
SSSSSSSSSSS DDDDDDDDDDD AAAAAAAAA
SSSSSSSSSSS DDDDDDDDDDD AAAAAAAAA
SSSSSSSSSSS DDDDDDDDDDD AAAAAAAAA
SSS DDD DDD AAA AAA
SSS DDD DDD AAA AAA
SSS DDD DDD AAA AAA
SSS DDD DDD AAA AAA
SSS DDD DDD AAA AAA
SSS DDD DDD AAA AAA
SSSSSSSSS DDD DDD AAA AAA
SSSSSSSSS DDD DDD AAA AAA
SSSSSSSSS DDD DDD AAA AAA
SSS DDD DDD AAAAAAAAAAAAAAA
SSS DDD DDD AAAAAAAAAAAAAAA
SSS DDD DDD AAAAAAAAAAAAAAA
SSS DDD DDD AAA AAA
SSS DDD DDD AAA AAA
SSS DDD DDD AAA AAA
SSSSSSSSSSS DDDDDDDDDDD AAA AAA
SSSSSSSSSSS DDDDDDDDDDD AAA AAA
SSSSSSSSSSS DDDDDDDDDDD AAA AAA

```
PPPPPPPP    RRRRRRR      000000      CCCCCCCC  EEEEEEEEEE    SSSSSSSS    SSSSSSSS
PPPPPPPP    RRRRRRRR     000000      CCCCCCCC  EEEEEEEEEE    SSSSSSSS    SSSSSSSS
PP      PP  RR      RR   00      00  CC        EE            SS          SS
PP      PP  RR      RR   00      00  CC        EE            SS          SS
PP      PP  RR      RR   00      00  CC        EE            SS          SS
PP      PP  RR      RR   00      00  CC        EE            SS          SS
PPPPPPPP    RRRRRRRR     00      00  CC        EEEEEEEE        SSSSSS      SSSSSS
PPPPPPPP    RRRRRRRR     00      00  CC        EEEEEEEE        SSSSSS      SSSSSS
PP          RR  RR       00      00  CC        EE                    SS          SS
PP          RR  RR       00      00  CC        EE                    SS          SS
PP          RR    RR     00      00  CC        EE                    SS          SS      ....
PP          RR    RR     00      00  CC        EE                    SS          SS      ....
PP          RR      RR     000000      CCCCCCCC  EEEEEEEEEE  SSSSSSSS    SSSSSSSS        ....
PP          RR      RR     000000      CCCCCCCC  EEEEEEEEEE  SSSSSSSS    SSSSSSSS        ....

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
```

Table of contents

```
0000     1          .TITLE  PROCESS PROCESS FORMATTING ROUTINES
0000     2          .SBTTL  COPYRIGHT NOTICE
0000     3          .IDENT  'V04-000'
0000     4 ;
0000     5 ;****************************************************************************
0000     6 ;*                                                                          *
0000     7 ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000     8 ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0000     9 ;*  ALL RIGHTS RESERVED.                                                    *
0000    10 ;*                                                                          *
0000    11 ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0000    12 ;*  ONLY IN ACCORDANCE WITH THE TERMS OF SUCH LICENSE AND WITH THE          *
           13 ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR ANY OTHER     *
           14 ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0000    15 ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF THE SOFTWARE IS HEREBY      *
0000    16 ;*  TRANSFERRED.                                                            *
0000    17 ;*                                                                          *
0000    18 ;*  THE INFORMATION IN THIS SOFTWARE IS SUBJECT TO CHANGE WITHOUT NOTICE    *
0000    19 ;*  AND SHOULD NOT BE CONSTRUED AS A COMMITMENT BY DIGITAL EQUIPMENT        *
0000    20 ;*  CORPORATION.                                                            *
0000    21 ;*                                                                          *
0000    22 ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE OR RELIABILITY OF ITS     *
0000    23 ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0000    24 ;*                                                                          *
0000    25 ;*                                                                          *
0000    26 ;****************************************************************************
0000    27 ;
```

```
0000     29          .SBTTL  PROGRAM DESCRIPTION
0000     30 ;++
0000     31 ;   FACILITY
0000     32 ;
0000     33 ;       SYSTEM DUMP ANALYZER
0000     34 ;
0000     35 ;   ABSTRACT
0000     36 ;
0000     37 ;       ROUTINES TO FORMAT PROCESS INFORMATION
0000     38 ;
0000     39 ;   ENVIRONMENT
0000     40 ;
0000     41 ;       NATIVE MODE, USER MODE
0000     42 ;
0000     43 ;   AUTHOR
0000     44 ;
0000     45 ;       TIM HALVORSEN, JULY 1978
0000     46 ;
0000     47 ;   MODIFIED BY
0000     48 ;
0000     49 ;       V03-013 MSH0072         Michael S. Harvey       3-Aug-1984
0000     50 ;               Correct some bugs causing errors getting data for
0000     51 ;               processes other than your own.
0000     52 ;
0000     53 ;       V03-012 EMB0101         Ellen M. Batbouta       14-Jun-1984
0000     54 ;               Decode MWAIT so as to correctly display the resource
0000     55 ;               a process is waiting for when a SHOW PROCESS or SHOW
0000     56 ;               SUMMARY command is issued.  Show the full device name
0000     57 ;               on a SHOW PROCESS/CHANNELS display.
0000     58 ;
0000     59 ;       V03-011 PRB0314         Paul Beck       28-Feb-1984  16:48
0000     60 ;               Use longword formatting for BUFIO byte count/limit.
0000     61 ;
0000     62 ;       V03-010 WMC0002         Wayne Cardoza           02-Dec-1983
0000     63 ;               Some PHD fields have changed to longwords
0000     64 ;
0000     65 ;       V03-009 WMC0001         Wayne Cardoza           11-Jul-1983
0000     66 ;               Support separate display of process P0 and P1 page tables.
0000     67 ;
0000     68 ;       V03-008 TCM0001         Trudy C. Matthews       12-Apr-1983
0000     69 ;               Make sure references to working set list offsets in the
0000     70 ;               process header are treated as unsigned words.
0000     71 ;
0000     72 ;       V03-007 CWH1004         CW Hobbs                26-Mar-1983
0000     73 ;               Run process names through !AF directive to keep escape
0000     74 ;               sequences from the displays.
0000     75 ;
0000     76 ;       V03-006 CWH1003         CW Hobbs                23-Mar-1983
0000     77 ;               Shuffle items in SHOW SUMMARY a little.
0000     78 ;
0000     79 ;       V03-005 CWH1002         CW Hobbs                1-Mar-1983
0000     80 ;               Display extended PIDs in heading and SHOW PROCESS /PCB
0000     81 ;               display.  Change SHOW SUMMARY to add /IMAGE qualifier
0000     82 ;               and alter items in summary line.
0000     83 ;
0000     84 ;       V03-004 MSH0002         Maryann Hinden          22-Oct-1982
0000     85 ;               Pass extra argument to SHOW_PROC_LOCK.
```

```
0000     86 ;
0000     87 ;       V03-003 LMP0033         L. Mark Pilant,         25-Jun-1982  15:36
0000     88 ;               Fix a problem introduced by KTA0103.  Also, fix a bug that
0000     89 ;               caused a BADPROC error message to appear if a display for all
0000     90 ;               processes terminated abnormally.
0000     91 ;
0000     92 ;       V03-002 LMP0031         L. Mark Pilant,         17-Jun-1982  13:25
0000     93 ;               Add support for SHOW PROCESS/CHANNELS.
0000     94 ;
0000     95 ;       V03-001 MSH0001         Maryann Hinden          10-Jun-1982
0000     96 ;               Add SHOW PROC/LOCK.
0000     97 ;
0000     98 ;--
```

```
                        .SBTTL  DECLARATIONS
                ;
                ;       SYMBOL DEFINITIONS
                ;
                        $PCBDEF                 ; PROCESS CONTROL BLOCK DEFINITIONS
                        $PHDDEF                 ; PROCESS HEADER DEFINITIONS
                        $WSLDEF                 ; WORKING SET LIST DEFINITIONS
                        $SECDEF                 ; SECTION TABLE DEFINITIONS
                        $IFDDEF                 ; IMAGE FILE DESCRIPTOR DEFINITIONS
                        $JIBDEF                 ; JOB INFORMATION BLOCK DEFINITIONS
                        $OPTDEF                 ; OPTION DEFINITIONS
                        $CCBDEF                 ; CHANNEL CONTROL BLOCK OFFSETS
                        $DDBDEF                 ; DEVICE DATA BLOCK OFFSETS
                        $DYNDEF                 ; DYNAMIC BLOCK TYPE CODES
                        $FCBDEF                 ; FCB OFFSETS
                        $RSNDEF                 ; RESOURCE NAME DEFINITIONS
                        $SBDEF                  ; SYSTEM BLOCK DEFINITIONS
                        $STSDEF                 ; STATUS FIELD DEFINITIONS
                        $UCBDEF                 ; UCB OFFSETS
                        $WCBDEF                 ; WCB OFFSETS

                ;
                ;       LOCAL MACRO DEFINITIONS
                ;

                        .MACRO  CSTABLE,LEN
                        CSL =   LEN
                        .ENDM


                        .MACRO  CS,TEXT
                        SAV = .
                        .ASCIC  \TEXT\
                        . = SAV + CSL
                        .ENDM
```

```
                     0000    136          .SBTTL  STORAGE DEFINITIONS
                     0000    137 ;
                     0000    138 ;        STORAGE DEFINITIONS
                     0000    139 ;
                     0000    140
                00000000     141          .PSECT  SDADATA,NOEXE,WRT
                     0000    142
                     0000    143          .default displacement,long
                     0000    144
                     0000    145 PCBADR::
00000004             0000    146          .BLKL   1                       ; ADDRESS OF PCB
                     0004    147
                     0004    148 PCB::
00000124             0004    149          .BLKB   PCB$C_LENGTH            ; BUFFER TO HOLD PCB
                     0124    150
                     0124    151 PHDADR::
00000128             0124    152          .BLKL   1                       ; ADDRESS OF PHD
                     0128    153
                     0128    154 PHD::
000002A4             0128    155          .BLKB   PHD$C_LENGTH            ; BUFFER TO HOLD PHD
                     02A4    156
                     02A4    157 JIBADR::
000002A8             02A4    158          .BLKL   1                       ; ADDRESS OF JIB
                     02A8    159 JIB::
0000031C             02A8    160          .BLKB   JIB$C_LENGTH            ; BUFFER TO HOLD JIB
                     031C    161
                     031C    162 CHANTBL_SIZE:
00000320             031C    163          .BLKL   1                       ; SIZE OF THE CHANNEL TABLE
                     0320    164 CHANTBL_ADDR:
00000324             0320    165          .BLKL   1                       ; ADDRESS OF THE CHANNEL TABLE COPY
                     0324    166 CHANNEL_DEVICE:
                     0324    167 CHANDEV_SIZE:
00000328             0324    168          .BLKL   1                       ; SIZE OF DEVICE NAME FOR CHANNEL
0000032D'            0328    169          .ADDRESS        CHANDEV_NAME+1  ; ADDRESS OF NAME STORAGE
                     032C    170 CHANDEV_NAME:
0000033C             032C    171          .BLKB   16                      ; STORAGE FOR THE DEVICE NAME
                     033C    172 FILE_NAME:
00000084             033C    173          .LONG   132                     ; FILE NAME DESCRIPTOR
00000344'            0340    174          .ADDRESS        FILE_NAMETXT
                     0344    175 FILE_NAMETXT:
000003C8             0344    176          .BLKB   132
                     03C8    177 NODE:
000003D8             03C8    178          .BLKB   SB$S_NODENAME           ; NODE NAME
                     03D8    179 FCB:
0000048C             03D8    180          .BLKB   FCB$C_LENGTH            ; STORAGE FOR THE FCB
                     048C    181 WCB:
000004BC             048C    182          .BLKB   WCB$C_LENGTH            ; STORAGE FOR THE WCB (NO MAP POINTERS)
                     04BC    183
                00000000     184          .PSECT  PROCESS,EXE,NOWRT
                     0000    185
                     0000    186          .DEFAULT DISPLACEMENT,LONG
```

```
                                0000   188          .SBTTL  READ-ONLY DATA DEFINITIONS
                                0000   189
                                0000   190 ;
                                0000   191 ;       READ-ONLY DATA DEFINITIONS
                                0000   192 ;
                                0000   193
                                0000   194 NOTAVAIL:
61 6E 20 65 67 61 6D 69 20 2D 2D 00'    0000   195          .ASCIC  '-- image name not available --'
6C 69 61 76 61 20 74 6F 6E 20 65 6D     000C
          2D 2D 20 65 6C 62 61          0018
                             1E         0000
                                001F   196
                                001F   197          .MACRO  $DEFINI NAME,P1,P2
                                001F   198          .ENDM
                                001F   199          .MACRO  $DEFEND NAME,P1,P2
                                001F   200          .ENDM
                                001F   201          .MACRO  $EQU SYMBOL,VALUE
                                001F   202 S = %LOCATE(< >,SYMBOL)+1
                                001F   203 L = %LENGTH(SYMBOL)-S
                                001F   204          .ASCIC  /%EXTRACT(S,L,SYMBOL)/
                                001F   205          .BLKB   8-<L+1>
                                001F   206          .ENDM   $EQU
                                001F   207
                                001F   208 STATE_TABLE:
        4C 41 47 45 4C 4C 49 00'    001F   209          .ASCIC  'ILLEGAL'                       ; FOR ZERO CASE
                             07     001F
                                0027   210          $STATEDEF                               ; TABLE OF PROCESS STATE NAMES
                                0097   211
                         00000003   0097   212          .MDELETE        $DEFINI,$DEFEND,$EQU
                                0097   213
                                0097   214          $STATEDEF
                                0097   215
                                0097   216 PROCESS_STATUS:
                                0097   217          TABLE   PCB$V_,<RES,DELPEN,FORCPEN,INQUAN,PSWAPM,RESPEN,SSFEXC,-
                                0097   218                  SSFEXCE,SSFEXCS,SSFEXCU,SSRWAIT,SUSPEN,WAKEPEN,WALL,-
                                0097   219                  BATCH,NOACNT,SWPVBN,ASTPEN,PHDRES,HIBER,LOGIN,NETWRK,-
                                0097   220                  PWRAST,NODELET>
                                015F   221
                                015F   222 ;
                                015F   223 ; Below is a complete list of the resources a process could be waiting for
                                015F   224 ; if it is in mwait state.  The mutex resource is a special case.  It is
                                015F   225 ; detected by examining the field, PCB$L_EFWM, in the pcb.  This field will
                                015F   226 ; contain the address of the mutex if the process is waiting for one.
                                015F   227 ; Otherwise this field will contain the number of the resource when in mwait.
                                015F   228 ;
                                015F   229
                                015F   230 ASSUME  RSN$_MAX  EQ   15
                                015F   231
                                015F   232 RESOURCE_WAIT:
                                015F   233          ADDR_TABLE      RSN$_,<-
                                015F   234                          <ASTWAIT,AST>,-
                                015F   235                          <MAILBOX,MBX>,-
                                015F   236                          <NPDYNMEM,NONPAGED>,-
                                015F   237                          <PGFILE,PAGING>,-
                                015F   238                          <PGDYNMEM,PAGED>,-
                                015F   239                          <BRKTHRU,BREAK>,-
                                015F   240                          <IACLOCK,IMGACT>,-
```

```
                     015F   241                     <JQUOTA,JOB_QUOTA>,-
                     015F   242                     <LOCKID,LOCKS>,-
                     015F   243                     <SWPFILE,SWAP>,-
                     015F   244                     <MPLEMPTY,MPAGE_EMPTY>,-
                     015F   245                     <MPWBUSY,MPAGE_BUSY>,-
                     015F   246                     <SCS,SCSWAIT>,-
                     015F   247                     <CLUSTRAN,CLUSTER>-
                     015F   248                     >
                     01D7   249
                     01D7   250 AST:
54 53 41 57 52 00'   01D7   251         .ASCIC      /RWAST/                 ; RESOURCE WAIT FOR AST COMPLETION
                05   01D7
                     01DD   252
                     01DD   253 MBX:
58 42 4D 57 52 00'   01DD   254         .ASCIC      /RWMBX/                 ; MAILBOX FULL
                05   01DD
                     01E3   255
                     01E3   256 NONPAGED:
47 50 4E 57 52 00'   01E3   257         .ASCIC      /RWNPG/                 ; NONPAGED DYNAMIC MEMORY
                05   01E3
                     01E9   258
                     01E9   259 PAGING:
46 46 50 57 52 00'   01E9   260         .ASCIC      /RWPFF/                 ; PAGE FILL FULL
                05   01E9
                     01EF   261
                     01EF   262 PAGED:
47 41 50 57 52 00'   01EF   263         .ASCIC      /RWPAG/                 ; PAGED DYNAMIC MEMORY
                05   01EF
                     01F5   264
                     01F5   265 BREAK:
4B 52 42 57 52 00'   01F5   266         .ASCIC      /RWBRK/                 ; BREAKTHROUGH
                05   01F5
                     01FB   267
                     01FB   268 IMGACT:
47 4D 49 57 52 00'   01FB   269         .ASCIC      /RWIMG/                 ; IMAGE ACTIVATION LOCK
                05   01FB
                     0201   270
                     0201   271 JOB_QUOTA:
4F 55 51 57 52 00'   0201   272         .ASCIC      /RWQUO/                 ; JOB POOLED QUOTA
                05   0201
                     0207   273
                     0207   274 LOCKS:
4B 43 4C 57 52 00'   0207   275         .ASCIC      /RWLCK/                 ; LOCK IDENTIFICATION DATABASE
                05   0207
                     020D   276
                     020D   277 SWAP:
50 57 53 57 52 00'   020D   278         .ASCIC      /RWSWP/                 ; SWAP FILE SPACE
                05   020D
                     0213   279
                     0213   280 MPAGE_EMPTY:
45 50 4D 57 52 00'   0213   281         .ASCIC      /RWMPE/                 ; MODIFIED PAGE LIST EMPTY
                05   0213
                     0219   282
                     0219   283 MPAGE_BUSY:
42 50 4D 57 52 00'   0219   284         .ASCIC      /RWMPB/                 ; MODIFIED PAGE LIST BUSY
                05   0219
                     021F   285
```

```
                        021F   286 SCSWAIT:
     53 43 53 57 52 00' 021F   287     .ASCIC      /RWSCS/                 ; SYSTEM COMMUNICATIONS
                     05 021F
                        0225   288
                        0225   289 CLUSTER:
     55 4C 43 57 52 00' 0225   290     .ASCIC      /RWCLU/                 ; CLUSTER STATE TRANSITION
                     05 0225
                        022B   291
                        022B   292 MWAIT:                                 ; Process is waiting for the resource,
     58 45 54 55 4D 00' 022B   293     .ASCIC      /MUTEX/                 ; mutex.
                     05 022B
                        0231   294
                        0231   295 GBL_TABLE:
                     00' 0231   296         .ASCIC  ''
                     00 0231
        20 4C 42 47 00' 0232   297         .ASCIC  'GBL '
                     04 0232
                        0237   298 CRF_TABLE:
                     00' 0237   299         .ASCIC  ''
                     00 0237
        20 46 52 43 00' 0238   300         .ASCIC  'CRF '
                     04 0238
                        023D   301 DZRO_TABLE:
                     00' 023D   302         .ASCIC  ''
                     00 023D
  20 4F 52 45 5A 44 00' 023E   303         .ASCIC  'DZERO '
                     06 023E
                        0245   304 WRT_TABLE:
                     00' 0245   305         .ASCIC  ''
                     00 0245
        20 54 52 57 00' 0246   306         .ASCIC  'WRT '
                     04 0246
                        024B   307
                        024B   308 VALID_TABLE:
                     00' 024B   309         .ASCIC  ''
                     00 024B
     44 49 4C 41 56 20 00' 024C   310         .ASCIC  ' VALID'
                     06 024C
                        0253   311 PAGTYP_TABLE:
  53 53 45 43 4F 52 50 00' 0253   312         .ASCIC  'PROCESS'
                     07 0253
  20 4D 45 54 53 59 53 00' 025B   313         .ASCIC  'SYSTEM '
                     07 025B
  20 4C 41 42 4F 4C 47 00' 0263   314         .ASCIC  'GLOBAL '
                     07 0263
  20 54 52 57 4C 42 47 00' 026B   315         .ASCIC  'GBLWRT '
                     07 026B
  20 4C 42 54 47 50 50 00' 0273   316         .ASCIC  'PPGTBL '
                     07 0273
  20 4C 42 54 47 50 47 00' 027B   317         .ASCIC  'GPGTBL '
                     07 027B
  20 20 20 20 20 20 20 00' 0283   318         .ASCIC  '       '
                     07 0283
  20 20 20 20 20 20 20 00' 028B   319         .ASCIC  '       '
                     07 028B
                        0293   320 MODIFY_TABLE:
                     00' 0293   321         .ASCIC  ''
```

```
                                00  0293
      44 45 49 46 49 44 4F 4D 20 00' 0294    322          .ASCIC  ' MODIFIED'
                                09  0294
                                    029E    323 WSLOCK_TABLE:
                                00' 029E    324          .ASCIC  ''
                                00  029E
            4B 43 4F 4C 53 57 20 00' 029F    325          .ASCIC  ' WSLOCK'
                                07  029F
                                    02A7    326 PFNLOCK_TABLE:
                                00' 02A7    327          .ASCIC  ''
                                00  02A7
         4B 43 4F 4C 4E 46 50 20 00' 02A8    328          .ASCIC  ' PFNLOCK'
                                08  02A8
                                    02B1    329 GOODPAGE_TABLE:
                                00' 02B1    330          .ASCIC  ''
                                00  02B1
      45 47 41 50 44 4F 4F 47 20 00' 02B2    331          .ASCIC  ' GOODPAGE'
                                09  02B2
                                    02BC    332
                                    02BC    333 CHANBUSY_TABLE:
                                00' 02BC    334          .ASCIC  ''
                                00  02BC
                  20 79 73 75 42 00' 02BD    335          .ASCIC  'Busy '
                                05  02BD
                                    02C3    336 CHANDPND_TABLE:
                                00' 02C3    337          .ASCIC  ''
