H 2

```
RRRRRRRRRRRR     EEEEEEEEEEEEEEE  MMM         MMM
RRRRRRRRRRRR     EEEEEEEEEEEEEEE  MMM         MMM
RRRRRRRRRRRR     EEEEEEEEEEEEEEE  MMM         MMM
RRR         RRR  EEE              MMMMMM   MMMMMM
RRR         RRR  EEE              MMMMMM   MMMMMM
RRR         RRR  EEE              MMMMMM   MMMMMM
RRR         RRR  EEE              MMM   MMM   MMM
RRR         RRR  EEE              MMM   MMM   MMM
RRR         RRR  EEE              MMM   MMM   MMM
RRRRRRRRRRRR     EEEEEEEEEEEE     MMM         MMM
RRRRRRRRRRRR     EEEEEEEEEEEE     MMM         MMM
RRRRRRRRRRRR     EEEEEEEEEEEE     MMM         MMM
RRR   RRR        EEE              MMM         MMM
RRR   RRR        EEE              MMM         MMM
RRR   RRR        EEE              MMM         MMM
RRR      RRR     EEE              MMM         MMM
RRR      RRR     EEE              MMM         MMM
RRR      RRR     EEE              MMM         MMM
RRR         RRR  EEEEEEEEEEEEEEE  MMM         MMM
RRR         RRR  EEEEEEEEEEEEEEE  MMM         MMM
RRR         RRR  EEEEEEEEEEEEEEE  MMM         MMM
```

```
**FILE**ID**REMGETIRP                                G 5                                                                  RE
                                                                                                                          VO




RRRRRRRR    EEEEEEEEEE  MM      MM    GGGGGGGG  EEEEEEEEEE  TTTTTTTTTT    IIIIII    RRRRRRRR    PPPPPPPP
RRRRRRRR    EEEEEEEEEE  MM      MM    GGGGGGGG  EEEEEEEEEE  TTTTTTTTTT    IIIIII    RRRRRRRR    PPPPPPPP
RR      RR  EE          MMMM  MMMM  GG          EE              TT          II      RR      RR  PP      PP
RR      RR  EE          MMMM  MMMM  GG          EE              TT          II      RR      RR  PP      PP
RR      RR  EE          MM  MM  MM  GG          EE              TT          II      RR      RR  PP      PP
RR      RR  EE          MM  MM  MM  GG          EE              TT          II      RR      RR  PP      PP
RRRRRRRR    EEEEEEEE    MM      MM  GG          EEEEEEEE        TT          II      RRRRRRRR    PPPPPPPP
RRRRRRRR    EEEEEEEE    MM      MM  GG          EEEEEEEE        TT          II      RRRRRRRR    PPPPPPPP
RR  RR      EE          MM      MM  GG  GGGGGG  EE              TT          II      RR  RR      PP
RR  RR      EE          MM      MM  GG  GGGGGG  EE              TT          II      RR  RR      PP
RR    RR    EE          MM      MM  GG      GG  EE              TT          II      RR    RR    PP              ....
RR    RR    EE          MM      MM  GG      GG  EE              TT          II      RR    RR    PP              ....
RR      RR  EEEEEEEEEE  MM      MM    GGGGGG    EEEEEEEEEE      TT        IIIIII    RR      RR  PP              ....
RR      RR  EEEEEEEEEE  MM      MM    GGGGGG    EEEEEEEEEE      TT        IIIIII    RR      RR  PP              ....


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            .TITLE  REMGETIRP - PROCESS IRP'S
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: REMOTE I/O ACP
0000    ;
0000    ; ABSTRACT:
0000    ;       THIS MODULE PERFORMS STATE TRANSITIONS FOR
0000    ;       LOGICAL LINKS AND FOR NSP.
