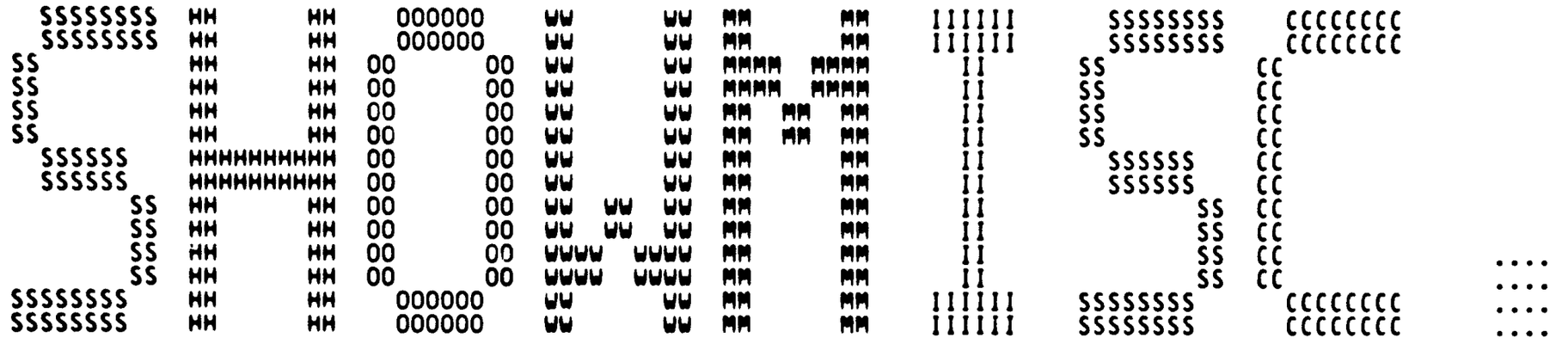
```
   CCCCCCCCCCCC  LLL                  IIIIIIIII     UUU         UUU   TTTTTTTTTTTTTTT  LLL
   CCCCCCCCCCCC  LLL                  IIIIIIIII     UUU         UUU   TTTTTTTTTTTTTTT  LLL
   CCCCCCCCCCCC  LLL                  IIIIIIIII     UUU         UUU   TTTTTTTTTTTTTTT  LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
CCC              LLL                     III        UUU         UUU         TTT        LLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL      IIIIIIIII     UUUUUUUUUUUUUUU         TTT        LLLLLLLLLLLLLLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL      IIIIIIIII     UUUUUUUUUUUUUUU         TTT        LLLLLLLLLLLLLLL
   CCCCCCCCCCCC  LLLLLLLLLLLLLLL      IIIIIIIII     UUUUUUUUUUUUUUU         TTT        LLLLLLLLLLLLLLL
```

FILEID**SHOWMISC



```
;     1          0001  0 MODULE showmisc (IDENT = 'V04-000',
;     2          0002  0                   ADDRESSING_MODE (EXTERNAL = GENERAL)) =
;     3          0003  0
;     4          0004  1 BEGIN
;     5          0005  1 !
;     6          0006  1 !
;     7          0007  1 !**************************************************************************
;     8          0008  1 !*                                                                        *
;     9          0009  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                              *
;    10          0010  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                *
;    11          0011  1 !*  ALL RIGHTS RESERVED.                                                  *
;    12          0012  1 !*                                                                        *
;    13          0013  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
;    14          0014  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
;    15          0015  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    16          0016  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    17          0017  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    18          0018  1 !*  TRANSFERRED.                                                           *
;    19          0019  1 !*                                                                        *
;    20          0020  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE  *
;    21          0021  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT  *
;    22          0022  1 !*  CORPORATION.                                                           *
;    23          0023  1 !*                                                                        *
;    24          0024  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
;    25          0025  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                *
;    26          0026  1 !*                                                                        *
;    27          0027  1 !*                                                                        *
;    28          0028  1 !**************************************************************************
;    29          0029  1 !
;    30          0030  1 !
;    31          0031  1 !++
;    32          0032  1 !
;    33          0033  1 ! FACILITY:  SHOW utility
;    34          0034  1 !
;    35          0035  1 ! ABSTRACT:
;    36          0036  1 !       This module contains the option routines for SHOW RMS, SHOW
;    37          0037  1 !       WORKING_SET, SHOW ACCOUNTING
;    38          0038  1 !
;    39          0039  1 ! ENVIRONMENT:
;    40          0040  1 !       VAX native, user mode.
;    41          0041  1 !
;    42          0042  1 ! AUTHOR:  Gerry Smith           CREATION DATE:  25-Jun-1982
;    43          0043  1 !
;    44          0044  1 ! MODIFIED BY:
;    45          0045  1 !
;    46          0046  1 !       V03-007 RAS0281         Ron Schaefer            27-Mar-1984
;    47          0047  1 !               Add Network Block Count display to SHOW RMS;
;    48          0048  1 !               adjust display alignments.
;    49          0049  1 !
;    50          0050  1 !       V03-006 MCN0155         Maria del C. Nasr       01-Mar-1984
;    51          0051  1 !               The offset to the disallow flags is from the start of
;    52          0052  1 !               the PCB structure, and not the longword.
;    53          0053  1 !
;    54          0054  1 !       V03-005 MCN0150         Maria del C. Nasr       09-Feb-1984
;    55          0055  1 !               Make display for SHOW WORKING_SET values larger.
;    56          0056  1 !
;    57          0057  1 !       V03-004 GAS0148         Gerry Smith     27-Jun-1983
```

```
;    58          0058  1 !               Update SHOW ACCOUNTING to display all classes
;    59          0059  1 !               of accounting.
;    60          0060  1 !
;    61          0061  1 !       V03-003 GAS0125         Gerry Smith     20-Apr-1983
;    62          0062  1 !               Fix displays for SHOW RMS_DEFAULT
;    63          0063  1 !
;    64          0064  1 !       V03-002 GAS0107         Gerry Smith      8-Feb-1983
;    65          0065  1 !               Extend the output field width for SHOW WORKING_SET.
;    66          0066  1 !
;    67          0067  1 !       V03-001 GAS0105         Gerry Smith     31-Jan-1983
;    68          0068  1 !               Fix the output of SHOW RMS so that the multi-buffer
;    69          0069  1 !               counts come out correctly.
;    70          0070  1 !
;    71          0071  1 !--
```

```
;    73          0072  1
;    74          0073  1
;    75          0074  1 ! Include files
;    76          0075  1 !
;    77          0076  1
;    78          0077  1 LIBRARY 'SYS$LIBRARY:LIB';              ! VAX/VMS system definitions
;    79          0078  1 REQUIRE 'SRC$:SHOWDEF';                ! SHOW common definitions
;    80          0177  1
```

```
;    82          0178  1 !
;    83          0179  1 ! Table of contents
;    84          0180  1 !
;    85          0181  1
;    86          0182  1 FORWARD ROUTINE
;    87          0183  1     show$accounting : NOVALUE,
;    88          0184  1     show$rms_default : NOVALUE,
;    89          0185  1     show$working_set : NOVALUE;
;    90          0186  1
;    91          0187  1 EXTERNAL
;    92          0188  1     exe$gl_acmflags,
;    93          0189  1     sys$gb_dfmbfsur,
;    94          0190  1     sys$gb_dfmbfsmt,
;    95          0191  1     sys$gb_dfmbfsdk,
;    96          0192  1     sys$gb_dfmbfrel,
;    97          0193  1     sys$gb_dfmbfidx,
;    98          0194  1     sys$gb_dfmbfhsh,
;    99          0195  1     sys$gb_dfmbc,
;   100          0196  1     sys$gb_dfnbc,
;   101          0197  1     sys$gw_rmsextend,
;   102          0198  1     sys$gb_rmsprolog,
;   103          0199  1     pio$gb_dfmbfsur,
;   104          0200  1     pio$gb_dfmbfsmt,
;   105          0201  1     pio$gb_dfmbfsdk,
;   106          0202  1     pio$gb_dfmbfrel,
;   107          0203  1     pio$gb_dfmbfidx,
;   108          0204  1     pio$gb_dfmbfhsh,
;   109          0205  1     pio$gb_dfmbc,
;   110          0206  1     pio$gb_dfnbc,
;   111          0207  1     pio$gw_rmsextend,
;   112          0208  1     pio$gb_rmsprolog;
;   113          0209  1
;   114          0210  1 EXTERNAL ROUTINE
;   115          0211  1     show$write_line : NOVALUE;
```

```
;  117          0212  1 !
;  118          0213  1 ! Declare some OWN storage, string descriptors which describe the various
;  119          0214  1 ! types of accounting.
;  120          0215  1 !
;  121          0216  1 OWN
;  122          0217  1     acc_type : VECTOR[10] INITIAL
;  123          0218  1                (%ASCID '      PROCESS         any process termination',
;  124          0219  1                 %ASCID '      IMAGE           image execution',
;  125          0220  1                 %ASCID '      INTERACTIVE     interactive job termination',
;  126          0221  1                 %ASCID '      LOGIN_FAILURE   login failures',
;  127          0222  1                 %ASCID '      SUBPROCESS      subprocess termination',
;  128          0223  1                 %ASCID '      DETACHED        detached job termination',
;  129          0224  1                 %ASCID '      BATCH           batch job termination',
;  130          0225  1                 %ASCID '      NETWORK         network job termination',
;  131          0226  1                 %ASCID '      PRINT           all print jobs',
;  132          0227  1                 %ASCID '      MESSAGE         user messages'
;  133          0228  1                );
;  134          0229  1
```

```
;  136          0230  1 GLOBAL ROUTINE show$accounting : NOVALUE =
;  137          0231  2 BEGIN
;  138          0232  2
;  139          0233  2 !---
;  140          0234  2 !
;  141          0235  2 ! This routine performs the SHOW ACCOUNTING function.  A copy of the
;  142          0236  2 ! accounting flags is brought into local memory, and based on the
;  143          0237  2 ! bit setting, the current accounting options are displayed.
;  144          0238  2 !
;  145          0239  2 !---
;  146          0240  2
;  147          0241  2 LOCAL
;  148          0242  2     acmflags : BITVECTOR[32];           ! Temp storage for accounting flags
;  149          0243  2
;  150          0244  2 !
;  151          0245  2 ! Get a copy of the accounting flags, so that the flag settings are consistent
;  152          0246  2 ! for the duration of this invocation of SHOW ACCOUNTING.
;  153          0247  2 !
;  154          0248  2 acmflags = .exe$gl_acmflags;
;  155          0249  2
;  156          0250  2 !
;  157          0251  2 ! Determine if any accounting is turned on.  If not, then simply print a
;  158          0252  2 ! line saying that, and return.
;  159          0253  2 !
;  160          0254  2 IF .acmflags EQL 0
;  161          0255  2 THEN show$write_line(%ASCID 'Accounting is not currently enabled.', 0)
;  162          0256  2
;  163          0257  2 !
;  164          0258  2 ! If something is set, then write a header line and then print whatever
;  165          0259  2 ! activities are shown in ACMFLAGS.
;  166          0260  2 !
;  167          0261  2 ELSE
;  168          0262  2     BEGIN
;  169          0263  3     show$write_line(%ASCID 'Accounting is currently enabled to log the following activities:!/', 0);
;  170          0264  3     INCR i FROM 0 TO 9 DO
;  171          0265  3     IF .acmflags[.i]
;  172          0266  3     THEN show$write_line(.acc_type[.i], 0);
;  173          0267  2     END;
;  174          0268  2
;  175          0269  2 RETURN;
;  176          0270  1 END;


                                                                .TITLE  SHOWMISC
                                                                .IDENT  \V04-000\

                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

20 20 53 53 45 43 4F 52 50 20 20 20 20 20 20 00000 P.AAB:    .ASCII  \      PROCESS         any process termina\  ;
65 63 6F 72 70 20 79 6E 61 20 20 20 20 20 20 0000F                                                               ;
                  61 6E 69 6D 72 65 74 20 73 73 0001E                                                            ;
                                        6E 6F 69 74 00028   .ASCII  \tion\                                       ;
                                              010E002C 0002C P.AAA: .LONG   17694764                             ;
                                              00000000' 00030       .ADDRESS P.AAB                               ;
20 20 20 20 45 47 41 4D 49 20 20 20 20 20 20 00034 P.AAD:    .ASCII  \      IMAGE           image execution\      ;
65 78 65 20 65 67 61 6D 69 20 20 20 20 20 20 00043                                                               ;
                              6E 6F 69 74 75 63 00052                                                            ;
```

```
                                                010E0024  00058 P.AAC:  .LONG   17694756
                                                00000000' 0005C         .ADDRESS P.AAD
49 54 43 41 52 45 54 4E 49 20 20 20 20 20 20 20 00060 P.AAF:  .ASCII  \      INTERACTIVE    interactive job ter\
69 74 63 61 72 65 74 6E 69 20 20 20 20 45 56    0006F
                72 65 74 20 62 6F 6A 20 65 76    0007E
                      6E 6F 69 74 61 6E 69 6D    00088         .ASCII  \mination\
                                                010E0030  00090 P.AAE:  .LONG   17694768
                                                00000000' 00094         .ADDRESS P.AAF
49 41 46 5F 4E 49 47 4F 4C 20 20 20 20 20 20 20 00098 P.AAH:  .ASCII  \      LOGIN_FAILURE  login failures\<0>
69 61 66 20 6E 69 67 6F 6C 20 20 45 52 55 4C    000A7
                      00 73 65 72 75 6C 00 00    000B6
                                                010E0023  000BC P.AAG:  .LONG   17694755
                                                00000000' 000C0         .ADDRESS P.AAH
53 45 43 4F 52 50 42 55 53 20 20 20 20 20 20 20 000C4 P.AAJ:  .ASCII  \      SUBPROCESS     subprocess terminat\
73 65 63 6F 72 70 62 75 73 20 20 20 20 20 53    000D3
                74 61 6E 69 6D 72 65 74 20 73    000E2
                                  00 6E 6F 69    000EC         .ASCII  \ion\<0>
                                                010E002B  000F0 P.AAI:  .LONG   17694763
                                                00000000' 000F4         .ADDRESS P.AAJ
20 44 45 48 43 41 54 45 44 20 20 20 20 20 20 20 000F8 P.AAL:  .ASCII  \      DETACHED       detached job termin\
20 64 65 68 63 61 74 65 64 20 20 20 20 20 20 20 00107
                6E 69 6D 72 65 74 20 62 6F 6A    00116
                      00 00 00 6E 6F 69 74 61    00120         .ASCII  \ation\<0><0><0>
                                                010E002D  00128 P.AAK:  .LONG   17694765
                                                00000000' 0012C         .ADDRESS P.AAL
20 20 20 20 48 43 54 41 42 20 20 20 20 20 20 20 00130 P.AAN:  .ASCII  \      BATCH          batch job terminati\
62 6F 6A 20 68 63 74 61 62 20 20 20 20 20 20 20 0013F
                69 74 61 6E 69 6D 72 65 74 20    0014E
                                  00 00 6E 6F    00158         .ASCII  \on\<0><0>
                                                010E002A  0015C P.AAM:  .LONG   17694762
                                                00000000' 00160         .ADDRESS P.AAN
20 20 4B 52 4F 57 54 45 4E 20 20 20 20 20 20 20 00164 P.AAP:  .ASCII  \      NETWORK        network job termina\
6A 20 6B 72 6F 77 74 65 6E 20 20 20 20 20 20 20 00173
                61 6E 69 6D 72 65 74 20 62 6F    00182
                                  6E 6F 69 74    0018C         .ASCII  \tion\
                                                010E002C  00190 P.AAO:  .LONG   17694764
                                                00000000' 00194         .ADDRESS P.AAP
20 20 20 20 54 4E 49 52 50 20 20 20 20 20 20 20 00198 P.AAR:  .ASCII  \      PRINT          all print jobs\<0>
74 6E 69 72 70 20 6C 6C 61 20 20 20 20 20 20 20 001A7
                      00 73 62 6F 6A 20 20 20    001B6
                                                010E0023  001BC P.AAQ:  .LONG   17694755
                                                00000000' 001C0         .ADDRESS P.AAR
20 20 45 47 41 53 53 45 4D 20 20 20 20 20 20 20 001C4 P.AAT:  .ASCII  \      MESSAGE        user messages\<0>
73 73 65 6D 20 72 65 73 75 20 20 20 20 20 20 20 001D3
                      00 73 65 67 61 20 20 20    001E2
                                  00 20 20 20    001E7         .ASCII  <0>
                                                010E0022  001E8 P.AAS:  .LONG   17694754
                                                00000000' 001EC         .ADDRESS P.AAT
6E 20 73 69 20 67 6E 69 74 6E 75 6F 63 63 41 20 001F0 P.AAV:  .ASCII  \Accounting is not currently enabled.\
6F 65 20 79 6C 74 6E 65 72 72 75 63 20 74 6F 20 001FF
                      2E 64 65 6C 62 61 6E 65    0020E
                                                010E0024  00214 P.AAU:  .LONG   17694756
                                                00000000' 00218         .ADDRESS P.AAV
63 20 73 69 20 67 6E 69 74 6E 75 6F 63 63 41 20 0021C P.AAX:  .ASCII  \Accounting is currently enabled to log t\
65 6C 62 61 6E 65 20 79 6C 74 6E 65 72 72 75 63 0022B
                74 20 67 6F 6C 20 6F 74 20 64    0023A
63 61 20 67 6E 69 77 6F 6C 6C 6F 66 20 65 68 20 00244         .ASCII  \he following activities:!/\<0><0>
```

```
        00  00  2F  21  3A  73  65  69  74  69  76  69  74  00253
                                           010E0042  00260 P.AAW:   .LONG    17694786                         ;
                                           00000000' 00264          .ADDRESS P.AAX                            ;

                                                                    .PSECT  $OWN$,NOEXE,2

00000000' 00000000' 00000000' 00000000' 00000000' 00000000' 00000 ACC_TYPE:
                                                                    .ADDRESS P.AAA, P.AAC, P.AAE, P.AAG, P.AAI, -   ;
                    00000000' 00C00000' 00000000' 00000000' 00018            P.AAK, P.AAM, P.AAO, P.AAQ, P.AAS      ;

                                                                    .EXTRN  EXE$GL_ACMFLAGS
                                                                    .EXTRN  SYS$GB_DFMBFSUR
                                                                    .EXTRN  SYS$GB_DFMBFSMT
                                                                    .EXTRN  SYS$GB_DFMBFSDK
                                                                    .EXTRN  SYS$GB_DFMBFREL
                                                                    .EXTRN  SYS$GB_DFMBFIDX
                                                                    .EXTRN  SYS$GB_DFMBFHSH
                                                                    .EXTRN  SYS$GB_DFMBC, SYS$GB_DFNBC
                                                                    .EXTRN  SYS$GW_RMSEXTEND
                                                                    .EXTRN  SYS$GB_RMSPROLOG
                                                                    .EXTRN  PIO$GB_DFMBFSUR
                                                                    .EXTRN  PIO$GB_DFMBFSMT
                                                                    .EXTRN  PIO$GB_DFMBFSDK
                                                                    .EXTRN  PIO$GB_DFMBFREL
                                                                    .EXTRN  PIO$GB_DFMBFIDX
                                                                    .EXTRN  PIO$GB_DFMBFHSH
                                                                    .EXTRN  PIO$GB_DFMBC, PIO$GB_DFNBC
                                                                    .EXTRN  PIO$GW_RMSEXTEND
                                                                    .EXTRN  PIO$GB_RMSPROLOG
                                                                    .EXTRN  SHOW$WRITE_LINE

                                                                    .PSECT  $CODE$,NOWRT,2

                                              001C 00000            .ENTRY  SHOW$ACCOUNTING, Save R2,R3,R4   ; 0230
                            54 00000000G  00   9E 00002            MOVAB   SHOW$WRITE_LINE, R4              ;
                            53 00000000G  00   D0 00009            MOVL    EXE$GL_ACMFLAGS, ACMFLAGS        ; 0248
                                          0A   12 00010            BNEQ    1$                               ; 0254
                                          7E   D4 00012            CLRL    -(SP)                            ; 0255
                                 0000'    CF   9F 00014            PUSHAB  P.AAU                            ;
                            64            02   FB 00018            CALLS      SHOW$WRITE_LINE               ;
                                               04 0001B            RET                                      ;
                                          7E   D4 0001C 1$:        CLRL    -(SP)                            ; 0263
                                 0000'    CF   9F 0001E            PUSHAB  P.AAW                            ;
                            64            02   FB 00022            CALLS   #2, SHOW$WRITE_LINE              ;
                                          52   D4 00025            CLRL    I                                ; 0264
                 0A         53            52   E1 00027 2$:        BBC     I, ACMFLAGS, 3$                  ; 0265
                                          7E   D4 0002B            CLRL    -(SP)                            ; 0266
                                 0000'CF42     DD 0002D            PUSHL   ACC_TYPE[I]                      ;
                            64            02   FB 00032            CALLS   #2, SHOW$WRITE_LINE              ;
                 EE         52            09   F3 00035 3$:        AOBLEQ  #9, I, 2$                        ; 0265
                                               04 00039            RET                                      ; 0270

; Routine Size:  58 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  178          0271  1 GLOBAL ROUTINE show$rms_default : NOVALUE =
;  179          0272  2 BEGIN
;  180          0273  2
;  181          0274  2 !---
;  182          0275  2 !
;  183          0276  2 ! This routine performs the SHOW RMS_DEFAULT funtion.  The current settings
;  184          0277  2 ! of the RMS default parameters are output.
;  185          0278  2 !
;  186          0279  2 !---
;  187          0280  2
;  188          0281  2 LOCAL
;  189          0282  2     status,
;  190          0283  2     desc : VECTOR[2],
;  191          0284  2     arglist : VECTOR[8],
;  192          0285  2     buffer : VECTOR[128,BYTE];
;  193          0286  2
;  194          0287  2 !
;  195          0288  2 ! Print the heading lines
;  196          0289  2 !
;  197          0290  2 show$write_line (%ASCID '          MULTI-  |                    MULTIBUFFER COUNTS                    | NETWORK', 0,
;  198          0291  2                  %ASCID '          BLOCK   | Indexed  Relative          Sequential              |  BLOCK', 0,
;  199          0292  2                  %ASCID '          COUNT   |                      Disk   Magtape  Unit Record |  COUNT', 0);
;  200          0293  2
;  201          0294  2 !
;  202          0295  2 ! Format and print the process RMS defaults
;  203          0296  2 !
;  204          0297  2 arglist[0] = .pio$gb_dfmbc;
;  205          0298  2 arglist[1] = .pio$gb_dfmbfidx;
;  206          0299  2 arglist[2] = .pio$gb_dfmbfrel;
;  207          0300  2 arglist[3] = .pio$gb_dfmbfsdk;
;  208          0301  2 arglist[4] = .pio$gb_dfmbfsmt;
;  209          0302  2 arglist[5] = .pio$gb_dfmbfsur;
;  210          0303  2 arglist[6] = .pio$gb_dfnbc;
;  211          0304  2
;  212          0305  2 show$write_line(%ASCID 'Process  !4SB     | !4SB      !4SB      !4SB     !4SB      !4SB       | !4SB',
;  213          0306  2                  arglist);
;  214          0307  2
;  215          0308  2 !
;  216          0309  2 ! Format and print the system RMS defaults
;  217          0310  2 !
;  218          0311  2 arglist[0] = .sys$gb_dfmbc;
;  219          0312  2 arglist[1] = .sys$gb_dfmbfidx;
;  220          0313  2 arglist[2] = .sys$gb_dfmbfrel;
;  221          0314  2 arglist[3] = .sys$gb_dfmbfsdk;
;  222          0315  2 arglist[4] = .sys$gb_dfmbfsmt;
;  223          0316  2 arglist[5] = .sys$gb_dfmbfsur;
;  224          0317  2 arglist[6] = .sys$gb_dfnbc;
;  225          0318  2
;  226          0319  2 show$write_line(%ASCID 'System   !4SB     | !4SB      !4SB      !4SB     !4SB      !4SB       | !4SB',
;  227          0320  2                  arglist);
;  228          0321  2
;  229          0322  2 !
;  230          0323  2 ! Now display the prologue and extend default quantities
;  231          0324  2 !
;  232          0325  2 arglist[0] = .pio$gb_rmsprolog;
;  233          0326  2 arglist[1] = .pio$gw_rmsextend;
;  234          0327  2 show$write_line(%ASCID '!/          Prolog    Extend Quantity !/Process  !4SB          !5UW', arglist);
```

```
;  235          0328  2
;  236          0329  2 arglist[0] = .sys$gb_rmsprolog;
;  237          0330  2 arglist[1] = .sys$gw_rmsextend;
;  238          0331  2 show$write_line(%ASCID 'System   '4SB          !5UW', arglist);
;  239          0332  2
;  240          0333  2 RETURN;
;  241          0334  1 END;                                    ! End of show$rms


                                                        .PSECT  $PLIT$,NOWRT,NOEXE,2

49 54 4C 55 4D 20 20 20 20 20 20 20 20 20 20 00268 P.AAZ:  .ASCII  \          MULTI-  :                MULTI\
20 20 20 20 20 20 20 20 20 20 20 7C 20 20 2D 00277
                  49 54 4C 55 4D 20 20 20 20 20 00286
20 20 53 54 4E 55 4F 43 20 52 45 46 46 55 42 00290        .ASCII  \BUFFER COUNTS                : NETWORK\<0>
20 7C 20 20 20 20 20 20 20 20 20 20 20 20 20 0029F
                        00 4B 52 4F 57 54 45 4E 002AE
                                    00 00 002B6        .ASCII  <0><0>
                              010E004D 002B8 P.AAY:  .LONG   17694797
                              00000000' 002BC        .ADDRESS P.AAZ
4B 43 4F 4C 42 20 20 20 20 20 20 20 20 20 20 002C0 P.ABB:  .ASCII  \          BLOCK   : Indexed  Relative   \
52 20 20 64 65 78 65 64 6E 49 20 7C 20 20 20 002CF
                  20 20 20 65 76 69 74 61 6C 65 002DE
6E 65 75 71 65 53 20 20 20 20 20 20 20 20 20 002E8        .ASCII  \          Sequential          : BLOCK\
20 7C 20 20 20 20 20 20 20 20 20 6C 61 69 74 002F7
                              4B 43 4F 4C 42 20 00306
                              010E004C 0030C P.ABA:  .LONG   17694796
                              00000000' 00310        .ADDRESS P.ABB
54 4E 55 4F 43 20 20 20 20 20 20 20 20 20 20 00314 P.ABD:  .ASCII  \          COUNT   :                    \
20 20 20 20 20 20 20 20 20 20 20 7C 20 20 20 00323
                  20 20 20 20 20 20 20 20 20 00332
20 65 70 61 74 67 61 4D 20 20 20 6B 73 69 44 0033C        .ASCII  \Disk   Magtape  Unit Record :  COUNT\
20 7C 20 64 72 6F 63 65 52 20 74 69 6E 55 20 0034B
                              54 4E 55 4F 43 20 0035A
                              010E004C 00360 P.ABC:  .LONG   17694796
                              00000000' 00364        .ADDRESS P.ABD
20 20 42 53 34 21 20 20 73 73 65 63 6F 72 50 00368 P.ABF:  .ASCII  \Process  !4SB     : !4SB      !4SB     !\
20 20 20 20 20 20 42 53 34 21 20 7C 20 20 20 00377
                  21 20 20 20 20 20 42 53 34 21 00386
20 20 20 20 42 53 34 21 20 20 20 20 42 53 34 00390        .ASCII  \4SB    !4SB      !4SB       : !4SB\<0>
20 7C 20 20 20 20 20 20 20 42 53 34 21 20 20 0039F
                              00 42 53 34 21 003AE
                                    00 003B3        .ASCII  <0>
                              010E004A 003B4 P.ABE:  .LONG   17694794
                              00000000' 003B8        .ADDRESS P.ABF
20 20 42 53 34 21 20 20 20 6D 65 74 73 79 53 003BC P.ABH:  .ASCII  \System   !4SB     : !4SB      !4SB     !\
20 20 20 20 20 20 42 53 34 21 20 7C 20 20 20 003CB
                  21 20 20 20 20 20 42 53 34 21 003DA
20 20 20 20 42 53 34 21 20 20 20 20 42 53 34 003E4        .ASCII  \4SB    !4SB      !4SB       : !4SB\<0>
20 7C 20 20 20 20 20 20 20 42 53 34 21 20 20 003F3
                              00 42 53 34 21 00402
                                    00 00407        .ASCII  <0>
                              010E004A 00408 P.ABG:  .LONG   17694794
                              00000000' 0040C        .ADDRESS P.ABH
6F 72 50 20 20 20 20 20 20 20 20 20 20 2F 21 00410 P.ABJ:  .ASCII  \!/          Prolog    Extend Quantity !/\
51 20 64 6E 65 74 78 45 20 20 20 20 67 6F 6C 0041F
```

```
                           2F 21 20 79 74 69 74 6E 61 75 0042E
20 20 42 53 34 21 20 20 73 73 65 63 6F 72 50 00438                   .ASCII  \Process  !4SB          !5UW\<0>        :
      00 57 55 35 21 20 20 20 20 20 20 20 20 00447                                                                   :
                                        010E0043 00454 P.ABI:        .LONG   17694787                                :
                                        00000000' 00458              .ADDRESS P.ABJ                                  :
20 20 42 53 34 21 20 20 20 6D 65 74 73 79 53 0045C P.ABL:            .ASCII  \System   !4SB          !5UW\<0>        :
      00 57 55 35 21 20 20 20 20 20 20 20 20 0046B                                                                   :
                                        010E001B 00478 P.ABK:        .LONG   17694747                                :
                                        00000000' 0047C              .ADDRESS P.ABL                                  :


                                                                     .PSECT  $CODE$,NOWRT,2

                                          0004 00000                 .ENTRY  SHOW$RMS_DEFAULT, Save R2              : 0271
                    52 00000000G    00   9E 00002                    MOVAB   SHOW$WRITE_LINE, R2                    :
                    5E      FF58    CE   9E 00009                    MOVAB   -168(SP), SP                           :
                                    7E   D4 0000E                    CLRL    -(SP)                                  : 0290
                           0000'    CF   9F 00010                    PUSHAB  P.ABC                                  : 0291
                                    7E   D4 00014                    CLRL    -(SP)                                  : 0290
                           0000'    CF   9F 00016                    PUSHAB  P.ABA                                  :
                                    7E   D4 0001A                    CLRL    -(SP)                                  :
                           0000'    CF   9F 0001C                    PUSHAB  P.AAY                                  :
                    62              06   FB 00020                    CALLS   #6, SHOW$WRITE_LINE                    :
              D8    AD 00000000G    00   D0 00023                    MOVL    PIO$GB_DFMBC, ARGLIST                  : 0297
              DC    AD 00000000G    00   D0 0002B                    MOVL    PIO$GB_DFMBFIDX, ARGLIST+4             : 0298
              E0    AD 00000000G    00   D0 00033                    MOVL    PIO$GB_DFMBFREL, ARGLIST+8             : 0299
              E4    AD 00000000G    00   D0 0003B                    MOVL    PIO$GB_DFMBFSDK, ARGLIST+12            : 0300
              E8    AD 00000000G    00   D0 00043                    MOVL    PIO$GB_DFMBFSMT, ARGLIST+16            : 0301
              EC    AD 00000000G    00   D0 0004B                    MOVL    PIO$GB_DFMBFSUR, ARGLIST+20            : 0302
              F0    AD 00000000G    00   D0 00053                    MOVL    PIO$GB_DFNBC, ARGLIST+24               : 0303
                              D8    AD   9F 0005B                    PUSHAB  ARGLIST                                : 0305
                           0000'    CF   9F 0005E                    PUSHAB  P.ABE                                  :
                    62              02   FB 00062                    CALLS   #2, SHOW$WRITE_LINE                    :
              D8    AD 00000000G    00   D0 00065                    MOVL    SYS$GB_DFMBC, ARGLIST                  : 0311
              DC    AD 00000000G    00   D0 0006D                    MOVL    SYS$GB_DFMBFIDX, ARGLIST+4             : 0312
              E0    AD 00000000G    00   D0 00075                    MOVL    SYS$GB_DFMBFREL, ARGLIST+8             : 0313
              E4    AD 00000000G    00   D0 0007D                    MOVL    SYS$GB_DFMBFSDK, ARGLIST+12            : 0314
              E8    AD 00000000G    00   D0 00085                    MOVL    SYS$GB_DFMBFSMT, ARGLIST+16            : 0315
              EC    AD 00000000G    00   D0 0008D                    MOVL    SYS$GB_DFMBFSUR, ARGLIST+20            : 0316
              F0    AD 00000000G    00   D0 00095                    MOVL    SYS$GB_DFNBC, ARGLIST+24               : 0317
                              D8    AD   9F 0009D                    PUSHAB  ARGLIST                                : 0319
                           0000'    CF   9F 000A0                    PUSHAB  P.ABG                                  :
                    62              02   FB 000A4                    CALLS   #2, SHOW$WRITE_LINE                    :
              D8    AD 00000000G    00   D0 000A7                    MOVL    PIO$GB_RMSPROLOG, ARGLIST              : 0325
              DC    AD 00000000G    00   D0 000AF                    MOVL    PIO$GW_RMSEXTEND, ARGLIST+4            : 0326
                              D8    AD   9F 000B7                    PUSHAB  ARGLIST                                : 0327
                           0000'    CF   9F 000BA                    PUSHAB  P.ABI                                  :
                    62              02   FB 000BE                    CALLS   #2, SHOW$WRITE_LINE                    :
              D8    AD 00000000G    00   D0 000C1                    MOVL    SYS$GB_RMSPROLOG, ARGLIST              : 0329
              DC    AD 00000000G    00   D0 000C9                    MOVL    SYS$GW_RMSEXTEND, ARGLIST+4            : 0330
                              D8    AD   9F 000D1                    PUSHAB  ARGLIST                                : 0331
                           0000'    CF   9F 000D4                    PUSHAB  P.ABK                                  :
                    62              02   FB 000D8                    CALLS   #2, SHOW$WRITE_LINE                    :
                                         04 000DB                    RET                                            : 0334

; Routine Size:  220 bytes,    Routine Base:  $CODE$ + 003A
```

```
;  243          0335  1 GLOBAL ROUTINE show$working_set : NOVALUE =
;  244          0336  2 BEGIN
;  245          0337  2
;  246          0338  2 !---
;  247          0339  2 !
;  248          0340  2 ! The current working set parameters are output.
;  249          0341  2 !
;  250          0342  2 !---
;  251          0343  2
;  252          0344  2 LOCAL
;  253          0345  2     status,                                     ! Status return
;  254          0346  2     default : VOLATILE,                         ! WS default
;  255          0347  2     quota : VOLATILE,                           ! WS quota
;  256          0348  2     extent : VOLATILE,                          ! WS extent
;  257          0349  2     max_quota : VOLATILE,                       ! Maximum authorized quota
;  258          0350  2     max_extent : VOLATILE,                      ! Maximum authorized extent
;  259          0351  2     pcb_flags : BITVECTOR[32],                  ! PCB status flags
;  260          0352  2     item_list : $BBLOCK[7*4*4],                 ! $GETJPI item list
;  261          0353  2     iosb : VECTOR[2],                           ! $GETJPI final status block
;  262          0354  2     arglist : VECTOR[3],                        ! Argument list for $FAOL
;  263          0355  2     desc : VECTOR[2],                           ! Descriptor for $FAOL buffer
;  264          0356  2     buffer : VECTOR[128,BYTE];                  ! $FAOL buffer
;  265          0357  2
;  266          0358  2 LITERAL
;  267          0359  2     disallow_bit = $BITPOSITION (pcb$v_disaws);
;  268          0360  2
;  269        P 0361  2 setup_list( item_list,
;  270        P 0362  2                 jpi$_sts,       4, pcb_flags,       0,
;  271        P 0363  2                 jpi$_dfwscnt,   4, default,         0,
;  272        P 0364  2                 jpi$_wsquota,   4, quota,           0,
;  273        P 0365  2                 jpi$_wsextent,  4, extent,          0,
;  274        P 0366  2                 jpi$_wsauth,    4, max_quota,       0,
;  275        P 0367  2                 jpi$_wsauthext, 4, max_extent,      0,
;  276          0368  2                 0,0,0,0);
;  277          0369  2
;  278        P 0370  2 status = $GETJPIW(ITMLST = item_list,
;  279          0371  2                   IOSB = iosb);
;  280          0372  2 IF .status THEN status = (.iosb[0])<0,16>;
;  281          0373  2 IF NOT .status
;  282          0374  2 THEN
;  283          0375  2     SIGNAL_STOP(.status);
;  284          0376  2
;  285          0377  2 arglist[0] = .default;
;  286          0378  2 arglist[1] = .quota;
;  287          0379  2 arglist[2] = .extent;
;  288          0380  2
;  289          0381  2 show$write_line(%ASCID '  Working Set      /Limit= !UL   /Quota= !UL    /Extent= !UL',
;  290          0382  2                 arglist);
;  291          0383  2
;  292          0384  2 arglist[0] = .max_quota;
;  293          0385  2 arglist[1] = .max_extent;
;  294          0386  2
;  295          0387  2 IF .pcb_flags[disallow_bit]
;  296          0388  2 THEN show$write_line(%ASCID '  Adjustment disabled   Authorized Quota= !UL  Authorized Extent= !UL',
;  297          0389  2                           arglist)
;  298          0390  2 ELSE show$write_line(%ASCID '  Adjustment enabled    Authorized Quota= !UL  Authorized Extent= !UL',
;  299          0391  2                           arglist);
```

```
;   300          0392  2
;   301          0393  2 RETURN;
;   302          0394  1 END;                                  ! End of show$work



                                                              .PSECT  $PLIT$,NOWRT,NOEXE,2

20 20 74 65 53 20 67 6E 69 6B 72 6F 57 20 20 00480 P.ABN:   .ASCII  \  Working Set      /Limit= !UL   /Quota=\   ;
4C 55 21 20 3D 74 69 6D 69 4C 2F 20 20 20 20 0048F                                                              ;
               3D 61 74 6F 75 51 2F 20 20 20 0049E                                                              ;
74 6E 65 74 78 45 2F 20 20 20 20 4C 55 21 20 004A8          .ASCII  \ !UL    /Extent= !UL\                       ;
                                  4C 55 21 20 3D 004B7                                                          ;
                              010E003C 004BC P.ABM:   .LONG   17694780                                           ;
                              00000000' 004C0         .ADDRESS P.ABN                                             ;
69 64 20 74 6E 65 6D 74 73 75 6A 64 41 20 20 004C4 P.ABP:   .ASCII  \  Adjustment disabled   Authorized Quota\  ;
72 6F 68 74 75 41 20 20 20 64 65 6C 62 61 73 004D3                                                              ;
               61 74 6F 75 51 20 64 65 7A 69 004E2                                                              ;
7A 69 72 6F 68 74 75 41 20 20 4C 55 21 20 3D 004EC          .ASCII  \= !UL  Authorized Extent= !UL\<0><0>        ;
00 4C 55 21 20 3D 74 6E 65 74 78 45 20 64 65 004FB                                                              ;
                                             00 0050A                                                          ;
                                             00 0050B         .ASCII  <0>                                        ;
                              010E0045 0050C P.ABO:   .LONG   17694789                                           ;
                              00000000' 00510         .ADDRESS P.ABP                                             ;
6E 65 20 74 6E 65 6D 74 73 75 6A 64 41 20 20 00514 P.ABR:   .ASCII  \  Adjustment enabled    Authorized Quota\  ;
72 6F 68 74 75 41 20 20 20 20 64 65 6C 62 61 00523                                                              ;
               61 74 6F 75 51 20 64 65 7A 69 00532                                                              ;
7A 69 72 6F 68 74 75 41 20 20 4C 55 21 20 3D 0053C          .ASCII  \= !UL  Authorized Extent= !UL\<0><0>        ;
00 4C 55 21 20 3D 74 6E 65 74 78 45 20 64 65 0054B                                                              ;
                                             00 0055A                                                          ;
                                             00 0055B         .ASCII  <0>                                        ;
                              010E0045 0055C P.ABQ:   .LONG   17694789                                           ;
                              00000000' 00560         .ADDRESS P.ABR                                             ;

                                                              .EXTRN  SYS$GETJPIW

                                                              .PSECT  $CODE$,NOWRT,2

                                     0004 00000         .ENTRY  SHOW$WORKING_SET, Save R2                      ; 0335
                52 00000000G   00  9E 00002         MOVAB   SHOW$WRITE_LINE, R2                                ;
                5E      FEDC   CE  9E 00009         MOVAB   -292(SP), SP                                       ;
                50      FF7C   CD  9E 0000E         MOVAB   ITEM_LIST, PTR                                     ; 0368
                80 03050004    8F  D0 00013         MOVL    #50659332, (PTR)+                                  ;
                80             6E  9E 0001A         MOVAB   PCB_FLAGS, (PTR)+                                  ;
                               80  D4 0001D         CLRL    (PTR)+                                             ;
                80 04030004    8F  D0 0001F         MOVL    #67305476, (PTR)+                                  ;
                80        FC   AD  9E 00026         MOVAB   DEFAULT, (PTR)+                                    ;
                               80  D4 0002A         CLRL    (PTR)+                                             ;
                80 04020004    8F  D0 0002C         MOVL    #67239940, (PTR)+                                  ;
                80        F8   AD  9E 00033         MOVAB   QUOTA, (PTR)+                                      ;
                               80  D4 00037         CLRL    (PTR)+                                             ;
                80 04160004    8F  D0 00039         MOVL    #68550660, (PTR)+                                  ;
                80        F4   AD  9E 00040         MOVAB   EXTENT, (PTR)+                                     ;
                               80  D4 00044         CLRL    (PTR)+                                             ;
                80 04010004    8F  D0 00046         MOVL    #67174404, (PTR)+                                  ;
                80        F0   AD  9E 0004D         MOVAB   MAX_QUOTA, (PTR)+                                  ;
                               80  D4 00051         CLRL    (PTR)+                                             ;
```

```
                80 04170004    8F   D0 00053          MOVL    #68616196, (PTR)+                       ;
                80         EC  AD   9E 00058          MOVAB   MAX_EXTENT, (PTR)+                      ;
                                80   D4 0005E          CLRL    (PTR)+                                  ;
                                80   7C 00060          CLRQ    (PTR)+                                  ;
                                80   D4 00062          CLRL    (PTR)+                                  ;
                                7E   7C 00064          CLRQ    -(SP)                                   ; 0371
                          FF74  CD   9F 00066          PUSHAB  IOSB                                    ;
                          FF7C  CD   9F 0006A          PUSHAB  ITEM_LIST                               ;
                                7E   7C 0006F          CLRQ    -(SP)                                   ;
                                7E   D4 00071          CLRL    -(SP)                                   ;
00000000G       00              07   FB 00072          CALLS   #7, SYS$GETJPIW                         ;
                08              50   E9 00079          BLBC    STATUS, 1$                              ; 0372
                50         FF74 CD   D0 0007C          MOVL    IOSB, STATUS                            ;
                09              50   E8 00081          BLBS    STATUS, 2$                              ; 0373
                                50   DD 00084 1$:       PUSHL   STATUS                                  ; 0375
00000000G       00              01   FB 00086          CALLS   #1, LIB$STOP                            ;
    008C        CE         FC   AD   D0 0008D 2$:       MOVL    DEFAULT, ARGLIST                        ; 0377
    0090        CE         F8   AD   D0 00093          MOVL    QUOTA, ARGLIST+4                        ; 0378
    FF70        CD         F4   AD   D0 00099          MOV     EXTENT, ARGLIST+8                       ; 0379
                          008C  CE   9F 0009F          PUSHAB  ARGLIST                                 ; 0381
                          0000' CF   9F 000A3          PUSHAB  P.ABM                                   ;
                62              02   FB 000A7          CALLS   #2, SHOW$WRITE_LINE                     ;
    008C        CE         F0   AD   D0 000AA          MOVL    MAX_QUOTA, ARGLIST                      ; 0384
    0090        CE         EC   AD   D0 000B0          MOVL    MAX_EXTENT, ARGLIST+4                   ; 0385
                0A         03   AE   E9 000B6          BLBC    PCB_FLAGS+3, 3$                         ; 0387
                          008C  CE   9F 000BA          PUSHAB  ARGLIST                                 ; 0388
                          0000' CF   9F 000BE          PUSHAB  P.ABO                                   ;
                                08   11 000C2          BRB     4$                                      ;
                          008C  CE   9F 000C4 3$:      PUSHAB  ARGLIST                                 ; 0390
                          0000' CF   9F 000C8          PUSHAB  P.ABO                                   ;
                62              02   FB 000CC 4$:      CALLS   #2, SHOW$WRITE_LINE                     ;
                                     04 000CF          RET                                             ; 0394

; Routine Size:  208 bytes,    Routine Base:  $CODE$ + 0116
```

```
;  304          0395  1 END
;  305          0396  0 ELUDOM


                                                        .EXTRN  LIB$STOP

;                               PSECT SUMMARY

;       Name                    Bytes                   Attributes

;  $PLIT$                       1380  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $OWN$                          40  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                        486  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)


;                           Library Statistics

;                                    -------- Symbols --------      Pages       Processing
;       File                         Total    Loaded   Percent      Mapped      Time

;  _$255$DUA28:[SYSLIB]LIB.L32;1     18619        22         0      1000          00:01.9


;                                   COMMAND QUALIFIERS

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

; Size:          486 code + 1420 data bytes
; Run Time:          00:14.1
; Elapsed Time:      00:41.5
; Lines/CPU Min:      1689
; Lexemes/CPU-Min:  18772
; Memory Used:  130 pages
; Compilation Complete
```

SHOWTERM
LIS

SHOWMISC
LIS

SHOWPROC
LIS

SHOWSYS
LIS

SHOWMAIN
LIS

SHOWMSCP
LIS

SHOWQUE
LIS

SHOWMSG
LIS