                                00  02C3
                  20 64 6E 70 44 00' 02C4    338          .ASCIC  'Dpnd '
                                05  02C4
                                    02CA    339 CHANSEC_TABLE:
                                00' 02CA    340          .ASCIC  ''
                                00  02CA
66 20 6E 6F 69 74 63 65 73 28 20 00' 02CB    341          .ASCIC  ' (section file)'
                        29 65 6C 69  02D7
                                0F  02CB
                                    02DB    342
                                    02DB    343 AST_TABLE:
                                    02DB    344          CSTABLE 8
                                    02DB    345          CS      NONE
                                    02E3    346          CS      K
                                    02EB    347          CS      E
                                    02F3    348          CS      KE
                                    02FB    349          CS      S
                                    0303    350          CS      KS
                                    030B    351          CS      ES
                                    0313    352          CS      KES
                                    031B    353          CS      U
                                    0323    354          CS      KU
                                    032B    355          CS      EU
                                    0333    356          CS      KEU
                                    033B    357          CS      SU
                                    0343    358          CS      KSU
                                    034B    359          CS      ESU
                                    0353    360          CS      KESU
                                    035B    361
                                    035B    362 HDRSTR1:
                                    035B    363          STRING  <Process index: !XW   Name: !AF   Extended PID: !XL>
```

```
                     0395   365          .SBTTL  DISPLAY_PROCS -- DISPLAY PROCESS DATA
                     0395   366 ;---
                     0395   367 ;
                     0395   368 ;       DISPLAY_PROCS
                     0395   369 ;
                     0395   370 ;       SHOWS INFORMATION ABOUT EACH PROCESS IN THE SYSTEM.
                     0395   371 ;
                     0395   372 ;   INPUTS:
                     0395   373 ;
                     0395   374 ;       NONE
                     0395   375 ;
                     0395   376 ;   OUTPUTS:
                     0395   377 ;
                     0395   378 ;       NONE
                     0395   379 ;
                     0395   380 ;---
                     0395   381
                     0395   382 DISPLAY_PROCS::
                001C 0395   383          .WORD   ^M<R2,R3,R4>
                     0397   384
                     0397   385          REQMEM  @SCH$GL_PCBVEC,R2       ; VECTOR OF PCB ADDRESSES
                     03A7   386          REQMEM  @SCH$GL_MAXPIX,R3       ; GET MAXIMUM PROCESS INDEX
                     03B7   387
              54     D4 03B7   388          CLRL    R4                      ; INITIALIZE CURRENT INDEX
00000000'EF     D4 03B9   389          CLRL    PROC_NAME               ;SET TO PREVIOUS PROCESS IN CASE OF ERROR
                     03BF   390 10$:
  51   6244     DE 03BF   391          MOVAL   (R2)[R4],R1             ; ADDRESS OF POINTER TO PCB
                     03C3   392          REQMEM  (R1),PCBADR             ; GET ADDRESS OF PCB
                     03D3   393          REQMEM  @PCBADR,PCB,#PCB$C_LENGTH       ; GET PCB
00000064'EF     B5 03EC   394          TSTW    PCB+PCB$L_PID           ; CHECK IF PROCESS INDEX = 0
          04     12 03F2   395          BNEQ    20$                     ; BRANCH IF NOT
          54     D5 03F4   396          TSTL    R4                      ; CHECK IF NULL PROCESS SLOT
          50     12 03F6   397          BNEQ    90$                     ; SKIP IF NOT
                     03F8   398 20$:
          00000070'EF     D5 03F8   399          TSTL    PCB+PCB$L_PHD           ; SWAPPED OUT?
                  39     13 03FE   400          BEQL    30$                     ; BRANCH IF SO
                     0400   401          REQMEM  @PCB+PCB$L_PHD,PHD,#PHD$C_LENGTH ; READ PHD
       50   00000000'EF     DE 0419   402          MOVAL   P0BR,R0                 ; 4 LONGWORDS
       51   00000128'EF     9E 0420   403          MOVAB   PHD,R1
          60   00C8 C1     7D 0427   404          MOVQ    PHD$L_P0BR(R1),(R0)     ; SETUP RELOCATION REGISTERS
        08 A0   00D0 C1     7D 042C   405          MOVQ    PHD$L_P1BR(R1),8(R0)
  04 A0   04 A0   18   00     EF 0432   406          EXTZV   #PHD$V_P0LR,#PHD$S_P0LR,4(R0),4(R0)
00000000'EF   00000064'EF     D0 0439   407 30$:    MOVL    PCB+PCB$L_PID,PROC_PID  ; SET PID OF PROCESS TO READ
         59'AF   00     FB 0444   408          CALLS   #0,B^DISPLAY_PROCESS    ; DISPLAY PROCESS INFORMATION
                     0448   409 90$:
         02 54   53     F3 0448   410          AOBLEQ  R3,R4,100$              ; CONTINUE UNTIL DONE
                03     11 044C   411          BRB     110$
              FF6E     31 044E   412 100$:    BRW     10$
                     0451   413 110$:    STATUS  SUCCESS
                  04 0458   414          RET
```

```
                                0459   416          .SBTTL  DISPLAY_PROCESS -- DISPLAY PROCESS INFORMATION
                                0459   417 ;---
                                0459   418 ;
                                0459   419 ;       DISPLAY_PROCESS
                                0459   420 ;
                                0459   421 ;       THIS ROUTINE FORMATS ALL INFORMATION ABOUT A SINGLE
                                0459   422 ;       PROCESS GIVEN THE PROCESS CONTROL BLOCK.
                                0459   423 ;
                                0459   424 ;   INPUTS:
                                0459   425 ;
                                0459   426 ;       PCB = PROCESS CONTROL BLOCK FOR PROCESS
                                0459   427 ;
                                0459   428 ;   OUTPUTS:
                                0459   429 ;
                                0459   430 ;       NONE
                                0459   431 ;
                                0459   432 ;---
                                0459   433
                                0459   434 DISPLAY_PROCESS::
                         003C   0459   435          .WORD   ^M<R2,R3,R4,R5>
      0074 8F    00    6E    00   2C   045B   436          MOVC5   #0,(SP),#0,#JIB$C_LENGTH,JIB    ; PRE-ZERO JIB IN CASE NOT THERE
                  000002A8'EF        0462
 000002A4'EF      00000084'EF   D0   0467   437          MOVL    PCB+PCB$L_JIB,JIBADR    ; CHECK FOR JIB
                           19   13   0472   438          BEQL    5$                      ; BR IF NONE
                                0474   439          REQMEM  @JIBADR,JIB,#JIB$C_LENGTH       ; GET JIB
                                048D   440 5$:
            52   00000004'EF    DE   048D   441          MOVAL   PCB,R2
            53   00000128'EF    DE   0494   442          MOVAL   PHD,R3
                                049B   443          ALLOC   80,R4                   ; 80 BYTE BUFFER FOR HEADER
                     64 A2      DD   04AD   444          PUSHL   PCB$L_EPID(R2)          ; EXTENDED PROCESS ID
                     71 A2      9F   04B0   445          PUSHAB  PCB$T_LNAME+1(R2)       ; PROCESS NAME
                  7E 70 A2      9A   04B3   446          MOVZBL  PCB$T_LNAME(R2),-(SP)   ; PROCESS NAME
                     60 A2      DD   04B7   447          PUSHL   PCB$L_PID(R2)           ; PROCESS INDEX
                        54      DD   04BA   448          PUSHL   R4
                        54      DD   04BC   449          PUSHL   R4
                   FE99 CF      9F   04BE   450          PUSHAB  HDRSTR1                 ; FAO CONTROL STRING
 00000000'GF           07      FB   04C2   451          CALLS   #7,G^SYS$FAO            ; FORMAT TITLE STRING
                        54      DD   04C9   452          PUSHL   R4
 00000000'EF            01      FB   04CB   453          CALLS   #1,SET_HEADING          ; SET NEW HEADING
                                04D2   454          SKIP    PAGE
                                04D9   455
 54   00000000'EF      00   D0   04D9   456          MOVL    OPTIONS,R4
 54   FFFFFFBF         8F   D3   04E0   457          BITL    #^C<OPT$M_SYSPROC>,R4   ; CHECK IF ANY OPTION SPECIFIED
                        04   13   04E7   458          BEQL    10$                     ; DEFAULT IS PCB
                  07 54      04   E1   04E9   459          BBC     #OPT$V_PCB,R4,20$       ; CHECK IF /PCB SPECIFIED
                                04ED   460 10$:
 000005B0'EF            00   FB   04ED   461          CALLS   #0,SHOW_PCB             ; SHOW SOME PCB FIELDS
                                04F4   462 20$:
 54   0000042F         8F   D3   04F4   463          BITL    #<OPT$M_PHD! -          ; IF ANY OF THESE OPTIONS CHOSEN
                                04FB   464                  OPT$M_REGS! -
                                04FB   465                  OPT$M_WSL! -
                                04FB   466                  OPT$M_F  ! -
                                04FB   467                  OPT$M_PRT! -
                                04FB   468                  OPT$M_CHAN>,R4
                        51   13   04FB   469          BEQL    70$                     ; SKIP READING PHD IF NONE
                     6C A2      D5   04FD   470          TSTL    PCB$L_PHD(R2)           ; PROCESS RESIDENT?
                        03   12   0500   471          BNEQ    25$                     ; XFER IF SO
```

```
                 008B   31  0502   472          BRW     100$                            ; ELSE GIVE A WARNING
                        0505   473 25$:     REQMEM  @PCB$L_PHD(R2),(R3),#PHD$C_LENGTH  ; READ PROCESS HEADER
                        0517   474
          07 54    05   E1  0517   475          BBC     #OPT$V_PHD,R4,30$               ; CHECK IF /PHD SPECIFIED
000007BA'EF    00    FB  051B   476          CALLS   #0,SHOW_PHD                     ; SHOW PROCESS HEADER
          07 54    03   E1  0522   477 30$:     BBC     #OPT$V_REGS,R4,40$              ; CHECK IF /REGS SPECIFIED
00000934'EF    00    FB  0526   478          CALLS   #0,SHOW_REGS                    ; SHOW SAVED REGISTERS
          07 54    00   E1  052D   479 40$:     BBC     #OPT$V_WSL,R4,50$               ; CHECK IF /WSL SPECIFIED
00000A31'EF    00    FB  0531   480          CALLS   #0,SHOW_WSL                     ; SHOW WORKING SET LIST
          07 54    02   E1  0538   481 50$:     BBC     #OPT$V_PST,R4,60$               ; CHECK IF /PST SPECIFIED
00000C35'EF    00    FB  053C   482          CALLS   #0,SHOW_PST                     ; SHOW PROCESS SECTION TABLE
          07 54    01   E1  0543   483 60$:     BBC     #OPT$V_PPT,R4,70$               ; CHECK IF /PPT SPECIFIED
00000DD7'EF    00    FB  0547   484          CALLS   #0,SHOW_PPT                     ; SHOW PROCESS PAGE TABLES
          14 54    07   E1  054E   485 70$:     BBC     #OPT$V_RMS,R4,80$               ; CHECK IF /RMS SPECIFIED
          09 54    08   E0  0552   486          BBS     #OPT$V_RMSD,R4,71$              ; CHECK IF /RMS=... SPECIFIED
00000000'EF    00    FB  0556   487          CALLS   #0,SHOW_RMS                     ; SHOW CURRENTLY SELECTED RMS STRUCTS
                 07   11  055D   488          BRB     80$                             ; GO ON TO NEXT OPTION
00000000'EF    00    FB  055F   489 71$:     CALLS   #0,SHOW_RMS_DIS                 ; SHOW RMS STURCTS SPECIFIED ON
                        0566   490                                                  ;  THIS COMMAND
          13 54    0C   E1  0566   491 80$:     BBC     #OPT$V_LCK,R4,85$               ; CHECK IF /LOCK SPECIFIED
     00000000'          9F  056A   492          PUSHAB  PCBADR                          ; PASS ON ADDRESS OF PCB IN SYSTEM
     00000004'GF        9F  0570   493          PUSHAB  PCB                             ; PASS ON ADDRESS OF PCB COPY IN SDA
00000000'EF    02    FB  0576   494          CALLS   #2,SHOW_PROC_LOCK               ; SHOW LOCKS OWNED BY PROCESS
                        057D   495
          07 54    0A   E1  057D   496 85$:     BBC     #OPT$V_CHAN,R4,90$              ; CHECK IF /CHANNELS SPECIFIED
00000F94'EF    00    FB  0581   497          CALLS   #0,SHOW_CHANNELS                ; DISPLAY PROCESS CHANNEL INFORMATION
                        0588   498
                        0588   499 90$:     STATUS  SUCCESS                         ; EXIT
                    04  058F   500          RET
                        0590   501
                        0590   502 100$:    SKIP    1                               ; IN CASE ALL PROCESSES ARE DISPLAYED
     50 00000000'8F    D0  0599   503          MOVL    #<<MSG$_NOTRES&^XFFFFFFF8>!STS$K_WARNING>,R0    ; SET STATUS
                        05A0   504          SIGNAL  1                               ; SIGNAL WARNING
                     9E 11  05AE   505          BRB     70$                             ; MUST RETURN SUCCESS TO TPARSE
```

```
                        05B0   507          .SBTTL  SHOW_PCB -- SHOW PROCESS CONTROL BLOCK
                        05B0   508 ;---
                        05B0   509 ;
                        05B0   510 ;        SHOW_PCB
                        05B0   511 ;
                        05B0   512 ;        THIS ROUTINE FORMATS SOME INTERESTING FIELDS IN THE
                        05B0   513 ;        PROCESS CONTROL BLOCK FOR EASY READING.
                        05B0   514 ;
                        05B0   515 ;    INPUTS:
                        05B0   516 ;
                        05B0   517 ;        PCB = CURRENT PROCESS CONTROL BLOCK
                        05B0   518 ;
                        05B0   519 ;    OUTPUTS:
                        05B0   520 ;
                        05B0   521 ;        NONE
                        05B0   522 ;
                        05B0   523 ;---
                        05B0   524
                        05B0   525 SHOW_PCB:
                  0004  05B0   526          .WORD   ^M<R2>
                        05B2   527
52  00000004'EF   DE    05B2   528          MOVAL   PCB,R2
                        05B9   529          ENSURE  12
                        05D1   530          ALLOC   80                      ; 80 BYTE OUTPUT BUFFER
          24 A2   DD    05E0   531          PUSHL   PCB$L_STS(R2)           ; PROCESS STATUS
        FAB0 CF   9F    05E3   532          PUSHAB  PROCESS_STATUS          ; BIT DEFINITION TABLE
00000000'EF   02   FB   05E7   533          CALLS   #2,TRANSLATE_BITS       ; TRANSLATE BITS INTO NAMES
           5E   DD    05EE   534          PUSHL   SP                      ; ADDRESS OF RESULT DESCRIPTOR
          24 A2   DD    05F0   535          PUSHL   PCB$L_STS(R2)
                        05F3   536          PRINT   2,<Process status:  !XL   !AS>
                        0600   537          SKIP    1
                        0609   538
000002A4'EF   DD    0609   539          PUSHL   JIBADR                  ; ADDRESS OF JIB
00000000'EF   DD    060F   540          PUSHL   PCBADR                  ; ADDRESS OF PCB
                        0615   541          PRINT   2,<-
                        0615   542 PCB address               !XL    -
                        0615   543 JIB address               !XL>
                        0622   544
          20 A2   DD    0622   545          PUSHL   PCB$L_WSSWP(R2)
          6C A2   DD    0625   546          PUSHL   PCB$L_PHD(R2)
                        0628   547          PRINT   2,<-
                        0628   548 PHD address               !XL    -
                        0628   549 Swapfile disk address    !XL>
                        0635   550
000002EC'EF   DD    0635   551          PUSHL   JIB+JIB$W_PRCCNT
000002FC'EF   DD    063B   552          PUSHL   JIB+JIB$L_MPID          ; MASTER PID
                        0641   553          PRINT   2,<-
                        0641   554 Master internal PID      !XL    -
                        0641   555 Subprocess count          !8UW>
                        064E   556
          1C A2   DD    064E   557          PUSHL   PCB$L_OWNER(R2)         ; CREATOR PID
          60 A2   DD    0651   558          PUSHL   PCB$L_PID(R2)
                        0654   559          PRINT   2,<-
                        0654   560 Internal PID               !XL    -
                        0654   561 Creator internal PID      !XL>
                        0661   562
          68 A2   DD    0661   563          PUSHL   PCB$L_EOWNER(R2)        ; CREATOR EXTENDED PID
```

```
                64 A2    DD  0664   564          PUSHL   PCB$L_EPID(R2)          ; EXTENDED PID
                         0667   565          PRINT   2,<-
                         0667   566 Extended PID              !XL    -
                         0667   567 Creator extended PID      !XL>
                         0674   568
                32 A2    DD  0674   569          PUSHL   PCB$W_TMBU(R2)
             02 2C A2    B1  0677   570          CMPW    PCB$W_STATE(R2),#SCH$C_MWAIT    ; CHECK FOR MWAIT
                   25    12  067B   571          BNEQ    15$                             ; NOT EQUAL, TRANSLATE TO STATE NAME
                   52    DD  067D   572 10$:      PUSHL   R2                              ; SAVE R2, DETERMINE RESOURCE WAIT STATE
             52    4C A2   D0  067F   573          MOVL    PCB$L_EFWM(R2),R2               ; RESOURCE WAIT
          53    FAD8 CF    9E  0683   574          MOVAB   RESOURCE_WAIT,R3                ; ADDRESS OF TABLE OF STATES
           00000000'GF      16  0688   575          JSB     G^TRANSLATE_ADDRESS             ; DETERMINE STATE
              52    8E    D0  068E   576          MOVL    (SP)+,R2                        ; RESTORE R2
                   0B    12  0691   577          BNEQ    12$                             ; MATCH FOUND, IF NOT EQUAL
                4C A2    D5  0693   578          TSTL    PCB$L_EFWM(R2)                  ; CHECK FOR MUTEX ADDRESS
                   0A    18  0696   579          BGEQ    15$                             ; BRANCH IF NOT ADDRESS
              FB8F CF    9F  0698   580          PUSHAB  MWAIT                           ; ADDRESS OF MUTEX COUNTED STRING
                   0D    11  069C   581          BRB     20$                             ; DISPLAY
                   50    DD  069E   582 12$:      PUSHL   R0                              ; ADDRESS OF RESOURCE WAITING FOR
                   09    11  06A0   583          BRB     20$                             ; DISPLAY
                         06A2   584 15$:
          51    2C A2    3C  06A2   585          MOVZWL  PCB$W_STATE(R2),R1
           F974 CF41     7F  06A6   586          PUSHAQ  STATE_TABLE[R1]
                         06AB   587 20$:
                         06AB   588          PRINT   2,<-
                         06AB   589 State                        !5AC    -
                         06AB   590 Termination mailbox          !XW>
                         06B8   591
          51    0D A2    9A  06B8   592          MOVZBL  PCB$B_ASTEN(R2),R1
           FC1A CF41     7F  06BC   593          PUSHAQ  AST_TABLE[R1]
          7E    0B A2    9A  06C1   594          MOVZBL  PCB$B_PRI(R2),-(SP)
       6E    1F    6E    C3  06C5   595          SUBL3   (SP),#31,(SP)
                         06C9   596          PRINT   2,<-
                         06C9   597 Current priority             !8UB    -
                         06C9   598 AST's enabled                    !4AC>
                         06D6   599
          51    0C A2    9A  06D6   600          MOVZBL  PCB$B_ASTACT(R2),R1
           FBFC CF41     7F  06DA   601          PUSHAQ  AST_TABLE[R1]
          7E    2F A2    9A  06DF   602          MOVZBL  PCB$B_PRIB(R2),-(SP)
       6E    1F    6E    C3  06E3   603          SUBL3   (SP),#31,(SP)
                         06E7   604          PRINT   2,<-
                         06E7   605 Base priority                !8UB    -
                         06E7   606 AST's active                     !4AC>
                         06F4   607
                38 A2    DD  06F4   608          PUSHL   PCB$W_ASTCNT(R2)
             00BC C2     DD  06F7   609          PUSHL   PCB$W_MEM(R2)
             00BE C2     DD  06FB   610          PUSHL   PCB$W_GRP(R2)
                         06FF   611          PRINT   3,<-
                         06FF   612 UIC                      [!30W,!30W]     -
                         06FF   613 AST's remaining           !8UW>
                         070C   614
                3C A2    DD  070C   615          PUSHL   PCB$W_BIOLM(R2)         ; BUFFERED I/O LIMIT
                3A A2    DD  070F   616          PUSHL   PCB$W_BIOCNT(R2)        ; BUFFERED I/O COUNT
                0E A2    DD  0712   617          PUSHL   PCB$W_MTXCNT(R2)        ; MUTEX COUNT
                         0715   618          PRINT   3,<-
                         0715   619 Mutex count                  !8UW    -
                         0715   620 Buffered I/O count/limit   !6UW/!UW>
```

```
                     0722   621
          40 A2   DD 0722   622          PUSHL   PCB$W_DIOLM(R2)
          3E A2   DD 0725   623          PUSHL   PCB$W_DIOCNT(R2)
          2E A2   DD 0728   624          PUSHL   PCB$B_WEFC(R2)
                     072B   625          PRINT   3,<-
                     072B   626 Waiting EF cluster       !8UB    -
                     072B   627 Direct I/O count/limit     !6UW/.UW>
                     0738   628
000002CC'EF       DD 0738   629          PUSHL   JIB+JIB$L_BYTLM
000002C8'EF       DD 073E   630          PUSHL   JIB+JIB$L_BYTCNT
          28 A2   DD 0744   631          PUSHL   PCB$L_WTIME(R2)
                     0747   632          PRINT   3,<-
                     0747   633 Starting wait time       !XL     -
                     0747   634 BUFIO byte count/limit     !6UL/!UL>
                     0754   635
000002D8'EF       DD 0754   636          PUSHL   JIB+JIB$W_FILCNT
          4C A2   DD 075A   637          PUSHL   PCB$L_EFWM(R2)
                     075D   638          PRINT   2,<-
                     075D   639 Event flag wait mask     !XL     -
                     075D   640 # open files allowed left  !6UW>
                     076A   641
000002DC'EF       DD 076A   642          PUSHL   JIB+JIB$W_TQCNT
          50 A2   DD 0770   643          PUSHL   PCB$L_EFCS(R2)
                     0773   644          PRINT   2,<-
                     0773   645 Local EF cluster 0       !XL     -
                     0773   646 Timer entries allowed left !6UW>
                     0780   647
          30 A2   DD 0780   648          PUSHL   PCB$W_APTCNT(R2)
          54 A2   DD 0783   649          PUSHL   PCB$L_EFCU(R2)
                     0786   650          PRINT   2,<-
                     0786   651 Local EF cluster 1       !XL     -
                     0786   652 Active page table count    !6UW>
                     0793   653
          36 A2   DD 0793   654          PUSHL   PCB$W_PPGCNT(R2)
          58 A2   DD 0796   655          PUSHL   PCB$L_EFC2P(R2)
                     0799   656          PRINT   2,<-
                     0799   657 Global cluster 2 pointer !XL     -
                     0799   658 Process WS page count      !6UW>
                     07A6   659
          34 A2   DD 07A6   660          PUSHL   PCB$W_GPGCNT(R2)
          5C A2   DD 07A9   661          PUSHL   PCB$L_EFC3P(R2)
                     07AC   662          PRINT   2,<-
                     07AC   663 Global cluster 3 pointer !XL     -
                     07AC   664 Global WS page count       !6UW>
                     07B9   665
                04   07B9   666          RET
```

```
                        07BA    668          .SBTTL  SHOW_PHD -- SHOW PROCESS HEADER
                        07BA    669 ;---
                        07BA    670 ;
                        07BA    671 ;       SHOW_PHD
                        07BA    672 ;
                        07BA    673 ;       FORMAT THE FIELDS FROM THE PROCESS HEADER.
                        07BA    674 ;
                        07BA    675 ;   INPUTS:
                        07BA    676 ;
                        07BA    677 ;       PHD = PROCESS HEADER BLOCK
                        07BA    678 ;
                        07BA    679 ;   OUTPUTS:
                        07BA    680 ;
                        07BA    681 ;       NONE
                        07BA    682 ;
                        07BA    683 ;---
                        07BA    684
                        07BA    685          .ENABL  LSB
                        07BA    686
                        07BA    687 SHOW_PHD:
                  0004  07BA    688          .WORD   ^M<R2>
                        07BC    689
     52 00000128'EF   DE 07BC    690          MOVAL   PHD,R2                  ; ADDRESS OF PROCESS HEADER
                        07C3    691          SKIP    2
                        07CC    692          ENSURE  12
                        07E4    693          PRINT   0,<Process header>
                        07F1    694          PRINT   0,<--------------->
                        07FE    695          SKIP    1
           38 A2        DD 0807 696          PUSHL   PHD$L_CPUTIM(R2)
           28 A2        DD 080A 697          PUSHL   PHD$L_FREPOVA(R2)
                           080D 698          PRINT   2,<-
                           080D 699 First free P0 address       !8XL!4* -
                           080D 700 Accumulated CPU time        !8XL>
           3C A2        DD 081A 701          PUSHL   PHD$W_QUANT(R2)
           2C A2        DD 081D 702          PUSHL   PHD$L_FREPTECNT(R2)
                           0820 703          PRINT   2,<-
                           0820 704 Free PTEs between P0/P1     !8UL!4* -
                           0820 705 CPU since last quantum      !8XW>
000002EE'EF             DD 082D 706          PUSHL   JIB+JIB$W_PRCLIM
           30 A2        DD 0833 707          PUSHL   PHD$L_FREP1VA(R2)
                           0836 708          PRINT   2,<-
                           0836 709 First free P1 address       !8XL!4* -
                           0836 710 Subprocess quota            !8UW>
           40 A2        DD 0843 711          PUSHL   PHD$W_ASTLM(R2)
000002E4'EF             DD 0846 712          PUSHL   JIB+JIB$L_PGFLCNT
                           084C 713          PRINT   2,<-
                           084C 714 Free page file pages        !8UL!4* -
                           084C 715 AST limit                   !8UW>
           42 A2        DD 0859 716          PUSHL   PHD$W_PHVINDEX(R2)
           34 A2        DD 085C 717          PUSHL   PHD$B_DFPFC(R2)
                           085F 718          PRINT   2,<-
                           085F 719 Page fault cluster size     !8UB!4* -
                           085F 720 Process header index        !8XW>
           44 A2        DD 086C 721          PUSHL   PHD$L_BAK(R2)
           35 A2        DD 086F 722          PUSHL   PHD$B_PGTBPFC(R2)
                           0872 723          PRINT   2,<-
                           0872 724 Page table cluster size     !8UB!4* -
```

```
                    0872   725 Backup address vector      !8XL>
      48 A2    DD   087F   726          PUSHL   PHD$L_WSLX(R2)
      36 A2    DD   0882   727          PUSHL   PHD$W_FLAGS(R2)
                    0885   728          PRINT   2,<-
                    0885   729 Flags                      !8XW!4* -
                    0885   730 WSL index save area        !8XL>
      6C A2    DD   0892   731          PUSHL   PHD$W_PTCNTLCK(R2)
      54 A2    DD   0895   732          PUSHL   PHD$L_DIOCNT(R2)
                    0898   733          PRINT   2,<-
                    0898   734 Direct I/O count           !8UL!4* -
                    0898   735 PTs having locked WSLs      .8UW>
      6E A2    DD   08A5   736          PUSHL   PHD$W_PTCNTVAL(R2)
      58 A2    DD   08A8   737          PUSHL   PHD$L_BIOCNT(R2)
                    08AB   738          PRINT   2,<-
                    08AB   739 Buffered I/O count          !8UL!4* -
                    08AB   740 PTs having valid WSLs       !8UW>
      70 A2    DD   08B8   741          PUSHL   PHD$W_PTCNTACT(R2)
      5C A2    DD   08BB   742          PUSHL   PHD$L_CPULIM(R2)
                    08BE   743          PRINT   2,<-
                    08BE   744 Limit on CPU time           !8XL!4* -
                    08BE   745 Active page tables          !8UW>
      72 A2    DD   08CB   746          PUSHL   PHD$W_PTCNTMAX(R2)
000002E0'EF    DD   08CE   747          PUSHL   JIB+JIB$L_PGFLQUOTA
                    08D4   748          PRINT   2,<-
                    08D4   749 Maximum page file count     !8UL!4* -
                    08D4   750 Maximum active PTs          !8UW>
      74 A2    DD   08E1   751          PUSHL   PHD$W_WSFLUID(R2)
      4C A2    DD   08E4   752          PUSHL   PHD$L_PAGEFLTS(R2)
                    08E7   753          PRINT   2,<-
                    08E7   754 Total page faults           !8UL!4* -
                    08E7   755 Guaranteed fluid WS pages   !8UW>
      76 A2    DD   08F4   756          PUSHL   PHD$W_EXTDYNWS(R2)
000002DA'EF    DD   08F7   757          PUSHL   JIB+JIB$W_FILLM
                    08FD   758          PRINT   2,<-
                    08FD   759 File limit                  !8UW!4* -
                    08FD   760 Extra dynamic WS entries    !8UW>
      64 A2    DD   090A   761          PUSHL   PHD$L_PTWSLELCK(R2)
000002DE'EF    DD   090D   762          PUSHL   JIB+JIB$W_TQLM
                    0913   763          PRINT   2,<-
                    0913   764 Timer queue limit           !8UW!4* -
                    0913   765 Locked WSLE counts array    !8XW>
      68 A2    DD   0920   766          PUSHL   PHD$L_PTWSLEVAL(R2)
      1C A2    DD   0923   767          PUSHL   PHD$L_PAGFIL(R2)
                    0926   768          PRINT   2,<-
                    0926   769 Paging file index           !8XL!4* -
                    0926   770 Valid WSLE counts array     !8XW>
                    0933   771
                04  0933   772          RET
```

```
                     0934   774          .SBTTL  SHOW_REGS -- SHOW SAVED PROCESS REGISTERS
                     0934   775 ;---
                     0934   776 ;
                     0934   777 ;       SHOW_REGS
                     0934   778 ;
                     0934   779 ;       DISPLAY THE SAVED PROCESS REGISTERS
                     0934   780 ;
                     0934   781 ;   INPUTS:
                     0934   782 ;
                     0934   783 ;       PHD = CURRENT PROCESS HEADER
                     0934   784 ;
                     0934   785 ;   OUTPUTS:
                     0934   786 ;
                     0934   787 ;       NONE
                     0934   788 ;
                     0934   789 ;---
                     0934   790
                     0934   791 SHOW_REGS:
                0004 0934   792          .WORD   ^M<R2>
                     0936   793
                     0936   794          SKIP    2
                     093F   795          ENSURE  7
                     0957   796          PRINT   0,<Saved process registers>
                     0964   797          PRINT   0,<----------------------->
                     0971   798          SKIP    1
    52 00000128'EF   DE 097A   799          MOVAL   PHD,R2                  ; ADDRESS OF PROCESS HEADER
          0094 C2    DD 0981   800          PUSHL   PHD$L_R3(R2)
          0090 C2    DD 0985   801          PUSHL   PHD$L_R2(R2)
          008C C2    DD 0989   802          PUSHL   PHD$L_R1(R2)
          0088 C2    DD 098D   803          PUSHL   PHD$L_R0(R2)
                     0991   804          PRINT   4,<R0   = !XL    R1   - !XL    R2   = !XL    R3   = !XL>
          00A4 C2    DD 099E   805          PUSHL   PHD$L_R7(R2)
          00A0 C2    DD 09A2   806          PUSHL   PHD$L_R6(R2)
          009C C2    DD 09A6   807          PUSHL   PHD$L_R5(R2)
          0098 C2    DD 09AA   808          PUSHL   PHD$L_R4(R2)
                     09AE   809          PRINT   4,<R4   = !XL    R5   = !XL    R6   = !XL    R7   = !XL>
          00B4 C2    DD 09BB   810          PUSHL   PHD$L_R11(R2)
          00B0 C2    DD 09BF   811          PUSHL   PHD$L_R10(R2)
          00AC C2    DD 09C3   812          PUSHL   PHD$L_R9(R2)
          00A8 C2    DD 09C7   813          PUSHL   PHD$L_R8(R2)
                     09CB   814          PRINT   4,<R8   = !XL    R9   = !XL    R10  = !XL    R11  = !XL>
          00C4 C2    DD 09D8   815          PUSHL   PHD$L_PSL(R2)
          00C0 C2    DD 09DC   816          PUSHL   PHD$L_PC(R2)
          00BC C2    DD 09E0   817          PUSHL   PHD$L_R13(R2)
          00B8 C2    DD 09E4   818          PUSHL   PHD$L_R12(R2)
                     09E8   819          PRINT   4,<AP   = !XL    FP   = !XL    PC   = !XL    PSL  = !XL>
          0084 C2    DD 09F5   820          PUSHL   PHD$L_USP(R2)
          0080 C2    DD 09F9   821          PUSHL   PHD$L_SSP(R2)
            7C A2    DD 09FD   822          PUSHL   PHD$L_ESP(R2)
            78 A2    DD 0A00   823          PUSHL   PHD$L_KSP(R2)
                     0A03   824          PRINT   4,<KSP  = !XL    ESP  = !XL    SSP  = !XL    USP  = !XL>
          00D4 C2    DD 0A10   825          PUSHL   PHD$L_P1LR(R2)
          00D0 C2    DD 0A14   826          PUSHL   PHD$L_P1BR(R2)
 7E 00CC C2 18    00 EF 0A18   827          EXTZV   #PHD$V_POLR,#PHD$S_POLR,PHD$L_POLRASTL(R2),-(SP)
          00C8 C2    DD 0A1F   828          PUSHL   PHD$L_P0BR(R2)
                     0A23   829          PRINT   4,<P0BR = !XL    P0LR = !XL    P1BR = !XL    P1LR = !XL>
                     0A30   830
```

```
04  0A30   831          RET
```

```
                        0A31    833          .SBTTL  SHOW_WSL -- SHOW WORKING SET LIST
                        0A31    834 ;---
                        0A31    835 ;
                        0A31    836 ;       SHOW_WSL
                        0A31    837 ;
                        0A31    838 ;       THIS ROUTINE FORMATS INFORMATION ABOUT THE PROCESSES
                        0A31    839 ;       WORKING SET LIST FROM THE PROCESS HEADER.
                        0A31    840 ;
                        0A31    841 ;   INPUTS:
                        0A31    842 ;
                        0A31    843 ;       PHD = PROCESS HEADER BLOCK
                        0A31    844 ;
                        0A31    845 ;   OUTPUTS:
                        0A31    846 ;
                        0A31    847 ;       NONE
                        0A31    848 ;
                        0A31    849 ;---
                        0A31    850
                        0A31    851          .ENABL  LSB
                        0A31    852
                        0A31    853 SHOW_WSL:
                007C    0A31    854          .WORD   ^M<R2,R3,R4,R5,R6>
                        0A33    855
                        0A33    856          SKIP    2
                        0A3C    857          ENSURE  8
                        0A54    858          PRINT   0,<Working set information>
                        0A61    859          PRINT   0,<----------------------->
                        0A6E    860          SKIP    1
        52  00000128'EF  DE     0A77    861          MOVAL   PHD,R2
             53    08 A2  3C     0A7E    862          MOVZWL  PHD$W_WSLIST(R2),R3     ; BIAS TO WORKING SET LIST
                    53  D7     0A82    863          DECL    R3                      ; ADJUST BIAS
         7E    0A A2    53  C3     0A84    864          SUBL3   R3,PHD$W_WSAUTH(R2),-(SP) ; CURRENT AUTHORIZED SIZE
                  08 A2  DD     0A89    865          PUSHL   PHD$W_WSLIST(R2)        ; 1ST WSL ENTRY
                         0A8C    866          PRINT   1,<First WSL entry          !XW         Current authorized working se
         7E    1A A2    53  C3     0A99    867          SUBL3   R3,PHD$W_DFWSCNT(R2),-(SP) ; DEFAULT WORKING SET SIZE
                  0C A2  DD     0A9E    868          PUSHL   PHD$W_WSLOCK(R2)        ; 1ST LOCKED WSL ENTRY
                         0AA1    869          PRINT   1,<First locked entry       !XW         Default (initial) working set
         7E    18 A2    53  C3     0AAE    870          SUBL3   R3,PHD$W_WSQUOTA(R2),-(SP) ; MAXIMUM WORKING SET ALLOWED
                  0E A2  DD     0AB3    871          PUSHL   PHD$W_WSDYN(R2)         ; 1ST DYNAMIC ENTRY
                         0AB6    872          PRINT   1,<First dynamic entry      !XW         Maximum working set allowed (
                  10 A2  DD     0AC3    873          PUSHL   PHD$W_WSNEXT(R2)        ; LAST ENTRY REPLACED
                         0AC6    874          PRINT   1,<Last entry replaced      !XW>
                  12 A2  DD     0AD3    875          PUSHL   PHD$W_WSLAST(R2)        ; LAST ENTRY IN LIST
                         0AD6    876          PRINT   1,<Last entry in list       !XW>
                  08 A2  B5     0AE3    877          TSTW    PHD$W_WSLIST(R2)        ; CHECK IF REASONABLE
                     01  14     0AE6    878          BGTR    5$                      ; BRANCH IF OK
                         04     0AE8    879          RET
                         0AE9    880 5$:
                         0AE9    881          SKIP    2
                         0AF2    882          ENSURE  8
                         0B0A    883          PRINT   0,<Working set list>
                         0B17    884          PRINT   0,<----------------->
             56    12 A2  3C     0B24    885          MOVZWL  PHD$W_WSLAST(R2),R6     ; LONGWORD OFFSET TO LAST ENTRY
                    56  D6     0B28    886          INCL    R6                      ; LONGWORDS IN PHD+WSL
            56    56  02  78     0B2A    887          ASHL    #2,R6,R6                ; R6 = LENGTH IN BYTES
                    56  DD     0B2E    888          PUSHL   R6                      ; LENGTH TO ALLOCATE
00000000'EF     01  FB     0B30    889          CALLS   #1,ALLOCATE             ; ALLOCATE SPACE FOR PHD+WSL
```

```
                    52   51    D0  0B37   890          MOVL    R1,R2                           ; ADDRESS OF PHD/WSL BUFFER
                                 0B3A   891          GETMEM  @PCB+PCB$L_PHD,(R2),R6          ; GET ENTIRE WORKING SET LIST
                      25 50      E9  0B4B   892          BLBC    R0,90$                          ; SKIP IF CANNOT BE READ
                                 0B4E   893
          00000C13'EF    00      FB  0B4E   894          CALLS   #0,WSL_TITLE                    ; PRINT THE HEADING LINES
00000000'EF   00000C13'EF      9E  0B55   895          MOVAB   WSL_TITLE,HEADING_ROUTINE       ; SET HEADING ROUTINE
                 53    08 A2   3C  0B60   896          MOVZWL  PHD$W_WSLIST(R2),R3             ; STARTING INDEX OF LIST
                 54    12 A2   3C  0B64   897          MOVZWL  PHD$W_WSLAST(R2),R4             ; LAST ENTRY TO DUMP
                         53    D7  0B68   898          DECL    R3                              ; ADJUST FOR FIRST ITERATION
                         55    D4  0B6A   899          CLRL    R5                              ; EMPTY WSLE COUNT
                                 0B6C   900 10$:
                         53    D6  0B6C   901          INCL    R3                              ; INCREMENT INDEX
                    54   53    D1  0B6E   902          CMPL    R3,R4                           ; CHECK IF DONE
                         12    15  0B71   903          BLEQ    20$                             ; BRANCH IF OK
                                 0B73   904 90$:
              00000000'EF      D4  0B73   905          CLRL    HEADING_ROUTINE                 ; CLEAR HEADING ROUTINE ADDRESS
                         56    DD  0B79   906          PUSHL   R6                              ; LENGTH OF BUFFER
                         52    DD  0B7B   907          PUSHL   R2                              ; ADDRESS OF PHD+WSL BUFFER
          00000000'EF    02    FB  0B7D   908          CALLS   #2,DEALLOCATE                   ; DEALLOCATE BUFFER
                               04  0B84   909          RET
                                 0B85   910 20$:
                50   6243      D0  0B85   911          MOVL    (R2)[R3],R0                     ; GET ENTRY
                04 50    00      E0  0B89   912          BBS     #WSL$V_VALID,R0,30$             ; BRANCH IF VALID
                         55    D6  0B8D   913          INCL    R5                              ; INCREMENT EMPTY ENTRIES
                         DB    11  0B8F   914          BRB     10$
                                 0B91   915 30$:
                         55    D5  0B91   916          TSTL    R5                              ; CHECK IF EMPTY ENTRIES
                         28    13  0B93   917          BEQL    40$                             ; BRANCH IF NOT
                         50    DD  0B95   918          PUSHL   R0                              ; SAVE PENDING WSLE
                         55    DD  0B97   919          PUSHL   R5
                                 0B99   920          SKIP    1
                                 0BA2   921          PRINT   1,<!_ ----  !UL empty entries>
                                 0BAF   922          SK_P    1
                            55   D4  0BB8   923          CLRL    R5                              ; RESET EMPTY WSLE COUNT
                            50 8ED0  0BBA   924          POPL    R0                              ; RESTORE PENDING WSLE TO SHOW
                                 0BBD   925 40$:
      51    50    01    06    EF  0BBD   926          EXTZV   #WSL$V_GOODPAGE,#1,R0,R1
                F6EA CF41    9F  0BC2   927          PUSHAB  GOODPAGE_TABLE[R1]
      51    50    01    04    EF  0BC7   928          EXTZV   #WSL$V_PFNLOCK,#1,R0,R1
                F6D6 CF41    9F  0BCC   929          PUSHAB  PFNLOCK_TABLE[R1]
      51    50    01    05    EF  0BD1   930          EXTZV   #WSL$V_WSLOCK,#1,R0,R1
                F6C3 CF41    9F  0BD6   931          PUSHAB  WSLOCK_TABLE[R1]
      51    50    01    08    EF  0BDB   932          EXTZV   #WSL$V_MODIFY,#1,R0,R1
                F6AE CF41    9F  0BE0   933          PUSHAB  MODIFY_TABLE[R1]
      51    50    03    01    EF  0BE5   934          EXTZV   #WSL$V_PAGTYP,#WSL$S_PAGTYP,R0,R1
                F664 CF41    7F  0BEA   935          PUSHAQ  PAGTYP_TABLE[R1]
      51    50    01    00    EF  0BEF   936          EXTZV   #WSL$V_VALID,#1,R0,R1
                F652 CF41    9F  0BF4   937          PUSHAB  VALID_TABLE[R1]
7E    50    000001FF    8F    CB  0BF9   938          BICL3   #^X1FF,R0,-(SP)                  ; VIRTUAL ADDRESS
                         53    DD  0C01   939          PUSHL   R3                              ; INDEX
                                 0C03   940          PRINT   8,<!_ !XW  !XL    !AC !5(AC)>
                      FF59   31  0C10   941          BRW     10$
                                 0C13   942
                                 0C13   943          .DSABL  LSB
                                 0C13   944
                                 0C13   945 ;
                                 0C13   946 ;        THIS ROUTINE PRINTS THE HEADING LINES FOR EACH PAGE
```

```
                0C13   947 ;       OF THE WORKING SET LIST DISPLAY.
                0C13   948 ;
                0C13   949 WSL_TITLE:
0000            0C13   950          .WORD   0
                0C15   951
                0C15   952          SKIP    1
                0C1E   953          PRINT   0,<!_INDEX   ADDRESS    STATUS>
                0C2B   954          SKIP    1
          04    0C34   955          RET
```

```
                        0C35   957          .SBTTL  SHOW_PST -- SHOW PROCESS SECTION TABLE
                        0C35   958 ;---
                        0C35   959 ;
                        0C35   960 ;       SHOW_PST
                        0C35   961 ;
                        0C35   962 ;       DISPLAY THE PROCESS SECTION TABLE
                        0C35   963 ;
                        0C35   964 ;   INPUTS:
                        0C35   965 ;
                        0C35   966 ;       PHD = PROCESS HEADER BLOCK
                        0C35   967 ;
                        0C35   968 ;   OUTPUTS:
                        0C35   969 ;
                        0C35   970 ;       NONE
                        0C35   971 ;
                        0C35   972 ;---
                        0C35   973
                        0C35   974          .ENABL  LSB
                        0C35   975
                        0C35   976 SHOW_PST:
                0OFC    0C35   977          .WORD   ^M<R2,R3,R4,R5,R6,R7>
                        0C37   978
                        0C37   979          SKIP    2
                        0C40   980          ENSURE  5
                        0C58   981          PRINT   0,<Process section table information>
                        0C65   982          PRINT   0,<------------------------------->
                        0C72   983          SKIP    1
        52  00000128'EF  DE    0C7B   984          MOVAL   PHD,R2
                  24 A2  DD    0C82   985          PUSHL   PHD$W_PSTLAST(R2)       ; LAST PST ENTRY ALLOCATED
                         0C85   986          PRINT   1,<! Last entry allocated     !XW>
                  26 A2  DD    0C92   987          PUSHL   PHD$W_PSTFREE(R2)
                         0C95   988          PRINT   1,<! First free PST entry     !XW>
                  24 A2  B5    0CA2   989          TSTW    PHD$W_PSTLAST(R2)       ; CHECK IF REASONABLE
                     01  19    0CA5   990          BLSS    5$                      ; BRANCH IF OK
                         04    0CA7   991          RET
                         0CA8   992 5$:
                         0CA8   993          SKIP    2
                         0CB1   994          ENSURE  8
                         0CC9   995          PRINT   0,<Process section table>
                         0CD6   996          PRINT   0,<--------------------->
        00000DB5'EF      00    FB    0CE3   997          CALLS   #0,PST_TITLE            ; PRINT TITLE LINE
00000000'EF  00000DB5'EF  9E    0CEA   998          MOVAB   PST_TITLE,HEADING_ROUTINE       ; SET HEADING ROUTINE
           50    24 A2  32    0CF5   999          CVTWL   PHD$W_PSTLAST(R2),R0    ; LAST ENTRY ALLOCATED
 57  20 A2  00000070'EF  C1    0CF9  1000          ADDL3   PCB+PCB$L_PHD,PHD$L_PSTBASOFF(R2),R7  ; BASE ADDRESS
             54    6740  DE    0D02  1001          MOVAL   (R7)[R0],R4             ; ADDRESS OF LOWEST ENTRY
                57    54  C2    0D06  1002          SUBL2   R4,R7                   ; LENGTH OF PST IN USE
                      57  DD    0D09  1003          PUSHL   R7                      ; LENGTH TO ALLOCATE
        00000000'EF   01  FB    0D0B  1004          CALLS   #1,ALLOCATE             ; ALLOCATE PST BUFFER
                56    51  D0    0D12  1005          MOVL    R1,R6                   ; SAVE ADDRESS OF PST BUFFER
                          0D15  1006          GETMEM  (R4),(R6),R7            ; GET USED PART OF PST
                          0D22  1007          RETIFERR                        ; SKIP IF ERROR READING
                          0D26  1008
                53    57  D0    0D26  1009          MOVL    R7,R3                   ; SET LENGTH OF PST TO SCAN
              54    6647  9E    0D29  1010          MOVAB   (R6)[R7],R4             ; FIRST ENTRY + 1
                      55  D4    0D2D  1011          CLRL    R5                      ; INITIALIZE PST INDEX
                          0D2F  1012 10$:
                54    20  C2    0D2F  1013          SUBL2   #SEC$C_LENGTH,R4        ; DECREMENT ADDRESS
```

```
                55   08   C2  0D32  1014          SUBL2   #SEC$C_LENGTH/4,R5       ; DECREMENT INDEX
                53   20   C2  0D35  1015          SUBL2   #SEC$C_LENGTH,R3         ; DECREMENT BUFFER LENGTH
                     19   18  0D38  1016          BGEQ    20$                      ; BRANCH IF OK
          00000000'EF     D4  0D3A  1017          CLRL    HEADING_ROUTINE          ; CLEAR HEADING ROUTINE ADDRESS
                     57   DD  0D40  1018          PUSHL   R7                       ; LENGTH TO DEALLOCATE
                     56   DD  0D42  1019          PUSHL   R6                       ; ADDRESS OF PST BUFFER
     00000000'EF     02   FB  0D44  1020          CALLS   #2,DEALLOCATE            ; DEALLOCATE PST BUFFER
                              0D4B  1021          STATUS  SUCCESS
                     04       0D52  1022          RET
                              0D53  1023  20$:
     51  14 A4  01   00       EF  0D53  1024          EXTZV   #SEC$V_GBL,#1,SEC$W_FLAGS(R4),R1
                F4D3 CF41       9F  0D59  1025          PUSHAB  GBL_TABLE[R1]
     51  14 A4  01   01       EF  0D5E  1026          EXTZV   #SEC$V_CRF,#1,SEC$W_FLAGS(R4),R1
                F4CE CF41       9F  0D64  1027          PUSHAB  CRF_TABLE[R1]
     51  14 A4  01   02       EF  0D69  1028          EXTZV   #SEC$V_DZRO,#1,SEC$W_FLAGS(R4),R1
                F4C9 CF41       9F  0D6F  1029          PUSHAB  DZRO_TABLE[R1]
     51  14 A4  01   03       EF  0D74  1030          EXTZV   #SEC$V_WRT,#1,SEC$W_FLAGS(R4),R1
                F4C6 CF41       9F  0D7A  1031          PUSHAB  WRT_TABLE[R1]
                   06 A4        DD  0D7F  1032          PUSHL   SEC$W_SECXBL(R4)         ; BACKWARD LINK
                   04 A4        DD  0D82  1033          PUSHL   SEC$W_SECXFL(R4)         ; FORWARD LINK
                   18 A4        DD  0D85  1034          PUSHL   SEC$L_REFCNT(R4)         ; REFERENCE COUNT
                      64        DD  0D88  1035          PUSHL   SEC$L_CCB(R4)            ; CCB ADDRESS
     7E  08 A4  08    18       EF  0D8A  1036          EXTZV   #SEC$V_PFC,#SEC$S_PFC,SEC$L_VPXPFC(R4),-(SP)
                   10 A4        DD  0D90  1037          PUSHL   SEC$L_VBN(R4)            ; STARTING BLOCK NUMBER
                   0C A4        DD  0D93  1038          PUSHL   SEC$L_WINDOW(R4)         ; WINDOW CONTROL BLOCK
                   1C A4        DD  0D96  1039          PUSHL   SEC$L_PAGCNT(R4)         ; PAGE COUNT
     51  08 A4  16   00       EF  0D99  1040          EXTZV   #SEC$V_VPX,#SEC$S_VPX,SEC$L_VPXPFC(R4),R1
           7E  51   09       78  0D9F  1041          ASHL    #9,R1,-(SP)              ; VIRTUAL ADDRESS OF SECTION
                      55        DD  0DA3  1042          PUSHL   R5                       ; SECTION TABLE INDEX
                              0DA5  1043          PRINT   14,<!_ !XW     !XL !XL   !XL  !XL   !4UL  !XL  !4UL     !XW  !XW   !
                FF7A  31       0DB2  1044          BRW     10$
                              0DB5  1045
                              0DB5  1046          .DSABL  LSB
                              0DB5  1047
                              0DB5  1048 ;
                              0DB5  1049 ;        SUBROUTINE TO PRINT THE SUB-HEADING LINE
                              0DB5  1050 ;
                              0DB5  1051 PST_TITLE:
                0000          0DB5  1052          .WORD   0
                              0DB7  1053
                              0DB7  1054          SKIP    1
                              0DC0  1055          PRINT   0,<!_INDEX      ADDRESS   PAGES     WINDOW     VBN    CLUSTER  CHANN
                              0DCD  1056          SKIP    1
                     04       0DD6  1057          RET
```

```
                        0DD7  1059          .SBTTL  SHOW_PPT -- SHOW PROCESS PAGE TABLES
                        0DD7  1060 ;---
                        0DD7  1061 ;
                        0DD7  1062 ;       SHOW_PPT
                        0DD7  1063 ;
                        0DD7  1064 ;       THIS ROUTINE DISPLAYS THE CONTENTS OF THE P0 AND
                        0DD7  1065 ;       P1 PAGE TABLES FOR THE CURRENT PROCESS.
                        0DD7  1066 ;
                        0DD7  1067 ;   INPUTS:
                        0DD7  1068 ;
                        0DD7  1069 ;       PCB = PROCESS CONTROL BLOCK
                        0DD7  1070 ;       PHD = PROCESS HEADER BLOCK
                        0DD7  1071 ;
                        0DD7  1072 ;   OUTPUTS:
                        0DD7  1073 ;
                        0DD7  1074 ;       NONE
                        0DD7  1075 ;
                        0DD7  1076 ;---
                        0DD7  1077
                        0DD7  1078 SHOW_PPT:
                003C    0DD7  1079          .WORD   ^M<R2,R3,R4,R5>
                        0DD9  1080
                        0DD9  1081          .ENABL  LSB
        00000000'EF   00   FB   0DD9  1082          CALLS   #0,INIT_PFN             ; SETUP ACCESS TO PFN DATA BASE
        52   00000004'EF   DE   0DE0  1083          MOVAL   PCB,R2
        53   00000128'EF   DE   0DE7  1084          MOVAL   PHD,R3
00000000'EF   00006000 8F   D3   0DEE  1085          BITL    #OPT$M_PPT_LEN!OPT$M_PPT_RNG,OPTIONS ; SPECIFIC ADDRESSES?
                    03   13   0DF9  1086          BEQL    5$                      ; NO
                 00FB   31   0DFB  1087          BRW     100$
00000000'EF   00001800 8F   D3   0DFE  1088 5$:       BITL    #OPT$M_P0_PPT!OPT$M_P1_PPT,OPTIONS ; SPECIFIC PAGE TABLES?
                    08   13   0E09  1089          BEQL    10$                     ; NO
     70 00000000'EF   0B   E1   0E0B  1090          BBC     #OPT$V_P0_PPT,OPTIONS,20$ ; P0 NOT REQUESTED
                        0E13  1091 10$:       SKIP    2
                        0E1C  1092          ENSURE  8
                        0E34  1093          PRINT   0,<P0 page table>
                        0E41  1094          PRINT   0,<-------------->
                        0E4E  1095          SKIP    1
                  6C A2   DD   0E57  1096          PUSHL   PCB$L_PHD(R2)           ; ADDRESS OF PROCESS HEADER
                00C8 C3   DD   0E5A  1097          PUSHL   PHD$L_P0BR(R3)          ; ADDRESS OF P0 PAGE TABLE
                    00   DD   0E5E  1098          PUSHL   #0                      ; STARTING ADDRESS BEING MAPPED
  7E   00CC C3   18   00   EF   0E60  1099          EXTZV   #PHD$V_P0LR,#PHD$S_P0LR,PHD$L_P0LRASTL(R3),-(SP) ; NUM ENTRIES
        00000000'EF   04   FB   0E67  1100          CALLS   #4,DUMP_PTE             ; FORMAT PAGE TABLE
                        0E6E  1101
00000000'EF   00001800 8F   D3   0E6E  1102          BITL    #OPT$M_P0_PPT!OPT$M_P1_PPT,OPTIONS ; SPECIFIC PAGE TABLES?
                    08   13   0E79  1103          BEQL    20$                     ; NO
     75 00000000'EF   0C   E1   0E7B  1104          BBC     #OPT$V_P1_PPT,OPTIONS,90$ ; P1 NOT REQUESTED
                        0E83  1105 20$:       SKIP    2
                        0E8C  1106          ENSURE  8
                        0EA4  1107          PRINT   0,<P1 page table>
                        0EB1  1108          PRINT   0,<-------------->
                        0EBE  1109          SKIP    1
           51   00D4 C3   D0   0EC7  1110          MOVL    PHD$L_P1LR(R3),R1       ; GET P1LR
                    2A   13   0ECC  1111          BEQL    90$                     ; SKIP IF ZERO
  51   00200000 8F   51   C3   0ECE  1112          SUBL3   R1,#^X200000,R1         ; TOTAL PAGES ALLOCATED
        50   51   09   78   0ED6  1113          ASHL    #9,R1,R0                ; SIZE OF P1 SPACE IN BYTES
                  6C A2   DD   0EDA  1114          PUSHL   PCB$L_PHD(R2)           ; ADDRESS OF PROCESS HEADER
           52   00D4 C3   D0   0EDD  1115          MOVL    PHD$L_P1LR(R3),R2       ; GET NUMBER OF PAGES
```

```
                00D0 D342   DF  0EE2  1116          PUSHAL  @PHD$L_P1BR(R3)[R2]      ; STARTING ADDRESS OF TABLE
          7E    80000000 8F   50  C3  0EE7  1117          SUBL3   R0,#^X80000000,-(SP)    ; STARTING ADDRESS BEING MAPPED
                        51    DD  0EEF  1118          PUSHL   R1                      ; TOTAL PAGES IN TABLE
                00000000'EF   04  FB  0EF1  1119          CALLS   #4,DUMP_PTE             ; FORMAT PAGE TABLE
                                  0EF8  1120
                                04  0EF8  1121 90$:      RET
                                  0EF9  1122
                                  0EF9  1123 100$:      SKIP    2
                                  0F02  1124          ENSURE  8
                                  0F1A  1125          PRINT   0,<Process page table>
                                  0F27  1126          PRINT   0,<------------------->
                                  0F34  1127          SKIP    1
                          6C A2   DD  0F3D  1128          PUSHL   PCB$L_PHD(R2)           ; ADDRESS OF PROCESS HEADER
                51    00000000'EF   D0  0F40  1129          MOVL    ESP,R1                  ; EXPRESSION STACK
                        54    61    7D  0F47  1130          MOVQ    (R1),R4                 ; R5 = LOW LIMIT
             03 00000000'EF   0D   E0  0F4A  1131          BBS     #OPT$V_PPT_LEN,OPTIONS,110$ ; R4 IS ALREADY LENGTH
                        54    55    C2  0F52  1132          SUBL    R5,R4                   ; GET LENGTH
                    55    01FF 8F   AA  0F55  1133 110$:    BICW    #^X1FF,R5               ; ROUND DOWN
                54    000003FF 8F   C0  0F5A  1134          ADDL    #^X3FF,R4
                 54    54    F7 8F   78  0F61  1135          ASHL    #-9,R4,R4               ; NUMBER OF PTE'S
                          00C8 C3   DD  0F66  1136          PUSHL   PHD$L_P0BR(R3)          ; ASSUME P0
                 51    55    F9 8F   78  0F6A  1137          ASHL    #-7,R5,R1               ; OFFSET IN PAGE TABLE
                       12 55    1E   E1  0F6F  1138          BBC     #30,R5,120$             ; IT WAS P0
                    6E    00D0 C3   D0  0F73  1139          MOVL    PHD$L_P1BR(R3),(SP)     ; USE P1
          51    55    40000000 8F   C3  0F78  1140          SUBL3   #^X40000000,R5,R1
                 51    51    F9 8F   78  0F80  1141          ASHL    #-7,R1,R1               ; OFFSET IN PAGE TABLE
                          6E    51    C0  0F85  1142 120$:    ADDL    R1,(SP)
                                55    DD  0F88  1143          PUSHL   R5                      ; BASE
                                54    DD  0F8A  1144          PUSHL   R4                      ; NUMBER OF ENTRIES
                00000000'EF   04  FB  0F8C  1145          CALLS   #4,DUMP_PTE             ; FORMAT PAGE TABLE
                                04  0F93  1146          RET
                                  0F94  1147
                                  0F94  1148          .DSABL  LSB
```

```
                                0F94  1150          .SBTTL  SHOW_CHANNELS -- DISPLAY ACTIVE CHANNELS
                                0F94  1151 ;---
                                0F94  1152 ;
                                0F94  1153 ;       SHOW_CHANNELS
                                0F94  1154 ;
                                0F94  1155 ;       THIS ROUTINE DISPLAYS THE INFORMATION IN THE PROCESS CHANNEL TABLE
                                0F94  1156 ;       PERTAINING TO THE ACTIVE CHANNELS.  FOR A CRASH FILE, YOU GET YOUR
                                0F94  1157 ;       CHOICE OF JUST THE DEVICE NAME OR A DEVICE NAME WITH THE FILE-ID.
                                0F94  1158 ;       FOR A RUNNING SYSTEM, THE FILE NAME (IF AVAILABLE) IS ALSO DISPLAYED.
                                0F94  1159 ;
                                0F94  1160 ;   INPUTS:
                                0F94  1161 ;
                                0F94  1162 ;       PCB = PROCESS CONTROL BLOCK
                                0F94  1163 ;       PHD = PROCESS HEADER BLOCK
                                0F94  1164 ;
                                0F94  1165 ;   OUTPUTS:
                                0F94  1166 ;
                                0F94  1167 ;       NONE
                                0F94  1168 ;
                                0F94  1169 ;---
                                0F94  1170
                                0F94  1171          .ENABL  LSB
                                0F94  1172
                                0F94  1173 SHOW_CHANNELS:
                         07FC   0F94  1174          .WORD   ^M<R2,R3,R4,R5,R6,R7,R8,R9,R10>
                                0F96  1175
                                0F96  1176          TRYMEM  @CTL$GL_CCBBASE         ;SEE IF CCB TABLE EXISTS
                      0A 50     E8  0FA3  1177          BLBS    R0,10$                  ;XFER IF AVAILABLE
                                0FA6  1178          SKIP    1                       ;ELSE FORCE ONE BLANK LINE
                            04  0FAF  1179          RET                             ;AND RETURN NOW
                 0000031C'EF    D4  0FB0  1180 10$:     CLRL    CHANTBL_SIZE            ;RESET SIZE
                                0FB6  1181          REQMEM  @CTL$GW_CHINDX,CHANTBL_SIZE,#2  ;GET CHANNEL HIGH WATER MARK
           57   0000031C'EF    D0  0FCB  1182          MOVL    CHANTBL_SIZE,R7         ;GET THE TABLE SIZE
                00000320'EF    9F  0FD2  1183          PUSHAB  CHANTBL_ADDR            ;GET MEMORY FOR CCB TABLE COPY
                0000031C'EF    9F  0FD8  1184          PUSHAB  CHANTBL_SIZE
           00000000'GF    02   FB  0FDE  1185          CALLS   #2,G^LIB$GET_VM
                      01 50    E8  0FE5  1186          BLBS    R0,20$                  ;XFER IF SUCCESSFUL
                               04  0FE8  1187          RET                             ;ELSE RETURN NOW
                                   0FE9  1188 20$:      SKIP    2
                                   0FF2  1189          PRINT   0,<!28* Process active channels>
                                   0FFF  1190          PRINT   0,<!28* ---------------------->
           000012E2'EF    00   FB  100C  1191          CALLS   #0,CHAN_TITLE           ;PRINT COLUMN HEADINGS
 00000000'EF   000012E2'EF    9E  1013  1192          MOVAB   CHAN_TITLE,HEADING_ROUTINE      ;SET HEADING ROUTINE ADDRESS
     57   00   00000320'FF    00  101E  1193          MOVC5   #0,@CHANTBL_ADDR,#0,R7,@CHANTBL_ADDR    ;CLEAR TABLE
                00000320'FF        1027
                                   102C  1194          REQMEM  @CTL$GL_CCBBASE,R6      ;GET CCB TABLE END ADDRESS
                      56   57    C2  103C  1195          SUBL2   R7,R6                   ;CALC START OF THE TABLE
                                   103F  1196          REQMEM  (R6),@CHANTBL_ADDR,R7   ;COPY CCB TABLE
                56   00000320'EF   57  C1  1050  1197          ADDL3   R7,CHANTBL_ADDR,R6      ;PROCESS FROM THE TOP DOWN
                      58   56    D0  1058  1198          MOVL    R6,R8                   ;SAVE FOR CALCULATING CHANNEL NUMBER
                         0261    31  105B  1199          BRW     160$                    ;JUMP INTO THE LOOP
                        09 A6    95  105E  1200 30$:     TSTB    CCB$B_AMOD(R6)          ;IS THE CHANNEL ACTIVE?
                           03    12  1061  1201          BNEQ    40$                     ;XFER IF SO
                         0259    31  1063  1202 35$:     BRW     160$                    ;ELSE GO TRY THE NEXT ONE
                      55   66    D0  1066  1203 40$:     MOVL    CCB$L_UCB(R6),R5        ;GET THE UCB ADDRESS
                           F8    13  1069  1204          BEQL    35$                     ;IF EQL THEN TRY NEXT ONE
                                   106B  1205          REQMEM  UCB$L_DDB(R5),R4        ;GET THE DDB ADDRESS
```

```
                                1078  1206 ;
                                1078  1207 ;  Read node name into local memory.  The node name will be displayed along
                                1078  1208 ;  with the device name.
                                1078  1209 ;
                000003C8'EF   D4  1078  1210          CLRL    NODE                            ; ZERO THE LENGTH FIELD
            53  00000000'EF   D0  107E  1211          MOVL    SCS$GA_LOCALSB,R3               ; LOCAL SYSTEM BLOCK
                               1085  1212          REQMEM  DDB$L_SB(R4),R0                 ; GET ADDRESS OF SYSTEM BLOCK
                    53  50     D1  1092  1213          CMPL    R0,R3                           ; IS THIS THE LOCAL NODE?
                        2E     13  1095  1214          BEQL    43$                             ; BRANCH, LOCAL NODE
        50  50  00000044 8F     C1  1097  1215          ADDL3   #SB$T_NODENAME,R0,R0            ; POINT TO NODE NAME
                               109F  1216          GETMEM  (R0),NODE,#SB$S_NODENAME        ; READ INTO LOCAL STORAGE
            50  000003C8'EF   9A  10B0  1217          MOVZBL  NODE,R0                         ; GET LENGTH OF NAME
          000003C9'EF40    24   90  10B7  1218          MOVB    #^A/$/,NODE+1[R0]               ; APPEND "$" TO NODE NAME
                000003C8'EF   96  10BF  1219          INCB    NODE                            ; INCREMENT LENGTH
                               10C5  1220 43$:
                               10C5  1221          REQMEM  DDB$T_NAME(R4),CHANDEV_NAME,#16 ;COPY THE DEVICE NAME
    00000324'EF   0000032C'EF   9A  10D7  1222          MOVZBL  CHANDEV_NAME,CHANDEV_SIZE       ;SET THE SIZE OF THE DEVICE NAME
                               10E2  1223
                               10E2  1224          REQMEM  UCB$W_UNIT(R5),R0       ;GET THE DEVICE UNIT NUMBER
                        7E     7C  10EF  1225          CLRQ    -(SP)                   ;DESCRIPTOR FOR THE OUTPUT STRING
                    54  5E     D0  10F1  1226          MOVL    SP,R4                   ;SAVE ADDRESS FOR LATER
        64  0F  00000324'EF   A3  10F4  1227          SUBW3   CHANDEV_SIZE,#15,(R4)   ;SET THE SIZE
    0000032D'8F   00000324'EF   C1  10FC  1228          ADDL3   CHANDEV_SIZE,#CHANDEV_NAME+1,4(R4)      ;SET THE ADDRESS
                      04 A4         1107
                        50  DD     1109  1229          PUSHL   R0                      ;STASH THE UNIT NUMBER
                        54  DD     110B  1230          PUSHL   R4                      ;FORMAT THE DEVICE NAME
                        54  DD     110D  1231          PUSHL   R4
                00001308'EF   9F  110F  1232          PUSHAB  CHANDEV_FMT
            00000000'GF   04   FB  1115  1233          CALLS   #4,G^SYS$FAO
                    50  8E     7D  111C  1234          MOVQ    (SP)+,R0                ;GET THE DESCRIPTOR & CLEAN THE STACK
            00000324'EF   50   A0  111F  1235          ADDW2   R0,CHANDEV_SIZE         ;SET FULL DEVICE NAME SIZE
        00  000003D8'EF   00   2C  1126  1236          MOVC5   #0,FCB,#0,#FCB$C_LENGTH,FCB     ;CLEAR THE FCB
        000003D8'EF   00B4 8F         112E
    30  00  0000048C'EF   00   2C  1136  1237          MOVC5   #0,WCB,#0,#WCB$C_LENGTH,WCB     ; AND THE WCB
                0000048C'EF         113F
              7E    F182 CF   9E  1144  1238          MOVAB   CHANSEC_TABLE,-(SP)     ;NO SECTION SO FAR
                    55  5E     D0  1149  1239          MOVL    SP,R5                   ;SAVE ADDRESS FOR LATER
                0000033C'EF   9F  114C  1240          PUSHAB  FILE_NAME               ;SET ADDRESS OF FILE NAME DESCRIPTOR
                0000033C'EF   D4  1152  1241          CLRL    FILE_NAME               ;NO FILE NAME SO FAR
    00000340'EF   00000344'EF   9E  1158  1242          MOVAB   FILE_NAMETXT,FILE_NAME+4        ;RESET ADDRESS
                  54  04 A6     D0  1163  1243          MOVL    CCB$L_WIND(R6),R4       ;GET THE WINDOW ADDRESS
                        05     13  1167  1244          BEQL    45$                     ;XFER IF NO WINDOW PRESENT
                    05  54     E9  1169  1245          BLBC    R4,50$                  ;XFER IF WINDOW NOW INTERLOCKED
                        54     D4  116C  1246          CLRL    R4                      ;ELSE NO WINDOW ADDRESS
                     00A9      31  116E  1247 45$:     BRW     70$                     ;SKIP THE FOLOWING TEST
                        54     D5  1171  1248 50$:     TSTL    R4                      ;REAL WINDOW OR PST OFFSET
                        20     19  1173  1249          BLSS    60$                     ;XFER IF REAL WINDOW
                    54  54     32  1175  1250          CVTWL   R4,R4                   ;ELSE EXTEND OFFSET
50  00000148'EF   00000070'EF   C1  1178  1251          ADDL3   PCB+PCB$L_PHD,PHD+PHD$L_PSTBASOFF,R0    ;CALC PST ADDRESS
                54  0C A044    DE  1184  1252          MOVAL   SEC$L_WINDOW(R0)[R4],R4 ;GET ADDRESS OF THE WINDOW ADDRESS
                               1189  1253          REQMEM  (R4),R4                 ;GET THE WINDOW ADDRESS
                               1195  1254 60$:     REQMEM  (R4),WCB,#WCB$C_LENGTH  ;COPY WINDOW (MINUS MAP POINTERS)
            12  00000496'EF   91  11A6  1255          CMPB    WCB+WCB$B_TYPE,#DYN$C_WCB       ;CHECK BLOCK TYPE
                        6B     12  11AD  1256          BNEQ    70$                     ;XFER IF NOT A WCB
            53  000004A4'EF   D0  11AF  1257          MOVL    WCB+WCB$L_FCB,R3        ;ELSE GET FCB ADDRESS
                        62     13  11B6  1258          BEQL    70$                     ;XFER IF NONE
                               11B8  1259          REQMEM  (R3),FCB,#FCB$C_LENGTH  ;ELSE COPY THE FCB
```

```
                07   000003E2'EF   91   11CD  1260          CMPB    FCB+FCB$B_TYPE,#DYN$C_FCB       ;CHECK BLOCK TYPE
                               44   12   11D4  1261          BNEQ    70$                             ;XFER IF NOT AN FCB
                     00000498'EF   B5   11D6  1262          TSTW    WCB+WCB$L_PID                   ;IS FILE A SECTION?
                               05   12   11DC  1263          BNEQ    65$                             ;XFER IF NOT
                  65   F0E9 CF    9E   11DE  1264          MOVAB   CHANSEC_TABLE+1,(R5)            ;ELSE SET NEW ADDRESS
           0000033C'EF   0084 8F    3C   11E3  1265 65$:     MOVZWL  #132,FILE_NAME                  ;ELSE RESET DESCRIPTOR SIZE
                               7E   D4   11EC  1266          CLRL    -(SP)                           ;MAKE ROOM FOR SECONDARY STATUS
                               5E   DD   11EE  1267          PUSHL   SP                              ;CONVERT FILE-ID TO A FILE NAME
                               00   DD   11F0  1268          PUSHL   #0
                     0000033C'EF   9F   11F2  1269          PUSHAB  FILE_NAME
                     0000033C'EF   9F   11F8  1270          PUSHAB  FILE_NAME
                     000003FC'EF   9F   11FE  1271          PUSHAB  FCB+FCB$W_FID
                     00000324'EF   9F   1204  1272          PUSHAB  CHANNEL_DEVICE
                00000000'GF    06   FB   120A  1273          CALLS   #6,G^LIB$FID_TO_NAME
                          51   8E   D0   1211  1274          MOVL    (SP)+,R1                        ;RETRIEVE SECONDARY STATUS
                            03 50   E9   1214  1275          BLBC    R0,70$                          ;CAN'T USE FILE NAME IF ANY ERRORS
                            10 51   E8   1217  1276          BLBS    R1,80$
                     0000033C'EF   D4   121A  1277 70$:     CLRL    FILE_NAME                       ;NO FILE NAME DESCRIPTOR
                     000003FC'EF   D5   1220  1278          TSTL    FCB+FCB$W_FID                   ;ANY FILE-ID?
                               1F   12   1226  1279          BNEQ    100$                            ;XFER IF SO
                               32   11   1228  1280          BRB     110$                            ;ELSE SKIP FOLLOWING
           0000033C'EF   5B 8F    3A   122A  1281 80$:     LOCC    #^A\[\,FILE_NAME,@FILE_NAME+4   ;STRIP DEVICE NAME
                     00000340'FF         1232
                               07   13   1237  1282          BEQL    90$                             ;XFER IF NONE
                0000033C'EF    50   7D   1239  1283          MOVQ    R0,FILE_NAME                    ;ELSE SAVE NEW DESCRIPTOR
                  15 00000000'EF   E8   1240  1284 90$:     BLBS    CURRENT_SYSTEM,110$             ;XFER IF THE RUNNING SYSTEM
                7E    00000400'EF   3C   1247  1285 100$:    MOVZWL  FCB+FCB$W_FID_RVN,-(SP)         ;ELSE SAVE THE FILE-ID
                7E    000003FE'EF   3C   124E  1286          MOVZWL  FCB+FCB$W_FID_SEQ,-(SP)
                7E    000003FC'EF   3C   1255  1287          MOVZWL  FCB+FCB$W_FID_NUM,-(SP)
                     00000324'EF   9F   125C  1288 110$:     PUSHAB  CHANNEL_DEVICE                  ;SET THE DEVICE NAME
                     000003C8'EF   9F   1262  1289          PUSHAB  NODE                            ;SET THE NODE NAME
                   7E   F050 CF    9E   1268  1290          MOVAB   CHANBUSY_TABLE,-(SP)            ;SET AS NOT BUSY
                          0A A6    B5   126D  1291          TSTW    CCB$W_IOC(R6)                   ;RIGHT?
                               05   13   1270  1292          BEQL    120$                            ;XFER IF SO
                6E    F047 CF    9E   1272  1293          MOVAB   CHANBUSY_TABLE+1,(SP)           ;ELSE NOTE BUSY
                7E    F048 CF    9E   1277  1294 120$:     MOVAB   CHANDPND_TABLE,-(SP)            ;NO DEACCESS PENDING
                          0C A6    D5   127C  1295          TSTL    CCB$L_DIRP(R6)                  ;RIGHT?
                               05   13   127F  1296          BEQL    130$                            ;XFER IF SO
                6E    F03F CF    9E   1281  1297          MOVAB   CHANDPND_TABLE+1,(SP)           ;ELSE NOTE DEACCESS
                               54   DD   1286  1298 130$:     PUSHL   R4                              ;SAVE THE WINDOW ADDRESS
                7E    58       56   C3   1288  1299          SUBL3   R6,R8,-(SP)                     ;SET THE CHANNEL NUMBER
                     000003FC'EF   D5   128C  1300          TSTL    FCB+FCB$W_FID                   ;ANY FILE-ID TO PRINT?
                               1E   13   1292  1301          BEQL    150$                            ;XFER IF NOT
                     0000033C'EF   B5   1294  1302          TSTW    FILE_NAME                       ;DO I HAVE A FILE NAME?
                               07   13   129A  1303          BEQL    140$                            ;YES, SKIP NEXT TEST
              0F 00000000'EF    E8   129C  1304          BLBS    CURRENT_SYSTEM,150$             ;DON'T DISPLAY FID IF CURRENT SYSTEM
                                     12A3  1305 140$:     PRINT   8,<  !XW         !XL            !5AC!5AC         !AC!AS:(!UW,!UW,!UW)!AS!AC>
                               0D   11   12B0  1306          BRB     160$                            ;GO GET THE NEXT CHANNEL BLOCK
                                     12B2  1307 150$:     PRINT   5,<  !XW         !XL            !5AC!5AC         !AC!AS:!AS!AC>
                          56   10   C2   12BF  1308 160$:     SUBL2   #CCB$C_LENGTH,R6                ;NEXT CCB PLEASE...
           00000320'EF    56   D1   12C2  1309          CMPL    R6,CHANTBL_ADDR                 ;DONE WITH ALL OF THEM?
                               03   1F   12C9  1310          BLSSU   170$                            ;XFER IF SO
                         FD90   31   12CB  1311          BRW     30$                             ;ELSE TRY ANOTHER
                     00000320'EF   9F   12CE  1312 170$:     PUSHAB  CHANTBL_ADDR                    ;RELEASE MEMORY ALLOCATED FOR CCB TABLE
                     0000031C'EF   9F   12D4  1313          PUSHAB  CHANTBL_SIZE
           00000000'GF    02   FB   12DA  1314          CALLS   #2,G^LIB$FREE_VM
                               04   12E1  1315          RET                                     ; AND RETURN
```

```
                        12E2  1316
                        12E2  1317 CHAN_TITLE:
                0000    12E2  1318          .WORD   0
                        12E4  1319          SKIP    1
                        12ED  1320          PRINT   0,<Channel      Window          Status      Device/file accessed
                        12FA  1321          PRINT   0,<-------      ------          ------      --------------------
                  04    1307  1322          RET
                        1308  1323
                        1308  1324 CHANDEV_FMT:
57 55 21 00001310'010E0000' 1308  1325          .ASCID  \!UW\                   ;FOR FORMATTING THE DEVICE UNIT NUMBER
                        1313  1326
                        1313  1327          .DSABL  LSB
```

```
                                1313  1329          .SBTTL  PROCESS_SUMMARY -- DISPLAY LIST OF RUNNING PROCESSES
                                1313  1330 ;---
                                1313  1331 ;
                                1313  1332 ;       PROCESS_SUMMARY
                                1313  1333 ;
                                1313  1334 ;       PRINT A LIST OF THE PROCESSES CURRENTLY RUNNING
                                1313  1335 ;
                                1313  1336 ;   INPUTS:
                                1313  1337 ;
                                1313  1338 ;       NONE
                                1313  1339 ;
                                1313  1340 ;   OUTPUTS:
                                1313  1341 ;
                                1313  1342 ;       NONE
                                1313  1343 ;
                                1313  1344 ;---
                                1313  1345
                                1313  1346          .ENABL  LSB
                                1313  1347
                                1313  1348 PROCESS_SUMMARY::
                         03FC   1313  1349          .WORD   ^M<R2,R3,R4,R5,R6,R7,R8,R9>
                                1315  1350
                                1315  1351          SUBHD   <Current process summary>
                                1322  1352          SKIP    PAGE
          0000147F'EF   00  FB  1329  1353          CALLS   #0,SUMMARY_TITLE        ; PRINT SUBHEADING LINE
00000000'EF   0000147F'EF   9E  1330  1354          MOVAB   SUMMARY_TITLE,HEADING_ROUTINE   ; SET HEADING ROUTINE
                                133B  1355
                                133P  1356          REQMEM  @SCH$GL_PCBVEC,R2       ; VECTOR OF PCB ADDRESSES
                                134B  1357          REQMEM  @SCH$GL_MAXPIX,R3       ; GET MAXIMUM PROCESS INDEX
                57   52   D0  135B  1358          MOVL    R2,R7                   ; ADDRESS OF NULL PROCESS SLOT
          58    000002A8'EF   9E  135E  1359          MOVAB   JIB,R8
          54    00C00C 4'EF   9E  1365  1360          MOVAB   PCB,R4
          55    00000128'EF   9E  136C  1361          MOVAB   PHD,R5
                                1373  1362 10$:
                                1373  1363          REQMEM  (R2),R9                 ; GET ADDRESS OF NEXT PCB
                                137F  1364          GETMEM  (R9),(R4),#PCB$C_LENGTH ; READ ENTIRE PCB
                  0A 50   E9    1390  1365          BLBC    R0,12$                  ; SKIP IF ERROR READING
                  60 A4   B5    1393  1366          TSTW    PCB$L_PID(R4)           ; CHECK IF NULL PROCESS
                     08   12    1396  1367          BNEQ    15$                     ; BRANCH IF NOT
                57   52   D1    1398  1368          CMPL    R2,R7                   ; IS THIS THE REAL NULL PROCESS?
                     03   13    139B  1369          BEQL    15$                     ; OK IF NULL PROCESS
                   00A0   31    139D  1370 12$:     BRW     50$                     ; BYPASS THIS ENTRY
                                13A0  1371 15$:
                                13A0  1372 ;        MOVC5   #0,(SP),#0,#JIB$C_LENGTH,(R8)   ; PRE-ZERO JIB IN CASE NOT THERE
     50    20202020   8F   D0  13A0  1373          MOVL    #^A"    ",R0            ; SET R0 TO SPACES
           0C A8      50   D0  13A7  1374          MOVL    R0,JIB$T_USERNAME(R8)   ; AND FILL THE USERNAME WITH SPACES
           10 A8      50   D0  13AB  1375          MOVL    R0,JIB$T_USERNAME+4(R8)
           14 A8      50   D0  13AF  1376          MOVL    R0,JIB$T_USERNAME+8(R8)
              0080    C4   D5  13B3  1377          TSTL    PCB$L_JIB(R4)           ; CHECK FOR JIB
                      13   13  13B7  1378          BEQL    18$                     ; BR IF NONE
                                13B9  1379          TRYMEM  @PCB$L_JIB(R4),(R8),#JIB$C_LENGTH  ; GET JIB
                                13CC  1380 18$:
     50    34 A4     36 A4   A1  13CC  1381          ADDW3   PCB$W_PPGCNT(R4),PCB$W_GPGCNT(R4),R0
                  7E   50   3C  13D2  1382          MOVZWL  R0,-(SP)                ; WORKING SET SIZE (PHYS. MEM.)
                     6C A4   DD  13D5  1383          PUSHL   PCB$L_PHD(R4)           ; PHD ADDRESS
                        59   DD  13D8  1384          PUSHL   R9                      ; PCB ADDRESS
     51    1F    0B A4   83  13DA  1385          SUBB3   PCB$B_PRI(R4),#31,R1    ; CURRENT PRIORITY
```

```
                7E    51    9A  13DF  1386          MOVZBL  R1,-(SP)
             02   2C A4    B1  13E2  1387          CMPW    PCB$W_STATE(R4),#SCH$C_MWAIT    ; CHECK FOR MWAIT
                    62    13  13E6  1388          BEQL    80$                             ; EQUAL, DETERMINE RESOURCE WAIT STATE
             51   2C A4    3C  13E8  1389 19$:     MOVZWL  PCB$W_STATE(R4),R1              ; CURRENT STATE
              EC2E CF41    7F  13EC  1390          PUSHAQ  STATE_TABLE[R1]
                        13F1  1391 20$:
                 0C A8    9F  13F1  1392          PUSHAB  JIB$T_USERNAME(R8)              ; ADDRESS OF USERNAME
                    0C    DD  13F4  1393          PUSHL   #12                             ; LENGTH OF USERNAME
                 71 A4    9F  13F6  1394          PUSHAB  PCB$T_LNAME+1(R4)               ; PROCESS NAME
                7E 70 A4    9A  13F9  1395          MOVZBL  PCB$T_LNAME(R4),-(SP)           ; PROCESS NAME LENGTH
                 60 A4    DD  13FD  1396          PUSHL   PCB$L_PID(R4)                   ; INTERNAL PROCESS IDENTIFICATION
                 64 A4    DD  1400  1397          PUSHL   PCB$L_EPID(R4)                  ; EXTENDED PROCESS IDENTIFICATION
                        1403  1398          PRINT   10,< !XL !XW !15AF !12AF !5AC !2UB !XL !XL !5UL>
          28 00000000'EF    00  E1  1410  1399          BBC     #OPT$V_IMAGE,OPTIONS,50$; CHECK IF /IMAGE SPECIFIED
                    EBE4 CF   9F  1418  1400          PUSHAB  NOTAVAIL                        ; PRESET TO USE "NOT AVAILABLE" STRING
             0000149C'EF    00  FB  141C  1401          CALLS   #0,READ_IMAGE                   ; READ IMAGE FILE NAME
                    03 50   E9  1423  1402          BLBC    R0,25$                          ; BRANCH IF ERROR READING
                 6E    56   D0  1426  1403          MOVL    R6,(SP)                         ; ADDRESS OF NAME BUFFER
                        1429  1404 25$:
                    00 BE   95  1429  1405          TSTB    @(SP)                           ; DID IT GIVE US A NULL?
                    05      12  142C  1406          BNEQ    35$
                 5E    04   C0  142E  1407          ADDL2   #4,SP                           ; POP THE STRING OFF, DON'T PRINT
                    0D      11  1431  1408          BRB     50$
                        1433  1409 35$:     PRINT   1,<          !AC>
                        1440  1410 50$:
                 52    04   C0  1440  1411          ADDL2   #4,R2                           ; NEXT PCB ADDRESS
                    53      D7  1443  1412          DECL    R3                              ; DECREMENT ITERATION COUNT
                    29      19  1445  1413          BLSS    90$                             ; BRANCH IF FINISHED
                   FF29     31  1447  1414          BRW     10$                             ; CONTINUE UNTIL DONE
                        144A  1415
                    0C      BB  144A  1416 80$:     PUSHR   #^M<R2,R3>                      ; SAVE NEXT PCB ADDRESS AND COUNTER
              52    4C A4    D0  144C  1417          MOVL    PCB$L_EFWM(R4),R2               ; RESOURCE WAIT
           53    EDOB CF    9E  1450  1418          MOVAB   RESOURCE_WAIT,R3                ; ADDRESS OF TABLE OF STATES
            00000000'GF    16  1455  1419          JSB     G^TRANSLATE_ADDRESS             ; DETERMINE STATE
                    0C      BA  145B  1420          POPR    #^M<R2,R3>                      ; RESTORE PCB ADDRESS AND COUNTER
                    0C      12  145D  1421          BNEQ    85$                             ; MATCH FOUND, IF NOT EQUAL
                 4C A4      D5  145F  1422          TSTL    PCB$L_EFWM(R4)                  ; CHECK FOR MUTEX ADDRESS
                    84      18  1462  1423          BGEQ    19$                             ; BRANCH IF NOT ADDRESS
                EDC3 CF     9F  1464  1424          PUSHAB  MWAIT                           ; ADDRESS OF MUTEX COUNTED STRING
                   FF86     31  1468  1425          BRW     20$                             ; RETURN TO MAIN PATH
                    50      DD  146B  1426 85$:     PUSHL   R0                              ; ADDRESS OF RESOURCE WAITING FOR
                   FF81     31  146D  1427          BRW     20$                             ; RETURN TO MAIN PATH
                        1470  1428
                        1470  1429 90$:
          00000000'EF    00  FB  1470  1430          CALLS   #0,SET_PROCESS                  ; RESET PROCESS PAGE REGISTERS
                        1477  1431          STATUS  SUCCESS
                    04      147E  1432          RET
                        147F  1433
                        147F  1434          .DSABL  LSB
                        147F  1435
                        147F  1436 ;
                        147F  1437 ;        SUBROUTINE TO PRINT TITLE LINE
                        147F  1438 ;
                        147F  1439 SUMMARY_TITLE:
                      0000  147F  1440          .WORD   0
                        1481  1441
                        1481  1442          PRINT   0,< Extended Indx Process name    Username    State Pri   PCB      P
```

```
                148E  1443          PRINT   0,<-- PID -- ---- -------------- ---------- ----- --- -------- ---
 04             149B  1444          RET
```

```
                                149C  1446          .SBTTL  READ_IMAGE, READ IMAGE FILE NAME STRING
                                149C  1447 ;---
                                149C  1448 ;
                                149C  1449 ;       SUBROUTINE TO READ THE IMAGE FILE NAME FOR THE
                                149C  1450 ;       PROCESS DESCRIBED BY THE SPECIFIED PCB AND PHD.
                                149C  1451 ;
                                149C  1452 ;       R4 = ADDRESS OF PCB
                                149C  1453 ;       R5 = ADDRESS OF PHD
                                149C  1454 ;       CALL
                                149C  1455 ;       R0 = SUCCESS/FAILURE FLAG
                                149C  1456 ;       R6 = ADDRESS OF BUFFER CONTAINING ASCIC FILE NAME
                                149C  1457 ;
                                149C  1458 ;---
                                149C  1459
                                149C  1460 READ_IMAGE:
                          0004  149C  1461          .WORD   ^M<R2>
                                149E  1462
    56  00000000'EF     9E    149E  1463          MOVAB   L^BUFFER,R6             ; SET BUFFER ADDRESS
                  66    94    14A5  1464          CLRB    (R6)                    ; PRESET TO NULL STRING
                01 60 A4    B1    14A7  1465          CMPW    PCB$L_PID(R4),#1        ; CHECK IF INDEX = 0 OR 1
                      03    1A    14AB  1466          BGTRU   15$                     ; IF NULL,SWAPPER SKIP THIS
                    0087    31    14AD  1467          BRW     50$                     ; IF NULL,SWAPPER SKIP THIS
                   6C A4    D5    14B0  1468 15$:     TSTL    PCB$L_PHD(R4)           ; CHECK IF ZERO
                      15    13    14B3  1469          BEQL    18$                     ; BRANCH IF SWAPPED OUT
                            14B5  1470          TRYMEM  @PCB$L_PHD(R4),(R5),#PHD$C_LENGTH  ; READ PROCESS HEADER
                03 50     E8    14C7  1471          BLBS    R0,20$                  ; BRANCH IF OK
                   50     D4    14CA  1472 18$:     CLRL    R0
                          04    14CC  1473          RET
    50  00000000'EF     7E    14CD  1474 20$:     MOVAQ   P0BR,R0
            60   00C8 C5    7D    14D4  1475          MOVQ    PHD$L_P0BR(R5),(R0)     ; SETUP RELOCATION REGISTERS
         08 A0   00D0 C5    7D    14D9  1476          MOVQ    PHD$L_P1BR(R5),8(R0)
04 A0    04 A0   18   00    EF    14DF  1477          EXTZV   #PHD$V_P0LR,#PHD$S_P0LR,4(R0),4(R0)
     00000000'EF    60 A4    D0    14E6  1478          MOVL    PCB$L_PID(R4),PROC_PID  ; SET PROCESS PID TO READ
          4C 00000000'EF    E8    14EE  1479          BLBS    VERSION_FLAGS,10$       ; BRANCH IF RELEASE 2
                            14F5  1480 ;
                            14F5  1481 ;       VMS RELEASE 1 - IMAGE FILE NAME RESIDES IN IFD BUFFER
                            14F5  1482 ;
    51  00000000'EF     D0    14F5  1483          MOVL    MMG$IMGHDRBUF,R1        ; ADDRESS OF IMAGE BUFFER
                            14FC  1484          TRYMEM  4(R1),R2                ; GET ADDRESS OF IFD
                  2E 50     E9    1509  1485          BLBC    R0,90$                  ; BRANCH IF ERROR
                            150C  1486          TRYMEM  IFD$W_FILNAMOFF(R2)     ; GET OFFSET TO FILE NAME
                  21 50     E9    1516  1487          BLBC    R0,90$                  ; BRANCH IF ERROR
                51   51     32    1519  1488          CVTWL   R1,R1
                     19     13    151C  1489          BEQL    50$                     ; BRANCH IF ZERO OFFSET
                            151E  1490          TRYMEM  (R2)[R1],(R6),#32       ; GET ASCIC IMAGE FILE NAME
                  0B 50     E9    152C  1491          BLBC    R0,90$                  ; BRANCH IF ERROR
                20   66     91    152F  1492          CMPB    (R6),#32                ; CHECK IF OVER 32 CHARACTERS
                     03     1B    1532  1493          BLEQU   50$                     ; BRANCH IF OK
                66   20     90    1534  1494          MOVB    #32,(R6)                ; TRUNCATE TO 32 CHARACTERS
                50   01     D0    1537  1495 50$:     MOVL    #1,R0                   ; SET NO IMAGE NAME
          00000000'EF     D4    153A  1496 90$:     CLRL    PROC_PID                ; CLEAR PID ON EXIT SO THAT MEMORY
                            1540  1497                                            ; REQUESTS FOR SYSTEM SPACE WILL GO
                            1540  1498                                            ; FASTER (DON'T NEED THIS CONTEXT)
                          04    1540  1499          RET
                            1541  1500 ;
                            1541  1501 ;       VMS RELEASE 2 - IMAGE FILE DESCRIBED BY IFD$Q_CURPROG
                            1541  1502 ;
```

```
                     1541  1503 10$:     TRYMEM  @CTL$GL_IMGHDRBF        ; READ ADDRESS OF IMAGE HEADER
E9 50    E9  154E   1504                 BLBC    R0,90$                  ; BRANCH IF ERROR
   51    D5  1551   1505                 TSTL    R1                      ; ANY IMAGE HEADER?
   E2    13  1553   1506                 BEQL    50$                     ; BRANCH IF NONE
                     1555  1507         TRYMEM  4(R1)                   ; READ IFD POINTER
D8 50    E9  155F   1508                 BLBC    R0,90$                  ; BRANCH IF ERROR
                     1562  1509         TRYMEM  IFD$Q_CURPROG(R1),(R6),#8       ; READ DESCRIPTOR
C7 50    E9  1570   1510                 BLBC    R0,90$                  ; BRANCH IF ERROR
50 66    9A  1573   1511                 MOVZBL  (R6),R0                 ; CHECK LENGTH OF STRING
   BF    13  1576   1512                 BEQL    50$                     ; BRANCH IF NULL
                     1578  1513         TRYMEM  @4(R6),1(R6),R0         ; READ FILE NAME STRING
B0 50    E9  1587   1514                 BLBC    R0,90$                  ; BRANCH IF ERROR
   AB    11  158A   1515                 BRB     50$                     ; SUCCESS
```

```
158C  1517
158C  1518          .END
```

```
$$$                     = 0000009F R     04
ALLOCATE                  ********   X   03
ARGS                    = 00000003
AST                       000001D7 R     03
AST_TABLE                 000002DB R     03
BREAK                     000001F5 R     03
BUFFER                    ********   X   03
CCB$B_AMOD              = 00000009
CCB$C_LENGTH            = 00000010
CCB$L_DIRP              = 0000000C
CCB$L_UCB               = 00000000
CCB$L_WIND              = 00000004
CCB$W_IOC               = 0000000A
CHANBUSY_TABLE            000002BC R     03
CHANDEV_FMT               00001308 R     03
CHANDEV_NAME              0000032C R     02
CHANDEV_SIZE              00000324 R     02
CHANDPND_TABLE            000002C3 R     03
CHANNEL_DEVICE            00000324 R     02
CHANSEC_TABLE             000002CA R     03
CHANTBL_ADDR              00000320 R     02
CHANTBL_SIZE              0000031C R     02
CHAN_TITLE                000012E2 R     03
CLUSTER                   00000225 R     03
CRF_TABLE                 00000237 R     03
CSL                     = 00000008
CTL$GL_CCBBASE            ********   X   03
CTL$GL_IMGHDRBF           ********   X   03
CTL$GW_CHINDX             ********   X   03
CURRENT_SYSTEM            ********   X   03
DDB$L_SB                = C0000034
DDB$T_NAME              = 00000014
DEALLOCATE                ********   X   03
DISPLAY_PROCESS           00000459 RG    03
DISPLAY_PROCS             00000395 RG    03
DUMP_PTE                  ********   X   03
DYN$C_FCB               = 00000007
DYN$C_WCB               = 00000012
DZRO_TABLE                0000023D R     03
ESP                       ********   X   03
FCB                       000003D8 R     02
FCB$B_TYPE              = 0000000A
FCB$C_LENGTH            = 000000B4
FCB$W_FID               = 00000024
FCB$W_FID_NUM           = 00000024
FCB$W_FID_RVN           = 00000028
FCB$W_FID_SEQ           = 00000026
FILE_NAME                 0000033C R     02
FILE_NAMETXT              00000344 R     02
GBL_TABLE                 00000231 R     03
GETMEM                    ********   X   03
GOODPAGE_TABLE            000002B1 R     03
HDRSTR1                   0000035B R     03
HEADING_ROUTINE           ********   X   03
IFD$Q_CORPROG           = 00000014
IFD$W_FILNAMOFF         = 00000002
IMGACT                    000001FB R     03
INIT_PFN                  ********   X   03
JIB                       000002A8 RG    02
JIB$C_LENGTH            = 00000074
JIB$L_BYTCNT            = 00000020
JIB$L_BYTLM             = 00000024
JIB$L_MPID              = 00000054
JIB$L_PGFLCNT           = 0000003C
JIB$L_PGFLQUOTA         = 00000038
JIB$T_USERNAME          = 0000000C
JIB$W_FILCNT            = 00000030
JIB$W_FILLM             = 00000032
JIB$W_PRCCNT            = 00000044
JIB$W_PRCLIM            = 00000046
JIB$W_TQCNT             = 00000034
JIB$W_TQLM              = 00000036
JIBADR                    000002A4 RG    02
JOB_QUOTA                 00000201 R     03
L                       = 00000003
LIB$FID_TO_NAME           ********   X   03
LIB$FREE_VM               ********   X   03
LIB$GET_VM                ********   X   03
LIB$SIGNAL                ********   X   03
LINE_COUNT                ********   X   03
LOCKS                     00000207 R     03
MBX                       000001DD R     03
MMG$IMGHDRBUF             ********   X   03
MODIFY_TABLE              00000293 R     03
MPAGE_BUSY                00000219 R     03
MPAGE_EMPTY               00000213 R     03
MSG$_NOTRES               ********   X   03
MSG$_SUCCESS              ********   X   03
MWAIT                     0000022B R     03
NEW_PAGE                  ********   X   03
NODE                      000003C8 R     02
NONPAGED                  000001E3 R     03
NOTAVAIL                  00000000 R     03
OPT$M_CHAN              = 00000400
OPT$M_P0_PPT            = 00000800
OPT$M_P1_PPT            = 00001000
OPT$M_PHD               = 00000020
OPT$M_PPT               = 00000002
OPT$M_PPT_LEN             00002000
OPT$M_PPT_RNG           = 00004000
OPT$M_PST               = 00000004
OPT$M_REGS              = 00000008
OPT$M_SYSPROC           = 00000040
OPT$M_WSL               = 00000001
OPT$V_CHAN              = 0000000A
OPT$V_IMAGE             = 00000000
OPT$V_LCK               = 00000009
OPT$V_P0_PPT            = 0000000B
OPT$V_P1_PPT            = 0000000C
OPT$V_PCB               = 00000004
OPT$V_PHD               = 00000005
OPT$V_PPT               = 00000001
OPT$V_PPT_LEN           = 0000000D
OPT$V_PST               = 00000002
```

```
OPT$V_REGS                      =  00000003
OPT$V_RMS                       =  00000007
OPT$V_RMSD                      =  00000008
OPT$V_WSL                       =  00000000
OPTIONS                            ********    X   03
POBR                               ********    X   03
PAGED                              000001EF R      03
PAGE_SIZE                          ********    X   03
PAGING                             000001E9 R      03
PAGTYP_TABLE                       00000253 R      03
PCB                                00000004 RG     02
PCB$B_ASTACT                    =  0000000C
PCB$B_ASTEN                     =  0000000D
PCB$B_PRI                       =  0000000B
PCB$B_PRIB                      =  0000002F
PCB$B_WEFC                      =  0000002E
PCB$C_LENGTH                    =  00000120
PCB$L_EFC2P                     =  00000058
PCB$L_EFC3P                     =  0000005C
PCB$L_EFCS                      =  00000050
PCB$L_EFCU                      =  00000054
PCB$L_EFWM                      =  0000004C
PCB$L_EOWNER                    =  00000068
PCB$L_EPID                      =  00000064
PCB$L_JIB                       =  00000080
PCB$L_OWNER                     =  0000001C
PCB$L_PHD                       =  0000006C
PCB$L_PID                       =  00000060
PCB$L_STS                       =  00000024
PCB$L_WSSWP                     =  00000020
PCB$L_WTIME                     =  00000028
PCB$T_LNAME                     =  00000070
PCB$V_ASTPEN                    =  00000011
PCB$V_BATCH                     =  0000000E
PCB$V_DELPEN                    =  00000001
PCB$V_FORCPEN                   =  00000002
PCB$V_HIBER                     =  00000013
PCB$V_INQUAN                    =  00000003
PCB$V_LOGIN                     =  00000014
PCB$V_NETWRK                    =  00000015
PCB$V_NOACNT                    =  0000000F
PCB$V_NODELET                   =  00000017
PCB$V_PHDRES                    =  00000012
PCB$V_PSWAPM                    =  00000004
PCB$V_PWRAST                    =  00000016
PCB$V_RES                       =  00000000
PCB$V_RESPEN                    =  00000005
PCB$V_SSFEXC                    =  00000006
PCB$V_SSFEXCE                   =  00000007
PCB$V_SSFEXCS                   =  00000008
PCB$V_SSFEXCU                   =  00000009
PCB$V_SSRWAIT                   =  0000000A
PCB$V_SUSPEN                    =  0000000B
PCB$V_SWPVBN                    =  00000010
PCB$V_WAKEPEN                   =  0000000C
PCB$V_WALL                      =  0000000D
PCB$W_APTCNT                    =  00000030
PCB$W_ASTCNT                    =  00000038
PCB$W_BIOCNT                    =  0000003A
PCB$W_BIOLM                     =  0000003C
PCB$W_DIOCNT                    =  0000003E
PCB$W_DIOLM                     =  00000040
PCB$W_GPGCNT                    =  00000034
PCB$W_GRP                       =  000000BE
PCB$W_MEM                       =  000000BC
PCB$W_MTXCNT                    =  0000000E
PCB$W_PPGCNT                    =  00000036
PCB$W_STATE                     =  0000002C
PCB$W_TMBU                      =  00000032
PCBADR                             00000000 RG     02
PFNLOCK_TABLE                      000002A7 R      03
PHD                                00000128 RG     02
PHD$B_DFPFC                     =  00000034
PHD$B_PGTBPFC                   =  00000035
PHD$C_LENGTH                    =  0000017C
PHD$L_BAK                       =  00000044
PHD$L_BIOCNT                    =  00000058
PHD$L_CPULIM                    =  0000005C
PHD$L_CPUTIM                    =  00000038
PHD$L_DIOCNT                    =  00000054
PHD$L_ESP                       =  0000007C
PHD$L_FREPOVA                   =  00000028
PHD$L_FREP1VA                   =  00000030
PHD$L_FREPTECNT                 =  0000002C
PHD$L_KSP                       =  00000078
PHD$L_POBR                      =  000000C8
PHD$L_POLRASTL                  =  000000CC
PHD$L_P1BR                      =  000000D0
PHD$L_P1LR                      =  000000D4
PHD$L_PAGEFLTS                  =  0000004C
PHD$L_PAGFIL                    =  0000001C
PHD$L_PC                        =  000000C0
PHD$L_PSL                       .  000000C4
PHD$L_PSTBASOFF                 =  00000020
PHD$L_PTWSLELCK                 =  00000064
PHD$L_PTWSLEVAL                 =  00000068
PHD$L_R0                        =  00000088
PHD$L_R1                        =  0000008C
PHD$L_R10                       =  000000B0
PHD$L_R11                       =  000000B4
PHD$L_R12                       =  000000B8
PHD$L_R13                       =  000000BC
PHD$L_R2                        =  00000090
PHD$L_R3                        =  00000094
PHD$L_R4                        =  00000098
PHD$L_R5                        =  0000009C
PHD$L_R6                        =  000000A0
PHD$L_R7                        =  000000A4
PHD$L_R8                        =  000000A8
PHD$L_R9                        =  000000AC
PHD$L_SSP                       =  00000080
PHD$L_USP                       =  00000084
PHD$L_WSLX                      =  00000048
PHD$S_POLR                      =  00000018
```

```
PHD$V_POLR                      = 00000000
PHD$W_ASTLM                     = 00000040
PHD$W_DFWSCNT                   = 0000001A
PHD$W_EXTDYNWS                  = 00000076
PHD$W_FLAGS                     = 00000036
PHD$W_PHVINDEX                  = 00000042
PHD$W_PSTFREE                   = 00000026
PHD$W_PSTLAST                   = 00000024
PHD$W_PTCNTACT                  = 00000070
PHD$W_PTCNTLCK                  = 0000006C
PHD$W_PTCNTMAX                  = 00000072
PHD$W_PTCNTVAL                  = 0000006E
PHD$W_QUANT                     = 0000003C
PHD$W_WSAUTH                    = 0000000A
PHD$W_WSDYN                     = 0000000E
PHD$W_WSFLUID                   = 00000074
PHD$W_WSLAST                    = 00000012
PHD$W_WSLIST                    = 00000008
PHD$W_WSLOCK                    = 0000000C
PHD$W_WSNEXT                    = 00000010
PHD$W_WSQUOTA                   = 00000018
PHDADR                            00000124 RG    02
PRINT                             ******** X     03
PROCESS_STATUS                    00000097 R     03
PROCESS_SUMMARY                   00001313 RG    03
PROC_NAME                         ******** X     03
PROC_PID                          ******** X     03
PST_TITLE                         00000DB5 R     03
READ_IMAGE                        0000149C R     03
REQMEM                            ******** X     03
RESOURCE_WAIT                     0000015F R     03
RSN$_ASTQAIT                    = 00000001
RSN$_BRKTHRU                    = 00000006
RSN$_CLUSTRAN                   = 0000000E
RSN$_IACLOCK                    = 00000007
RSN$_JQUOTA                     = 00000008
RSN$_LOCKID                     = 00000009
RSN$_MAILBOX                    = 00000002
RSN$_MAX                        = 0000000F
RSN$_MPLEMPTY                   = 0000000B
RSN$_MPWBUSY                    = 0000000C
RSN$_NPDYNMEM                   = 00000003
RSN$_PGDYNMEM                   = 00000005
RSN$_PGFILE                     = 00000004
RSN$_SCS                        = 0000000D
RSN$_SWPFILE                    = 0000000A
S                               = 00000006
SAV                             = 00000353 R     03
SB$S_NODENAME                   = 00000010
SB$T_NODENAME                   = 00000044
SCH$C_MWAIT                     = 00000002
SCH$GL_MAXPIX                     ******** X     03
SCH$GL_PCBVEC                     ******** X     03
SCS$GA_LOCALSB                    ******** X     03
SCSWAIT                           0000021F R     03
SEC$C_LENGTH                    = 00000020
SEC$L_CCB                       = 00000000
SEC$L_PAGCNT                    = 0000001C
SEC$L_REFCNT                    = 00000018
SEC$L_VBN                       = 00000010
SEC$L_VPXPFC                    = 00000008
SEC$L_WINDOW                    = 0000000C
SEC$S_PFC                       = 00000008
SEC$S_VPX                       = 00000016
SEC$V_CRF                       = 00000001
SEC$V_DZRO                      = 00000002
SEC$V_GBL                       = 00000000
SEC$V_PFC                       = 00000018
SEC$V_VPX                       = 00000000
SEC$V_WRT                       = 00000003
SEC$W_FLAGS                     = 00000014
SEC$W_SECXBL                    = 00000006
SEC$W_SECXFL                    = 00000004
SET_HEADING                       ********     X   03
SET_PROCESS                       ********     X   03
SHOW_CHANNELS                     00000F94 R       03
SHOW_PCB                          000005B0 R       03
SHOW_PHD                          000007BA R       03
SHOW_PPT                          00000DD7 R       03
SHOW_PROC_LOCK                    ********     X   03
SHOW_PST                          00000C35 R       03
SHOW_REGS                         00000934 R       03
SHOW_RMS                          ********     X   03
SHOW_RMS_DIS                      ********     X   03
SHOW_WSL                          00000A31 R       03
SKIP_LINES                        ********     X   03
STATE_TABLE                       0000001F R       03
STS$K_WARNING                   = 00000000
SUMMARY_TITLE                     0000147F R       03
SWAP                              0000120D R       03
SYS$FAO                           ********     X   03
TRANSLATE_ADDRESS                 ********     X   03
TRANSLATE_BITS                    ********     X   03
TRYMEM                            ********     X   03
UCB$L_DDB                       = 00000028
UCB$W_UNIT                      = 00000054
VALID_TABLE                       0000024B R       03
VERSION_FLAGS                     ********     X   03
WCB                               0000048C R       02
WCB$B_TYPE                      = 0000000A
WCB$C_LENGTH                    = 00000030
WCB$L_FCB                       = 00000018
WCB$L_PID                       = 0000000C
WRT_TABLE                         00000245 R       03
WSL$S_PAGTYP                    = 00000003
WSL$V_GOODPAGE                  = 00000006
WSL$V_MODIFY                    = 00000008
WSL$V_PAGTYP                    = 00000001
WSL$V_PFNLOCK                   = 00000004
WSL$V_VALID                     = 00000000
WSL$V_WSLOCK                    = 00000005
WSLOCK_TABLE                      0000029E R       03
WSL_TITLE                         00000C13 R       03
```

Psect synopsis

PSECT name	Allocation	PSECT No.	Attributes
. ABS .	00000000 (0.)	00 (0.)	NOPIC USR CON ABS LCL NOSHR NOEXE NORD NOWRT NOVEC BYTE
ABS	00000000 (0.)	01 (1.)	NOPIC USR CON ABS LCL NOSHR EXE RD WRT NOVEC BYTE
SDADATA	000004BC (1212.)	02 (2.)	NOPIC USR CON REL LCL NOSHR NOEXE RD WRT NOVEC BYTE
PROCESS	0000158C (5516.)	03 (3.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE
LITERALS	00001208 (4616.)	04 (4.)	NOPIC USR CON REL LCL NOSHR EXE RD NOWRT NOVEC BYTE

Performance indicators

Phase	Page faults	CPU Time	Elapsed Time
Initialization	29	00:00:00.05	00:00:00.90
Command processing	107	00:00:00.45	00:00:03.12
Pass 1	530	00:00:13.56	00:00:48.08
Symbol table sort	0	00:00:01.65	00:00:04.86
Pass 2	286	00:00:04.07	00:00:17.61
Symbol table output	39	00:00:00.18	00:00:00.18
Psect synopsis output	3	00:00:00.03	00:00:00.20
Cross-reference output	0	00:00:00.00	00:00:00.00
Assembler run totals	996	00:00:19.99	00:01:14.96

The working set limit was 2250 pages.
139106 bytes (272 pages) of virtual memory were used to buffer the intermediate code.
There were 80 pages of symbol table space allocated to hold 1422 non-local and 171 local symbols.
1518 source lines were read in Pass 1, producing 54 object records in Pass 2.
42 pages of virtual memory were used to define 41 macros.

Macro library statistics

Macro library name	Macros defined
_255DUA28:[SDA.OBJ]SDALIB.MLB;1	15
_255DUA28:[SYS.OBJ]LIB.MLB;1	14
_255DUA28:[SYSLIB]STARLET.MLB;2	8
TOTALS (all libraries)	37

1597 GETS were required to define 37 macros.

There were no errors, warnings or information messages.

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

PROCESS
LIS

RMS
LIS

QAST
LIS

RELEASE
LIS

POOL
LIS