C 8

```
   OOOOOOOOO     PPPPPPPPPPPP        CCCCCCCCCCCC     OOOOOOOOO     MMM          MMM
   OOOOOOOOO     PPPPPPPPPPPP        CCCCCCCCCCCC     OOOOOOOOO     MMM          MMM
   OOOOOOOOO     PPPPPPPPPPPP        CCCCCCCCCCCC     OOOOOOOOO     MMM          MMM
OOO         OOO  PPP         PPP  CCC              OOO         OOO  MMMMMM    MMMMMM
OOO         OOO  PPP         PPP  CCC              OOO         OOO  MMMMMM    MMMMMM
OOO         OOO  PPP         PPP  CCC              OOO         OOO  MMMMMM    MMMMMM
OOO         OOO  PPP         PPP  CCC              OOO         OOO  MMM    MMM   MMM
OOO         OOO  PPP         PPP  CCC              OOO         OOO  MMM    MMM   MMM
OOO         OOO  PPP         PPP  CCC              OOO         OOO  MMM    MMM   MMM
OOO         OOO  PPPPPPPPPPPP     CCC              OOO         OOO  MMM          MMM
OOO         OOO  PPPPPPPPPPPP     CCC              OOO         OOO  MMM          MMM
OOO         OOO  PPPPPPPPPPPP     CCC              OOO         OOO  MMM          MMM
OOO         OOO  PPP              CCC              OOO         OOO  MMM          MMM
OOO         OOO  PPP              CCC              OOO         OOO  MMM          MMM
OOO         OOO  PPP              CCC              OOO         OOO  MMM          MMM
OOO         OOO  PPP              CCC              OOO         OOO  MMM          MMM
OOO         OOO  PPP              CCC              OOO         OOO  MMM          MMM
OOO         OOO  PPP              CCC              OOO         OOO  MMM          MMM
   OOOOOOOOO     PPP                 CCCCCCCCCCCC     OOOOOOOOO     MMM          MMM
   OOOOOOOOO     PPP                 CCCCCCCCCCCC     OOOOOOOOO     MMM          MMM
   OOOOOOOOO     PPP                 CCCCCCCCCCCC     OOOOOOOOO     MMM          MMM
```

```
_$2


Sym
---
ALL


ASC


BOD
BOD
BOD
BOD
BOD
BOD
BOD
BOD
BUG
BYP
CAN
CAN
CAN
CHE
CHE


CLU
CLU
CLU
CLU
CLU
CLU
CLU

CLU
CLU
CLU
CLU
CLU
CLU
CLU


CLU
CLU
```

```
**FILE**ID**STATUS                                        F  9



  SSSSSSSS  TTTTTTTTTT    AAAAAA    TTTTTTTTTT  UU      UU    SSSSSSSS
  SSSSSSSS  TTTTTTTTTT    AAAAAA    TTTTTTTTTT  UU      UU    SSSSSSSS
SS              TT      AA      AA      TT      UU      UU  SS
SS              TT      AA      AA      TT      UU      UU  SS
SS              TT      AA      AA      TT      UU      UU  SS
SS              TT      AA      AA      TT      UU      UU  SS
  SSSSSS        TT      AA      AA      TT      UU      UU    SSSSSS
  SSSSSS        TT      AA      AA      TT      UU      UU    SSSSSS
        SS      TT      AAAAAAAAAA      TT      UU      UU          SS
        SS      TT      AAAAAAAAAA      TT      UU      UU          SS
        SS      TT      AA      AA      TT      UU      UU          SS      ....
        SS      TT      AA      AA      TT      UU      UU          SS      ....
SSSSSSSS        TT      AA      AA      TT      UUUUUUUUUU  SSSSSSSS        ....
SSSSSSSS        TT      AA      AA      TT      UUUUUUUUUU  SSSSSSSS        ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL               II     SS
LL               II     SS
LL               II     SS
LL               II     SS
LL               II       SSSSSS
LL               II       SSSSSS
LL               II             SS
LL               II             SS
LL               II             SS
LL               II             SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

OP
VO

```
0 MODULE  OPC$STATUS                 (
0                                     LANGUAGE (BLISS32),
0                                     IDENT = 'V04-000'
0                                   ) =
0
0 !*****************************************************************************
0 !*                                                                           *
0 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
0 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
0 !*  ALL RIGHTS RESERVED.                                                     *
0 !*                                                                           *
0 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
0 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE    *
0 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER    *
0 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY    *
0 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY    *
0 !*  TRANSFERRED.                                                             *
0 !*                                                                           *
0 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE    *
0 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT    *
0 !*  CORPORATION.                                                             *
0 !*                                                                           *
0 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS    *
0 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
0 !*                                                                           *
0 !*                                                                           *
0 !*****************************************************************************
0
0 !++
0 ! FACILITY:
0 !
0 !       OPCOM
0 !
0 ! ABSTRACT:
0 !
0 !       This module contains the specialized logic to service
0 !       a particular type of request sent by a user to OPCOM.
0 !
0 ! Environment:
0 !
0 !       VAX/VMS operating system.
0 !
0 ! Author:
0 !
0 !       Steven T. Jeffreys
0 !
0 ! Creation date:
0 !
0 !       March 10, 1981
0 !
0 ! Revision history:
0 !
0 !       V03-002 CWH3001         CW Hobbs                        30-Jul-1983
0 !               Various and sundry things to make OPCOM distributed
0 !               across the cluster.
0 !
0 !       V03-001 STJ3036         Steven T. Jeffreys,             07-Oct-1982
```

```
;    58         0058  0 !                   Handle implicitly canceled requests.
;    59         0059  0 !
;    60         0060  0 !--
;    61         0061  0
;    62         0062  1 BEGIN                                                  ! Start of STATUS
;    63         0063  1
;    64         0064  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    65         0065  1 LIBRARY 'LIB$:OPCOMLIB';
;    66         0066  1
;    67         0067  1 FORWARD ROUTINE
;    68         0068  1         STATUS_HANDLER   : NOVALUE;
;    69         0069  1
;    70         0070  1 BUILTIN
;    71         0071  1
;    72         0072  1         INSQUE,                                        ! Insert entry onto a queue
;    73         0073  1         REMQUE;                                        ! Remove entry from a queue
```

```
1 GLOBAL ROUTINE STATUS_HANDLER (BUFFER_DESC) : NOVALUE =
1
1 !++
1 ! Functional description:
1 !
1 !       This routine is the handler for all STATUS messages received by OPCOM.
1 !
1 ! Input:
1 !
1 !       BUFFER_DESC : The address of a quadword buffer descriptor that
1 !                     describes the buffer containing the message.
1 !
1 ! Implicit Input:
1 !
1 !       None.
1 !
1 ! Output:
1 !
1 !       None.
1 !
1 ! Implict output:
1 !
1 !       Some accounting data will be updated
1 !       to reflect the receipt of the message.
1 !
1 ! Side effects:
1 !
1 !       None.
1 !
1 ! Routine value:
1 !
1 !       None.
1 !--
1
2 BEGIN                                          ! Start of STATUS_HANDLER
2
2 MAP
2
2       BUFFER_DESC     : $ref_bblock;
2
2 EXTERNAL ROUTINE
2       CHECK_REQUEST,                           ! Common sanity checks
2       DEALLOCATE_RQCB,                         ! Dispose of an RQCB
2       FIND_OPERATOR,                           ! Locate a given operator
2       FORMAT_MESSAGE,                          ! Format an OPCOM message
2       IMPLICITLY_CANCELED,                     ! See if the request is dead
2       IMPLIED_CANCEL,                          ! Perform cancellation
2       INTERPRET_MASK,                          ! Interpret attention mask
2       NOTIFY_OPERATOR,                         ! Notify a given operator
2       VALID_OPERATOR;                          ! See if operator device is valid
2
2 LOCAL
2       MESSAGE_VECTOR  : VECTOR [5,LONG],       ! Message info
2       STATUS_DESC     : $desc_block,           ! Status message descriptor
2       STATUS_BUF      : $bblock [OPC$K_MAXREAD], ! Status message buffer
2       RQCB            : $ref_bblock,           ! RQCB data structure
```

```
2               OPER_RQCB       : $ref_bblock,          ! ditto (for known operator)
2               RQST_RQCB       : $ref_bblock,          ! ditto (for request RQCB)
2               OCD             : $ref_bblock,          ! OCD data structure
2               MCB             : $ref_bblock,          ! MCB data structure
2               RQST_COUNT      : LONG,                 ! Count of requests
2               STATUS          : LONG;
2
2       !
2       ! Check the minimum request size.  If not
2       ! enough data, then ignore the message.
2       !
3       IF .BUFFER_DESC [DSC$W_LENGTH] LSS (OPC$K_COMHDRSIZ + OPC$K_STATUS_MIN_SIZE)
2       THEN
2           RETURN;
2       !
2       ! Do some common sanity checking, and get an RQCB.
2       !
2       IF NOT CHECK_REQUEST (.BUFFER_DESC, RQCB)
2       THEN
2           RETURN;
2       !
2       ! Make sure this is a valid operator device.
2       !
2       IF NOT VALID_OPERATOR (.BUFFER_DESC, .RQCB)
2       THEN
3           BEGIN
3           DEALLOCATE_RQCB (.RQCB);                    ! Dismiss the request
3           RETURN;
3           END;
2       !
2       ! See if this is a valid operator.  If not, then dismiss the request.
2       !
2       IF NOT FIND_OPERATOR (.RQCB, OPER_RQCB)
2       THEN
3           BEGIN
3           MESSAGE_VECTOR [0] = OPC$_ILLRQST;          ! Set message code
3           MESSAGE_VECTOR [1] = 0;                     ! Use current time of day
3           FORMAT_MESSAGE (.RQCB, MESSAGE_VECTOR);     ! Format the message
3           NOTIFY_OPERATOR (.RQCB);                    ! Notify the requestor
3           DEALLOCATE_RQCB (.RQCB);                    ! Dispose of the RQCb
3           RETURN;                                     ! Dismiss the request
2           END;
2       !
2       ! The operator is known to opcom.  Send the operator
2       ! the standard status message.
2       !
2       STATUS_DESC [0,0,32,0] = OPC$K_MAXREAD;         ! Set buffer size
2       STATUS_DESC [DSC$A_POINTER] = STATUS_BUF;       ! Set buffer address
2       IF NOT INTERPRET_MASK (OPER_RQCB [RQCB_L_ATTNMASK1], STATUS_DESC, STATUS_DESC)
2       THEN
3           BEGIN
3           DEALLOCATE_RQCB (.RQCB);
3           RETURN;
3           END
2       ELSE
3           BEGIN
3           MESSAGE_VECTOR [0] = OPC$_OPERSTS;          ! Set message code
```

```
;  189        0188  3      MESSAGE_VECTOR [1] = 0;                      ! Use current date and time
;  190        0189  3      MESSAGE_VECTOR [2] = RQCB [RQCB_L_OPER_LEN]; ! Set operator device name
;  191        0190  3      MESSAGE_VECTOR [3] = STATUS_DESC;            ! Set status message descriptor
;  192        0191  3      FORMAT_MESSAGE (.RQCB, MESSAGE_VECTOR);      ! Format the message
;  193        0192  3      NOTIFY_OPERATOR (.RQCB);                     ! Send status message to the requestor
;  194        0193  2      END;
;  195        0194  2  !
;  196        0195  2  ! Scan through the list of request queued for this OCD.
;  197        0196  2  ! If the requestor is enabled to handle the request, inform
;  198        0197  2  ! the operator of the request.
;  199        0198  2  !
;  200        0199  2  OCD = .OPER_RQCB [RQCB_L_OCD];                     ! Get OCD address
;  201        0200  2  RQST_COUNT = .OCD [OCD_W_RQSTCOUNT];               ! Get count of requests
;  202        0201  2  RQST_RQCB = .OCD [OCD_L_RQSTFLINK];                ! Get address of first RQCB
;  203        0202  2  WHILE (.RQST_COUNT GTR 0) DO
;  204        0203  3      BEGIN
;  205        0204  3      !
;  206        0205  3      ! Compare the attention masks of the requests against
;  207        0206  3      ! the attention masks of the operator.  If any bits are
;  208        0207  3      ! in common, then notify the operator of the request.
;  209        0208  3      !
;  210        0209  4      IF ((.OPER_RQCB [RQCB_L_ATTNMASK1] AND .RQST_RQCB [RQCB_L_ATTNMASK1]) NEQ 0)
;  211        0210  4      OR ((.OPER_RQCB [RQCB_L_ATTNMASK2] AND .RQST_RQCB [RQCB_L_ATTNMASK2]) NEQ 0)
;  212        0211  3      THEN
;  213        0212  3          IF NOT IMPLICITLY_CANCELED (.RQST_RQCB)
;  214        0213  3          THEN
;  215        0214  4              BEGIN
;  216        0215  4              MCB = .RQCB [RQCB_L_MCB];              ! Save MCB address
;  217        0216  4              RQCB [RQCB_L_MCB] = .RQST_RQCB [RQCB_L_MCB];
;  218        0217  4              NOTIFY_OPERATOR (.RQCB);               ! Inform the operator
;  219        0218  4              RQCB [RQCB_L_MCB] = .MCB;              ! Restore MCB address
;  220        0219  3              END;
;  221        0220  3     RQST_COUNT = .RQST_COUNT - 1;                   ! Decrement the request cout
;  222        0221  3     RQST_RQCB  = .RQST_RQCB [RQCB_L_FLINK];         ! Get address of next request
;  223        0222  2     END;
;  224        0223  2
;  225        0224  2  IMPLIED_CANCEL ();                                 ! Process any implicit cancellations
;  226        0225  2  DEALLOCATE_RQCB (.RQCB);                           ! Dismiss the request.
;  227        0226  2
;  228        0227  1  END;                                               ! End of STATUS_HANDLER

                                                                           .TITLE   OPC$STATUS
                                                                           .IDENT   \V04-000\

                                                                           .EXTRN   CHECK_REQUEST, DEALLOCATE_RQCB
                                                                           .EXTRN   FIND_OPERATOR, FORMAT_MESSAGE
                                                                           .EXTRN   IMPLICITLY_CANCELED
                                                                           .EXTRN   IMPLIED_CANCEL, INTERPRET_MASK
                                                                           .EXTRN   NOTIFY_OPERATOR
                                                                           .EXTRN   VALID_OPERATOR

                                                                           .PSECT   $CODE$,NOWRT,2

                                                       00FC 00000          .ENTRY   STATUS_HANDLER, Save R2,R3,R4,R5,R6,R7          ; 0074
                                      57      0000G  CF   9E 00002          MOVAB    NOTIFY_OPERATOR, R7                             ;
                                      5E      F5DC   CE   9E 00007          MOVAB    -2596(SP), SP                                   ;
```

```
0044    8F         04   BC   B1 0000C        CMPW     @BUFFER_DESC, #68                                ; 0142
                        01   1E 00012        BGEQU    1$                                               ;
                             04 00014        RET                                                       ;
                        5E   DD 00015 1$:    PUSHL    SP                                               ; 0148
                   04   AC   DD 00017        PUSHL    BUFFER_DESC                                      ;
0000G   CF              02   FB 0001A        CALLS    #2, CHECK_REQUEST                                ;
        01              50   E8 0001F        BLBS     R0, 2$                                           ;
                             04 00022        RET                                                       ;
        53              6E   D0 00023 2$:    MOVL     RQCB, R3                                         ; 0154
                        53   DD 00026        PUSHL    R3                                               ;
                   04   AC   DD 00028        PUSHL    BUFFER_DESC                                      ;
0000G   CF              02   FB 0002B        CALLS    #2, VALID_OPERATOR                               ;
        27              50   E9 00030        BLBC     R0, 3$                                           ;
                   04   AE   9F 00033        PUSHAB   OPER_RQCB                                        ; 0163
                        53   DD 00036        PUSHL    R3                                               ;
0000G   CF              02   FB 00038        CALLS    #2, FIND_OPERATOR                                ;
        1D              50   E8 0003D        BLBS     R0, 4$                                           ;
  EC    AD 0005807C     8F   D0 00040        MOVL     #360572, MESSAGE_VECTOR                          ; 0166
                   F0   AD   D4 00048        CLRL     MESSAGE_VECTOR+4                                 ; 0167
                   EC   AD   9F 0004B        PUSHAB   MESSAGE_VECTOR                                   ; 0168
                        53   DD 0004E        PUSHL    R3                                               ;
0000G   CF              02   FB 00050        CALLS    #2, FORMAT_MESSAGE                               ;
                        53   DD 00055        PUSHL    R3                                               ; 0169
        67              01   FB 00057        CALLS    #1, NOTIFY_OPERATOR                              ;
                 008A   31 0005A 3$:         BRW      9$                                               ; 0170
  E4    AD       0A00   8F   3C 0005D 4$:    MOVZWL   #2560, STATUS_DESC                               ; 0177
  E8    AD         08   AE   9E 00063        MOVAB    STATUS_BUF, STATUS_DESC+4                        ; 0178
                   E4   AD   9F 00068        PUSHAB   STATUS_DESC                                      ; 0179
                   E4   AD   9F 0006B        PUSHAB   STATUS_DESC                                      ;
        54         0C   AE   D0 0006E        MOVL     OPER_RQCB, R4                                    ;
                   5C   A4   9F 00072        PUSHAB   92(R4)                                           ;
0000G   CF              03   FB 00075        CALLS    #3, INTERPRET_MASK                               ;
        6A              50   E9 0007A        BLBC     R0, 9$                                           ;
  EC    AD 000580C3     8F   D0 0007D        MOVL     #360643, MESSAGE_VECTOR                          ; 0187
                   F0   AD   D4 00085        CLRL     MESSAGE_VECTOR+4                                 ; 0188
  F4    AD         7C   A3   9E 00088        MOVAB    124(R3), MESSAGE_VECTOR+8                        ; 0189
  F8    AD         E4   AD   9E 0008D        MOVAB    STATUS_DESC, MESSAGE_VECTOR+12                   ; 0190
                   EC   AD   9F 00092        PUSHAB   MESSAGE_VECTOR                                   ; 0191
                        53   DD 00095        PUSHL    R3                                               ;
0000G   CF              02   FB 00097        CALLS    #2, FORMAT_MESSAGE                               ;
                        53   DD 0009C        PUSHL    R3                                               ; 0192
        67              01   FB 0009E        CALLS    #1, NOTIFY_OPERATOR                              ;
        50         24   A4   D0 000A1        MOVL     36(R4), OCD                                      ; 0199
        55         3A   A0   3C 000A5        MOVZWL   58(OCD), RQST_COUNT                              ; 0200
        52         3C   A0   D0 000A9        MOVL     60(OCD), RQST_RQCB                               ; 0201
                        55   D5 000AD 5$:    TSTL     RQST_COUNT                                       ; 0202
                        31   15 000AF        BLEQ     8$                                               ;
  5C    A2         5C   A4   D3 000B1        BITL     92(R4), 92(RQST_RQCB)                            ; 0209
                        07   12 000B6        BNEQ     6$                                               ;
  60    A2         60   A4   D3 000B8        BITL     96(R4), 96(RQST_RQCB)                            ; 0210
                        1C   13 000BD        BEQL     7$                                               ;
                        52   DD 000BF 6$:    PUSHL    RQST_RQCB                                        ; 0212
0000G   CF              01   FB 000C1        CALLS    #1, IMPLICITLY_CANCELED                          ;
        12              50   E8 000C6        BLBS     R0, 7$                                           ;
        56         6C   A3   D0 000C9        MOVL     108(R3), MCB                                     ; 0215
  6C    A3         6C   A2   D0 000CD        MOVL     108(RQST_RQCB), 108(R3)                          ; 0216
                        53   DD 000D2        PUSHL    R3                                               ; 0217
                        67          01 FB 000D4         CALLS   #1, NOTIFY_OPERATOR                          ;
              6C        A3          56 D0 000D7         MOVL    MCB, 108(R3)                                 ; 0218
                                    55 D7 000DB 7$:     DECL    RQST_COUNT                                   ; 0220
                        52          62 D0 000DD         MOVL    (RQST_RQCB), RQST_RQCB                       ; 0221
                                    CB 11 000E0         BRB     5$                                           ; 0202
           0000G        CF          00 FB 000E2 8$:     CALLS   #0, IMPLIED_CANCEL                           ; 0224
                                    53 DD 000E7 9$:     PUSHL   R3                                           ; 0225
           0000G        CF          01 FB 000E9         CALLS   #1, DEALLOCATE_RQCB                          ;
                                       04 000EE         RET                                                  ; 0227

; Routine Size:  239 bytes,    Routine Base:  $CODE$ + 0000


;  229          0228  1
;  230          0229  1 END                                                    ! End of STATUS
;  231          0230  0 ELUDOM





;                                 PSECT SUMMARY
;
;       Name                             Bytes                     Attributes
;
;   $CODE$                                    239  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)




;                             Library Statistics
;
;                                                    -------- Symbols --------         Pages        Processing
;       File                                         Total    Loaded   Percent         Mapped       Time
;
;   _$255$DUA28:[SYSLIB]LIB.L32;1                    18619         6         0         1000           00:01.8
;   _$255$DUA28:[OPCOM.OBJ]OPCOMLIB.L32;1              633        13         2           43           00:00.8





;                                  COMMAND QUALIFIERS

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

; Size:           239 code + 0 data bytes
; Run Time:          00:07.6
; Elapsed Time:      00:30.1
; Lines/CPU Min:       1820
; Lexemes/CPU-Min: 10195
; Memory Used:  103 pages
; Compilation Complete
```

0292 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY