F 6

```
RRRRRRRRRRRR     TTTTTTTTTTTTTTT  PPPPPPPPPPPP        AAAAAAAAA     DDDDDDDDDDDD
RRRRRRRRRRRR     TTTTTTTTTTTTTTT  PPPPPPPPPPPP        AAAAAAAAA     DDDDDDDDDDDD
RRRRRRRRRRRR     TTTTTTTTTTTTTTT  PPPPPPPPPPPP        AAAAAAAAA     DDDDDDDDDDDD
RRR         RRR        TTT        PPP         PPP  AAA         AAA  DDD         DDD
RRR         RRR        TTT        PPP         PPP  AAA         AAA  DDD         DDD
RRR         RRR        TTT        PPP         PPP  AAA         AAA  DDD         DDD
RRR         RRR        TTT        PPP         PPP  AAA         AAA  DDD         DDD
RRR         RRR        TTT        PPP         PPP  AAA         AAA  DDD         DDD
RRR         RRR        TTT        PPP         PPP  AAA         AAA  DDD         DDD
RRRRRRRRRRRR           TTT        PPPPPPPPPPPP     AAA         AAA  DDD         DDD
RRRRRRRRRRRR           TTT        PPPPPPPPPPPP     AAA         AAA  DDD         DDD
RRRRRRRRRRRR           TTT        PPPPPPPPPPPP     AAA         AAA  DDD         DDD
RRR   RRR              TTT        PPP              AAAAAAAAAAAAAAA  DDD         DDD
RRR   RRR              TTT        PPP              AAAAAAAAAAAAAAA  DDD         DDD
RRR   RRR              TTT        PPP              AAAAAAAAAAAAAAA  DDD         DDD
RRR      RRR           TTT        PPP              AAA         AAA  DDD         DDD
RRR      RRR           TTT        PPP              AAA         AAA  DDD         DDD
RRR      RRR           TTT        PPP              AAA         AAA  DDD         DDD
RRR         RRR        TTT        PPP              AAA         AAA  DDDDDDDDDDDD
RRR         RRR        TTT        PPP              AAA         AAA  DDDDDDDDDDDD
RRR         RRR        TTT        PPP              AAA         AAA  DDDDDDDDDDDD
```

```
**FILE**ID**VMSRT                                                H  4                                                                      VMS
                                                                                                                                           V04

VV      VV  MM      MM    SSSSSSSS  RRRRRRRR    TTTTTTTTTT
VV      VV  MM      MM    SSSSSSSS  RRRRRRRR    TTTTTTTTTT
VV      VV  MMMM  MMMM  SS          RR      RR      TT
VV      VV  MMMM  MMMM  SS          RR      RR      TT
VV      VV  MM  MM  MM  SS          RR      RR      TT
VV      VV  MM  MM  MM  SS          RR      RR      TT
VV      VV  MM      MM    SSSSSS    RRRRRRRR        TT
VV      VV  MM      MM    SSSSSS    RRRRRRRR        TT
VV      VV  MM      MM          SS  RR  RR          TT
VV      VV  MM      MM          SS  RR  RR          TT
  VV  VV    MM      MM          SS  RR    RR        TT              ....
  VV  VV    MM      MM          SS  RR    RR        TT              ....
    VV      MM      MM  SSSSSSSS    RR      RR      TT              ....
    VV      MM      MM  SSSSSSSS    RR      RR      TT              ....

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  VMSRT - VMS PROTOCOL WITH CTERM HOOKS
0000         .IDENT  'V04-000'
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    ;++
0000    ;
0000    ; FACILITY:
0000    ;
0000    ;       SET HOST, aka RTPAD
0000    ;
0000    ; ABSTRACT:
0000    ;
0000    ;       This module handles all of the VMS specific remote terminal
0000    ;       protocol. It also contains hooks for the CTERM protocol.
0000    ;
0000    ; ENVIRONMENT:  VMS user mode
0000    ;
0000    ;--
0000    ;
0000    ; AUTHOR: Jake VanNoy, CREATION DATE: 14-Jan-1982
0000    ; ORIGINAL AUTHOR: W M Cardoza
0000    ;
0000    ; MODIFIED BY:
0000    ;
0000    ;       V03-008 JLV0364         Jake VanNoy             11-JUL-1984
0000    ;               Add code to skip $SETIMR in ^Y routine if SETIMR
0000    ;               already pending.
0000    ;               Fix broken branch due to change in QUIT.
0000    ;
0000    ;       V03-007 MHB0135         Mark Bramhall           10-Apr-1984
0000    ;               Fix register conventions when calling VMS_INDREAD.
0000    ;
0000    ;       V03-006 JLV0331         Jake VanNoy             28-FEB-1984
0000    ;               Integrate trace facility into permanent code.
```

```
            0000  ;               Add secondary status to REM-F-NETMBX failure to
            0000  ;               show why link was blown away.
            0000  ;
            0000  ;       V03-005 JLV0325         Jake VanNoy             10-JAN-1984
            0000  ;               Add terminator in fake IOSB generated by VMS_INDREAD.
            0000  ;
            0000  ;       V03-004 JLV0296         Jake VanNoy             28-JUL-1983
            0000  ;               add OUTBANDABO channel.
            0000  ;
            0000  ;       V03-003 JLV0266         Jake VanNoy             26-MAY-1983
            0000  ;               Restore code to prevent purge typeahead on "Username:"
            0000  ;
            0000  ;       V03-002 MHB0092         Mark Bramhall            3-Mar-1983
            0000  ;               Moved MAXMSG to $RTPADDEF.
            0000  ;
            0000  ;       V03-001 JLV0            Jake VanNoy             14-Jan-1983
            0000  ;               Module created from VMS protocol code previously
            0000  ;               held in RTPAD module. Hook for CTERM were included.
            0000  ;
            0000  ;**
            0000
            0000          .SBTTL  DECLARATIONS
            0000  ;
            0000  ; INCLUDE FILES:
            0000  ;
            0000
            0000          $DVIDEF
            0000          $RDPDEF
            0000          $RTPADDEF
            0000          $TSADEF
            0000
            0000  ;
            0000  ; EQUATED SYMBOLS:
            0000  ;
00000001    0000  ASSEM_TRACE = 1                 ; INCLUDE TRACE CODE
            0000
00000026    0000  AST$T_BUF = CTP$B_PRO_MSGTYPE ; ***
            0000
            0000  ;
            0000  ;       OP CODES
            0000  ;
00000001    0000          OP_READ = 1
00000002    0000          OP_WRITE = 2
00000003    0000          OP_SETMODE = 3
00000004    0000          OP_SENSEMODE = 4
00000005    0000          OP_CANCEL = 5
00000006    0000          OP_BRDCST = 6
00000100    0000          OP_PRMPT = ^X100
            0000
```

```
                            00000000  .PSECT  RTPAD,NOWRT
                                0000
                                0000          .SBTTL VMSRT - START UP VMS TO VMS PROTOCOL
                                0000  ;++
                                0000  ; FUNCTIONAL DESCRIPTION:
                                0000  ;
                                0000  ;       PERFORMS INITIALIZATION FUNCTIONS FOR VMS TO VMS PROTOCOL
                                0000  ;
                                0000  ; CALLING SEQUENCE:
                                0000  ;
                                0000  ;       CALLS   #0,VMSRT
                                0000  ;
                                0000  ; INPUT PARAMETERS:
                                0000  ;
                                0000  ;       NONE
                                0000  ;
                                0000  ; IMPLICIT INPUTS:
                                0000  ;
                                0000  ;       CHANNEL NUMBERS, ETC.
                                0000  ;
                                0000  ; OUTPUT PARAMETERS:
                                0000  ;
                                0000  ;       NONE
                                0000  ;
                                0000  ; IMPLICIT OUTPUTS:
                                0000  ;
                                0000  ;       NONE
                                0000  ;
                                0000  ; COMPLETION CODES:
                                0000  ;
                                0000  ;
                                0000  ; SIDE EFFECTS:
                                0000  ;
                                0000  ;       SETS UP MAILBOX READS FOR LINK, UNSOLICITED TERMINAL INPUT
                                0000  ;       ENABLES ^C, ^Y  AST'S
                                0000  ;
                                0000  ;--
                                0000  ;
                                0000
                           000C 0000  .ENTRY  VMSRT,  ^M<R2,R3>
                                0002
          00000000'EF      B5   0002          TSTW    FINALACS                ;
                   13      13   0008          BEQL    10$                     ; Branch if no access control string
                                000A          $PUTMSG_S MSGVEC = ACSIGNORE    ; Warn him we are ignoring it
                                001D  10$:
                                001D          ;
                                001D          ; Assume CTERM BIND ACCEPT message
                                001D          ;
      50   00000000'EF     9E   001D          MOVAB   CT_BIND_ACC_MSG,R0      ; Address of buffer (BIND ACCEPT)
         51    0000'8F     3C   0024          MOVZWL  #CT_BIND_MSGLEN,R1      ; Length of message
                                0029
       50 00000000'EF      E8   0029          BLBS    CTERM_FLAG,25$          ; Branch if cterm
                                0030  ;
                                0030  ; *** WHAT USES THIS TERMUNIT GARBAGE??? NOTE CODE IS BUGGY...
                                0030  ;
      50   00000000'EF     9E   0030          MOVAB   DEVNAM,R0               ;ADDRESS OF COUNTED STRING
 00000002'EF    01 A0      B0   0037          MOVW    1(R0),TERMUNIT+2        ;USE PART OF DEV NAME AS HIGH UNIT
 50   00000000'EF   C0  003F          ADDL2   DEVNAMLEN,R0               ;GET TO LAST CHAR (CONTROLLER)
  51   60    41 8F   83  0046          SUBB3   #^A/A/,(R0),R1             ;GET CONTROLLER NUMBER
           51    51   9A  004B          MOVZBL  R1,R1
       51  51    04   78  004E          ASHL    #4,R1,R1
 00000000'EF     51   A0  0052          ADDW2   R1,TERMUNIT                ;ADD CONTROLLER NUMBER TO UNIT
                          0059          ;
                          0059          ; VAX config (BIND) message
                          0059          ;
      50    0000'CF   9E  0059          MOVAB   W^CHAR_BLOCK,R0            ; Characteristics
      51    00B2'CF   9E  005E          MOVAB   W^CONFIG_CHAR,R1           ; CONFIG characteristics
           81    80   7D  0063          MOVQ    (R0)+,(R1)+                ;  first 8 bytes
           81    80   D0  0066          MOVL    (R0)+,(R1)+                ;  last four bytes
                          0069
 50   00000084'EF   9E  0069          MOVAB   CONFIG_MSG,R0              ; Address of buffer
  51    0014'8F    3C  0070          MOVZWL  #CONFIG_MSGLEN,R1          ; Length of message
    00000000'EF    95  0075          TSTB    PROTO_ECO                  ; Eco is 0 for 8 bytes of char
                 03    12  007B          BNEQ    25$                        ; Long form of charateristics?
           51    04   C2  007D          SUBL    #4,R1                      ; No, old form had 8 bytes of char
                          0080          ;
                          0080          ; Send BIND ACCEPT back to HOST
                          0080          ;
                          0080  25$:
       52    26 A0   9E  0080          MOVAB   AST$T_BUF(R0),R2           ; Address of message
              093D    30  0084          BSBW    WRITE_TO_NET_SYNC          ; Write message to NET
                          0087          ;
                          0087          ; Read from NET mailbox
                          0087          ;
      50    0036'CF   9E  0087          MOVAB   W^LINKMAIL,R0              ; Address of data area
                          008C          $QIO_S  CHAN = MAILCHAN -          ; Link mailbox read
                          008C                  FUNC = #IO$_READVBLK -
                          008C                  IOSB = AST$Q_IOSB(R0) -
                          008C                  ASTADR = ASTHANDLER -
                          008C                  ASTPRM = R0 -
                          008C                  P1 = AST$T_BUF(R0) -
                          008C                  P2 = #40
                          00B5          ONERROR QUIT                       ; Die if error
                          00DD          ;
                          00DD          ; Read from associated terminal mailbox
                          00DD          ;
   07 00000000'EF   E8  00DD          BLBS    CTERM_FLAG,30$             ; Branch if CTERM
    0D4E'CF     00   FB  00E4          CALLS   #0,W^UNSMSGDONE            ; Call read routine
                 05    11  00E9          BRB     40$                        ; Branch
                          00EB  30$:
    0000'CF     00   FB  00EB          CALLS   #0,W^CTERM_UNSMSGDONE      ; Call read routine
                          00F0  40$:
                          00F0
                          00F0          ;
                          00F0          ; Set up ^Y and assign channels needed for out of band processing
                          00F0          ;
                          00F0          $QIO_S  CHAN = CNTRLCHAN -               ;WE WILL ALWAYS HANDLE ^Y
                          00F0                  FUNC = #IO$_SETMODE!IO$M_CTRLYAST -
                          00F0                  P1 = CNTRLC_AST -
                          00F0                  P2 = #IO$M_CTRLYAST
                          0119
                          0119          $ASSIGN_S -                        ;CHANNEL FOR INCLUDE OUT OF BAND AST'S
                          0119                  DEVNAM = TTYDESC, -
                          0119                  CHAN = OUTBANDINC
```

```
                           012E
                           012E          $ASSIGN_S -                          ;CHANNEL FOR EXCLUDE OUT OF BAND AST'S
                           012E                  DEVNAM = TTYDESC, -
                           012E                  CHAN = OUTBANDEXC
                           0143
                           0143          $ASSIGN_S -                          ;CHANNEL FOR ABORT OUT OF BAND AST'S
                           0143                  DEVNAM = TTYDESC, -
                           0143                  CHAN = OUTBANDABO
                           0158
      3E 00000000'EF   E9  0158          BLBC    CTERM_FLAG,99$               ; Branch if not CTERM
                           015F          $ASSIGN_S -
                           015F                  DEVNAM = TTYDESC, -
                           015F                  CHAN = CTRLO_CHAN            ; ^O channel
                           0174
                           0174          $QIOW_S CHAN=CTRLO_CHAN,-
                           0174                  FUNC=#IO$_SETMODE!IO$M_OUTBAND!IO$M_INCLUDE,-
                           0174                  P1=  CTERM_CTRLO_AST,-
                           0174                  P2=  #CTRLO_MASK
                           019D
                           019D  99$:
                 0E71  30  019D          BSBW    GETBUF                       ; Get a buffer for reading link
    60   000006DA'EF   9E  01A0          MOVAB   LINKRECV,(R0)                ; Insert state
                           01A7          $QIO_S  CHAN = LINKCHAN -
                           01A7                  FUNC = #IO$_READVBLK -
                           01A7                  IOSB = AST$Q_IOSB(R0) -
                           01A7                  ASTADR = ASTHANDLER -
                           01A7                  ASTPRM = R0 -
                           01A7                  P1 = AST$T_BUF(R0) -
                           01A7                  P2 = #MAXMSG
                           01D4          ONERROR QUIT                         ;
                           01FC  ;
                           01FC  ; The following pice of code was formerly done before reads to net were done
                           01FC  ; This caused disconects sent as a result of driver not found to be lost
                           01FC  ;
000000F0'EF  00000000'EF   B0  01FC          MOVW    TERMUNIT,INIT_MSG+4+RDP$W_UNIT+AST$T_BUF
                           0207          ;
                           0207          ; Check for CTERM protocol. Note that the VAX protocol now
                           0207          ; sends what is interpretted as an unsolicted data message
                           0207          ; which in turn starts up a process on the other end.
                           0207          ; The CTERM protocol module first expects to read and respond
                           0207          ; with INIT messages before proceeding to the unsolicted
                           0207          ; data message.
                           0207          ;
      15 00000000'EF   E8  0207          3LBS    CTERM_FLAG,100$              ; branch if VAX
    50   000000BE'EF   9E  020F          MOVAB   INIT_MSG,R0                  ; Assume VAX protocol
    51   0000000A'8F   D0  0215          MOVL    #INIT_MSGLEN,R1              ; Length of message
            52   26 A0   9E  021C          MOVAB   AST$T_BUF(R0),R2             ; Address of buffer
                 07A1  30  0220          BSBW    WRITE_TO_NET_SYNC            ; Write message to NET
                           0223  100$:
                       04  0223          RET                                  ;
```

```
                        0224            .SBTTL ASTHANDLER - DISPATCH AST'S
                        0224    ;++
                        0224    ; FUNCTIONAL DESCRIPTION:
                        0224    ;
                        0224    ;       CALLS THE APPROPRIATE AST HANDLING ROUTINE
                        0224    ;
                        0224    ; CALLING SEQUENCE:
                        0224    ;
                        0224    ;       AST
                        0224    ;
                        0224    ; INPUT PARAMETERS:
                        0224    ;
                        0224    ;       THE AST PARAMETER IS A POINTER TO THE ADDRESS OF THE AST HANDLING
                        0224    ;       ROUTINE
                        0224    ;
                        0224    ; IMPLICIT INPUTS:
                        0224    ;
                        0224    ;       NONE
                        0224    ;
                        0224    ; OUTPUT PARAMETERS:
                        0224    ;
                        0224    ;       NONE
                        0224    ;
                        0224    ; IMPLICIT OUTPUTS:
                        0224    ;
                        0224    ;       NONE
                        0224    ;
                        0224    ; COMPLETION CODES:
                        0224    ;
                        0224    ;
                        0224    ; SIDE EFFECTS:
                        0224    ;
                        0224    ;       NONE
                        0224    ;
                        0224    ;--
                        0224    ;
                        0224    ASTHANDLER::
                  0000  0224            .WORD   0
50    04 AC        D0  0226            MOVL    4(AP),R0                ; GET AST PARAMETER
00 B0    00        FB  022A            CALLS   #0,@(R0)                ; CALL SERVICE ROUTINE
                   04  022E            RET
                   022F   315           .SBTTL PROCMSG - PROCESS A LINK MESSAGE
                   022F   316   ;++
                   022F   317   ; FUNCTIONAL DESCRIPTION:
                   022F   318   ;
                   022F   319   ;       ACT ON A QIO REQUEST PREVIOUSLY RECEIVED ON THE LINK
                   022F   320   ;
                   022F   321   ; CALLING SEQUENCE:
                   022F   322   ;
                   022F   323   ;       CALLS   #0,PROCMSG
                   022F   324   ;
                   022F   325   ; INPUT PARAMETERS:
                   022F   326   ;
                   022F   327   ;       R0 IS A POINTER TO AN AST CONTROL BLOCK
                   022F   328   ;
                   022F   329   ; IMPLICIT INPUTS:
                   022F   330   ;
                   022F   331   ;       READQIO
                   022F   332   ;       WRITEQIO
                   022F   333   ;
                   022F   334   ; OUTPUT PARAMETERS:
                   022F   335   ;
                   022F   336   ;       NONE
                   022F   337   ;
                   022F   338   ; IMPLICIT OUTPUTS:
                   022F   339   ;
                   022F   340   ;       READQIO
                   022F   341   ;       WRITEQIO
                   022F   342   ;       RETSTATUS
                   022F   343   ;
                   022F   344   ; COMPLETION CODES:
                   022F   345   ;
                   022F   346   ;
                   022F   347   ; SIDE EFFECTS:
                   022F   348   ;
                   022F   349   ;       A QIO TO THE TERMINAL MAY BE PERFORMED.
                   022F   350   ;
                   022F   351   ;--
                   022F   352   ;
                   022F   353   PROCMSG:
            00FC   022F   354           .WORD   ^M<R2,R3,R4,R5,R6,R7>
                   0231   355
                   0231   356   ; CTERM module calls CTERM_PROCMSG directly
                   0231   357
   57    0C A0  3C 0231   358           MOVZWL  AST$W_OPCODE(R0),R7     ; Fetch internal opcode
                   0235   359
         28 A0  A8 0235   360           BISW2   AST$T_BUF+RDP$W_MOD(R0),-
         26 A0     0238   361                   AST$T_BUF+RDP$W_OPCODE(R0)      ;ADD OP MOD
      56    50  D0 023A   362           MOVL    R0,R6
                   023D   363           ;
                   023D   364           ; Case instruction uses a byte, note that prompt flag
                   023D   365           ; is in second byte.
                   023D   366           ;
 04    01   57  8F 023D   367           CASEB   R7,#OP_READ,#OP_SENSEMODE
           0009' 0241     368   4$:      .WORD  READMSG-4$
           00F4' 0243     369            .WORD  WRITEMSG-4$
           0162' 0245     370            .WORD  SETMSG-4$
           0387' 0247     371            .WORD  SENSEMSG-4$
```

```
                    04  0249          RET
                        024A  ;
                        024A  ; A READ QIO REQUEST
                        024A  READMSG:
     60  0000095B'EF    9E  024A          MOVAB   QIODONE,AST$L_STATE(R0) ;NEW STATE
         52    26 A0    3C  0251          MOVZWL  AST$T_BUF+RDP$W_OPCODE(R0),R2
                        0255
                        0255  ; *** Note that the following two instructions are obsolete in a roundabout
                        0255  ; sort of way. LOGINOUT is now smart enough (in V4) to not do a purge typeahead
                        0255  ; to a remote terminal. Connections to a V3.x system will then have the
                        0255  ; type ahead purged.
                        0255
     52  000005D7'EF    AA  0255          BICW    FIRST_READ, R2          ; Don't purge type ahead on first read
         000005D7'EF    B4  025C          CLRW    FIRST_READ              ;  but only on the first read...
                        0262
         56    30 A0    3C  0262          MOVZWL  AST$T_BUF+RDP$L_TT_BCNT(R0),R6
                  54    D4  0266          CLRL    R4
                  55    D4  0268          CLRL    R5
       57    0100 8F    B3  026A          BITW    #OP_PRMPT,R7
                  0E    13  026F          BEQL    10$
         54    39 A0    9E  0271          MOVAB   AST$T_BUF+RDP$T_TT_TERM+1(R0),R4        ;TERM MSG SIZE ADDR+1
         55    FF A4    9A  0275          MOVZBL  -1(R4),R5       ;TERM MSG SIZE
            54    55    C0  0279          ADDL2   R5,R4   ;PROMPT SIZE ADDRESS
            55    84    3C  027C          MOVZWL  (R4)+,R5
                        027F  10$:
0000001E'EF    38 A0    9A  027F          MOVZBL  AST$T_BUF+RDP$T_TT_TERM(RC),RTERMDES    ;TERMINATION CHARS
                  04    12  0287          BNEQ    30$
                        0289  ; USF DEFAULT
                  51    D4  0289          CLRL    R1
                  07    11  028B          BRB     40$
                        028D  30$:
     51  0000001E'EF    9E  028D          MOVAB   RTERMDES,R1
                        0294  40$:
00000022'EF    39 A0    9E  0294          MOVAB   AST$T_BUF+RDP$T_TT_TERM+1(R0),RTERMDES+4
0000C5A7'EF    2A A0    D0  029C          MOVL    AST$T_BUF+RDP$L_REFID(R0),READQIO
         00000000'EF    95  02A4          TSTB    INDFLAG
                  77    12  02AA          BNEQ    100$                    ; Branch if reading from a file
                        02AC  45$:
            53    50    D0  02AC          MOVL    R0,R3                   ; Save in case error later
                        02AF  ;
                        02AF  ; Test for read verify
                        02AF  ;
         00000000'8F    E1  02AF          BBC     #IO$V_EXTEND,-
            0C 26 A0        02B5                  AST$T_BUF+RDP$W_OPCODE(R0),46$  ; Branch if not read verify
                        02B8
         04 A4    54    C0  02B8          ADDL    R4,4(R4)                ; relocate address
         0C A4    54    C0  02BC          ADDL    R4,12(R4)               ; relocate address
         14 A4    54    C0  02C0          ADDL    R4,20(R4)               ; relocate address
                        02C4  46$:
                        02C4          $QIO_S  CHAN = READCHAN -
                        02C4                  FUNC = R2 -
                        02C4                  IOSB = AST$Q_IOSB(R0) -
                        02C4                  ASTADR = ASTHANDLER -
                        02C4                  ASTPRM = R0 -
                        02C4                  P1 = AST$T_BUF+RDP$T_TT_RDATA+2(R0) -
                        02C4                  P2 = R6 -
                        02C4                  P3 = AST$T_BUF+RDP$L_TT_TIMOUT(R0) -
```

```
                                 02C4    429                  P4 = R1 -
                                 02C4    430                  P5 = R4 -
                                 02C4    431                  P6 = R5
                                 02EE    432          IF_NO_QUOTA     QUIT
                  03 50    E8    031A    433          BLBS    R0,47$                  ; branch if ok
                   0293    31    031D    434          BRW     QIO_ERR                 ; Handle error
                                 0320    435 47$:
                   02FE    31    0320    436          BRW     PROCMSG_EXIT
                                 0323    437
                                 0323    438          ;
                                 0323    439          ; Read from a file, not the terminal
                                 0323    440          ;
                                 0323    441
                                 0323    442 100$:
                     51    DD    0323    443          PUSHL   R1                      ; Save
             51   3A A0    9E    0325    444          MOVAB   AST$T_BUF+RDP$T_TT_RDATA+2(R0),R1
                                 0329    445                                          ; Address for input
                53   56    D0    0329    446          MOVL    R6,R3                   ; Size for input
                   0C0B    30    032C    447          BSBW    VMS_INDREAD             ; Try the indirect file
                     51  8ED0    032F    448          POPL    R1                      ; Save
                   FF77    31    0332    449          BRW     45$                     ; If routine returns here,
                                 0335    450                                          ; RMS got EOF, must read from TTY.
                                 0335    451
                                 0335    452 ;
                                 0335    453 ; A WRITE QIO REQUEST
                                 0335    454 WRITEMSG:
        60    0000095B'EF    9E  0335    455          MOVAB   QIODONE,AST$L_STATE(R0) ;NEW STATE
             52     26 A0    3C  033C    456          MOVZWL  AST$T_BUF+RDP$W_OPCODE(R0),R2
             53     30 A0    3C  0340    457          MOVZWL  AST$T_BUF+RDP$L_TT_BCNT(R0),R3
      000005AB'EF   2A A0    D0  0344    458          MOVL    AST$T_BUF+RDP$L_REFID(R0),WRITEQIO
                                 034C    459          $QIO_S  CHAN = WRITECHAN -
                                 034C    460                  FUNC = R2 -
                                 034C    461                  IOSB = AST$Q_IOSB(R0) -
                                 034C    462                  ASTADR = ASTHANDLER -
                                 034C    463                  ASTPRM = R0 -
                                 034C    464                  P1 = AST$T_BUF+RDP$T_TT_WDATA(R0) -
                                 034C    465                  P2 = R3 -
                                 034C    466                  P4 = AST$T_BUF+RDP$L_TT_CARCON(R0)
                                 0374    467          IF_NO_QUOTA     QUIT
                   027E    31    03A0    468          BRW     PROCMSG_EXIT
                                 03A3    469 ;
                                 03A3    470 ; A SET MODE QIO REQUEST
                                 03A3    471 SETMSG:
        26 A0     0000'8F    B3  03A3    472          BITW    #IO$M_OUTBAND,AST$T_BUF+RDP$W_OPCODE(R0)
                       03    12  03A9    473          BNEQ    49$
                     009B    31  03AB    474          BRW     60$
                                 03AE    475 ; HANDLE AN OUT OF BAND AST REQUEST
                57     50    D0  03AE    476 49$:     MOVL    R0,R7                   ;SAVE R0
            53     30 A0     9E  03B1    477          MOVAB   AST$T_BUF+RDP$B_TT_OUTBAND(R0),R3       ;START OF DATA (MASKS)
               04      63    91  03B5    478          CMPB    (R3),#4                 ;FOR NOW, IT MUST BE A SINGLE LONGWORD
                       03    13  03B8    479          BEQL    51$
                     0265    31  03BA    480          BRW     OUTBAND_ERR
      000005CB'EF   01 A3    D1  03BD    481 51$:     CMPL    1(R3),INCMASK+4
                       31    13  03C5    482          BEQL    53$                     ;DON'T BOTHER WITH THE QIO
      000005CB'EF   01 A3    D0  03C7    483          MOVL    1(R3),INCMASK+4         ;GET THE INCLUDE MASK
                                 03CF    484          $QIOW_S CHAN = OUTBANDINC -
                                 03CF    485                  FUNC = #IO$_SETMODE!IO$M_OUTBAND!IO$M_INCLUDE -
```

```
                                  03CF                         P1 = OUTBANDAST -
                                  03CF                         P2 = #INCMASK
            04    05 A3     91    03F8   53$:      CMPB      5(R3),#4              ;AGAIN, A SINGLE LONGWORD
                     03     13    03FC             BEQL      55$
                   0221     31    03FE             BRW       OUTBAND_ERR
     000005D3'EF  06 A3     D1    0401   55$:      CMPL      6(R3),EXCMASK+4
                     31     13    0409             BEQL      58$                   ;DON'T BOTHER WITH THE QIO
     000005D3'EF  06 A3     D0    040B             MOVL      6(R3),EXCMASK+4       ;EXCLUDE MASK
                                  0413             $QIOW_S CHAN = OUTBANDEXC -
                                  0413                       FUNC = #IO$_SETMODE!IO$M_OUTBAND -
                                  0413                       P1 = OUTBANDAST -
                                  0413                       P2 = #EXCMASK
                  50   57   D0    043C   58$:      MOVL      R7,R0                 ;RESTORE R0
       0000095B'EF     00   FB    043F             CALLS     #0,QIODONE
                   01D8     31    0446             BRW       PROCMSG_EXIT
   26 A0    0000'8F         B3    0449   60$:      BITW      #IO$M_HANGUP,AST$T_BUF+RDP$W_OPCODE(R0)
                     29     13    044F             BEQL      70$
                                  0451   ; HANGUP IS TREATED A PROGRAM EXIT
                                  0451             QUIT      #SS$_NORMAL
                                  047A   70$:
      60   0000095B'EF      9E    047A             MOVAB     QIODONE,AST$L_STATE(R0)         ;NEW STATE
            52    26 A0     3C    0481             MOVZWL    AST$T_BUF+RDP$W_OPCODE(R0),R2
     000005A7'EF  2A A0     D0    0485             MOVL      AST$T_BUF+RDP$L_REFID(R0),READQIO
   26 A0    0000'8F         B3    048D             BITW      #IO$M_CTRLCAST,AST$T_BUF+RDP$W_OPCODE(R0)
                     03     12    0493             BNEQ      72$
                   00A2     31    0495             BRW       80$
                                  0498   ; CONTROL-C ENABLE OR DISABLE
                  30 A0     D5    0498   72$:      TSTL      AST$T_BUF+RDP$L_TT_ASTPRM(R0)
                     6E     13    049B             BEQL      75$             ;DISABLE
         000005A6'EF        95    049D             TSTB      CNTRCFLAG       ;IS THERE ALREADY ONE ENABLED?
                     03     13    04A3             BEQL      74$
                   009A     31    04A5             BRW       90$             ;YES
         000005A6'EF        96    04A8   74$:      INCB      CNTRCFLAG
                                  04AE             $QIO_S    CHAN = CNTRLCHAN -       ;ENABLE
                                  04AE                       FUNC = #IO$_SETMODE!IO$M_CTRLCAST -
                                  04AE                       IOSB = AST$Q_IOSB(R0) -
                                  04AE                       ASTADR = ASTHANDLER -
                                  04AE                       ASTPRM = R0 -
                                  04AE                       P1 = CNTRLC_AST -
                                  04AE                       P2 = #IO$M_CTRLCAST
                                  04DC             IF_NO_QUOTA     QUIT
                   0116     31    0508             BRW       PROCMSG_EXIT
                                  050B   75$:      $QIO_S    CHAN = CNTRLCHAN -       ;DISABLE
                                  050B                       FUNC = #IO$_SETMODE!IO$M_CTRLCAST -
                                  050B                       IOSB = AST$Q_IOSB(R0) -
                                  050B                       ASTADR = ASTHANDLER -
                                  050B                       ASTPRM = R0
         000005A6'EF        94    0531             CLRB      CNTRCFLAG       ;NO ^C'S ENABLED
                   00E7     31    0537             BRW       PROCMSG_EXIT
   26 A0    0000'8F         B3    053A   80$:      BITW      #IO$M_CTRLYAST,AST$T_BUF+RDP$W_OPCODE(R0)
                     1E     13    0540             BEQL      100$
                                  0542   ; CONTROL-Y ENABLE OR DISABLE
04 A0   00000000 00000000'8F  7D  0542   90$:      MOVQ      #SS$_NORMAL,AST$Q_IOSB(R0)            ;PRETEND WE DID THE QIO
                                  054E             $DCLAST_S ASTADR = ASTHANDLER -
                                  054E                         ASTPRM = R0
                   00C1     31    055D             BRW       PROCMSG_EXIT
                                  0560   100$:
```

```
             53    50   D0 0560               MOVL     R0,R3                         ;WE NEED THE BUFFER ADDRESS LATER
             54    08   D0 0563               MOVL     #8,R4                         ;ASSUME SHORT FORM
 00000000'EF    30 A0   7D 0566               MOVQ     AST$T_BUF+RDP$Q_TT_CHAR(R0),CHARBUF
        00000000'EF     95 056E               TSTB     PROTO_ECO                     ;IS IT LEVEL 0
                   0B   13 0574               BEQL     105$
             54    04   C0 0576               ADDL     #4,R4                         ;LONG FORM
 00000008'EF    44 A0   D0 0579               MOVL     AST$T_BUF+RDP$L_TT_CHAR2(R0),CHARBUF+8
                           0581       105$:   $QIO_S   CHAN = READCHAN -
                           0581                        FUNC = R2 -
                           0581                        IOSB = AST$Q_IOSB(R0) -
                           0581                        ASTADR = ASTHANDLER -
                           0581                        ASTPRM = R0 -
                           0581                        P1 = CHARBUF -
                           0581                        P2 = R4 -
                           0581                        P3 = AST$T_BUF+RDP$L_TT_SPEED(R0) -
                           0581                        P4 = AST$T_BUF+RDP$L_TT_FILL(R0) -
                           0581                        P5 = AST$T_BUF+RDP$L_TT_PARITY(R0)
                           05B0       ; TAKE CARE OF NOT ALWAYS GETTING AST ON ERROR
              6E 50     E8 05B0               BLBS     R0,PROCMSG_EXIT               ;NO ERROR
                           05B3
                           05B3       QIO_ERR:
          04 A3    50   B0 05B3               MOVW     R0,AST$Q_IOSB(R3)             ;MAKE SURE STATUS IS IN IOSB
                           05B7               $DCLAST_S ASTADR = ASTHANDLER -
                           05B7                          ASTPRM = R3
                   59   11 05C6               BRB      PROCMSG_EXIT
                           05C8       ;
                           05C8       ; A SENSE MODE QIO REQUEST
                           05C8       SENSEMSG:
             53    50   D0 05C8               MOVL     R0,R3                         ;WE NEED THE BUFFER ADDRESS LATER
     60  0000095B'EF    9E 05CB               MOVAB    QIODONE,AST$L_STATE(R0)  ;NEW STATE
             52    26 A0   3C 05D2            MOVZWL   AST$T_BUF+RDP$W_OPCODE(R0),R2
 000005A7'EF    2A A0   D0 05D6               MOVL     AST$T_BUF+RDP$L_REFID(R0),READQIO
                38 A0   7C 05DE               CLRQ     AST$T_BUF+RDP$Q_TT_SCHAR(R0)
                40 A0   D4 05E1               CLRL     AST$T_BUF+RDP$Q_TT_SCHAR+8(R0)
                           05E4               $QIO_S   CHAN = READCHAN -
                           05E4                        FUNC = R2 -
                           05E4                        IOSB = AST$Q_IOSB(R0) -
                           05E4                        ASTADR = ASTHANDLER -
                           05E4                        ASTPRM = R0 -
                           05E4                        P1 = AST$T_BUF+RDP$Q_TT_SCHAR(R0) -
                           05E4                        P2 = #12
                           0609       ; TAKE CARE OF NOT ALWAYS GETTING AST ON ERROR
              15 50     E8 0609               BLBS     R0,PROCMSG_EXIT               ;NO ERROR
          04 A3    50   B0 060C               MOVW     R0,AST$Q_IOSB(R3)             ;MAKE SURE STATUS IS IN IOSB
                           0610               $DCLAST_S ASTADR = ASTHANDLER -
                           0610                          ASTPRM = R3
                   00   11 061F               BRB      PROCMSG_EXIT
                           0621
                           0621       PROCMSG_EXIT:
                        04 0621               RET
                           0622
                           0622       OUTBAND_ERR:
                           0622               $PUTMSG_S MSGVEC = BADOUTBAND
                           0635               QUIT
                           065A
```

```
                                065A              .SBTTL LINKRECV - PROCESS A RECEIVED MESSAGE
                                065A      ;++
                                065A      ; FUNCTIONAL DESCRIPTION:
                                065A      ;
                                065A      ;       PROCESS THE AST INDICATING THAT A MESSAGE WAS RECEIVED ON THE LINK
                                065A      ;
                                065A      ; CALLING SEQUENCE:
                                065A      ;
                                065A      ;       CALLS   #0,LINKRECV
                                065A      ;
                                065A      ; INPUT PARAMETERS:
                                065A      ;
                                065A      ;       R0 POINTS TO AN AST CONTROL BLOCK
                                065A      ;
                                065A      ; IMPLICIT INPUTS:
                                065A      ;
                                065A      ;       WRITEQIO
                                065A      ;       READQIO
                                065A      ;
                                065A      ; OUTPUT PARAMETERS:
                                065A      ;
                                065A      ;       NONE
                                065A      ;
                                065A      ; IMPLICIT OUTPUTS:
                                065A      ;
                                065A      ;       AN ENTRY MAY BE ADDED TO THE QUEUE OF PENDING READS OR WRITES.
                                065A      ;       RETSTATUS
                                065A      ;
                                065A      ; COMPLETION CODES:
                                065A      ;
                                065A      ;
                                065A      ; SIDE EFFECTS:
                                065A      ;
                                065A      ;       A NEW READ OF THE LINK IS INITIATED.  IF THERE IS A ERROR ON THIS QIO,
                                065A      ;       A $WAKE IS ISSUED TO CAUSE A PROGRAM EXIT.
                                065A      ;
                                065A      ;       A PREVIOUS I/O MAY BE CANCELED
                                065A      ;
                                065A      ;--
                                065A      ;
                                065A      .IF DF ASSEM_TRACE
                                065A      TRACE_RECV:
                      3F   BB   065A              PUSHR   #^M<R0,R1,R2,R3,R4,R5>
           51   06 A0  3C   065C              MOVZWL  AST$Q_IOSB+2(R0),R1
           52   26 A0  9E   0660              MOVAB   AST$T_BUF(R0),R2
   53    00000000'EF   9E   0664              MOVAB   DBG$LINKRECV,R3          ; Message received from net
                   F992'   30   066B              BSBW    DBG$TRACE_IO             ; ... log input
                      3F   BA   066E              POPR    #^M<R0,R1,R2,R3,R4,R5>
                      77   11   0670              BRB     TRACE_CONTINUE
                                0672      .ENDC
                                0672
                                0672      LINKRECV_ERR:
           52   04 A0  3C   0672              MOVZWL  AST$Q_IOSB(R0),R2                ;SAVE ERROR STATUS
        0000'8F    52  B1   0676              CMPW    R2,#SS$_ABORT
                   1F  12   067B              BNEQ    20$
   00000597'EF     52  D0   067D              MOVL    R2,LINKERR               ;SAVE ERROR
                                0684              $SETIMR_S DAYTIM = THREESEC -   ;JUST IN CASE MAILBOX DOESN'T GET REASON
```

```
                     0684   656                      ASTADR = LINKGONE
              04     069B   657 10$:     RET                                  ;LINK BROKE - LINK MBX WILL GET REASON
     0000'CF  95     069C   658 20$:     TSTB     W^WAKEFLAG                  ; has QUIT already happened
          F9  12     06A0   659          BNEQ     10$                         ; if so, ignore error
                     06A2   660          $PUTMSG_S MSGVEC = DECNETERR         ;LINK ERROR
                     06B5   661          QUIT     R2
                     06DA   662
                     06DA   663 LINKRECV::
            008C     06DA   664          .WORD    ^M<R2,R3,R7>
    92 04 A0  E9     06DC   665          BLBC     AST$Q_IOSB(R0),LINKRECV_ERR      ;ERROR ON LINK READ ??
                     06E0   666
                     06E0   667 .IF DF ASSEM_TRACE
          01  E1     06E0   668          BBC      #RTLOG$V_TRACE,-
     0000'CF         06E2   669                   W^RTLOG_FLAGS,-
          03         06E5   670                   TRACE_CONTINUE              ; branch if not tracing
        FF71  31     06E6   671          BRW      TRACE_RECV                  ; branch if tracing
                     06E9   672 TRACE_CONTINUE:
                     06E9   673 .ENDC
 06 00000000'EF  E9  06E9   674          BLBC     CTERM_FLAG,20$
       F90D'  30     06F0   675          BSBW     CTERM_LINKRECV
        0072  31     06F3   676          BRW      100$
                     06F6   677 20$:
  51   26 A0  3C     06F6   678          MOVZWL   AST$T_BUF+RDP$W_OPCODE(R0),R1
          57  D4     06FA   679          CLRL     R7
52  00001067'EF  9E  06FC   680          MOVAB    TERMOPS,R2
      53  82  3C     0703   681          MOVZWL   (R2)+,R3                    ;COUNT
      62  51  B1     0706   682 30$:     CMPW     R1,(R2)
      06  12         0709   683          BNEQ     40$
  57   02 A2  3C     070B   684          MOVZWL   2(R2),R7                    ;INTERNAL OP CODE
      06  11         070F   685          BRB      50$
      52  04  C0     0711   686 40$:     ADDL2    #4,R2                       ;SKIP
      EF  53  F5     0714   687          SOBGTR   R3,30$
                     0717   688 50$:
   0C A0  57  B0     0717   689          MOVW     R7,AST$W_OPCODE(R0)         ; Save for later
      06  57  91     071B   690          CMPB     R7,#OP_BRDCST
          08  12     071E   691          BNEQ     60$
  000008BC'EF  16    0720   692          JSB      BROADCAST                   ;GO BROADCAST IT
          40  11     0726   693          BRB      100$
      05  57  91     0728   694 60$:     CMPB     R7,#OP_CANCEL
          08  12     072B   695          BNEQ     70$
  000007BD'EF  16    072D   696          JSB      CANCELIO                    ;GO CANCEL IT
          33  11     0733   697          BRB      100$
      02  57  91     0735   698 70$:     CMPB     R7,#OP_WRITE
          18  12     0738   699          BNEQ     90$
                     073A   700 ; A WRITE QIO
  000005AB'EF  D5    073A   701          TSTL WRITEQIO
          07  12     0740   702          BNEQ     80$
                     0742   703 ; NO WRITE IN PROGRESS
 FAE8 CF  00  FB     0742   704          CALLS    #0,PROCMSG
          1F  11     0747   705          BRB      100$
                     0749   706 80$:
000005BB'FF  60  0E  0749   707          INSQUE  (R0),@WRITEQ+4  ;QUEUE IT
          16  11     0750   708          BRB      100$
                     0752   709 90$:
                     0752   710 ; A READ OR SETMODE OR SENSEMODE
  000005A7'EF  D5    0752   711          TSTL READQIO
          07  12     0758   712          BNEQ     95$
```

```
                            075A  ; NO READ IN PROGRESS
       FAD0 CF    00    FB  075A          CALLS   #0,PROCMSG
                  07    11  075F          BRB     100$
                            0761  95$:
   000005B3'FF    60    0E  0761          INSQUE  (R0),@READQ+4   ;QUEUE IT
                            0768  100$:
            08A6  30        0768          BSBW    GETBUF          ;GET A BUFFER FOR READING LINK
        60   FF6B CF    9E  076B          MOVAB   LINKRECV,(R0)
                            0770          $QIO_S  CHAN = LINKCHAN -       ;READ LINK AGAIN
                            0770                  FUNC = #IO$_READVBLK -
                            0770                  IOSB = AST$Q_IOSB(R0) -
                            0770                  ASTADR = ASTHANDLER -
                            0770                  ASTPRM = R0 -
                            0770                  P1 = AST$T_BUF(R0) -
                            0770                  P2 = #MAXMSG
              1E 50     E8  079B          BLBS    R0,110$                 ;WAS THERE A LINK ERROR
   00000597'EF    50    D0  079E          MOVL    R0,LINKERR              ;SAVE ERROR
                            07A5          $SETIMR_S DAYTIM = THREESEC -   ;JUST IN CASE MAILBOX DOESN'T GET REASON
                            07A5                    ASTADR = LINKGONE
                        04  07BC  110$:   RET
                            07BD  ;
                            07BD  ;
                            07BD  ;
                            07BD  ; CANCEL AN I/O
                            07BD  ;
                            07BD  CANCELIO:
          52      50    D0  07BD          MOVL    R0,R2
                            07C0          $CANCEL_S CHAN = CNTRLCHAN      ;DISABLE ^C ON A CANCEL
      000005A6'EF       94  07CE          C'RB    CNTRCFLAG               ;NO CONTROL-C'S ENABLED
          50      52    D0  07D4          MOVL    R2,R0
 000005A7'EF    2A A0   D1  07D7          CMPL    AST$T_BUF+RDP$L_REFID(R0),READQIO
                  15    12  07DF          BNEQ    20$
                            07E1  ; CANCEL THE READ
      0000105F'EF       16  07E1          JSB     BUFFREE
                            07E7          $CANCEL_S  CHAN = READCHAN
                        05  07F5          RSB
 000005AB'EF    2A A0   D1  07F6  20$:    CMPL    AST$T_BUF+RDP$L_REFID(R0),WRITEQIO
                  15    12  07FE          BNEQ    30$
                            0800  ; CANCEL THE WRITE
      0000105F'EF       16  0800          JSB     BUFFREE
                            0806          $CANCEL_S  CHAN = WRITECHAN
                        05  0814          RSB
                            0815  30$:
   51   000005AF'EF     9E  0815          MOVAB   READQ,R1        ;GET QUEUE OF PENDING READS
            51    61    9E  081C  31$:    MOVAB   (R1),R1
   000005AF'8F    51    D1  081F          CMPL    R1,#READQ
                  09    13  0826          BEQL    32$             ;END OF QUEUE
               2A A1    D1  0828          CMPL    AST$T_BUF+RDP$L_REFID(R1),-
               2A A0        082B                  AST$T_BUF+RDP$L_REFID(R0)
                  ED    12  082D          BNEQ    31$             ;TRY THE NEXT ONE
                  23    11  082F          BRB     39$             ;GO CANCEL IT
   51   000005B7'EF     9E  0831  32$:    MOVAB   WRITEQ,R1       ;GET QUEUE OF PENDING WRITES
            51    61    D0  0838  33$:    MOVL    (R1),R1
   000005B7'8F    51    D1  083B          CMPL    R1,#WRITEQ
                  09    13  0842          BEQL    34$             ;END OF QUEUE
               2A A1    D1  0844          CMPL    AST$T_BUF+RDP$L_REFID(R1),-
               2A A0        0847                  AST$T_BUF+RDP$L_REFID(R0)
```

```
                     ED   12  0849           BNEQ     33$               ;TRY THE NEXT ONE
                     07   11  084B           BRB      39$               ;SAVE THE CANCEL - WRITE NOT DONE YET
            0000105F'EF   16  084D  34$:     JSB      BUFFREE           ;QIO ALREADY DONE - THROW OUT THE CANCEL
                          05  0853           RSB
                              0854  39$:
                  51   61   0F  0854           REMQUE   (R1),R1 ;REMOVE THE ENTRY
     00000571'EF    2A A1   D0  0857           MOVL     AST$T_BUF+RDP$L_REFID(R1),CANMSG+RDP$L_REFID
            0000105F'EF   16  085F           JSB      BUFFREE ;WE DON'T NEED THE CANCEL ANYMORE
                  50   51   D0  0865           MOVL     R1,R0
            0000105F'EF   16  0868           JSB      BUFFREE           ;WE DON'T NEED THE QIO EITHER
                              086E           $QIO_S   CHAN = LINKCHAN -        ;SEND THE CANCEL COMPLETE MSG
                              086E                    FUNC = #IO$_WRITEVBLK -
                              086E                    P1 = CANMSG -
                              086E                    P2 = #RDP$K_HEADERLEN+8
                              0893           ONERROR QUIT
                          05  08BB           RSB
                              08BC  ;
                              08BC  ;
                              08BC  ;
                              08BC  ; BROADCAST TO THE TERMINAL
                              08BC  ;
                              08BC  BROADCAST:
     0000057F'EF    30 A0   D0  08BC           MOVL     AST$T_BUF+RDP$L_TT_BCNT(R0),BRDDESC      ;COUNT
     00000583'EF    38 A0   9E  08C4           MOVAB    AST$T_BUF+RDP$T_TT_WDATA(R0),BRDDESC+4   ;ADDRESS
0A0D 8F   00000583'FF       B1  08CC           CMPW     @BRDDESC+4,#^XA0D                   ;CHECK FOR CR-LF
                     06   12  08D5           BNEQ     10$
          00000583'FF     B4  08D7           CLRW     @BRDDESC+4                          ;REMOVE IT - BRDCST ADDS ANOTHER ONE
               52    50   D0  08DD  10$:     MOVL     R0,R2
                              08E0           $BRDCST_S MSGBUF = BRDDESC -
                              08E0                      DEVNAM = TTYDESC
           30 A2     50   7D  08F7           MOVQ     R0,AST$T_BUF+RDP$Q_STATUS(R2)    ;RETURN AN IOSB
      62   00000A9F'EF    9E  08FB           MOVAB    LNKWRTDONE,AST$L_ST, E(R2)             ;NEW STATE
      26 A2    FFFE 8F    B0  0902           MOVW     #RDP$C_END,AST$T_BUF+RDP$W_OPCODE(R2)   ;A STATUS MESSAGE
                  28 A2   B4  0908           CLRW     AST$T_BUF+RDP$W_MOD(R2)
                              090B           $QIO_S   CHAN = LINKCHAN -        ;WRITE MESSAGE ON LINK
                              090B                    FUNC = #IO$_WRITEVBLK -
                              090B                    IOSB = AST$Q_IOSB(R2) -
                              090B                    ASTADR = ASTHANDLER -
                              090B                    ASTPRM = R2 -
                              090B                    P1 = AST$T_BUF(R2) -
                              090B                    P2 = #RDP$K_HEADERLEN+8
                              0932           ONERROR QUIT
                          05  095A           RSB
```

```
                                095B              .SBTTL QIODONE - PROCESS A COMPLETED TERMINAL QIO
                                095B          ;++
                                095B          ; FUNCTIONAL DESCRIPTION:
                                095B          ;
                                095B          ;       HANDLE THE AST INDICATING THAT A TERMINAL QIO HAS COMPLETED
                                095B          ;
                                095B          ; CALLING SEQUENCE:
                                095B          ;
                                095B          ;       CALLS   #0,QIODONE
                                095B          ;
                                095B          ; INPUT PARAMETERS:
                                095B          ;
                                095B          ;       R0 POINTS TO AN AST CONTROL BLOCK
                                095B          ;
                                095B          ; IMPLICIT INPUTS:
                                095B          ;
                                095B          ;       NONE
                                095B          ;
                                095B          ; OUTPUT PARAMETERS:
                                095B          ;
                                095B          ;       NONE
                                095B          ;
                                095B          ; IMPLICIT OUTPUTS:
                                095B          ;
                                095B          ;       RETSTATUS
                                095B          ;
                                095B          ; COMPLETION CODES:
                                095B          ;
                                095B          ;
                                095B          ; SIDE EFFECTS:
                                095B          ;
                                095B          ;       A STATUS MESSAGE IS WRITTEN TO THE LINK.  IF THERE IS AN ERROR ON THIS
                                095B          ;       QIO, A $WAKE IS ISSUED TO CAUSE THE PROGRAM TO EXIT
                                095B          ;
                                095B          ;--
                                095B          ;
                                095B          QIODONE::
                           00BC 095B                  .WORD   ^M<R2,R3,R4,R5,R7>
                                095D
       0A 00000000'EF      E9   095D                  BLBC    CTERM_FLAG,10$
                  F699'    30   0964                  BSBW    CTERM_QIODONE
                     50    D5   0967                  TSTL    R0                          ; Was a message returned?
                     56    12   0969                  BNEQ    60$                         ; Branch if yes
                   0055    31   096B                  BRW     70$                         ; Exit if it not
                                096E          10$:
                  2A A0    D5   096E                  TSTL    AST$T_BUF+RDP$L_REFID(R0)                   ;CHECK FOR ZERO REF ID
                     0C    12   0971                  BNEQ    20$
          04 A0      01    9A   0973                  MOVZBL  #1,AST$Q_IOSB(R0)           ;NO ERRORS
     00000A9F'EF     00    FB   0977                  CALLS   #0,LNKWRTDONE   ;PRETEND WE SENT IT
                           04   097E                  RET
                                097F          20$:
          57    0C A0      3C   097F                  MOVZWL  AST$W_OPCODE(R0),R7         ; Fetch internal opcode
             01      57    91   0983                  CMPB    R7,#OP_READ
                     11    12   0986                  BNEQ    30$
                                0988          ; IT WAS A READ SO WE NEED THE COUNT
          51    06 A0      3C   0988                  MOVZWL  AST$Q_IOSB+2(R0),R1         ;CHARACTERS BEFORE THE TERMINATOR
          51    0A A0      A0   098C                  ADDW2   AST$Q_IOSB+6(R0),R1         ;TOTAL CHARACTERS
       38 A0    51   B0  0990    871          MOVW     R1,AST$T_BUF+RDP$T_TT_RDATA(R0) ;SAVE IN LINK MESSAGE
          51    02   A0  0994    872          ADDW2    #2,R1                          ;SIZE OF READ DATA PLUS COUNT
                13   11  0997    873          BRB      50$
                         0999    874 30$:
          02    57   91  0999    875          CMPB     R7,#OP_WRITE
                04   12  099C    876          BNEQ     40$
                51   D4  099E    877          CLRL     R1                             ;NO READ DATA
                0A   11  09A0    878          BRB      50$
                         09A2    879 40$:
                         09A2    880 ; SETMODE OR SENSEMODE
                51   D4  09A2    881          CLRL     R1                             ;ASSUME NO DATA
          03    57   91  09A4    882          CMPB     R7,#OP_SETMODE
                03   13  09A7    883          BEQL     50$
          51    0C   D0  09A9    884          MOVL     #12,R1                         ;12 BYTES OF DATA
    26 A0     FFFE 8F   B0  09AC    885 50$:     MOVW     #RDP$C_END,AST$T BUF+RDP$W_OPCODE(R0)   ;A STATUS MESSAGE
             28 A0   B4  09B2    886          CLRW     AST$T_BUF+RDP$W_MOD(R0)
     30 A0   04 A0   7D  09B5    887          MOVQ     AST$Q_IOSB(R0),AST$T_BUF+RDP$Q_STATUS(R0)
          51    12   A0  09BA    888          ADDW2    #RDP$T TT RDATA,R1
       52    26 A0   9E  09BD    889          MOVAB    AST$T_BUF(R0),R2                      ; SET ADDRESS OF WRITE MESSAGE
                         09C1    890 60$:
                6D   10  09C1    891          BSBB     WRITE_TO_NETX
                         09C3    892 70$:
                     04  09C3    893          RET
                         09C4    894
```

```
                          09C4         .SBTTL WRITE_TO_NET - WRITE TO LINK
                          09C4   ;
                          09C4   ;
                          09C4   ; INPUTS:
                          09C4   ;      R0 - AST BLOCK
                          09C4   ;      R1 - length of message
                          09C4   ;      R2 - address of message
                          09C4   ;      R3 - AST routine to call (if called at WRITE_TO_NET)
                          09C4   ;
                          09C4
                          09C4   WRITE_TO_NET_SYNC::      ; R0,R1,R2 inputs
                          09C4
               53    DD   09C4          PUSHL   R3                          ; Save R3
               53    D4   09C6          CLRL    R3                          ; No AST
               50    DD   09C8          PUSHL   R0                          ; Save block
               70    10   09CA          BSBB    WRITE_TO_NET                ; Write to net
                          09CC          $WAITFR_S EFN = #RT$C_LINKEFN       ; Wait for completion
                          09D5          ONERROR QUIT                        ; Failure?
               50  8ED0   09FD          POPL    R0                          ; Restore AST block
      50    04 A0    3C   0A00          MOVZWL  AST$Q_IOSB(R0),R0           ; Fetch status
                          0A04          ONERROR QUIT                        ; Failure?
               53  8ED0   0A2C          POPL    R3                          ; Restore
                     05   0A2F          RSB                                 ; Return
                          0A30
                          0A30   WRITE_TO_NETX::                            ; Standard completion routine
  60   00000A9F'EF   9E   0A30          MOVAB   LNKWRTDONE,AST$L_STATE(R0)        ; New state
     53    F7E9 CF   9E   0A37          MOVAB   ASTHANDLER,R3               ; AST routine
                          0A3C
                          0A3C   WRITE_TO_NET::
                          0A3C
                          0A3C   .IF DF ASSEM_TRACE
               01    E0   0A3C          BBS     #RTLOG$V_TRACE,-
      4D 0000'CF          0A3E                  W^RTLOG_FLAGS,TRACE_WRITE        ; branch if tracing
                          0A42   TRACE_CONTINUE2:
                          0A42   .ENDC
                          0A42          $QIO_S  CHAN = LINKCHAN -         ;WRITE MESSAGE ON LINK
                          0A42                  EFN  = #RT$C_LINKEFN,-
                          0A42                  FUNC = #IO$_WRITEVBLK -
                          0A42                  IOSB = AST$Q_IOSB(R0) -
                          0A42                  ASTADR = (R3) -
                          0A42                  ASTPRM = R0 -
                          0A42                  P1 = (R2) -
                          0A42                  P2 = R1
                          0A66          ONERROR QUIT
                     05   0A8E          RSB
                          0A8F
                          0A8F   .IF DF ASSEM_TRACE
                          0A8F   TRACE_WRITE:
               3F    BB   0A8F          PUSHR   #^M<R0,R1,R2,R3,R4,R5>  ; Trace code
  53    00000000'EF  9E   0A91          MOVAB   DBG$LINKWRITE,R3
            F565'    30   0A96          BSBW    DBG$TRACE_IO
               3F    B^   0A9B          POPR    #^M<R0,R1,R2,R3,R4,R5>
               A3    1.   0A9D          BRB     TRACE_CONTINUE2
                          0A9F   .ENDC
```

```
                             0A9F   951          .SBTTL LNKWRTDONE - A WRITE TO THE LINK HAS COMPLETED
                             0A9F   952 ;++
                             0A9F   953 ; FUNCTIONAL DESCRIPTION:
                             0A9F   954 ;
                             0A9F   955 ;        HANDLE THE AST INDICATING THAT A WRITE HAS COMPLETED ON THE LINK
                             0A9F   956 ;        BY FREEING THE BUFFER.
                             0A9F   957 ;
                             0A9F   958 ; CALLING SEQUENCE:
                             0A9F   959 ;
                             0A9F   960 ;        CALLS   #0,LNKWRTDONE
                             0A9F   961 ;
                             0A9F   962 ; INPUT PARAMETERS:
                             0A9F   963 ;
                             0A9F   964 ;        R0 POINTS TO AN AST CONTROL BLOCK
                             0A9F   965 ;
                             0A9F   966 ; IMPLICIT INPUTS:
                             0A9F   967 ;
                             0A9F   968 ;        NONE
                             0A9F   969 ;
                             0A9F   970 ; OUTPUT PARAMETERS:
                             0A9F   971 ;
                             0A9F   972 ;        NONE
                             0A9F   973 ;
                             0A9F   974 ; IMPLICIT OUTPUTS:
                             0A9F   975 ;
                             0A9F   976 ;        NONE
                             0A9F   977 ;
                             0A9F   978 ; COMPLETION CC^ES:
                             0A9F   979 ;
                             0A9F   980 ;
                             0A9F   981 ; SIDE EFFECTS:
                             0A9F   982 ;
                             0A9F   983 ;        A BUFFER IS FREED
                             0A9F   984 ;        THE QUEUE OF PENDING QIO'S IS CHECKED
                             0A9F   985 ;
                             0A9F   986 ;--
                             0A9F   987 ;
                             0A9F   988 LNKWRTDONE::
                       0000  0A9F   989         .WORD   0
          51 04 A0       E8  0AA1   990         BLBS    AST$Q_IOSB(R0),20$              ;ERROR ON LINK WRITE ??
         52   04 A0      3C  0AA5   991         MOVZWL  AST$Q_IOSB(R0),R2               ;SAVE ERROR STATUS
       0000'8F    52     B1  0AA9   992         CMPW    R2,#SS$_ABORT
                  01     12  0AAE   993         BNEQ    10$
                         04  0AB0   994         RET                             ;LINK BROKE - LINK MBX WILL GET REASON
                             0AB1   995 10$:    $PUTMSG_S MSGVEC = DECNETERR    ;LINK ERROR
                             0AC4   996         QUIT    R2
                             0AE9   997
   06 00000000'EF        E9  0AE9   998         BLBC    CTERM_FLAG,20$
               F50D'     30  0AF0   999         BSBW    CTERM_LNKWRTDONE
                0042     31  0AF3  1000         BRW     40$
                             0AF6  1001
        53    2A A0      D0  0AF6  1002 20$:    MOVL    AST$T_B''F+RDP$L_REFID(R0),R3   ;SAVE ID
                0562     30  0AFA  1003         BSBW    BUFFREE         ;RELEASE THE BUFFER
    000005A7'EF   53     D1  0AFD  1004         CMPL    R3,READQIO      ;WAS THIS A READ (OR SET MODE)
                  15     12  0B04  1005         BNEQ    30$             ;NO
       000005A7'EF       D4  0B06  1006         CLRL    READQIO         ;FORGET ABOUT THE PREVIOUS ONE
 50    000005AF'FF       0F  ^B0C  1007         REMQUE  @READQ,R0
```

```
                  06   1D  0B13  1008            BVS     30$              ;NO MORE
       F715 CF    00   FB  0B15  1009            CALLS   #0,PROCMSG       ;GO PROCESS IT
                       04  0B1A  1010            RET
   000005AB'EF    53   D1  0B1B  1011 30$:       CMPL    R3,WRITEQIO      ;WAS THIS A WRITE
                  14   12  0B22  1012            BNEQ    40$              ;NO
        000005AB'EF        D4  0B24  1013        CLRL    WRITEQIO         ;FORGET ABOUT THE PREVIOUS ONE
   50   000005B7'FF        0F  0B2A  1014        REMQUE  @WRITEQ,R0
                  05   1D  0B31  1015            BVS     40$              ;NO MORE
       F6F7 CF    00   FB  0B33  1016            CALLS   #0,PROCMSG       ;GO PROCESS IT
                       04  0B38  1017 40$:       RET
```

```
                           0B39          .SBTTL LNKMBXDONE - MESSAGE RECEIVED ON THE LINK MAILBOX
                           0B39   ;++
                           0B39   ; FUNCTIONAL DESCRIPTION:
                           0B39   ;
                           0B39   ;       HANDLE THE AST INDICATING THAT A MESSAGE WAS RECEIVED ON THE LINK
                           0B39   ;       MAILBOX
                           0B39   ;
                           0B39   ; CALLING SEQUENCE:
                           0B39   ;
                           0B39   ;       CALLS   #0,LNKMBXDONE
                           0B39   ;
                           0B39   ; INPUT PARAMETERS:
                           0B39   ;
                           0B39   ;       R0 POINTS TO AN AST CONTROL BLOCK
                           0B39   ;
                           0B39   ; IMPLICIT INPUTS:
                           0B39   ;
                           0B39   ;       NONE
                           0B39   ;
                           0B39   ; OUTPUT PARAMETERS:
                           0B39   ;
                           0B39   ;       NONE
                           0B39   ;
                           0B39   ; IMPLICIT OUTPUTS:
                           0B39   ;
                           0B39   ;       NONE
                           0B39   ;
                           0B39   ; COMPLETION CODES:
                           0B39   ;
                           0B39   ;
                           0B39   ; SIDE EFFECTS:
                           0B39   ;
                           0B39   ;       THE PROGRAM CAN BE ABORTED.
                           0B39   ;
                           0B39   ;--
                           0B39   ;
                           0B39   LNKMBXDONE:
                   0004    0B39           .WORD   ^M<R2>
                           0B3B           ;
                           0B3B           ; First, check valid disconnects
                           0B3B           ;
    52   26 A0      3C     0B3B           MOVZWL  AST$T_BUF(R0),R2          ; Fetch MSG code
 0000'8F     52     B1     0B3F           CMPW    R2,#MSG$_DISCON           ; Disconnect?
             0E     13     0B44           BEQL    10$                       ; Branch if yes
 0000'8F     52     B1     0B46           CMPW    R2,#MSG$_EXIT             ; Exit?
             07     13     0B4B           BEQL    10$                       ; Branch if yes
 0000'8F     52     B1     0B4D           CMPW    R2,#MSG$_ABORT            ; Log out - ignore it
             29     12     0B52           BNEQ    20$                       ; Not a valid shutdown message...
                           0B54   10$:
                           0B54           QUIT    #SS$_NORMAL               ; Exit, no status message
                           0B7D           ;
                           0B7D           ; Either a serious error or something like a CONFIRM,
                           0B7D           ; which isn't important.
                           0B7D           ;
                           0B7D   20$:
 0000'8F     52     B1     0B7D           CMPW    R2,#MSG$_THIRDPARTY       ; Third party disconnect?
             29     12     0B82           BNEQ    30$                       ; Branch if not
```

```
                          0B84          QUIT    #SS$_THIRDPARTY          ; Abort program
                          0BAD     30$:
   0000'8F    52    B1    0BAD          CMPW    R2,#MSG$_PATHLOST        ; Path lost?
              29    12    0BB2          BNEQ    40$                      ; Branch if not
                          0BB4          QUIT    #SS$_PATHLOST            ; Abort program
                          0BDD          ;
                          0BDD          ; Unimportant message, just requeue read
                          0BDD          ;
                          0BDD     40$:
                          0BDD          $QIO_S  CHAN = MAILCHAN -        ;LINK MAILBOX READ
                          0BDD                  FUNC = #IO$_READVBLK -
                          0BDD                  IOSB = LINKMAIL+AST$Q_IOSB -
                          0BDD                  ASTADR = ASTHANDLER -
                          0BDD                  ASTPRM = #LINKMAIL -
                          0BDD                  P1 = LINKMAIL+AST$T_BUF -
                          0BDD                  P2 = #40
                          0C0E          ONERROR QUIT
                    04    0C36          RET
```

```
                                     0C37         .SBTTL OUTBANDAST - OUT OF BAND CHARACTER AST RECEIVED
                                     0C37   ;++
                                     0C37   ; FUNCTIONAL DESCRIPTION:
                                     0C37   ;
                                     0C37   ;         HANDLES THE AST RESULTING FROM AN OUT OF BAND CHARACTER
                                     0C37   ;
                                     0C37   ; CALLING SEQUENCE:
                                     0C37   ;
                                     0C37   ;         CALLS   #0,OUTBANDAST
                                     0C37   ;
                                     0C37   ; INPUT PARAMETERS:
                                     0C37   ;
                                     0C37   ;         NONE
                                     0C37   ;
                                     0C37   ; IMPLICIT INPUTS:
                                     0C37   ;
                                     0C37   ;         AST PARAMETER - CHARACTER
                                     0C37   ;
                                     0C37   ; OUTPUT PARAMETERS:
                                     0C37   ;
                                     0C37   ;         NONE
                                     0C37   ;
                                     0C37   ; IMPLICIT OUTPUTS:
                                     0C37   ;
                                     0C37   ;         RETSTATUS
                                     0C37   ;
                                     0C37   ; COMPLETION CODES:
                                     0C37   ;
                                     0C37   ;
                                     0C37   ; SIDE EFFECTS:
                                     0C37   ;
                                     0C37   ;         A MESSAGE SENT ON LINK
                                     0C37   ;
                                     0C37   ;--
                                     0C37   ;
                                     0C37   OUTBANDAST:
                                0000 0C37         .WORD   0
      0000056C'EF    04 AC        90 0C39         MOVB    4(AP),OUTBANDCHAR        ;GET THE CHARACTER THAT CAUSED THE AST
    00000562'EF    FFFF 8F        B0 0C41         MOVW    #RDP$C_ATTN,OUTBANDMSG+RDP$W_OPCODE
         00000564'EF    06        B0 0C4A         MOVW    #RDP$C_TT_OUTBAND,OUTBANDMSG+RDP$W_MOD
   0000056A'EF   00000000'EF      B0 0C51         MOVW    TERMUNIT,OUTBANDMSG+RDP$W_UNIT
                                     0C5C         $QIO_S  CHAN = LINKCHAN -        ;SEND ON LINK
                                     0C5C                 FUNC = #IO$_WRITEVBLK -
                                     0C5C                 P1 = OUTBANDMSG -
                                     0C5C                 P2 = #RDP$K_HEADERLEN+1
                                     0C81         ONERROR QUIT
                                  04 0CA9         RET
```

```
                                   0CAA          .SBTTL LINKGONE - TIMER EXPIRED SO LINK IS GONE
                                   0CAA   ;++
                                   0CAA   ; FUNCTIONAL DESCRIPTION:
                                   0CAA   ;
                                   0CAA   ;      HANDLES THE AST RESULTING FROM THE TIMER STARTED TO WAIT FOR THE
                                   0CAA   ;        MAILBOX TO GET THE REASON WHY THE LINK IS GONE
                                   0CAA   ;
                                   0CAA   ; CALLING SEQUENCE:
                                   0CAA   ;
                                   0CAA   ;      CALLS   #0,LINKGONE
                                   0CAA   ;
                                   0CAA   ; INPUT PARAMETERS:
                                   0CAA   ;
                                   0CAA   ;      NONE
                                   0CAA   ;
                                   0CAA   ; IMPLICIT INPUTS:
                                   0CAA   ;
                                   0CAA   ;      LAST LINK MAILBOX MESSAGE
                                   0CAA   ;
                                   0CAA   ; OUTPUT PARAMETERS:
                                   0CAA   ;
                                   0CAA   ;      NONE
                                   0CAA   ;
                                   0CAA   ; IMPLICIT OUTPUTS:
                                   0CAA   ;
                                   0CAA   ;      RETSTATUS
                                   0CAA   ;
                                   0CAA   ; COMPLETION CODES:
                                   0CAA   ;
                                   0CAA   ;
                                   0CAA   ; SIDE EFFECTS:
                                   0CAA   ;
                                   0CAA   ;      A $WAKE WILL FORCE THE PROGRAM TO EXIT
                                   0CAA   ;
                                   0CAA   ;--
                                   0CAA   ;
                                   0CAA   LINKGONE:
                             0000  0CAA          .WORD   0
00000593'EF   0000005C'EF      3C  0CAC          MOVZWL  LINKMAIL+AST$T_BUF,MBXMSGTYP    ;GET REASON FROM THE LAST MAILBOX ME
                                   0CB7          $PUTMSG_S MSGVEC = MBXMSG               ;OUTPUT MESSAGE TYPE
                                   0CCA          QUIT
```

```
                              0CEF             .SBTTL UNSDATMBX - MESSAGE IN TERMINAL MAILBOX
                              0CEF   ;++
                              0CEF   ; FUNCTIONAL DESCRIPTION:
                              0CEF   ;
                              0CEF   ;        HANDLES THE AST RESULTING FROM UNSOLICITED TERMINAL DATA OR HANGUP
                              0CEF   ;
                              0CEF   ; CALLING SEQUENCE:
                              0CEF   ;
                              0CEF   ;        CALLS   #0,UNSDATMBX
                              0CEF   ;
                              0CEF   ; INPUT PARAMETERS:
                              0CEF   ;
                              0CEF   ;        NONE
                              0CEF   ;
                              0CEF   ; IMPLICIT INPUTS:
                              0CEF   ;
                              0CEF   ;        UNSDAT
                              0CEF   ;
                              0CEF   ; OUTPUT PARAMETERS:
                              0CEF   ;
                              0CEF   ;        NONE
                              0CEF   ;
                              0CEF   ; IMPLICIT OUTPUTS:
                              0CEF   ;
                              0CEF   ;        RETSTATUS
                              0CEF   ;
                              0CEF   ; COMPLETION CODES:
                              0CEF   ;
                              0CEF   ;
                              0CEF   ; SIDE EFFECTS:
                              0CEF   ;
                              0CEF   ;        A MESSAGE IS SENT ON THE LINK.  IF THIS QIO FAILS, A $WAKE FORCES THE
                              0CEF   ;        PROGRAM TO EXIT.
                              0CEF   ;
                              0CEF   ;--
                              0CEF   ;
                              0CEF   UNSDATMBX:
                       000C   0CEF             .WORD   ^M<R2,R3>
                              0CF1
            51   0A     D0    0CF1             MOVL    #RDP$K_HEADERLEN,R1        ; Assume we send this much
 52   00000144'EF       9E    0CF4             MOVAB   UNSDAT+AST$T_BUF,R2        ; base address of RDP
       53   0C A2       3C    0CFB             MOVZWL  RDP$K_HEADERLEN+2(R2),R3   ; Message code
     0000'8F     53     B1    0CFF             CMPW    R3,#MSG$_TRMUNSOLIC        ; Unsolicted data?
                 06     12    0D04             BNEQ    10$                        ; Branch if not
                 00     B0    0D06             MOVW    #RDP$C_TT_UNSOL,-
              02 A2           0D08                     RDP$W_MOD(R2)              ; Unsolicited data
                 23     11    0D0A             BRB     20$                        ; Back to common code
                              0D0C   10$:
     0000'8F     53     B1    0D0C             CMPW    R3,#MSG$_TRMBRDCST         ; Broadcast message?
                 11     12    0D11             BNEQ    18$                        ; branch if not
                              0D13
       51   20 A2       3C    0D13             MOVZWL  RDP$K_HEADERLEN+22(R2),R1  ; Length of broadcast
            51   22     C0    0D17             ADDL2   #RDP$K_HEADERLEN+24,R1     ; Add rest + header
         0A A2   51     B0    0D1A             MOVW    R1,RDP$K_HEADERLEN(R2)     ; Save it in msg
         02 A2   05     B0    0D1E             MOVW    #RDP$C_TT_BRDCST, -
                              0D22                     RDP$W_MOD(R2)              ; Broadcast
                 0B     11    0D22             BRB     20$
```

```
                              0D24  18$:
         0000'8F      53  B1  0D24         CMPW    R3,#MSG$_TRMHANGUP       ; Hangup?
                     25'  12  0D29         BNEQ    UNSMSGDONE+2             ; Ignore it
                      01  B0  0D2B         MOVW    #RDP$C_TT_HANGUP,-
                   02 A2      0D2D                 RDP$W_MOD(R2)            ; Hangup
                              0D2F  20$:
         62   FFFF 8F     B0  0D2F         MOVW    #RDP$C_ATTN,RDP$W_OPCODE(R2) ; message type
   08 A2  00000000'EF     B0  0D34         MOVW    TERMUNIT,RDP$W_UNIT(R2) ; Term unit num *** ?
                              0D3C
         50   011E'CF     9E  0D3C         MOVAB   W^UNSDAT,R0              ; Address of AST block
                 4E'AF    9E  0D41         MOVAB   B^UNSMSGDONE,-
                    60        0D44                 AST$L_STATE(R0)          ; Next state
         53   F4DB CF     9E  0D45         MOVAB   W^ASTHANDLER,R3          ; Ast address
                 FCEF     30  0D4A         BSBW    WRITE_TO_NET             ; Write message
                      04      0D4D         RET
                              0D4E
```

```
OP_
OP_
OUT
OUT
OUT
OUT
OUT
OUT
OUT
PRO
PRO
PRO
QIO
QIO
RAB
RAB
RAB
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
RDP
REA
REA
REA
REA
REC
REM
REM
REM
REM
REM
RET
RMS
```

```
                                 0D4E          .SBTTL UNSMSGDONE - DO A NEW TERMINAL MAILBOX READ
                                 0D4E    ;++
                                 0D4E    ; FUNCTIONAL DESCRIPTION:
                                 0D4E    ;
                                 0D4E    ;       WHEN THE WRITE TO THE LINK COMPLETES,DO A NEW TERMINAL MAILBOX READ.
                                 0D4E    ;
                                 0D4E    ; CALLING SEQUENCE:
                                 0D4E    ;
                                 0D4E    ;       CALLS   #0,UNSMSGDONE
                                 0D4E    ;
                                 0D4E    ; INPUT PARAMETERS:
                                 0D4E    ;
                                 0D4E    ;       NONE
                                 0D4E    ;
                                 0D4E    ; IMPLICIT INPUTS:
                                 0D4E    ;
                                 0D4E    ;       NONE
                                 0D4E    ;
                                 0D4E    ; OUTPUT PARAMETERS:
                                 0D4E    ;
                                 0D4E    ;       NONE
                                 0D4E    ;
                                 0D4E    ; IMPLICIT OUTPUTS:
                                 0D4E    ;
                                 0D4E    ;       NONE
                                 0D4E    ;
                                 0D4E    ; COMPLETION CODES:
                                 0D4E    ;
                                 0D4E    ;
                                 0D4E    ; SIDE EFFECTS:
                                 0D4E    ;
                                 0D4E    ;       IF THE MAILBOX READ QIO FAILS, A $WAKE IS ISSUED TO CAUSE THE PROGRAM
                                 0D4E    ;       TO EXIT
                                 0D4E    ;
                                 0D4E    ;--
                                 0D4E    ;
                                 0D4E    UNSMSGDONE:
                           0000  0D4E          .WORD   0
        0000011E'EF   9C AF   9E  0D50          MOVAB   UNSDATMBX,UNSDAT        ;NEW STATE
                                 0D58          $QIO_S  CHAN = TERMMBXCHAN -    ;SET UP UNSOLICITED DATA MBX READ
                                 0D58                  FUNC = #IO$_READVBLK -
                                 0D58                  IOSB = UNSDAT+AST$Q_IOSB -
                                 0D58                  ASTADR = ASTHANDLER -
                                 0D58                  ASTPRM = #UNSDAT -
                                 0D58                  P1 = UNSDAT+AST$T_BUF+RDP$K_HEADERLEN+2 -
                                 0D58                  P2 = #MAXMSG
                                 0D8D          ONERROR QUIT
                             04  0DB5          RET
```

```
VMS
Pse

PSE
---

$AB
RTP
_RT

Pha
---
Ini
Com
Pas
Sym
Pas
Sym
Pse
Cro
Ass

The
910
The
173
36

Mac
---
_$2
_$2
_$2
TOT

968

The

MAC
```

```
                                 0DB6                .SBTTL CNTRLC_AST - CONTROL-C & CONTROL-Y
                                 0DB6       ;++
                                 0DB6       ; FUNCT'ONAL DESCRIPTION:
                                 0DB6       ;
                                 0DB6       ;        HANDLE THE AST RESULTING FROM A CONTROL-C OR A CONTROL-Y
                                 0DB6       ;
                                 0DB6       ; CALLING SEQUENCE:
                                 0DB6       ;
                                 0DB6       ;        CALLS   #0,CNTRLC_AST
                                 0DB6       ;
                                 0DB6       ; INPUT PARAMETERS:
                                 0DB5       ;
                                 0DB6       ;        NONE
                                 0DB6       ;
                                 0DB6       ; IMPLICIT INPUTS:
                                 0DB6       ;
                                 0DB6       ;        CNTRLYTIM
                                 0DB6       ;
                                 0DB6       ; OUTPUT PARAMETERS:
                                 0DB6       ;
                                 0DB6       ;        NONE
                                 0DB6       ;
                                 0DB6       ; IMPLICIT OUTPUTS:
                                 0DB6       ;
                                 0DB6       ;        CNTRLYTIM
                                 0DB6       ;
                                 0DB6       ; COMPLETION CODES:
                                 0DB6       ;
                                 0DB6       ;
                                 0uB6       ; SIDE EFFECTS:
                                 CDB6       ;
                                 0DB6       ;        A MESSAGE IS SENT ON THE LINK AND FOR ^Y THE AST IS REENABLED.
                                 0DB6       ;        TWO QUICK (LESS THAN 3 SEC) ^Y'S WILL ABORT THIS PROGRAM.
                                 0DB6       ;--
                                 0DB6       ;
                           0000  0DB6       CNTRLC_AST::    .WORD   0
00000114'EF    FFFF 8F     B0    0DB8                MOVW    #RDP$C_ATTN,CNTRLCY+RDP$W_OPCODE
     0000'8F      04 AC    B1    0DC1                CMPW    4(AP),#IO$M_CTRLCAST
                     10    12    0DC7                BNEQ    10$
     00000116'EF     02    B0    0DC9                MOVW    #RDP$C_TT_CTRLC,CNTRLCY+RDP$W_MOD
                                 0DD0
           000005A6'EF     94    0DD0                CLRB    CNTRCFLAG                  ; NO CONTROL-C ENABLES
                   0083    31    0DD6                BRW     30$
     0000'8F      04 AC    B1    0DD9       10$:     CMPW    4(AP),#IO$M_CTRLYAST
                     26    13    0DDF                BEQL    20$
                                 0DE1                QUIT    4(AP)                      ; PROBABLY A HANGUP
                   0088    30    0E07       20$:     BSBW    CNTRLYTEST                 ; CHECK FOR RECENT ^Y
                                 0E0A                $QIO_S  CHAN = CNTRLCHAN -         ; RE-ENABLE IT
                                 0E0A                        FUNC = #IO$_SETMODE!IO$M_CTRLYAST -
                                 0E0A                        P1 = CNTRLC_AST -
                                 0E0A                        P2 = #IO$M_CTRLYAST
                                 0E30
                                 0E30       ; WE WILL START TIMING WAITING FOR A SECOND ONE
                                 0E30
           0000059B'EF     95    0E30                TSTB    CNTRLYTIM                  ; ONLY SET ONE TIMER AT A TIME
                     1D    12    0E36                BNEQ    25$                        ; SKIP IF TIMER ENABLED
           0000059B'EF     96    0E38                INCB    CNTRLYTIM                  ; INDICATE WE HAVE ONE ^Y
```

```
                                 0E3E  1365            $SETIMR_S DAYTIM = THREESEC -   ; TIME THREE SECONDS
                                 0E3E  1366                      ASTADR = YTIMEDONE
         00000116'EF    03   B0  0E55  1367 25$:       MOVW    #RDP$C_TT_CTRLY,CNTRLCY+RDP$W_MOD
                                 0E5C  1368 30$:
    0000011C'EF   00000000'EF   B0  0E5C  1369         MOVW    TERMUNIT,CNTRLCY+RDP$W_UNIT
                                 0E67  1370
          07 00000000'EF    E9  0E67  1371             BLBC    CTERM_FLAG,40$              ; Branch if VAX
            0000'CF    6C   FA  0E6E  1372             CALLG   (AP),@^CTERM_CTRL_CY        ; Notify cterm module
                       13   11  0E73  1373             BRB     50$                         ; Exit
                                 0E75  1374
         50   000000EE'EF   9E  0E75  1375 40$:        MOVAB   CNTRLCY_MSG,R0              ; AST block
                51   0A   D0  0E7C  1376               MOVL    #RDP$K_HEADERLEN,R1         ; and length
                                 0E7F  1377
            52    26 A0   9E  0E7F  1378               MOVAB   AST$T_BUF(R0),R2                  ; Address of message
                       53   D4  0E83  1379             CLRL    R3                          ; No ast
                     FBB4   30  0E85  1380             BSBW    WRITE_TO_NET                ; Write to net
                                 0E88  1381 50$:
                            04  0E88  1382             RET
                                 0E89  1383 ;
                                 0E89  1384 ;
                                 0E89  1385 ; ^Y TIMER HAS EXPIRED
                                 0E89  1386 ;
                                 0E89  1387 YTIMEDONE:
                          0000  0E89  1388             .WORD   0
         0000059B'EF        94  0E8B  1389             CLRB    CNTRLYTIM       ;NO RECENT ^Y
                            04  0E91  1390             RET
                                 0E92  1391 ;
                                 0E92  1392 ;
                                 0E92  1393 ; ASK ABORT QUESTION IF THERE WAS A RECENT ^Y
                                 0E92  1394 ;
                                 0E92  1395 CNTRLYTEST:
         0000059B'EF        95  0E92  1396             TSTB    CNTRLYTIM
                       01   12  0E98  1397             BNEQ    10$
                            05  0E9A  1398             RSB                     ;NO RECENT ^Y - CONTINUE
                                 0E9B  1399 10$:       $GETMSG_S MSGID = #REM$_CNTRLY -
                                 0E9B  1400                    MSGLEN = CNTRLYMSGBUF -
                                 0E9B  1401                    BUFADR = CNTRLYMSGBUF -
                                 0E9B  1402                    FLAGS = #1
                                 0EB8  1403            $FAO_S  CTRSTR = CNTRLYMSGBUF -
                                 0EB8  1404                    OUTLEN = CNTRLYQUESLEN -
                                 0EB8  1405                    OUTBUF = CNTRLYQUESBUF -
                                 0EB8  1406                    P1 = #REMOTENODE
                                 0ED7  1407            $QIOW_S CHAN = READCHAN -       ;ASK ABOUT THE ^Y JUST TYPED
                                 0ED7  1408                    EFN = #1 -
                                 0ED7  1409                    FUNC = #IO$_READPROMPT!IO$M_CVTLOW -
                                 0ED7  1410                    P1 = ANSBUF -
                                 0ED7  1411                    P2 = #10 -
                                 0ED7  1412                    P5 = #CNTRLYQUES -
                                 0ED7  1413                    P6 = CNTRLYQUESLEN
      59 8F   0000059C'EF   91  0F06  1414             CMPB    ANSBUF,#^A/Y/           ;DID HE SAY YES
                       01   13  0F0E  1415             BEQL    20$
                            05  0F10  1416             RSB                     ;HE SAID NO - SO CONTINUE
                                 0F11  1417 ; ABORT
                                 0F11  1418 20$:       QUIT    #SS$_NORMAL     ;NO STATUS MESSAGE
```

```
                                0F3A  1420          .SBTTL VMS_INDREAD - READ INDIRECT COMMAND FILE
                                0F3A  1421 ;++
                                0F3A  1422 ; FUNCTIONAL DESCRIPTION:
                                0F3A  1423 ;
                                0F3A  1424 ;        READS FROM AN INDIRECT COMMAND FILE
                                0F3A  1425 ;
                                0F3A  1426 ; CALLING SEQUENCE:
                                0F3A  1427 ;
                                0F3A  1428 ;        JSB     VMS_INDREAD
                                0F3A  1429 ;
                                0F3A  1430 ; INPUT PARAMETERS:
                                0F3A  1431 ;
                                0F3A  1432 ;        R0 - AST Block
                                0F3A  1433 ;        R1 - Address to put input data
                                0F3A  1434 ;        R2 - QIO read modifiers (only IO$M_CVTLOW checked for)
                                0F3A  1435 ;        R3 - Size of request in bytes
                                0F3A  1436 ;
                                0F3A  1437 ; IMPLICIT INPUTS:
                                0F3A  1438 ;
                                0F3A  1439 ;        SYSINRAB
                                0F3A  1440 ;
                                0F3A  1441 ; OUTPUT PARAMETERS:
                                0F3A  1442 ;
                                0F3A  1443 ;        NONE
                                0F3A  1444 ;
                                0F3A  1445 ; IMPLICIT OUTPUTS:
                                0F3A  1446 ;
                                0F3A  1447 ;        INDFLAG
                                0F3A  1448 ;
                                0F3A  1449 ; COMPLETION CODES:
                                0F3A  1450 ;
                                0F3A  1451 ; SIDE EFFECTS:
                                0F3A  1452 ;
                                0F3A  1453 ;        ON AN EOF, FURTHER READS FROM THE INDIRECT FILE ARE DISABLED.
                                0F3A  1454 ;        THE PROGRAM WILL EXIT ON AN INDIRECT FILE READ.
                                0F3A  1455 ;
                                0F3A  1456 ;--
                                0F3A  1457 ;
                                0F3A  1458 VMS_INDREAD::
                                0F3A  1459
            0043 8F       BB    0F3A  1460          PUSHR   #^M<R0,R1,R6>
 00000000'EF      51      D0    0F3E  1461          MOVL    R1,SYSINRAB+RAB$L_UBF     ; Data address
 00000000'EF      53      B0    0F45  1462          MOVW    R3,SYSINRAB+RAB$W_USZ     ; Requested size
                                0F4C  1463          $GET    RAB = SYSINRAB            ; Read a record
 00000000'8F      50      D1    0F59  1464          CMPL    R0,#RMS$_NORMAL
                  69      12    0F60  1465          BNEQ    10$                       ; Problem
            0043 8F       BA    0F62  1466          POPR    #^M<R0,R1,R6>
              04 A0       7C    0F66  1467          CLRQ    AST$Q_IOSB(R0)
 51    00000000'EF        3C    0F69  1468          MOVZWL  SYSINRAB+RAB$W_RSZ,R1
       00000000'EF        B0    0F70  1469          MOVW    SYSINRAB+RAB$W_RSZ,-
              06 A0             0F76  1470                  AST$Q_IOSB+2(R0)          ; Size of read
 51    00000000'EF        C0    0F78  1471          ADDL    SYSINRAB+RAB$L_UBF,R1     ; Find where to put terminator
              61   0D     90    0F7F  1472          MOVB    #^X0D,(R1)
           08 A0   0D     90    0F82  1473          MOVB    #^X0D,AST$Q_IOSB+4(R0)    ; Set <CR> as terminator
           0A A0   01     B0    0F86  1474          MOVW    #1,AST$Q_IOSB+6(R0)       ; Terminator size
  04 A0    0000'8F        B0    0F8A  1475          MOVW    #SS$_NORMAL,AST$Q_IOSB(R0)
 52    00000000'8F        D3    0F90  1476          BITL    #IO$M_CVTLOW,R2
```

```
EXE


Mod
---
NDX
NDX
NDX
IND
PER
NDX
NDX
POC
NDX
NDX
NDX
NDX
NDX
PAC
NDX
NDX
PAC
PAC
CON
CON
CON
XPC
SFA
XME
XCL
XFA
XGE
XZA
XPM
XOP
XPA
XPL
XVM
XTE
XVA
SAP
SAS
SBI
SCC
STR
STE
SMS
SJC
XRC
SSC
SVA
CLI
CLI
LIE
```

```
                 22   13  0F97  1477          BEQL    9$                              ;NO CASE CONVERSION
   56   0C000000'EF   D0  0F99  1478          MOVL    SYSINRAB+RAB$L_UBF,R6           ;BUFFER ADDRESS
   51   00000000'EF   3C  0FA0  1479          MOVZWL  SYSINRAB+RAB$W_RSZ,R1           ;CHARACTERS TO CHECK
          61 8F   66  91  0FA7  1480 5$:      CMPB    (R6),#^A/a/
                  09  19  0FAB  1481          BLSS    8$                              ;NOT LOWER CASE
          7A 8F   66  91  0FAD  1482          CMPB    (R6),#^A/z/
                  03  14  0FB1  1483          BGTR    8$                              ;NOT LOWER CASE
              66  20  82  0FB3  1484          SUBB    #^X20,(R6)                      ;MAKE IT UPPER CASE
                  56  D6  0FB6  1485 8$:      INCL    R6                              ;NEXT
             EC   51  F5  0FB8  1486          SOBGTR  R1,5$
                          0FBB  1487 9$:      $DCLAST_S ASTADR = ASTHANDLER, -        ;SIMULATE A COMPETION AST
                          0FBB  1488                    ASTPRM = R0
                      04  0FCA  1489          RET
   00000000'8F    50  D1  0FCB  1490 10$:     CMPL    R0,#RMS$_EOF            ;ARE WE JUST DONE WITH THE FILE
                  18  12  0FD2  1491          BNEQ    20$                     ;REAL PROBLEM
                          0FD4  1492          $CLOSE  FAB = SYSINFAB          ;DON'T NEED IT
        00000000'EF   94  0FE1  1493          CLRB    INDFLAG
            0043 8F   BA  0FE7  1494          POPR    #^M<R0,R1,R6>
                      05  0FEB  1495          RSB                             ;GO DO THE REAL QIO
                          0FEC  1496 20$:     QUIT
```

```
                                1011         .SBTTL GETBUF - GET A BUFFER
                                1011  ;++
                                1011  ; FUNCTIONAL DESCRIPTION:
                                1011  ;
                                1011  ;      GET A FREE BUFFER OR ALLOCATE ONE IF THERE ARE NONE.
                                1011  ;
                                1011  ; CALLING SEQUENCE:
                                1011  ;
                                1011  ;      JSB     GETBUF
                                1011  ;
                                1011  ; INPUT PARAMETERS:
                                1011  ;
                                1011  ;      NONE
                                1011  ;
                                1011  ; IMPLICIT INPUTS:
                                1011  ;
                                1011  ;      BUFQUEUE
                                1011  ;
                                1011  ; OUTPUT PARAMETERS:
                                1011  ;
                                1011  ;      R0 POINTS TO THE BUFFER
                                1011  ;
                                1011  ; IMPLICIT OUTPUTS:
                                1011  ;
                                1011  ;      NONE
                                1011  ;
                                1011  ; COMPLETION CODES:
                                1011  ;
                                1011  ;
                                1011  ; SIDE EFFECTS:
                                1011  ;
                                1011  ;      MORE VIRTUAL MEMORY MAY BE ALLOCATED
                                1011  ;
                                1011  ;--
                                1011  ;
                                1011  GETBUF::
  50    00000026'FF     0F      1011         REMQUE  @BUFQUEUE,R0      ;GET A BUFFER
                 01     1D      1018         BVS     10$               ; BRANCH IF NONE
                                101A  5$:
                        05      101A         RSB                       ; RETURN
                                101B
                                101B  ; ALLOCATE A BUFFER
                                101B
        0000002E'EF     9F      101B  10$:   PUSHAB  BUFADR            ;BUFFER ADDRESS WILL BE RETURNED HERE
        00000032'EF     9F      1021         PUSHAB  BUFSIZE           ;REQUESTED SIZE
  00000000'GF    02     FB      1027         CALLS   #2,G^LIB$GET_VM
                                102E         ONERROR QUIT
                                1056
                                1056  .IF DF debug
                                1056         movab   gotvm,R0          ; set message address
                                1056         bsbw    log_ascic         ; log message
                                1056  .endc
  50    0000002E'EF     D0      1056         MOVL    BUFADR,R0
                 BB     11      105D         BRB     5$                ; BRANCH TO EXIT
                                105F
                                105F        .SBTTL BUFFREE - FREE A BUFFER
                                105F  ;++
                                105F  ; FUNCTIONAL DESCRIPTION:
                                105F  ;
                                105F  ;       FREE A BUFFER.
                                105F  ;
                                105F  ; CALLING SEQUENCE:
                                105F  ;
                                105F  ;       JSB     BUFFREE
                                105F  ;
                                105F  ; INPUT PARAMETERS:
                                105F  ;
                                105F  ;       R0 POINTS TO THE BUFFER
                                105F  ;
                                105F  ; IMPLICIT INPUTS:
                                105F  ;
                                105F  ;       BUFQUEUE
                                105F  ;
                                105F  ; OUTPUT PARAMETERS:
                                105F  ;
                                105F  ;       NONE
                                105F  ;
                                105F  ; IMPLICIT OUTPUTS:
                                105F  ;
                                105F  ;       NONE
                                105F  ;
                                105F  ; COMPLETION CODES:
                                105F  ;
                                105F  ;
                                105F  ; SIDE EFFECTS:
                                105F  ,
                                105F  ;       NONE
                                105F  ;
                                105F  ;--
                                105F  ;
                                105F  BUFFREE::
      0C000026'EF    60    0E   105F        INSQUE  (R0),BUFQUEUE        ;PUT BUFFER IN FREE LIST
                           05   1066        RSB
```

```
                              1067  .SBTTL  READ ONLY DATA
                              1067
                       000F'  1067  TERMOPS:      .WORD   <2$-1$>/4
                  0001 0000'  1069  1$:           .WORD   IO$_READVBLK,OP_READ
                  0001 0000'  106C                .WORD   IO$_READLBLK,OP_READ
                  0001 0000'  1071                .WORD   IO$_READPBLK,OP_READ
                  0101 0000'  1075                .WORD   IO$_READPROMPT,OP_READ!OP_PRMPT
                  0001 0000'  1079                .WORD   IO$_TTYREADALL,OP_READ
                  0101 0000'  107D                .WORD   IO$_TTYREADPALL,OP_READ!OP_PRMPT
                  0002 0000'  1081                .WORD   IO$_WRITEVBLK,OP_WRITE
                  0002 0000'  1085                .WORD   IO$_WRITELBLK,OP_WRITE
                  0002 0000'  1089                .WORD   IO$_WRITEPBLK,OP_WRITE
                  0003 0000'  108D                .WORD   IO$_SETMODE,OP_SETMODE
                  0003 0000'  1091                .WORD   IO$_SETCHAR,OP_SETMODE
                  0004 0000'  1095                .WORD   IO$_SENSEMODE,OP_SENSEMODE
                  0004 0000'  1099                .WORD   IO$_SENSECHAR,OP_SENSEMODE
                  0005 0000'  109D                .WORD   IO$_ACPCONTROL,OP_CANCEL
                  0006 FFFF   10A1                .WORD   -1,OP_BRDCST
                              10A5  2$:
                              10A5
 00000000 00000000'00000002   10A5  ACSIGNORE: .LONG  2,REM$_ACSIGN,0
                              10B1
 00000000 00000000'00000002   10B1  BADOUTBAND: .LONG 2,REM$_BADOUTBAND,0
                              10BD
          FFFFFFFF FE363C80   10BD  THREESEC:  .LONG  -10*1000*1000*3,-1            ;THREE SECOND TIMER
                              10C5
```

```
                     10C5          .SBTTL  READ WRITE DATA
                 00000000          .PSECT  _RTPAD,LONG
                     0000
          0000000C   0000 CHARBUF:         .BLKB   12
                     000C
              0000   000C OUTBANDINC::  .WORD  0           ;OUT OF BAND (INCLUDE) AST CHANNEL
              0000   000E OUTBANDEXC::  .WORD  0           ;OUT OF BAND (EXCLUDE) AST CHANNEL
              0000   0010 OUTBANDABO::  .WORD  0           ;OUT OF BAND (ABORT)   AST CHANNEL
                     0012
          00000000   0012 CTRLO_CHAN:      .LONG   0       ; CHANNEL FOR ^O OUT OF BANDS
          00000000   0016 CTRLO_MASK:      .LONG   0
          00008000   001A                  .LONG   1@<^A/O/-^A/@/>
                     001E
          00000026   001E RTERMDES:  .BLKL  2              ;TERMINATOR CHARACTER DESCRIPTOR
                     0026
 00000026'00000026'  0026 BUFQUEUE:  .LONG  BUFQUEUE,BUFQUEUE     ;EMPTY BUFFER QUEUE
                     002E
          00000000   002E BUFADR:    .LONG  0              ;THE ADDRESS OF AN ALLOCATED BUFFER GOES HERE
                     0032
          00000440   0032 BUFSIZE:   .LONG  AST$T_BUF+MAXMSG      ;BUFFER SIZE
                     0036
          00000B39'  0036 LINKMAIL:        .LONG   LNKMBXDONE
          00000084   003A                  .BLKB   AST$T_BUF+40-4
```

```
                          0084  .SBTTL  Protocol Message buffers
                          0084
                          0084  ;
                          0084  ; BIND and configuration data message
                          0084  ;
                          0084
                000000AA  0084  CONFIG_MSG:       .BLKB   AST$T_BUF         ;CONFIGURATION MESSAGE
             00 01 01 01  00AA  CONFIG_MSG_ST:    .BYTE   1,1,1,0           ; BIND,V1, ECO=1,customer ECO
               0004 0007  00AE                    .WORD   7,4               ; VMS, support mask
                          00B2  CONFIG_CHAR:
                000000BE  00B2                    .BLKL   3                 ; Characteristics buffer
                00000014  00BE  CONFIG_MSGLEN = .-CONFIG_MSG_ST
                          00BE
                          00BE  ;
                          00BE  ; Unsolicited data message (init)
                          00BE  ;
                          00BE
                000000E4  00BE  INIT_MSG:         .BLKB   AST$T_BUF
                    FFFF  00E4  1$:               .WORD   RDP$C_ATTN
     0000 0000 0000 0000  00E6                    .WORD   RDP$C_TT_UNSOL,0,0,0    ; Fake unsolicited data
                0000000A  00EE  INIT_MSGLEN = .-1$
                          00EE
                          00EE  ;
                          00EE  ; CONTROL C or CONTROL Y (^C or ^Y) out of band message
                          00EE  ;
                          00EE
                00000114  00EE  CNTRLCY_MSG:      .BLKB   AST$T_BUF
                0000011E  0114  CNTRLCY:          .BLKB  RDP$K_HEADERLEN  ; LINK MESSAGE FOR CONTROL C OR Y
                0000000A  011E  CNTRLCY_MSGLEN = .-CNTRLCY
                          011E
               00000CEF'  011E  UNSDAT::          .LONG  UNSDATMBX        ; VMS AND CTERMRT ONLY???***
                00000562  0122                    .BLKB  AST$T_BUF+MAXMSG ; FOR UNSOLICITED DATA MAILBOX
                          0562
                          0562  ;
                          0562  ; Out of band message
                          0562  ;
                          0562
                0000056C  0562  OUTBANDMSG: .BLKB  RDP$K_HEADERLEN
                      00  056C  OUTBANDCHAR: .BYTE 0
                          056D  ;
                          056D  ; Cancel message
                          056D  ;
                          056D
       00000000 FFFFFFFE  056D  CANMSG:     .LONG  RDP$C_END,0
                    0000  0575              .WORD  0
                0000057F  0577              .BLKQ  1
                          057F  ;
                00000587  057F  BRDDESC:    .BLKL  2               ;DESCRIPTOR FOR BROADCASTS
                          0587  ;
00000001 00000000'00000004  0587  MBXMSG:           .LONG  4,REM$_NETMBX,1
                00000000  0593  MBXMSGTYP:        .LONG  0
                00000000  0597  LINKERR:          .LONG   0
                          059B  ;
                      00  059B  CNTRLYTIM: .BYTE  0               ;RECENT ^Y INDICATOR
                          059C  ;
                000005A6  059C  ANSBUF:     .BLKB  10              ;RECEIVE ANSWER TO ^Y QUESTION
                          05A6  ;
```

```
_$2



Sym
---
$IN
ALL
ASG
BOO
BUC
CHR
CHR
CHR
CLI
CLI
CLI
CLI
CLI
CLI
CLI
CLI
CLI
CLI
CLI
CLI
CLI
CLI
CMD

CON
CON
CON
CPO
DMP
DSP
DSR


DSR


DSR


DSR


DSR


DSR


DSR
```

```
                  00  05A6  1701 CNTRCFLAG:: .BYTE  0            ;INDICATE CONTROL-C ENABLES
                      05A7  1702
            00000000  05A7  1703 READQIO::  .LONG   0            ;ID OF CURRENT READ REQUEST
                      05AB  1704 ;
            00000000  05AB  1705 WRITEQIO:: .LONG   0            ;ID OF CURRENT WRITE REQUEST
                      05AF  1706 ;
 000005AF'000005AF'  05AF  1707 READQ::    .LONG   READQ,READQ  ;QUEUE OF PENDING READS
                      05B7  1708 ;
 000005B7'000005B7'  05B7  1709 WRITEQ::   .LONG   WRITEQ,WRITEQ        ;QUEUE OF PENDING WRITES
                      05BF  1710 ;
            000005C7  05BF  1711 REMOTENODE::    .BLKL   2              ;REMOTE NODE NAME
                      05C7  1712 ;
                      05C7  1713
                      05C7  1714 ;
            000005CF  05C7  1715 INCMASK:   .BLKL  2             ;OUT OF BAND INCLUDE MASK
            000005D7  05CF  1716 EXCMASK:   .BLKL  2             ;OUT OF BAND EXCLUDE MASK
                      05D7  1717 ;
               0000'  05D7  1718 FIRST_READ: .WORD IO$M_PURGE    ; Don't purge type ahead (on the first read)
                      05D9  1719 ;
                      05D9  1720 CNTRLYMSGBUF:
 000005E1'00000100   05D9  1721         .LONG   256,1$                 ;BUFFER TO HOLD CONTROL Y QUESTION (BEFORE F
            000006E1  05E1  1722 1$:     .BLKB   256
                      06E1  1723 ;
                      06E1  1724 CNTRLYQUESLEN:
            00000000  06E1  1725         .LONG   0                      ;FINAL LENGTH OF QUESTION
                      06E5  1726 ;
                      06E5  1727 CNTRLYQUESBUF:
 000006ED'00000100   06E5  1728         .LONG   256,CNTRLYQUES         ;BUFFER TO HOLD THE FINAL QUESTION
                      06ED  1729 CNTRLYQUES:
            000007ED  06ED  1730         .BLKB   256
                      07ED  1731
            00000000  07ED  1732 MAX_SENDSIZE::  .LONG   0              ; Maximum network send size ***
                      07F1  1733
                      07F1  1734
                      07F1  1735 .END    ; of module VMSRT
```

```
_$2
Sym
---
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
DSR
```

```
$$.TMP1             = 00000001               EXCMASK               000005CF R       03
$$.TMP2             = 000000EF               FINALACS              ********     X   02
$$T1                = 00000000               FIRST_READ            000005D7 R       03
$$T2                = 00000004               GETBUF                00001011 RG      02
ACSIGNORE             000010A5 R       02    INCMASK               000005C7 R       03
ANSBUF                0000059C R       03    INDFLAG               ********     X   02
ASSEM_TRACE         = 00000001               INIT_MSG              000000BE R       03
AST$L_STATE         = 00000000               INIT_MSGLEN         = 0000000A
AST$Q_IOSB          = 00000004               IO$M_CTRLCAST         ********     X   02
AST$T_BUF           = 00000026               IO$M_CTRLYAST         ********     X   02
AST$W_OPCODE        = 0000000C               IO$M_CVTLOW           ********     X   02
ASTHANDLER            00000224 RG      02    IO$M_HANGUP           ********     X   02
BADOUTBAND            000010B1 R       02    IO$M_INCLUDE          ********     X   02
BRDDESC               0000057F R       03    IO$M_OUTBAND          ********     X   02
BROADCAST             000008BC R       02    IO$M_PURGE            ********     X   03
BUFADR                0000002E R       03    IO$V_EXTEND           ********     X   02
BUFFREE               0000105F RG      02    IO$_ACPCONTROL        ********     X   02
BUFQUEUE              00000026 R       03    IO$_READLBLK          ********     X   02
BUFSIZE               00000032 R       03    IO$_READPBLK          ********     X   02
CANCELIO              000007BD R       02    IO$_READPROMPT        ********     X   02
CANMSG                0000056D R       03    IO$_READVBLK          ********     X   02
CHARBUF               00000000 R       03    IO$_SENSECHAR         ********     X   02
CHAR_BLOCK            ********     X   02    IO$_SENSEMODE         ********     X   02
CNTRCFLAG             000005A6 RG      03    IO$_SETCHAR           ********     X   02
CNTRLCHAN             ********     X   02    IO$_SETMODE           ********     X   02
CNTRLCY               00000114 R       03    IO$_TTYREADALL        ********     X   02
CNTRLCY_MSG           000000EE R       03    IO$_TTYREADPALL       ********     X   02
CNTRLCY_MSGLEN      = 0000000A               IO$_WRITELBLK         ********     X   02
CNTRLC_AST            00000DB6 RG      02    IO$_WRITEPBLK         ********     X   02
CNTRLYMSGBUF          000005D9 R       03    IO$_WRITEVBLK         ********     X   02
CNTRLYQUES            000006ED R       03    LIB$GET_VM            ********     X   02
CNTRLYQUESBUF         000006E5 R       03    LINKCHAN              ********     X   02
CNTRLYQUESLEN         000006E1 R       03    LINKERR               00000597 R       03
CNTRLYTEST            00000E92 R       02    LINKGONE              00000CAA R       02
CNTRLYTIM             0000059B R       03    LINKMAIL              00000036 R       03
CONFIG_CHAR           000000B2 R       03    LINKRECV              000006DA RG      02
CONFIG_MSG            00000084 R       03    LINKRECV_ERR          00000672 R       02
CONFIG_MSGLEN       = 00000014               LNKMBXDONE            00000B39 R       02
CONFIG_MSG_ST         000000AA R       03    LNKWRTDONE            00000A9F RG      02
CTERM_CTRLO_AST       ********     X   02    MAILCHAN              ********     X   02
CTERM_CTRL_CY         ********     X   02    MAXMSG              = 0000041A
CTERM_FLAG            ********     X   02    MAX_SENDSIZE          000007ED RG      03
CTERM_LINKRECV        ********     X   02    MBXMSG                00000587 R       03
CTERM_LNKWRTDONE      ********     X   02    MBXMSGTYP             00000593 R       03
CTERM_QIODONE         ********     X   02    MSG$_ABORT            ********     X   02
CTERM_UNSMSGDONE      ********     X   02    MSG$_DISCON           ********     X   02
CTP$B_PRO_MSGTYPE   = 00000026               MSG$_EXIT             ********     X   02
CTRLO_CHAN            00000012 R       03    MSG$_PATHLOST         ********     X   02
CTRLO_MASK            00000016 R       03    MSG$_THIRDPARTY       ********     X   02
CT_BIND_ACC_MSG       ********     X   02    MSG$_TRMBRDCST        ********     X   02
CT_BIND_MSGLEN        ********     X   02    MSG$_TRMHANGUP        ********     X   02
DBG$LINKRECV          ********     X   02    MSG$_TRMUNSOLIC       ********     X   02
DBG$LINKWRITE         ********     X   02    OP_BRDCST           = 00000006
DBG$TRACE_IO          ********     X   02    OP_CANCEL           = 00000005
DECNETERR             ********     X   02    OP_PRMPT            = 00000100
DEVNAM                ********     X   02    OP_READ             = 00000001
DEVNAMLEN             ********     X   02    OP_SENSEMODE        = 00000004
```

```
OP_SETMODE                      = 00000003
OP_WRITE                        = 00000002
OUTBANDABO                        00000010 RG    03
OUTBANDAST                        00000C37 R     02
OUTBANDCHAR                       0000056C R     03
OUTBANDEXC                        0000000E RG    03
OUTBANDINC                        0000000C RG    03
OUTBANDMSG                        00000562 R     03
OUTBAND_ERR                       00000622 R     02
PROCMSG                           0000022F R     02
PROCMSG_EXIT                      00000621 R     02
PROTO_ECO                         ********    X  02
QIODONE                           0000095B RG    02
QIO_ERR                           000005B3 R     02
RAB$L_UBF                         ********    X  02
RAB$W_RSZ                         ********    X  02
RAB$W_USZ                         ********    X  02
RDP$B_TT_OUTBAND                = 0000000A
RDP$C_ATTN                      = FFFFFFFF
RDP$C_END                       = FFFFFFFE
RDP$C_TT_BRDCST                 = 00000005
RDP$C_TT_CTRLC                  = 00000002
RDP$C_TT_CTRLY                  = 00000003
RDP$C_TT_HANGUP                 = 00000001
RDP$C_TT_OUTBAND                = 00000006
RDP$C_TT_UNSOL                  = 00000000
RDP$K_HEADERLEN                 = 0000000A
RDP$L_REFID                     = 00000004
RDP$L_TT_ASTPRM                 = 0000000A
RDP$L_TT_BCNT                   = 0000000A
RDP$L_TT_CARCON                 = 0000000E
RDP$L_TT_CHAR2                  = 0000001E
RDP$L_TT_FILL                   = 00000016
RDP$L_TT_PARITY                 = 0000001A
RDP$L_TT_SPEED                  = 00000012
RDP$L_TT_TIMOUT                 = 0000000E
RDP$Q_STATUS                    = 0000000A
RDP$Q_TT_CHAR                   = 0000000A
RDP$Q_TT_SCHAR                  = 00000012
RDP$T_TT_RDATA                  = 00000012
RDP$T_TT_TERM                   = 00000012
RDP$T_TT_WDATA                  = 00000012
RDP$W_MOD                       = 00000002
RDP$W_OPCODE                    = 00000000
RDP$W_UNIT                      = 00000008
READCHAN                          ********    X  02
READMSG                           0000024A R     02
READQ                             000005AF RG    03
READQIO                           000005A7 RG    03
RECORD_QUIT                       ********    X  02
REM$_ACSIGN                       ********    X  02
REM$_BADOUTBAND                   ********    X  02
REM$_CNTRLY                       ********    X  02
REM$_NETMBX                       ********    X  03
REMOTENODE                        000005BF RG    03
RETSTATUS                         ********    X  02
RMS$_EOF                          ********    X  02

RMS$_NORMAL                       ********    X  02
RT$C_LINKEFN                    = 00000001
RTERMDES                          0000001E R     03
RTLOG$V_TRACE                   = 00000001
RTLOG_FLAGS                       ********    X  02
SENSEMSG                          000005C8 R     02
SETMSG                            000003A3 R     02
SS$_ABORT                         ********    X  02
SS$_EXQUOTA                       ********    X  02
SS$_NORMAL                        ********    X  02
SS$_PATHLOST                      ********    X  02
SS$_THIRDPARTY                    ********    X  02
SYS$ASSIGN                        ********   GX  02
SYS$BRDCST                        ********   GX  02
SYS$CANCEL                        ********   GX  02
SYS$CLOSE                         ********   GX  02
SYS$DCLAST                        ********   GX  02
SYS$FAO                           ********    X  02
SYS$GET                           ********   GX  02
SYS$GETMSG                        ********   GX  02
SYS$PUTMSG                        ********   GX  02
SYS$QIO                           ********   GX  02
SYS$QIOW                          ********   GX  02
S S$SETAST                        ********   GX  02
SYS$SETIMR                        ********   GX  02
SYS$WAITFR                        ********   GX  02
SYS$WAKE                          ********   GX  02
SYSINFAB                          ********    X  02
SYSINRAB                          ********    X  02
TERMMBXCHAN                       ********    X  02
TERMOPS                           00001067 R     02
TERMUNIT                          ********    X  02
THREESEC                          000010BD R     02
TRACE_CONTINUE                    000006E9 R     02
TRACE_CONTINUE2                   00000A42 R     02
TRACE_RECV                        0000065A R     02
TRACE_WRITE                       00000A8F R     02
TTYDESC                           ********    X  02
UNSDAT                            0000011E RG    03
UNSDATMBX                         00000CEF R     02
UNSMSGDONE                        00000D4E R     02
VMSRT                             00000000 RG    02
VMS_INDREAD                       00000F3A RG    02
WAKEFLAG                          ********    X  02
WRITECHAN                         ********    X  02
WRITEMSG                          00000335 R     02
WRITEQ                            000005B7 RG    03
WRITEQIO                          000005AB RG    03
WRITE_TO_NET                      00000A3C RG    02
WRITE_TO_NETX                     00000A30 RG    02
WRITE_TO_NET_SYNC                 000009C4 RG    02
YTIMEDONE                         00000E89 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
RTPAD                           000010C5  (  4293.)  02 (  2.)  NOPIC   USR   CON   REL   LCL NOSHR   EXE   RD  NOWRT NOVEC BYTE
_RTPAD                          000007F1  (  2033.)  03 (  3.)  NOPIC   USR   CON   REL   LCL NOSHR   EXE   RD    WRT NOVEC LONG

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

Phase                     Page faults    CPU Time       Elapsed Time
-----                     -----------    --------       ------------
Initialization                    37     00:00:00.05    00:00:02.30
Command processing               123     00:00:00.46    00:00:03.35
Pass 1                           401     00:00:10.19    00:00:38.89
Symbol table sort                  0     00:00:00.98    00:00:02.22
Pass 2                           288     00:00:03.24    00:00:10.96
Symbol table output               26     00:00:00.11    00:00:00.30
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             880     00:00:15.05    00:00:58.04

The working set limit was 1950 pages.
91005 bytes (178 pages) of virtual memory were used to buffer the intermediate code.
There were 60 pages of symbol table space al ocated to hold 859 non-local and 100 local symbols.
1735 source lines were read in Pass 1, producing 40 object records in Pass 2.
36 pages of virtual memory were used to define 33 macros.

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

Macro library name                          Macros defined
------------------                          --------------
_$255$DUA28:[RTPAD.OBJ]RTPAD.MLB;1                 5
_$255$DUA28:[SYS.OBJ]LIB.MLB;1                     1
_$255$DUA28:[SYSLIB]STARLET.MLB;2                 24
TOTALS (all libraries)                            30

968 GETS were required to define 30 macros.

There were no errors, warnings or information messages.

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

```
_$2

Sym
---
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IND
IPC
LAS
LCC
LIB
LIB
LIB
LIB
LIB
LIB
LIB


LIB
LIB
LIB
LLI
LST
LST
LTY
MAK
MAX
MAX
MSF
NDX
NDX
NDX
NDX
NDX
NDX
NDX
```

0335 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY