L 12

```
   SSSSSSSSSSSS  YYY         YYY     SSSSSSSSSSSS
   SSSSSSSSSSSS  YYY         YYY     SSSSSSSSSSSS
   SSSSSSSSSSSS  YYY         YYY     SSSSSSSSSSSS
SSS              YYY         YYY  SSS
SSS              YYY         YYY  SSS
SSS              YYY         YYY  SSS
SSS                 YYY   YYY     SSS
SSS                 YYY   YYY     SSS
SSS                 YYY   YYY     SSS
   SSSSSSSSS           YYY           SSSSSSSSS
   SSSSSSSSS           YYY           SSSSSSSSS
   SSSSSSSSS           YYY           SSSSSSSSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
            SSS        YYY                    SSS
SSSSSSSSSSSS           YYY        SSSSSSSSSSSS
SSSSSSSSSSSS           YYY        SSSSSSSSSSSS
SSSSSSSSSSSS           YYY        SSSSSSSSSSSS
```

```
_$

Ps
--

YZ

Z$

Z$

Z$

Z$

Z$

Z$

Z$

Z$

Z$

Z$

Z$

Z$
```

```
**FILE**ID**MUTEX




          MM      MM  UU      UU  TTTTTTTTTT  EEEEEEEEEE  XX      XX
          MM      MM  UU      UU  TTTTTTTTTT  EEEEEEEEEE  XX      XX
          MMMM  MMMM  UU      UU      TT      EE          XX      XX
          MMMM  MMMM  UU      UU      TT      EE          XX      XX
          MM  MM  MM  UU      UU      TT      EE            XX  XX
          MM  MM  MM  UU      UU      TT      EE            XX  XX
          MM      MM  UU      UU      TT      EEEEEEEE        XX
          MM      MM  UU      UU      TT      EEEEEEEE        XX
          MM      MM  UU      UU      TT      EE            XX  XX
          MM      MM  UU      UU      TT      EE            XX  XX
          MM      MM  UU      UU      TT      EE          XX      XX       ....
          MM      MM  UU      UU      TT      EE          XX      XX       ....
          MM      MM  UUUUUUUUUU      TT      EEEEEEEEEE  XX      XX       ....
          MM      MM  UUUUUUUUUU      TT      EEEEEEEEEE  XX      XX       ....

          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
0000            .TITLE  MUTEX - MUTEX WAIT ROUTINES
0000            .IDENT  'V04-000'
0000
0000    ;
0000    ;****************************************************************************
0000    ;*                                                                          *
0000    ;*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
0000    ;*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
0000    ;*  ALL RIGHTS RESERVED.                                                    *
0000    ;*                                                                          *
0000    ;*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
0000    ;*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
0000    ;*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
0000    ;*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
0000    ;*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
0000    ;*  TRANSFERRED.                                                            *
0000    ;*                                                                          *
0000    ;*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
0000    ;*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
0000    ;*  CORPORATION.                                                            *
0000    ;*                                                                          *
0000    ;*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
0000    ;*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
0000    ;*                                                                          *
0000    ;*                                                                          *
0000    ;****************************************************************************
0000
0000    ;++
0000    ; FACILITY: EXECUTIVE, SCHEDULER
0000    ;
0000    ; ABSTRACT:
0000    ;       THIS MODULE CONTAINS THE ROUTINES WHICH IMPLEMENT THE MUTEX
0000    ;       LOCK AND UNLOCK SERVICES FOR INTERNAL EXECUTIVE USE.
0000    ;
0000    ;
0000    ; ENVIRONMENT:
0000    ;       MODE = KERNEL
0000    ;
0000    ;--
0000    ;
0000    ;       .PAGE
0000            .SBTTL  HISTORY                 ; DETAILED
0000    ;
0000    ; AUTHOR:       R. HUSTVEDT  CREATION DATE: 25-AUG-76
0000    ;
0000    ; MODIFIED BY:
0000    ;
0000    ;       V03-003 SSA0022         Stan Amway              2-Apr-1984
0000    ;               Backed out SSA0005. It was temporary.
0000    ;
0000    ;       V03-002 SSA0005         Stan Amway              10-Jan-1984
0000    ;               Added code to maintain PMS MWAIT transition counters.
0000    ;               The counters (in MDAT) and supporting code will be removed
0000    ;               before V4 release.
0000    ;
0000    ;       V03-001 ROW0168         Ralph O. Weber          3-MAR-1983
```

```
                    0000     58 ;            Change W^ references to G^.
                    0000     59 ;
```

```
MU
Sy

BU
DY
EX
EX
IO
IP
IP
IP
LK
MT
MT
NO
PC
PC
PC
PC
PC
PC
PC
PC
PR
PR
PR
PS
PS
RD
SC
SC
SC
SC
SC
SCI
SC
SC
SC
SC
SC
SC
SC
SC
SC
SC
SS
UN
WA
WA
WQ


PS
--
.
$A
AE
```

```
    0000                .SBTTL  DECLARATIONS
    0000
    0000        ;
    0000        ; INCLUDE FILES:
    0000        ;
    0000
    0000                $DYNDEF                                 ; STRUCTURE TYPE DEFINITIONS
    0000                $IPLDEF                                 ; IPL DEFINITIONS
    0000                $MTXDEF                                 ; MUTEX DEFINITIONS
    0000                $PCBDEF                                 ; PCB DEFINITIONS
    0000                $PRDEF                                  ; PROCESSOR REGISTER DEFINITIONS
    0000                $PRIDEF                                 ; PRIORITY INCR CLASS DEFS
    0000                $PSLDEF                                 ; PSL DEFINITIONS
    0000                $SSDEF                                  ; SYSTEM STATUS CODES
    0000                $STATEDEF                               ; SCHEDULER STATE DEFS
    0000                $WQHDEF                                 ; WAIT QUEUE HEADER DEFS
    0000        ;
    0000        ; EQUATED SYMBOLS
    0000        ;
    0000
00000000                .PSECT  AEXENONPAGED,BYTE               ; NONPAGED EXEC
```

```
                              0000              .SBTTL  SCH$RWAIT - RESOURCE WAIT
                              0000
                              0000       ;++
                              0000       ; FUNCTIONAL DESCRIPTION:
                              0C00       ;       SCH$RWAIT SUSPENDS THE EXECUTION OF A PROCESS UNTIL REQUIRED
                              0C00       ;       RESOURCES ARE AVAILABLE.
                              0000       ;
                              0000       ; CALLING SEQUENCE:
                              0000       ;       SETIPL/DSBINT #IPL$_SYNCH
                              0000       ;       PUSHL   <PSL>
                              0030       ;       BSB/JSB SCH$RWAIT
                              0000       ;
                              0000       ; INPUT PARAMETERS:
                              0000       ;       R0 - RESOURCE NUMBER FOR WHICH TO WAIT
                              0000       ;       R4 - PCB ADDRESS
                              0000       ;       00(SP) - PC AT WHICH TO RESUME
                              0000       ;       04(SP) - PSL WITH WHICH TO RESUME
                              0003       ;
                              0000       ; IMPLICIT INPUTS:
                              0000       ;       SCH$GQ_MWAIT - MUTEX WAIT QUEUE HEADER
                              0000       ;       PCB OF CURRENT PROCESS
                              0000       ;
                              0000       ; OUTPUTS:
                              0000       ;       R0-R3 PRESERVED
                              0000       ;
                              0000       ; IMPLICIT OUTPUTS:
                              0000       ;       *** TBS ***
                              0000       ;
                              0000       ; SIDE EFFECTS:
                              0000       ;       *** TBS ***
                              0000       ;
                              0000       ;--
                              0000
                              0000       SCH$RWAIT::                                     ;;; RESOURCE WAIT ENTRY POINT
00 00000000'GF     50   E6    0000              BBSSI   R0,G^SCH$GL_RESMASK,10$ ;;; SET WAITING FLAG
                   7E   11    0008       10$:   BRB     WAITR                   ;;; AND ENTER WAIT STATE
                              000A
                   000A                .SBTTL  SCH$LOCKWNOWAIT - LOCK MUTEX FOR WRITE WITHOUT WAITING
                   000A
                   000A       ;++
                   000A       ; FUNCTIONAL DESCRIPTION:
                   000A       ;        SCH$LOCKWNOWAIT LOCKS THE SPECIFIED MUTEX FOR EXCLUSIVE WRITE ACCESS
                   000A       ;        TO THE PROTECTED STRUCTURE.  IF ANOTHER PROCESS HAS ALREADY CLAIMED
                   000A       ;        THE MUTEX, THEN THIS ROUTINE RETURNS A FAILURE INDICATION.
                   000A       ;
                   000A       ;
                   000A       ;
                   000A       ; CALLING SEQUENCE:
                   000A       ;        BSB/JSB SCH$LOCKWNOWAIT
                   000A       ;
                   000A       ;
                   000A       ; INPUT PARAMETERS:
                   000A       ;        R0 - ADDRESS OF MUTEX
                   000A       ;        R4 - PCB ADDRESS OF CURRENT PROCESS
                   000A       ;
                   000A       ; IMPLICIT INPUTS:
                   000A       ;        SCH$GQ_MWAIT - MUTEX WAIT QUEUE HEADER
                   000A       ;        PCB OF CURRENT PROCESS
                   000A       ;        MUTEX LOCATED BY R0
                   000A       ;
                   000A       ; OUTPUTS:
                   000A       ;        R0 LOW BIT SET IF LOCKED SUCCESSFULLY
                   000A       ;           LOW BIT CLEAR IF MUTEX IN USE
                   000A       ;        R1-R3 PRESERVED
                   000A       ;        IPL = ASTDEL
                   000A       ;
                   000A       ; IMPLICIT OUTPUTS:
                   000A       ;        *** TBS ***
                   000A       ;
                   000A       ; SIDE EFFECTS:
                   000A       ;        *** TBS ***
                   000A       ;
                   000A       ;--
                   000A       SCH$LOCKWNOWAIT::
                   000A                SETIPL  #IPL$_SYNCH             ;;; RAISE TO SYNCH IPL
0B 60     10   E6  000D                BBSSI   #MTX$V_WRT,(R0),20$     ;;; SET WRITE PENDING
          60   B6  0011                INCW    MTX$W_OWNCNT(R0)        ;;; RAISE OWNER COUNT
          05   12  0013                BNEQ    10$                     ;;; RETURN FAILURE IF BUSY
   50     01   3C  0015                MOVZWL  #SS$_NORMAL,R0          ;;; INDICATE SUCCESSFUL COMPLETION
          32   11  0018                BRB     LKEX                    ;;; AND MERGE WITH COMMON EXIT CODE
          60   B7  001A       10$:     DECW    MTX$W_OWNCNT(R0)        ;;; CORRECT COUNT
          50   D4  001C       20$:     CLRL    R0                      ;;; SET FAILURE RETURN INDICATION
                   001E                SETIPL  #IPL$_ASTDEL            ;;; LOWER TO ASTDEL
               05  0021                RSB                            ; AND RETURN
```

```
                          0022          .SBTTL  SCH$IOLOCKW - LOCK I/O DATA BASE MUTEX FOR WRITE
                          0022  ;++
                          0022  ; FUNCTIONAL DESCRIPTION:
                          0022  ;       SCH$IOLOCKW RETURNS TO THE CALLER WHEN THE I/O DATABASE MUTEX
                          0022  ;       HAS BEEN LOCKED FOR WRITE ASSURING EXCLUSIVE ACCESS.
                          0022  ;
                          0022  ;
                          0022  ;
                          0022  ; CALLING SEQUENCE:
                          0022  ;       BSB/JSB SCH$IOLOCKW
                          0022  ;
                          0022  ;
                          0022  ; INPUT PARAMETERS:
                          0022  ;       R4 - PCB ADDRESS OF CURRENT PROCESS
                          0022  ;
                          0022  ; IMPLICIT INPUTS:
                          0022  ;       SCH$GQ_MWAIT - MUTEX WAIT QUEUE HEADER
                          0022  ;       PCB OF CURRENT PROCESS
                          0022  ;       I/O DATABASE MUTEX
                          0022  ;
                          0022  ; OUTPUTS:
                          0022  ;       R0 = ADDRESS OF I/O DATABASE MUTEX
                          0022  ;       R1-R3 PRESERVED
                          0022  ;       IPL = ASTDEL
                          0022  ;
                          0022  ; IMPLICIT OUTPUTS:
                          0022  ;       *** TBS ***
                          0022  ;
                          0022  ; SIDF EFFECTS:
                          0022  ;       *** TBS ***
                          0022  ;
                          0022  ;--
                          0022
                          0022 SCH$IOLOCKW::                            ; LOCK I/O DATA BASE FOR WRITE ACCESS
  50   00000000'EF   9E   0022          MOVAB   IOC$GL_MUTEX,R0         ; GET ADDRESS OF I/O DATABASE MUTEX
```

```
                  0029          .SBTTL  SCH$LOCKW - LOCK MUTEX FOR WRITE
                  0029  ;++
                  0029  ; FUNCTIONAL DESCRIPTION:
                  0029  ;       SCH$LOCKW RETURNS TO THE CALLER WHEN THE SPECIFIED MUTEX
                  0029  ;       HAS BEEN LOCKED FOR WRITE ASSURING EXCLUSIVE ACCESS TO THE
                  0029  ;       PROTECTED STRUCTURE.
                  0029  ;
                  0029  ;
                  0029  ;
                  0029  ; CALLING SEQUENCE:
                  0029  ;       BSB/JSB SCH$LOCKW
                  0029  ;
                  0029  ;
                  0029  ; INPUT PARAMETERS:
                  0029  ;       R0 - ADDRESS OF MUTEX
                  0029  ;       R4 - PCB ADDRESS OF CURRENT PROCESS
                  0029  ;
                  0029  ; IMPLICIT INPUTS:
                  0029  ;       SCH$GQ_MWAIT - MUTEX WAIT QUEUE HEADER
                  0029  ;       PCB OF CURRENT PROCESS
                  0029  ;       MUTEX LOCATED BY R0
                  0029  ;
                  0029  ; OUTPUTS:
                  0029  ;       R0-R3 PRESERVED
                  0029  ;       IPL = ASTDEL
                  0029  ;
                  0029  ; IMPLICIT OUTPUTS:
                  0029  ;       *** TBS ***
                  0029  ;
                  0029  ; SIDE EFFECTS:
                  0029  ;       *** TBS ***
                  0029  ;
                  0029  ;--
                  0029
                  0029  SCH$LOCKW::                                     ; LOCK MUTEX FOR WRITE
                  0029  10$:    SETIPL  #IPL$_SYNCH                     ;;; RAISE TO SYNCH IPL
08 60   10    E6  002C          BBSSI   #MTX$V_WRT,(R0),30$             ;;; SET WRITE PENDING
        60    B6  0030          INCW    MTX$W_OWNCNT(R0)                ;;; RAISE OWNER COUNT
        02    12  0032          BNEQ    20$                             ;;; WAIT IF BUSY
        16    11  0034          BRB     LKEX                            ;;; MERGE WITH COMMON EXIT CODE
                  0036
                  0036  20$:                                            ;;; MUST WAIT FOR EXCLUSIVE USE
        60    B7  0036          DECW    MTX$W_OWNCNT(R0)                ;;; CORRECT COUNT
        43    10  0038  30$:    BSBB    WAITM                           ;;; AND WAIT FOR MUTEX
        ED    11  003A          BRB     10$                             ; REPEAT LOCK ATTEMPT WHEN
                  003C                                                  ; RESCHEDULED
```

```
                          003C                  .SBTTL  SCH$IOLOCKR - LOCK I/O DATABASE MUTEX FOR READ
                          003C        ;++
                          003C        ; FUNCTIONAL DESCRIPTION:
                          003C        ;       SCH$IOLOCKR RETURNS TO THE CALLER WHEN NO WRITERS OWN THE I/O
                          003C        ;       DATABASE MUTEX THUS ASSURING THE I/O DATABASE WILL REMAIN UN-
                          003C        ;       CHANGED UNTIL THE MUTEX IS RELEASED. IPL IS RAISED TO PREVENT
                          003C        ;       AST DELIVERY WHILE THE MUTEX IS OWNED AND THE PROCESS WILL NOT
                          003C        ;       BE OUTSWAPPED.
                          003C        ;
                          003C        ; CALLING SEQUENCE:
                          003C        ;       BSB/JSB SCH$IOLOCKR
                          003C        ;
                          003C        ; INPUT PARAMETERS:
                          003C        ;       R4 - CURRENT PROCESS PCB ADDRESS
                          003C        ;
                          003C        ; IMPLICIT INPUTS:
                          003C        ;       SCH$GQ_MWAIT - MUTEX WAIT QUEUE HEADER
                          003C        ;       PCB OF CURRENT PROCESS
                          003C        ;       I/O DATABASE MUTEX
                          003C        ;
                          003C        ; OUTPUTS:
                          003C        ;       R0 = ADDRESS OF I/O DATABASE MUTEX
                          003C        ;       R1-R3 PRESERVED
                          003C        ;       IPL = ASTDEL
                          003C        ;
                          003C        ; IMPLICIT OUTPUTS:
                          003C        ;       *** TBS ***
                          003C        ;
                          003C        ; SIDE EFFECTS:
                          003C        ;       *** TBS ***
                          003C        ;
                          003C        ;--
                          003C
                          003C       SCH$IOLOCKR::                              ; LOCK I/O DATABASE FOR READ ACCESS
      50   00000000'EF  9E  003C              MOVAB   IOC$GL_MUTEX,R0        ; GET ADDRESS OF I/O DATA BASE MUTEX
```

```
                        0043          .SBTTL  SCH$LOCKR - LOCK MUTEX FOR READ
                        0043  ;++
                        0043  ; FUNCTIONAL DESCRIPTION:
                        0043  ;       SCH$LOCKR RETURNS TO THE CALLER WHEN NO WRITERS OWN THE
                        0043  ;       SPECIFIED MUTEX.  THUS THE STRUCTURE PROTECTED BY THE MUTEX
                        0043  ;       WILL REMAIN UNCHANGED UNTIL THE MUTEX IS RELEASED.  IPL IS
                        0043  ;       RAISED TO PREVENT AST DELIVERY WHILE THE MUTEX IS OWNED AND
                        0043  ;       THE PROCESS WILL NOT BE OUTSWAPPED.
                        0043  ;
                        0043  ; CALLING SEQUENCE:
                        0043  ;       BSB/JSB SCH$LOCKR
                        0043  ;
                        0043  ; INPUT PARAMETERS:
                        0043  ;       R0 - ADDRESS OF MUTEX
                        0043  ;       R4 - CURRENT PROCESS PCB ADDRESS
                        0043  ;
                        0043  ; IMPLICIT INPUTS:
                        0043  ;       SCH$GQ_MWAIT - MUTEX WAIT QUEUE HEADER
                        0043  ;       PCB OF CURRENT PROCESS
                        0043  ;       MUTEX
                        0043  ;
                        0043  ; OUTPUTS:
                        0043  ;       R0-R3 PRESERVED
                        0043  ;       IPL = ASTDEL
                        0043  ;
                        0043  ; IMPLICIT OUTPUTS:
                        0043  ;       *** TBS ***
                        0043  ;
                        0043  ; SIDE EFFECTS:
                        0043  ;       *** TBS ***
                        0043  ;
                        0043  ;--
                        0043
                        0043  SCH$LOCKR::                                     ; LOCK MUTEX FOR READ
                        0043          SETIPL  #IPL$_SYNCH                   ;;; RAISE TO SYNCH IPL
   30 60    10    E0    0046          BBS     #MTX$V_WRT,(R0),RDWAIT        ;;; WAIT IF WRITE PENDING OR
                        004A                                                ;;; IN PROGRESS
            60    B6    004A          INCW    MTX$W_OWNCNT(R0)              ;;; INCREASE OWNER COUNT
   0A A4    0C    91    004C  LKEX:   CMPB    #DYN$C_PCB,PCB$B_TYPE(R4) ; CHECK FOR PCB
            25    12    0050          BNEQ    20$                           ; BUG CHECK IF NOT PCB
         0E A4    B6    0052          INCW    PCB$W_MTXCNT(R4)              ;;; NOTE IN PCB ALSO
   01    0E A4    B1    0055          CMPW    PCB$W_MTXCNT(R4),#1           ; IS THIS THE FIRST MUTEX IT OWNS?
            18    12    0059          BNEQ    10$                           ; BR IF OWNS MORE THAN 1 MUTEX
28 A4    0B A4    90    005B          MOVB    PCB$B_PRI(R4),PCB$B_PRISAV(R4); SAVE CURRENT PRIORITY
29 A4    2F A4    90    0060          MOVB    PCB$B_PRIB(R4),PCB$B_PRIBSAV(R4) ; SAVE BASE PRIORITY
   0B A4    10    91    0065          CMPB    #16,PCB$B_PRI(R4)             ; IS THIS A REAL TIME PROCESS?
            08    1A    0069          BGTRU   10$                           ; BR IF SO
   0B A4    0F    90    006B          MOVB    #15,PCB$B_PRI(R4)             ; ELSE FORCE TO LOWEST RT PRIORITY
   2F A4    0F    90    006F          MOVB    #15,PCB$B_PRIB(R4)            ; AND SET PRIORITY BASE TO RT
                        0073  10$:    SETIPL  #IPL$_ASTDEL                  ;;; DROP TO ASTDEL IPL
            05          0076          RSB                                   ;;; AND RETURN
          00AC    31    0077  20$:    BRW     NOTPCB                        ;
                        007A
                        007A  RDWAIT:                                       ;;; MUST WAIT FOR READ
         C6 AF    DF    007A          PUSHAL  SCH$LOCKR                     ;;; RETRY AFTER WAIT
                        007D
                        007D  WAITM:                                        ;;; WAIT FOR MUTEX TO FREE
                        6E   DD  007D   345           PUSHL   (SP)                          ;;; FORM PC, PSL ON STACK
                     04 AE   DC  007F   346           MOVPSL  4(SP)                         ;;; BUILD PSL
      04 AE   05   10   02   F0  0082   347           INSV    #IPL$_ASTDEL,#PSL$V_IPL,#PSL$S_IPL,4(SP) ;;; SET IPL TO ASTDEL
                4C A4   50   D0  0088   348 WAITR:    MOVL    R0,PCB$L_EFWM(R4)             ;;; SAVE ADDRESS OF MUTEX
         00000000'GF    64   0E  008C   349           INSQUE  (R4),G^SCH$GQ_MWAIT           ;;; INSERT AT HEAD OF WAIT QUEUE
             00000008'GF    B6  0093   350           INCW    G^SCH$GQ_MWAIT+WQH$W_WQCNT         ;;; INCREMENT COUNT IN QUEUE
                2C A4   02   B0  0099   351           MOVW    #SCH$C_MWAIT,PCB$W_STATE(R4)      ;;; SET STATE
                    FF60'   31   009D   352           BRW     SCH$WAITL                     ;;; WAIT WITH STACK CLEAN, STATE SET
                                 00A0   353
```

```
                              00A0          .SBTTL  SCH$RAVAIL - DECLARE RESOURCE AVAILABILITY
                              00A0
                              00A0   ;++
                              00A0   ; FUNCTIONAL DESCRIPTION:
                              00A0   ;       SCH$RAVAIL I3 CALLED TO SIGNAL THE AVAILABILITY OF THE SPECIFIED
                              00A0   ;       RESOURCE AND RELEASE ANY WAITING PROCESSES.
                              00A0   ;
                              00A0   ; CALLING SEQUENCE:
                              00A0   ;       BSB/JSB SCH$RAVAIL
                              00A0   ;
                              00A0   ; INPUT PARAMETERS:
                              00A0   ;       R0 - RESOURCE NUMBER
                              00A0   ;
                              00A0   ; IMPLICIT OUTPUTS:
                              00A0   ;       *** TBS ***
                              00A0   ;
                              00A0   ; SIDE EFFECTS:
                              00A0   ;       *** TBS ***
                              00A0   ;
                              00A0   ;--
                              00A0
                              00A0   SCH$RAVAIL::                                    ; DECLARE RESOURCE AVAILABILITY
    7D 00000000'GF     50  E7 00A0          BBCCI   R0,G^SCH$GL_RESMASK,EXIT              ; CLEAR AND TEST WAITING FLAG
                              00A8          DSBINT  #IPL$_SYNCH                   ;;; BLOCK SYSTEM EVENTS
                       45  11 00AE          BRB     UNLOCK                        ;;; MERGE WITH COMMON CODE
```

```
                              00B0          .SBTTL  SCH$IOUNLOCK - UNLOCK I/O DATABASE MUTEX
                              00B0      ;++
                              00B0      ; FUNCTIONAL DESCRIPTION:
                              00B0      ;       SCH$IOUNLOCK RELEASES OWNERSHIP OF THE I/O DATABASE MUTEX AND
                              00B0      ;       RE-ACTIVATES ANY WAITING PROCESSES IF THE MUTEX HAS BECOME
                              00B0      ;       AVAILABLE AS A CONSEQUENCE OF THIS UNLOCK REQUEST.
                              00B0      ;
                              00B0      ; CALLING SEQUENCE:
                              00B0      ;       BSB/JSB SCH$IOUNLOCK
                              00B0      ;
                              00B0      ; INPUT PARAMETERS:
                              00B0      ;       R4 - PCB ADDRESS OF CURRENT PROCESS
                              00B0      ;
                              00B0      ; IMPLICIT INPUTS:
                              00B0      ;       SCH$GQ_MWAIT - MUTEXT WAIT QUEUE HEADER
                              00B0      ;       PCB OF CURRENT PROCESS
                              00B0      ;       I/O DATABASE MUTEX
                              00B0      ;
                              00B0      ; IMPLICIT OUTPUTS:
                              00B0      ;       *** TBS ***
                              00B0      ;
                              00B0      ; SIDE EFFECTS:
                              00B0      ;       *** TBS ***
                              00B0      ;
                              00B0      ;--
                              00B0
                              00B0      SCH$IOUNLOCK::                          ; UNLOCK I/O DATABASE MUTEX
      50   00000000'EF    9E  00B0              MOVAB   IOC$GL_MUTEX,R0         ; GET ADDRESS OF I/O DATABASE MUTEX
```

```
                                 00B7          .SBTTL  SCH$UNLOCK - UNLOCK MUTEX
                                 00B7  ;++
                                 00B7  ; FUNCTIONAL DESCRIPTION:
                                 00B7  ;          SCH$UNLOCK RELEASES OWNERSHIP OF THE SPECIFIED MUTEX AND
                                 00B7  ;          RE-ACTIVATES ANY WAITING PROCESSES IF THE MUTEX HAS BECOME
                                 00B7  ;          AVAILABLE AS A CONSEQUENCE OF THIS UNLOCK REQUEST.
                                 00B7  ;
                                 00B7  ; CALLING SEQUENCE:
                                 00B7  ;          BSB/JSB SCH$UNLOCK
                                 00B7  ;
                                 00B7  ; INPUT PARAMETERS:
                                 00B7  ;          R0 - MUTEX ADDRESS
                                 00B7  ;          R4 - PCB ADDRESS OF CURRENT PROCESS
                                 00B7  ;
                                 00B7  ; IMPLICIT INPUTS:
                                 00B7  ;          SCH$GQ_MWAIT - MUTEXT WAIT QUEUE HEADER
                                 00B7  ;          PCB OF CURRENT PROCESS
                                 00B7  ;          MUTEX
                                 00B7  ;
                                 00B7  ; IMPLICIT OUTPUTS:
                                 00B7  ;          *** TBS ***
                                 00B7  ;
                                 00B7  ; SIDE EFFECTS:
                                 00B7  ;          *** TBS ***
                                 00B7  ;
                                 00B7  ;--
                                 00B7
                                 00B7  SCH$UNLOCK::                                   ; UNLOCK MUTEX
                                 00B7          DSBINT  #IPL$_SYNCH                    ;;; RAISE TO SYNCH IPL
          0A A4    0C      91    00BD          CMPB    #DYN$C_PCB,PCB$B_TYPE(R4);  STRUCTURE MUST BE PCB
                   63      12    00C1          BNEQ    NOTPCB                         ;
             0E A4         B7    00C3          DECW    PCB$W_MTXCNT(R4)               ;;; NOTE UNLOCK IN PCB
                   25      12    00C6          BNEQ    10$                            ;;; MORE STILL OWNED
    2F A4       29 A4      90    00C8          MOVB    PCB$B_PRIBSAV(R4),PCB$B_PRIB(R4) ; RESTORE SAVED BASE PRIORITY
       51       28 A4      90    00CD          MOVB    PCB$B_PRISAV(R4),R1            ; GET ORIGINAL PRIORITY
          0B A4    51      90    00D1          MOVB    R1,PCB$B_PRI(R4)               ; RESTORE IT
      00000000'GF  51      90    00D5          MOVB    R1,G^SCH$GB_PRI                ; AND ANNOUNCE IT
52  00000000'GF    20  00  EA    00DC          FFS     #0,#32,G^SCH$GL_COMQS,R2; FIND PRIORITY OF NEXT COMPUTABLE PROCESS
             52    51      91    00E5          CMPB    R1,R2                          ; CHECK FOR DELAYED PREMPTION
                   03      1B    00E8          BLEQU   10$                            ; NO, CONTINUE
                                 00EA          SOFTINT #IPL$_SCHED                    ; ELSE RESCHEDULE WHEN IPL DROPS
                   60      B7    00ED  10$:    DECW    MTX$W_OWNCNT(R0)               ;;; DECREMENT OWNERSHIP COUNT
                   31      18    00EF          BGEQ    EXITN                          ;;; EXIT IF NOT LAST
          2D 60    10      E7    00F1          BBCCI   #MTX$V_WRT,(R0),EXITN          ;;; EXIT IF NO WRITE IN PROGRESS
                                 00F5                                                 ;;; OR PENDING
                   11      BB    00F5  UNLOCK: PUSHR   #^M<R0,R4>                     ;;; SAVE PCB ADDRESS
      53    00000000'GF    DE    00F7          MOVAL   G^SCH$GQ_MWAIT,R3              ;;; GET ADDRESS OF WAIT QUEU
             54    63      D0    00FE          MOVL    (R3),R4                        ;;; AND HEAD PCB
             52    02      9A    0101          MOVZBL  #PRI$_RESAVL,R2                ;;; SET PRIORITY INCREMENT CLASS
             54    53      D1    0104  10$:    CMPL    R3,R4                          ;;; CHECK FOR END OF QUEUE
                   17      13    0107          BEQL    30$                            ;;; YES, DONE
          4C A4    6E      D1    0109          CMPL    (SP),PCB$L_EFWM(R4)            ;;; IS PROCESS WAITING FOR THIS MUTEX
                   0C      12    010D          BNEQ    20$                            ;;; NO, SKIP IT
                   64      DD    010F          PUSHL   (R4)                           ;;; SAVE FLINK
                FEEC'      30    0111          BSBW    SCH$CHSE                       ;;; CHANGE TO EXECUTABLE STATE
             08 A3         B7    0114          DECW    WQH$W_WQCNT(R3)                ;;; DECREASE QUEUE LENGTH
                   10      BA    0117          POPR    #^M<R4>                        ;;; RESTORE FLINK
```

```
                 E9  11 0119   467         BRB     10$                     .;; AND CONTINUE
             54  64  D0 011B   468 20$:    MOVL    (R4),R4                 ;;; FLINK ON TO NEXT PCB
                 E4  11 011E   469         BRB     10$                     ;;; AND CONTINUE
                 11  BA 0120   470 30$:    POPR    #^M<R0,R4>              ;;; RESTORE REGISTERS
                        0122   471 EXITN:  ENBINT                          ;;; ENABLE INTERRUPTS
                     05 0125   472 EXIT:   RSB                             ; AND RETURN
                        0126   473
                        0126   474 NOTPCB: BUG_CHECK NOTPCB,FATAL          ; STRUCTURE NOT PCB
                        012A   475         .END
```

```
BUG$_NOTPCB                          ********    X    02
DYN$C_PCB                          = 0000000C
EXIT                                 00000125 R       02
EXITN                                00000122 R       02
IOC$GL_MUTEX                         ********    X    02
IPL$_ASTDEL                        = 00000002
IPL$_SCHED                         = 00000003
IPL$_SYNCH                         = 00000008
LKEX                                 0000004C R       02
MTX$V_WRT                          = 00000010
MTX$W_OWNCNT                       = 00000000
NOTPCB                               00000126 R       02
PCB$B_PRI                          = 0000000B
PCB$B_PRIB                         = 0000002F
PCB$B_PRIBSAV                      = 00000029
PCB$B_PRISAV                       = 00000028
PCB$B_TYPE                         = 0000000A
PCB$L_EFWM                         = 0000004C
PCB$W_MTXCNT                       = 0000000E
PCB$W_STATE                        = 0000002C
PR$_IPL                            = 00000012
PR$_SIRR                           = 00000014
PRI$_RESAVL                        = 00000002
PSL$S_IPL                          = 00000005
PSL$V_IPL                          = 00000010
RDWAIT                               0000007A R       02
SCH$CHSE                             ********    X    02
SCH$C_MWAIT                        = 00000002
SCH$GB_PRI                           ********    X    02
SCH$GL_COMQS                         ********    X    02
SCH$GL_RESMASK                       ********    X    02
SCH$GQ_MWAIT                         ********    X    02
SCH$IOLOCKR                          0000003C RG      02
SCH$IOLOCKW                          00000022 RG      02
SCH$IOUNLOCK                         000000B0 RG      02
SCH$LOCKR                            00000043 RG      02
SCH$LOCKW                            00000029 RG      02
SCH$LOCKWNOWAIT                      0000000A RG      02
SCH$RAVAIL                           000000A0 RG      02
SCH$RWAIT                            00000000 RG      02
SCH$UNLOCK                           000000B7 RG      02
SCH$WAITL                            ********    X    02
SS$_NORMAL                         = 00000001
UNLOCK                               000000F5 R       02
WAITM                                0000007D R       02
WAITR                                00000088 R       02
WQH$W_WQCNT                        = 00000008

                                              +----------------+
                                              ! 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
AEXENONPAGED                       0000012A  (    298.)  02 (  2.)  NOPIC    USR    CON    REL    LCL NOSHR   EXE   RD     WRT NOVEC BYTE
```

```
                                                  +-------------------------+
                                                  ! Performance indicators !
                                                  +-------------------------+

Phase                     Page faults    CPU Time        Elapsed Time
-----                     -----------    --------        ------------
Initialization                     29    00:00:00.06     00:00:01.92
Command processing                120    00:00:00.49     00:00:05.19
Pass 1                            285    00:00:08.42     00:00:27.52
Symbol table sort                   0    00:00:01.22     00:00:04.36
Pass 2                             92    00:00:01.82     00:00:05.21
Symbol table output                 6    00:00:00.08     00:00:00.57
Psect synopsis output               2    00:00:00.02     00:00:00.02
Cross-reference output              0    00:00:00.00     00:00:00.00
Assembler run totals              536    00:00:12.13     00:00:44.79

The working set limit was 1350 pages.
48294 bytes (95 pages) of virtual memory were used to buffer the intermediate code.
There were 50 pages of symbol table space allocated to hold 872 non-local and 12 local symbols.
475 source lines were read in Pass 1, producing 13 object records in Pass 2.
22 pages of virtual memory were used to define 21 macros.

                                                 +--------------------------+
                                                 ! Macro library statistics !
                                                 +--------------------------+

Macro library name                               Macros defined
------------------                               --------------
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                             12
_$255$DUA28:[SYSLIB]STARLET.MLB;2                           6
TOTALS (all libraries)                                     18

979 GETS were required to define 18 macros.

There were no errors, warnings or information messages.

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

0377 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY