine does the one time initialization for FCP.
;    90          0403  1 !
;    91          0404  1 ! CALLING SEQUENCE:
;    92          0405  1 !       INIT_FCP ()
;    93          0406  1 !
;    94          0407  1 ! INPUT PARAMETERS:
;    95          0408  1 !       NONE
;    96          0409  1 !
;    97          0410  1 ! IMPLICIT INPUTS:
;    98          0411  1 !       system I/O data base
;    99          0412  1 !
;   100          0413  1 ! OUTPUT PARAMETERS:
;   101          0414  1 !       NONE
;   102          0415  1 !
;   103          0416  1 ! IMPLICIT OUTPUTS:
;   104          0417  1 !       IO_CHANNEL: gets channel number of I/O channel
;   105          0418  1 !       QUEUE_HEAD: points to ACP queue block
;   106          0419  1 !
;   107          0420  1 ! ROUTINE VALUE:
;   108          0421  1 !       NONE
;   109          0422  1 !
;   110          0423  1 ! SIDE EFFECTS:
;   111          0424  1 !       FCP hooked up to system data base
;   112          0425  1 !
;   113          0426  1 !--
;   114          0427  1
;   115          0428  2 BEGIN
;   116          0429  2
;   117          0430  2 LITERAL
;   118          0431  2         EXEC_MODE       = 1;            ! exec mode
;   119          0432  2
;   120          0433  2 LOCAL
;   121          0434  2         SYSEXV          : VECTOR [2],   ! place to save system exception vectors
;   122          0435  2         MBX_CHANNEL,                    ! channel number of mailbox created
;   123          0436  2         WORKING_SET,                    ! desired ACP working set
;   124          0437  2         STATUS,                         ! system status return
;   125          0438  2         AQB             : REF BBLOCK,   ! pointer to scan AQB list
;   126          0439  2         CCB             : REF BBLOCK;   ! pointer to channel control block
;   127          0440  2
;   128          0441  2 EXTERNAL
;   129          0442  2         IO_CHANNEL,                     ! I/O channel number
;   130          0443  2         DISK_UCB        : REF BBLOCK,   ! UCB address of 'SYS$DISK'
;   131          0444  2         QUEUE_HEAD      : REF BBLOCK,   ! address of queue header
;   132          0445  2         IOC$GL_AQBLIST  : REF BBLOCK ADDRESSING_MODE (ABSOLUTE),
;   133          0446  2                                         ! system AQB listhead
;   134          0447  2         SCH$GL_CURPCB   : REF BBLOCK ADDRESSING_MODE (ABSOLUTE),
;   135          0448  2                                         ! address of our PCB
;   136          0449  2         ACP$GW_WORKSET  : WORD ADDRESSING_MODE (ABSOLUTE),
;   137          0450  2                                         ! user specified ACP working set
;   138          0451  2         ACP$GW_MAPCACHE : WORD ADDRESSING_MODE (ABSOLUTE),
;   139          0452  2                                         ! storage map cache size
```

```
;  140          0453  2          ACP$GW_HDRCACHE : WORD ADDRESSING_MODE (ABSOLUTE),
;  141          0454  2                           ! file header cache size
;  142          0455  2          ACP$GW_DIRCACHE : WORD ADDRESSING_MODE (ABSOLUTE),
;  143          0456  2                           ! directory cache size
;  144          0457  2          EXE$GL_FLAGS    : BITVECTOR ADDRESSING_MODE (ABSOLUTE);
;  145          0458  2                           ! system flags vector
;  146          0459  2
;  147          0460  2  EXTERNAL LITERAL
;  148          0461  2          EXE$V_INIT      : UNSIGNED (6); ! bit position of FCP init flag
;  149          0462  2
;  150          0463  2  EXTERNAL ROUTINE
;  151          0464  2          GET_CCB,                       ! get CCB address of channel
;  152          0465  2          INIT_POOL;                     ! initialize buffer pool
;  153          0466  2
;  154          0467  2
;  155          0468  2  ! Clear the exec exception vectors to avoid interference
;  156          0469  2  ! with FCP's error reporting logic.
;  157          0470  2  !
;  158          0471  2
;  159          0472  2  $SETEXV (VECTOR = 0, ACMODE = EXEC_MODE, PRVHND = SYSEXV[0]); ! primary exec handler
;  160          0473  2  $SETEXV (VECTOR = 1, ACMODE = EXEC_MODE, PRVHND = SYSEXV[1]); ! secondary exec handler
;  161          0474  2
;  162          0475  2  ! If the previous handler addresses are found to be in process space,
;  163          0476  2  ! restore them since they belong to the debugger.
;  164          0477  2  !
;  165          0478  2
;  166          0479  2  IF .SYSEXV[0] GTR 0
;  167          0480  2  THEN $SETEXV (VECTOR = 0, ADDRES = .SYSEXV[0], ACMODE = EXEC_MODE); ! primary exec handler
;  168          0481  2  IF .SYSEXV[1] GTR 0
;  169          0482  2  THEN $SETEXV (VECTOR = 1, ADDRES = .SYSEXV[1], ACMODE = EXEC_MODE); ! secondary exec handler
;  170          0483  2
;  171          0484  2  ! Find the queue header for this ACP by searching the system AQB list
;  172          0485  2  ! for an AQB with a matching PID. Then assign a channel to
;  173          0486  2  ! 'SYS$DISK' (as good a device as any) and record the channel number.
;  174          0487  2  ! Said channel number is used for all I/O - it is assigned to the
;  175          0488  2  ! right device simply by stuffing the UCB pointer in the CCB.
;  176          0489  2  ! Also save the UCB address in the channel so that it can be restored
;  177          0490  2  ! to properly deassign the channel when we exit.
;  178          0491  2  !
;  179          0492  2
;  180          0493  2  AQB = .IOC$GL_AQBLIST;
;  181          0494  2  UNTIL .AQB EQL 0 OR .AQB[AQB$L_ACPPID] EQL .SCH$GL_CURPCB[PCB$L_PID]
;  182          0495  2  DO AQB = .AQB[AQB$L_LINK];
;  183          0496  2
;  184          0497  2  IF .AQB NEQ 0
;  185          0498  2  THEN QUEUE_HEAD = .AQB
;  186          0499  2  ELSE $EXIT (CODE = SS$_NOAQB);          ! no queue header found - quit
;  187          0500  2
;  188          0501  2  ! Now adjust the working set. Get the working set size from the system
;  189          0502  2  ! parameters if non-zero. Else, compute it from the buffer cache sizes.
;  190          0503  2  ! This is done as follows: Add up the requested cache size. Add in the
;  191          0504  2  ! buffer descriptor overhead. Add in 48 pages of FCP code and other overhead.
;  192          0505  2  ! Add in additional charge for extra page tables.
;  193          0506  2  !
;  194          0507  2
;  195          0508  2  WORKING_SET = .ACP$GW_WORKSET;
;  196          0509  2  IF .WORKING_SET EQL 0
```

```
;  197          0510  2 THEN
;  198          0511  3     BEGIN
;  199          0512  3     WORKING_SET = MAXU (4, .ACP$GW_MAPCACHE
;  200          0513  3                         + .ACP$GW_HDRCACHE
;  201          0514  3                         + .ACP$GW_DIRCACHE);
;  202          0515  3     WORKING_SET = .WORKING_SET + (.WORKING_SET*161 + 4095) / 4096;
;  203          0516  3     WORKING_SET = .WORKING_SET + 48;
;  204          0517  3     WORKING_SET = .WORKING_SET + .WORKING_SET/128;
;  205          0518  2     END;
;  206          0519  2 IF .AQB[AQB$B_ACPTYPE] EQL AQB$K_F11V2
;  207          0520  2 THEN WORKING_SET = .WORKING_SET * 6;    ! allow extra space for structure level 2
;  208          0521  2
;  209          0522  2 $ADJWSL (PAGCNT = -10000);             ! adjust working set to minimum
;  210          0523  2 $ADJWSL (PAGCNT = .WORKING_SET);       ! then add size desired
;  211          0524  2
;  212          0525  2 ! Now lock appropriate areas into the working set. These are code and data that
;  213          0526  2 ! are used at raised IPL.
;  214          0527  2 !
;  215          0528  2
;  216          0529  2 STATUS = $LKWSET (INADR = UPLIT (L_CODE_START, L_CODE_END));
;  217          0530  2 IF NOT .STATUS THEN $EXIT (CODE = .STATUS);
;  218          0531  2
;  219          0532  2 STATUS = $LKWSET (INADR = UPLIT (L_DATA_START, L_DATA_END));
;  220          0533  2 IF NOT .STATUS THEN $EXIT (CODE = .STATUS);
;  221          0534  2
;  222          0535  2 INIT_POOL ();
;  223          0536  2
;  224          0537  2 IO_CHANNEL = 0;
;  225        P 0538  2 IF NOT $ASSIGN (DEVNAM = DESCRIPTOR ('SYS$SYSTEM'),
;  226        P 0539  2                 CHAN   = IO_CHANNEL,
;  227          0540  3                 ACMODE = EXEC_MODE)
;  228          0541  2 THEN BUG_CHECK (NOACPCHAN, FATAL, 'Failure to assign ACP channel');
;  229          0542  2
;  230          0543  2 CCB = GET_CCB (.IO_CHANNEL);
;  231          0544  2 DISK_UCB = .CCB[CCB$L_UCB];
;  232          0545  2
;  233          0546  2 ! Finally set the FCP inited bit in the system flags word to indicate that
;  234          0547  2 ! a file system now exists (significant only during system startup). If
;  235          0548  2 ! this bit was clear, we are the first up, so also create the bad block
;  236          0549  2 ! scanner mailbox. Also clear the transition bit in the AQB to tell
;  237          0550  2 ! MOUNT that we are alive and well.
;  238          0551  2 !
;  239          0552  2
;  240          0553  2 IF TESTBITCS (EXE$GL_FLAGS[EXE$V_INIT])
;  241          0554  2 THEN
;  242          0555  3     BEGIN
;  243        P 0556  3     $CREMBX (
;  244        P 0557  3                 CHAN   = MBX_CHANNEL,
;  245        P 0558  3                 MAXMSG = BBS$C_LENGTH,
;  246        P 0559  3                 BUFQUO = BBS$C_LENGTH * 100,
;  247        P 0560  3                 PROMSK = %X'FFOF',
;  248        P 0561  3                 ACMODE = EXEC_MODE,
;  249        P 0562  3                 LOGNAM = DESCRIPTOR ('ACP$BADBLOCK_MBX')
;  250          0563  3               );
;  251          0564  2     END;
;  252          0565  2
;  253          0566  2 QUEUE_HEAD[AQB$V_CREATING] = 0;
```

```
;  254          0567  2
;  255          0568  1 END;                                    ! end of routine INIT_FCP


                                                                .TITLE  INIFCP
                                                                .IDENT  \V04-000\

                                                                .PSECT  $CODE$,NOWRT,2

                                        00000000' 00000000'  00000 P.AAA:   .ADDRESS L_CODE_START, L_CODE_END                ;
                                        00000000' 00000000'  00008 P.AAB:   .ADDRESS L_DATA_START, L_DATA_END                ;
                        4D  45  54  53  59  53  24  53  59  53  00010 P.AAD:   .ASCII  \SYS$SYSTEM\                          ;
                                                                0001A         .BLKB   2
                                            0000000A  0001C P.AAC:   .LONG   10                                          ;
                                            00000000' 00020         .ADDRESS P.AAD                                       ;
42  4D  5F  4B  43  4F  4C  42  44  41  42  24  50  43  41  00024 P.AAF:   .ASCII  \ACP$BADBLOCK_MBX\                       ;
                                                        58  00033                                                        ;
                                            00000010  00034 P.AAE:   .LONG   16                                          ;
                                            00000000' 00038         .ADDRESS P.AAF                                       ;

                                                                .PSECT  $LOCKEDD9$,NOEXE,2

                                                00000 L_DATA_END:
                                                                .BLKB   0

                                                                .PSECT  $LOCKEDD0$,NOEXE,2

                                                00000 L_DATA_START:
                                                                .BLKB   0

                                                                .PSECT  $LOCKEDC9$,NOWRT,2

                                                00000 L_CODE_END:
                                                                .BLKB   0

                                                                .PSECT  $LOCKEDC0$,NOWRT,9

                                                00000 L_CODE_START:
                                                                .BLKB   0

                                                                .EXTRN  IO_CHANNEL, DISK_UCB
                                                                .EXTRN  QUEUE_HEAD, IOC$GL_AQBLIST
                                                                .EXTRN  SCH$GL_CURPCB, ACP$GW_WORKSET
                                                                .EXTRN  ACP$GW_MAPCACHE
                                                                .EXTRN  ACP$GW_HDRCACHE
                                                                .EXTRN  ACP$GW_DIRCACHE
                                                                .EXTRN  EXE$GL_FLAGS, EXE$V_INIT
                                                                .EXTRN  GET_CCB, INIT_POOL
                                                                .EXTRN  SYS$SETEXV, SYS$EXIT
                                                                .EXTRN  SYS$ADJWSL, SYS$LKWSET
                                                                .EXTRN  SYS$ASSIGN, BUG$_NOACPCHAN
                                                                .EXTRN  SYS$CREMBX

                                                                .PSECT  $CODE$,NOWRT,2

                                  03FC 00000                    .ENTRY  INIT_FCP, Save R2,R3,R4,R5,R6,R7,R8,R9       ; 0396
59      0000G   CF   9E 00002                                   MOVAB   IO_CHANNEL, R9                               ;
```

```
                58 00000000G   00  9E 00007          MOVAB   SYS$LKWSET, R8                                  ;
                57 00000000G   00  9E 0000E          MOVAB   SYS$ADJWSL, R7                                 ;
                56 00000000G   00  9E 00015          MOVAB   SYS$EXIT, R6                                   ;
                55 00000000G   00  9E 0001C          MOVAB   SYS$SETEXV, R5                                 ;
                5E             0C  C2 00023          SUBL2   #12, SP                                        ;
                         04    AE  9F 00026          PUSHAB  SYSEXV                                         ; 0472
                               01  DD 00029          PUSHL   #1                                             ;
                               7E  7C 0002B          CLRQ    -(SP)                                          ;
                65             04  FB 0002D          CALLS   #4, SYS$SETEXV                                 ;
                         08    AE  9F 00030          PUSHAB  SYSEXV+4                                       ; 0473
                               01  DD 00033          PUSHL   #1                                             ;
                7E             01  7D 00035          MOVQ    #1, -(SP)                                      ;
                65             04  FB 00038          CALLS   #4, SYS$SETEXV                                 ;
                         04    AE  D5 0003B          TSTL    SYSEXV                                         ; 0479
                               0B  15 0003E          BLEQ    1$                                             ;
                7E             01  7D 00040          MOVQ    #1, -(SP)                                      ; 0480
                         0C    AE  DD 00043          PUSHL   SYSEXV                                         ;
                               7E  D4 00046          CLRL    -(SP)                                          ;
                65             04  FB 00048          CALLS   #4, SYS$SETEXV                                 ;
                         08    AE  D5 0004B 1$:      TSTL    SYSEXV+4                                       ; 0481
                               0B  15 0004E          BLEQ    2$                                             ;
                7E             01  7D 00050          MOVQ    #1, -(SP)                                      ; 0482
                         10    AE  DD 00053          PUSHL   SYSEXV+4                                       ;
                               01  DD 00056          PUSHL   #1                                             ;
                65             04  FB 00058          CALLS   #4, SYS$SETEXV                                 ;
                53 00000000G   9F  D0 0005B 2$:      MOVL    @#IOC$GL_AQBLIST, AQB                          ; 0493
                               14  13 00062 3$:      BEQL    4$                                             ; 0494
                50 00000000G   9F  D0 00064          MOVL    @#SCH$GL_CURPCB, R0                            ;
          60    A0       0C    A3  D1 0006B          CMPL    12(AQB), 96(R0)                                ;
                               06  13 00070          BEQL    4$                                             ;
                53       10    A3  D0 00072          MOVL    16(AQB), AQB                                   ; 0495
                               EA  11 00076          BRB     3$                                             ;
                               53  D5 00078 4$:      TSTL    AQB                                            ; 0497
                               07  13 0007A          BEQL    5$                                             ;
                0000G  CF              53  D0 0007C          MOVL    AQB, QUEUE_HEAD                        ; 0498
                               08  11 00081          BRB     6$                                             ;
                7E       0314  8F  3C 00083 5$:      MOVZWL  #788, -(SP)                                    ; 0499
                66             01  FB 00088          CALLS   #1, SYS$EXIT                                   ;
                52 00000000G   9F  3C 0008B 6$:      MOVZWL  @#ACP$GW_WORKSET, WORKING_SET                  ; 0508
                               4B  12 00092          BNEQ    8$                                             ; 0509
                50 00000000G   9F  3C 00094          MOVZWL  @#ACP$GW_MAPCACHE, R0                          ; 0513
                51 00000000G   9F  3C 0009B          MOVZWL  @#ACP$GW_HDRCACHE, R1                          ;
                50             51  C0 000A2          ADDL2   R1, R0                                         ;
                54 00000000G   9F  3C 000A5          MOVZWL  @#ACP$GW_DIRCACHE, R4                          ; 0514
                50             54  C0 000AC          ADDL2   R4, R0                                         ;
                04             50  D1 000AF          CMPL    R0, #4                                         ; 0512
                               03  1E 000B2          BGEQU   7$                                             ;
                50             04  D0 000B4          MOVL    #4, R0                                         ;
                52             50  D0 000B7 7$:      MOVL    R0, WORKING_SET                                ;
          50    52 000000A1    8F  C5 000BA          MULL3   #161, WORKING_SET, R0                          ; 0515
                50     0FFF    C0  9E 000C2          MOVAB   4095(R0), R0                                   ;
                50 00001000    8F  C6 000C7          DIVL2   #4096, R0                                      ;
                52             50  C0 000CE          ADDL2   R0, WORKING_SET                                ;
                52             30  C0 000D1          ADDL2   #48, WORKING_SET                               ; 0516
          50    52 00000080    8F  C7 000D4          DIVL3   #128, WORKING_SET, R0                          ; 0517
                52             50  C0 000DC          ADDL2   R0, WORKING_SET                                ;
                02       15    A3  91 000DF 8$:      CMPB    21(AQB), #2                                    ; 0519
```

```
                                03  12 000E3          BNEQ    9$                                                      ;
                            52  06  C0 000E5          ADDL2   #6, WORKING_SET                                         ; 0520
                            7E  D4  000E8 9$:        CLRL    -(SP)                                                   ; 0522
                        7E  D8F0    8F  32 000EA      CVTWL   #-10000, -(SP)                                          ;
                            67  02  FB 000EF          CALLS   #2, SYS$ADJWSL                                          ;
                                7E  D4 000F2          CLRL    -(SP)                                                   ; 0523
                                52  DD 000F4          PUSHL   WORKING_SET                                             ;
                            67  02  FB 000F6          CALLS   #2, SYS$ADJWSL                                          ;
                                7E  7C 000F9          CLRQ    -(SP)                                                   ; 0529
                            FEC5    CF  9F 000FB      PUSHAB  P.AAA                                                   ;
                            68  03  FB 000FF          CALLS   #3, SYS$LKWSET                                          ;
                            52  50  D0 00102          MOVL    R0, STATUS                                              ;
                            05  52  E8 00105          BLBS    STATUS, 10$                                             ; 0530
                                52  DD 00108          PUSHL   STATUS                                                  ;
                            66  01  FB 0010A          CALLS   #1, SYS$EXIT                                            ;
                                7E  7C 0010D 10$:     CLRQ    -(SP)                                                   ; 0532
                            FEB9    CF  9F 0010F      PUSHAB  P.AAB                                                   ;
                            68  03  FB 00113          CALLS   #3, SYS$LKWSET                                          ;
                            52  50  D0 00116          MOVL    R0, STATUS                                              ;
                            05  52  E8 00119          BLBS    STATUS, 11$                                             ; 0533
                                52  DD 0011C          PUSHL   STATUS                                                  ;
                            66  01  FB 0011E          CALLS   #1, SYS$EXIT                                            ;
                    0000G   CF  00  FB 00121 11$:     CALLS   #0, INIT_POOL                                           ; 0535
                                69  D4 00126          CLRL    IO_CHANNEL                                              ; 0537
                            7E  01  7D 00128          MOVQ    #1, -(SP)                                               ; 0540
                                59  DD 0012B          PUSHL   R9                                                      ;
                            FEAF    CF  9F 0012D      PUSHAB  P.AAC                                                   ;
                00000000G   00  04  FB 00131          CALLS   #4, SYS$ASSIGN                                          ;
                            04  50  E8 00138          BLBS    R0, 12$                                                 ;
                                  FEFF 0013B          BUGW                                                            ; 0541
                                 0000* 0013D          .WORD   <BUG$_NOACPCHAN!4>                                      ;
                                69  DD 0013F 12$:     PUSHL   IO_CHANNEL                                              ; 0543
                        0000G   CF  01  FB 00141      CALLS   #1, GET_CCB                                             ;
                        0000G   CF  60  D0 00146      MOVL    (CCB), DISK_UCB                                         ; 0544
                1E 00000000G    9F  00G E2 0014B      BBSS    S^EXE$V_INIT, @#EXE$GL_FLAGS, 13$                       ; 0553
                                    FEA1    CF  9F 00153  PUSHAB  P.AAE                                               ; 0563
                                            01  DD 00157  PUSHL   #1                                                 ;
                                7E  FF0F    8F  3C 00159  MOVZWL  #65295, -(SP)                                       ;
                                7E  0708    8F  3C 0015E  MOVZWL  #1800, -(SP)                                        ;
                                            12  DD 00163  PUSHL   #18                                                 ;
                                    14      AE  9F 00165  PUSHAB  MBX_CHANNEL                                         ;
                                            7E  D4 00168  CLRL    -(SP)                                               ;
                        00000000G   00      07  FB 0016A  CALLS   #7, SYS$CREMBX                                     ;
                                50  0000G   CF  D0 00171 13$:     MOVL    QUEUE_HEAD, R0                            ; 0566
                            14  A0          08  8A 00176  BICB2   #8, 20(R0)                                        ;
                                            04  0017A  RET                                                           ; 0568

; Routine Size:  379 bytes,    Routine Base:  $CODE$ + 003C

;  256          0569  1
;  257          0570  1 END
;  258          0571  0 ELUDOM
```

```
;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  $LOCKEDC0$                       0  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(9)
;  $LOCKEDC9$                       0  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $LOCKEDD0$                       0  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $LOCKEDD9$                       0  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                         439  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        18         0        1000        00:01.9
;
;
;                                   COMMAND QUALIFIERS
;
;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:INIFCP/OBJ=OBJ$:INIFCP MSRC$:INIFCP/UPDATE=(ENH$:INIFCP)
;
; Size:         379 code + 60 data bytes
; Run Time:       00:10.7
; Elapsed Time:   00:33.4
; Lines/CPU Min:   3198
; Lexemes/CPU-Min: 20016
; Memory Used:  138 pages
; Compilation Complete
```

EXTFCB.LIS

DELFIL.LIS

DIRGET.LIS

EXTIDX.LIS

IODONE.LIS

LOCKDN.LIS

ENTER.LIS

GETREQ.LIS

GETTIM.LIS

DISPAT.LIS

INIFCP.LIS

DIRFCB.LIS

EXTHDR.LIS

DIRSCN.LIS

LOGDEL.LIS

LOCKDB.LIS

FIND.LIS

GETFIB.LIS

DIRACC.LIS

INIFCB.LIS

EXTDIR.LIS

EXTEND.LIS