0000    ;
0000    ; ENVIRONMENT:
0000    ;       MODE = KERNEL
0000    ;--
```

```
0000          .SBTTL  HISTORY
0000   ; AUTHOR:      SCOTT G. DAVIS, CREATION DATE: 06-JUL-1979
0000   ;
0000   ; MODIFIED BY:
0000   ;
0000   ;       V03-002 KDM0002         Kathleen D. Morse       28-Jun-1982
0000   ;               Added $DYNDEF.
0000   ;
```

```
                0000          .SBTTL  DECLARATIONS
                0000    ;
                0000    ; INCLUDE FILES:
                0000    ;
                0000
                0000          $AQBDEF
                0000          $DYNDEF
                0000          $IPLDEF
                0000          $IRPDEF
                0000          $PRDEF
                0000          $RDPDEF
                0000          $RBFDEF
                0000          $UCBDEF
                0000          $VCBDEF
                0000    ;
                0000    ; MACROS:
                0000    ;
                0000
                0000    ;
                0000    ; EQUATED SYMBOLS:
                0000    ;
                0000
                0000    ;
                0000    ; OWN STORAGE:
                0000    ;
                0000
            00000000          .PSECT  REM_PURE,NOWRT,NOEXE
                0000
00000000        0000   RANGE:  .LONG   0                        ; RANGE FOR WORKING-SET PURGE
7FFFFFFF        0004           .LONG   <1@31>-1                 ; DO IT ALL
                0008
                0008   WORK_VECTOR:                             ; Vector of items for work queue processing
00000000'       0008           .ADDRESS        REM$MBX_READ     ; Put up a mailbox read
00000000'       000C           .ADDRESS        REM$RECV_MSG     ; Put up a link read
                0010
            0000C000          .PSECT  REM_IMPURE      NOSHR,NOEXE,RD,WRT
                0000
00000002        0000   UNIT:   .BLKW   1                        ; For saving remote's unit no.
```

```
                       00000000         .PSECT  REM_CODE,NOWRT
                           0000
                           0000         .SBTTL  PROCESS QIO
                           0000 ;++
                           0000 ; FUNCTIONAL DESCRIPTION:
                           0000 ;
                           0000 ; REM$MAIN - purge working set and process IRP's from the AQB.
                           0000 ;               This routine determines what the I/O request type is
                           0000 ;               and processes CANCEL functions by itself.
                           0000 ;               For a regular IRP, the data in the associated buffered
                           0000 ;               io packet is sent to the remote for processing.
                           0000 ;               For a CANCEL function (ACPCONTROL sans complex buffer),
                           0000 ;               a message for each IRP using the affected channel
                           0000 ;               is sent to the remote, who does the actual cancel.
                           0000 ;
                           0000 ;--
                           0000
                           0000 REM$MAIN::
                           0000         $PURGWS_S       W^RANGE         ; Purge the working set
                           000B ;
                           000B ; TRY TO DEQUEUE A REQUEST
                           000B ;
                           000B 10$:
    52  0000'CF     D0     000B         MOVL    W^REM$GL_Q_HEAD,R2      ; Get address of queue head
      53  00 B2     0F     0010         REMQUE  @(R2),R3                ; Try to get a packet
            11      1C     0014         BVC     20$                     ; If VC there is one
                           0016 ;
                           0016 ; Nothing in queue - see whether it is time to go away
                           0016 ;
         0B A2      95     0016         TSTB    AQB$B_MNTCNT(R2)        ; Any "volumes" mounted?
            03      12     0019         BNEQ    15$                     ; If NEQ yes
         FFE2'      30     001B         BSBW    REM$CHK_ACPDONE         ; See if the ACP is all done
                           001E 15$:
                           001E ;
                           001E ; Go to sleep, my baaaby
                           001E ;
                           001E         $HIBER_S                        ; Hibernate
            E4      11     0025         BRB     10$                     ; Loop
                           0027 ;
                           0027 ; There was a request
                           0027 ;
                           0027 20$:
     50  0A A3      9E     0027         MOVAB   IRP$B_TYPE(R3),R0       ; Point at block type
            80      95     002B         TSTB    (R0)+                   ; Is it a work queue element?
            19      12     002D         BNEQ    22$                     ; If NEQ no
         5A 80      9A     002F         MOVZBL  (R0)+,R10               ; Get the work index
         5B 60      D0     0032         MOVL    (R0),R11                ; Get the device index, maybe
   5A  0004'CF4A    D0     0035         MOVL    W^WORK_VECTOR-4[R10],R10 ; Get address of processing routine
         50 53      D0     003B         MOVL    R3,R0                   ; Get the address for deallocation
      00000000'GF   16     003E         JSB     G^EXE$DEANONPAGED       ; Deallocate the IRP
            6A      16     0044         JSB     (R10)                   ; Process the element
            C3      11     0046         BRB     10$                     ; Try to dequeue something else
                           0048 22$:
     0A  0A A3      91     0048         CMPB    IRP$B_TYPE(R3),S^#DYN$C_IRP ; Is it an IRP?
            28      13     004C         BEQL    35$                     ; If EQL yes
     10  0A A3      91     004E         CMPB    UCB$B_TYPE(R3),S^#DYN$C_UCB ; Is it a UCB?
            1E      12     0052         BNEQ    30$                     ; If NEQ no - fatal error
```

```
                                      0054  ;
                                      0054  ; There is a UCB in my queue
                                      0054  ;
          5B   0080 C3     9A         0054            MOVZBL  UCB$B_ERTCNT(R3),R11    ; Get the index
                    B0     13         0059            BEQL    10$                     ; If EQL none - ignore
                                      005B  ;
                                      005B  ; The channels to this device may be gone - get rid of it, maybe
                                      005B  ;
                                      005B            $SETAST_S       #0              ; Disable AST's
                 FF99'     30         0064            BSBW    REM$KILL_UCB            ; Delete the UCB and break the link
                                      0067            $SETAST_S       #1              ; Enable AST's
                    99     11         0070            BRB     10$                     ; Try for a packet
                                      0072  ;*****
                                      0072  ; can't assign channel
                                      0072  ;*****
                                      0072
                                      0072  30$:
                                      0072            BUG_CHECK       NOTIRPAQB,FATAL ; Bad ACP queue entry
                                      0076  35$:
          57    20 A3      B0         0076            MOVW    IRP$W_FUNC(R3),R7       ; Get the I/O function code
          55    1C A3      D0         007A            MOVL    IRP$L_UCB(R3),R5        ; Get the UCB address
    00'   57   00'  00'    ED         007E            CMPZV   S^#IO$V_FCODE,S^#IO$S_FCODE,R7,S^#IO$_ACPCONTROL ; Control?
                    ED     12         0083            BNEQ    30$                     ; If NEQ no - Bad unsupported error
          E8 2A A3  03     E0         0085            BBS     #IRP$V_COMPLX,IRP$W_STS(R3),30$ ; If BS then real ACPCONTROL
38 A3   00000000 00000000'8F  7D      008A            MOVQ    #SS$_NORMAL, -          ; This is a cancel, so just post it
                                      0096                    IRP$L_IOST1(R3)         ; with success and so
                 FF67'     30         0096            BSBW    REM$POST                ; ignore it.
                 FF6F      31         0099            BRW     10$                     ; For another entry
```

```
                             009C
                             009C  ;++
                             009C  ;
                             009C  ; REM$ALLOC_IRP - allocate an IRP-size block for use as a message buffer
                             009C  ;
                             009C  ; OUTPUTS:
                             009C  ;
                             009C  ;       R2 - buffer address, with size and type filled in
                             009C  ;
                             009C  ;--
                             009C
                             009C  REM$ALLOC_IRP::
   54    00000000'GF   D0    009C          MOVL    G^SCH$GL_CURPCB,R4         ; Set up my PCB address
         00000000'GF   16    00A3          JSB     G^EXE$ALLOCIRP             ; Allocate a block (IRP's are easy to get)
                             00A9          SETIPL  #0                         ; Bring down the IPL
                       05    00AC          RSB                                ; Done
                             00AD
                             00AD          .END
```

```
AQB$B_MNTCNT               = 0000000B
BUG$_NOTIRPAQB               ********    X    04
DYN$C_IRP                  = 0000000A
DYN$C_UCB                  = 00000010
EXE$ALLOCIRP                 ********    X    04
EXE$DEANONPAGED              ********    X    04
IO$S_FCODE                   ********    X    04
IO$V_FCODE                   ********    X    04
IO$_ACPCONTROL               ********    X    04
IRP$B_TYPE                 = 0000000A
IRP$L_IOST1                = 00000038
IRP$L_UCB                  = 0000001C
IRP$V_COMPLX               = 00000003
IRP$W_FUNC                 = 00000020
IRP$W_STS                  = 0000002A
PR$_IPL                    = 00000012
RANGE                        00000000 R       02
REM$ALLOC_IRP                0000009C RG      04
REM$CHK_ACPDONE              ********    X    04
REM$GL_Q_HEAD                ********    X    04
REM$KILL_UCB                 ********    X    04
REM$MAIN                     00000000 RG      04
REM$MBX_READ                 ********    X    02
REM$POST                     ********    X    04
REM$RECV_MSG                 ********    X    02
SCH$GL_CURPCB                ********    X    04
SS$_NORMAL                   ********    X    04
SYS$HIBER                    ********   GX    04
SYS$PURGWS                   ********   GX    04
SYS$SETAST                   ********   GX    04
UCB$B_ERTCNT               = 00000080
UCB$B_TYPE                 = 0000000A
UNIT                         00000000 R       03
WORK_VECTOR                  00000008 R       02

                                                  +-----------------+
                                                  ! 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
REM_PURE                           00000010  (   16.)  02 (  2.)  NOPIC   USR   CON   REL   LCL NOSHR NOEXE   RD  NOWRT NOVEC BYTE
REM_IMPURE                         00000002  (    2.)  03 (  3.)  NOPIC   USR   CON   REL   LCL NOSHR NOEXE   RD    WRT NOVEC BYTE
REM_CODE                           000000AD  (  173.)  04 (  4.)  NOPIC   USR   CON   REL   LCL NOSHR   EXE   RD  NOWRT NOVEC BYTE

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

Phase                       Page faults    CPU Time       Elapsed Time
-----                       -----------    --------       ------------
Initialization                       31    00:00:00.09    00:00:00.32
Command processing                  150    00:00:00.65    00:00:03.51
Pass 1                              317    00:00:09.66    00:00:21.42
Symbol table sort                     0    00:00:01.58    00:00:06.15
```

```
Pass 2                              51      00:00:01.61     00:00:03.18
Symbol table output                  5      00:00:00.07     00:00:00.06
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               558      00:00:13.68     00:00:34.67

The working set limit was 1200 pages.
53234 bytes (104 pages) of virtual memory were used to buffer the intermediate code.
There were 60 pages of symbol table space allocated to hold 1082 non-local and 6 local symbols.
190 source lines were read in Pass 1, producing 17 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:[REM.OBJ]REM.MLB;1                            0
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                           10
_$255$DUA28:[SYSLIB]STARLET.MLB;2                         8
TOTALS (all libraries)                                   18

1184 GETS were required to define 18 macros.

There were no errors, warnings or information messages.

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

0312 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

REMLOCKDB
LIS

REMQIO
LIS

RMS

REMINI
LIS

RMS
MAP

REMPOST
LIS

REMGETIRP
LIS

REM

REMMACROS
MAR

REMACP
MAP

REMXPORTD
LIS

STOPREM
LIS

DAPDEF
MDL

REMACPDAT
LIS

REMPROTCL
LIS

REMSUBR
LIS

TRANSFER
LIS

STOPREM
MAP