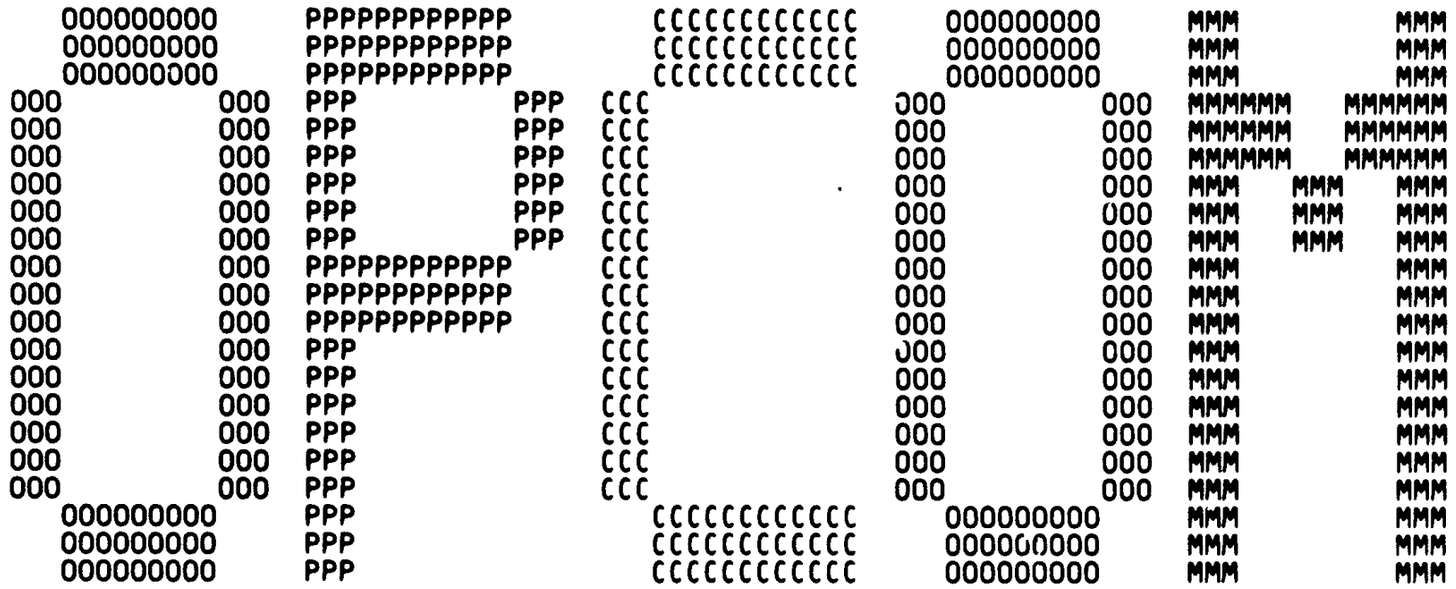

_$2

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
CLU

```
  000000    PPPPPPPP    RRRRRRRR    EEEEEEEEEE  NN      NN    AAAAAA    BBBBBBBB    LL          EEEEEEEEEE
  000000    PPPPPPPP    RRRRRRRR    EEEEEEEEEE  NN      NN    AAAAAA    BBBBBBBB    LL          EEEEEEEEEE
00      00  PP      PP  RR      RR  EE          NN      NN   AA      AA  BB      BB  LL          EE
00      00  PP      PP  RR      RR  EE          NN      NN   AA      AA  BB      BB  LL          EE
00      00  PP      PP  RR      RR  EE          NNNN    NN   AA      AA  BB      BB  LL          EE
00      00  PP      PP  RR      RR  EE          NNNN    NN   AA      AA  BB      BB  LL          EE
00      00  PPPPPPPP    RRRRRRRR    EEEEEEEE    NN  NN  NN   AA      AA  BBBBBBBB    LL          EEEEEEEE
00      00  PPPPPPPP    RRRRRRRR    EEEEEEEE    NN  NN  NN   AA      AA  BBBBBBBB    LL          EEEEEEEE
00      00  PP          RR  RR      EE          NN    NNNN   AAAAAAAAAA  BB      BB  LL          EE
00      00  PP          RR  RR      EE          NN    NNNN   AAAAAAAAAA  BB      BB  LL          EE
00      00  PP          RR    RR    EE          NN      NN   AA      AA  BB      BB  LL          EE                ....
00      00  PP          RR    RR    EE          NN      NN   AA      AA  BB      BB  LL          EE                ....
  000000    PP          RR      RR  EEEEEEEEEE  NN      NN   AA      AA  BBBBBBBB    LLLLLLLLLL  EEEEEEEEEE        ....
  000000    PP          RR      RR  EEEEEEEEEE  NN      NN   AA      AA  BBBBBBBB    LLLLLLLLLL  EEEEEEEEEE        ....

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
```

```
;     1          0001  0 MODULE  OPC$OPRENABLE   (
;     2          0002  0                         LANGUAGE (BLISS32),
;     3          0003  0                         IDENT = 'V04-000'
;     4          0004  0                         ) =
;     5          0005  0
;     6          0006  0 !*****************************************************************************
;     7          0007  0 !*                                                                           *
;     8          0008  0 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
;     9          0009  0 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                   *
;    10          0010  0 !*  ALL RIGHTS RESERVED.                                                     *
;    11          0011  0 !*                                                                           *
;    12          0012  0 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
;    13          0013  0 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
;    14          0014  0 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    15          0015  0 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    16          0016  0 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    17          0017  0 !*  TRANSFERRED.                                                             *
;    18          0018  0 !*                                                                           *
;    19          0019  0 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    20          0020  0 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    21          0021  0 !*  CORPORATION.                                                             *
;    22          0022  0 !*                                                                           *
;    23          0023  0 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    24          0024  0 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
;    25          0025  0 !*                                                                           *
;    26          0026  0 !*                                                                           *
;    27          0027  0 !*****************************************************************************
;    28          0028  0
;    29          0029  0 !++
;    30          0030  0 ! FACILITY:
;    31          0031  0 !
;    32          0032  0 !       OPCOM
;    33          0033  0 !
;    34          0034  0 ! ABSTRACT:
;    35          0035  0 !
;    36          0036  0 !       This module contains the specialized logic to service
;    37          0037  0 !       a particular type of request sent by a user to OPCOM.
;    38          0038  0 !
;    39          0039  0 ! Environment:
;    40          0040  0 !
;    41          0041  0 !       VAX/VMS operating system.
;    42          0042  0 !
;    43          0043  0 ! Author:
;    44          0044  0 !
;    45          0045  0 !       Steven T. Jeffreys
;    46          0046  0 !
;    47          0047  0 ! Creation date:
;    48          0048  0 !
;    49          0049  0 !       March 10, 1981
;    50          0050  0 !
;    51          0051  0 ! Revision history:
;    52          0052  0 !
;    53          0053  0 !       V03-003 CWH3003         CW Hobbs                        16-Sep-1983
;    54          0054  0 !               Moved a routine reference.
;    55          0055  0 !
;    56          0056  0 !       V03-002 CWH3002         CW Hobbs                        30-Jul-1983
;    57          0057  0 !               Various and sundry things to make OPCOM distributed
```

```
;    58          0058  0 !                 across the cluster.
;    59          0059  0 !
;    60          0060  0 ! V03-001        STJ3035         Steven T. Jeffreys,     07-Oct-1982
;    61          0061  0 !                 Force remote terminal operators to be temporary operators.
;    62          0062  0 !
;    63          0063  0 ! V02-002        STJ0166         Steven T. Jeffreys,     08-Feb-1982
;    64          0064  0 !                 Removed reference to library OPC$GET_VM.
;    65          0065  0 !
;    66          0066  0 !--
;    67          0067  0
;    68          0068  1 BEGIN                                           ! Start of OPRENABLE
;    69          0069  1
;    70          0070  1 LIBRARY 'SYS$LIBRARY:LIB.L32';
;    71          0071  1 LIBRARY 'LIB$:OPCOMLIB';
;    72          0072  1
;    73          0073  1 FORWARD ROUTINE
;    74          0074  1         OPRENABLE_HANDLER       : NOVALUE,
;    75          0075  1         OPRENABLE_CLM_HANDLER   : NOVALUE;
;    76          0076  1
;    77          0077  1 BUILTIN
;    78          0078  1
;    79          0079  1         INSQUE,                                 ! Insert entry onto a queue
;    80          0080  1         REMQUE;                                 ! Remove entry from a queue
;    81          0081  1
;    82          0082  1 EXTERNAL LITERAL
;    83          0083  1         RQCB_K_TYPE,                            ! RQCB structure type
;    84          0084  1         MIN_SCOPE,                              ! Minimum value for scope
;    85          0085  1         MAX_SCOPE;                              ! Maximum value for scope
;    86          0086  1
;    87          0087  1 EXTERNAL ROUTINE
;    88          0088  1         ALLOCATE_DS,                            ! Allocate a data structure
;    89          0089  1         CHECK_OPER_COVERAGE : NOVALUE,          ! Check operator coverage on requests
;    90          0090  1         CHECK_REQUEST,                          ! Common sanity checks
;    91          0091  1         CLUSMSG_CONV_CLM_RQCB,                  ! Convert CLMRQCB to a local RQCB
;    92          0092  1         CLUSMSG_RQCB_SEND,                      ! Send the RQCB to remote nodes
;    93          0093  1         CREATE_OCD      : NOVALUE,              ! Create a new OCD
;    94          0094  1         DEALLOCATE_MCB  : NOVALUE,              ! Dispose of an MCB
;    95          0095  1         DEALLOCATE_RQCB : NOVALUE,              ! Dispose of an RQCB
;    96          0096  1         DUMP_LOG_FILE,
;    97          0097  1         FIND_OCD,                               ! Find a given OCD
;    98          0098  1         FIND_OPERATOR,                          ! Find a given operator
;    99          0099  1         FORMAT_MESSAGE,                         ! Format a message and build an MCB
;   100          0100  1         INTERPRET_MASK,                         ! Interpret the attention mask
;   101          0101  1         LOG_MESSAGE,                            ! Write a message to a logfile
;   102          0102  1         NOTIFY_LISTED_OPERATORS,                ! Send message to list of operators
;   103          0103  1         NOTIFY_OPERATOR,                        ! Send message to a given operator
;   104          0104  1         UPD_OPER_CONTEXT,                       ! Update a given operator RQCB
;   105          0105  1         VALID_OPERATOR,                         ! See if operator device is valid
;   106          0106  1         WRITE_LOG_FILE;
;   107          0107  1
;   108          0108  1 EXTERNAL
;   109          0109  1         GLOBAL_STATUS   : BITVECTOR;            ! Global status flags
```

```
;  111     0110  1 GLOBAL ROUTINE OPRENABLE_HANDLER (BUFFER_DESC) : NOVALUE =
;  112     0111  1
;  113     0112  1 !++
;  114     0113  1 ! Functional description:
;  115     0114  1 !
;  116     0115  1 !       This routine is the handler for all OPRENABLE messages received by OPCOM.
;  117     0116  1 !
;  118     0117  1 !
;  119     0118  1 ! Input:
;  120     0119  1 !
;  121     0120  1 !       BUFFER_DESC : The address of a quadword buffer descriptor that
;  122     0121  1 !                     describes the buffer containing the message.
;  123     0122  1 !
;  124     0123  1 ! Implicit Input:
;  125     0124  1 !
;  126     0125  1 !       None.
;  127     0126  1 !
;  128     0127  1 ! Output:
;  129     0128  1 !
;  130     0129  1 !       None.
;  131     0130  1 !
;  132     0131  1 ! Implict output:
;  133     0132  1 !
;  134     0133  1 !       Some accounting data will be updated
;  135     0134  1 !       to reflect the receipt of the message.
;  136     0135  1 !
;  137     0136  1 ! Side effects:
;  138     0137  1 !
;  139     0138  1 !       None.
;  140     0139  1 !
;  141     0140  1 ! Routine value:
;  142     0141  1 !
;  143     0142  1 !       None.
;  144     0143  1 !--
;  145     0144  1
;  146     0145  2 BEGIN                                           ! Start of OPRENABLE_HANDLER
;  147     0146  2
;  148     0147  2 MAP
;  149     0148  2         BUFFER_DESC     : $ref_bblock;
;  150     0149  2
;  151     0150  2 LOCAL
;  152     0151  2         STATUS_DESC     : $desc_block,          ! Descriptor for status message
;  153     0152  2         STATUS_BUF      : $bblock [OPC$K_MAXREAD], ! Buffer for status message
;  154     0153  2         MESSAGE_VECTOR  : VECTOR [5,LONG],      ! Message vector
;  155     0154  2         MSG             : $ref_bblock,          ! Pointer to user request text
;  156     0155  2         ARG_LIST        : VECTOR [3],           ! Arguement list
;  157     0156  2         ENABLE_FLAG     : LONG,                 ! 1 if ENABLE, 0 if DISABLE
;  158     0157  2         FULL_DISABLE    : LONG,                 ! Boolean
;  159     0158  2         FULL_ENABLE     : LONG,                 ! Boolean
;  160     0159  2         FOUND           : LONG,                 ! Boolean
;  161     0160  2         LOST_COVERAGE   : LONG,                 ! Boolean
;  162     0161  2         OCD             : $ref_bblock,          ! OCD structure
;  163     0162  2         RQCB            : $ref_bblock,          ! ReQuest Context Block
;  164     0163  2         OPER_RQCB       : $ref_bblock,          ! Known operator RQCB
;  165     0164  2         STATUS          : LONG;
;  166     0165  2
;  167     0166  2 FULL_ENABLE = FALSE;
```

```
;  168          0167  2 FULL_DISABLE = FALSE;
;  169          0168  2 !
;  170          0169  2 ! If the request does not contain enough data,
;  171          0170  2 ! then it cannot be processed.  Ignore it.
;  172          0171  2 !
;  173          0172  3 IF .BUFFER_DESC [DSC$W_LENGTH] LSS (OPC$K_OPRENABLE_MIN_SIZE + OPC$K_COMHDRSIZ)
;  174          0173  2 THEN
;  175          0174  2     RETURN;
;  176          0175  2 !
;  177          0176  2 ! Call a special routine that does some sanity
;  178          0177  2 ! checking and startup code for all request handlers.
;  179          0178  2 !
;  180          0179  3 IF NOT (STATUS = CHECK_REQUEST (.BUFFER_DESC, RQCB))
;  181          0180  2 THEN
;  182          0181  2     RETURN;
;  183          0182  2 !
;  184          0183  2 ! Check the specified device and see if it is a
;  185          0184  2 ! valid operator device.  This routine will also
;  186          0185  2 ! format the operator device name and create a
;  187          0186  2 ! device name descriptor within the RQCB.
;  188          0187  2 !
;  189          0188  2 IF NOT VALID_OPERATOR (.BUFFER_DESC, .RQCB)
;  190          0189  2 THEN
;  191          0190  3     BEGIN
;  192          0191  3     DEALLOCATE_RQCB (.RQCB);
;  193          0192  3     RETURN;
;  194          0193  2     END;
;  195          0194  2 !
;  196          0195  2 !
;  197          0196  2 ! See if the operator is already known to OPCOM.
;  198          0197  2 ! This entails scanning down the appropriate operator
;  199          0198  2 ! list and comparing the device names for equality.
;  200          0199  2 !
;  201          0200  2 FOUND = FIND_OPERATOR (.RQCB, OPER_RQCB);
;  202          0201  2 !
;  203          0202  2 ! If the operator is known to OPCOM, then the requestor's UIC
;  204          0203  2 ! must agree with the UIC of the known operator.  If it does not,
;  205          0204  2 ! then dismiss the request without notifying the requestor, as
;  206          0205  2 ! this would only allow a user to cause an operator to be plauged
;  207          0206  2 ! with error messages for a request he did not send.
;  208          0207  2 !
;  209          0208  2 IF .FOUND
;  210          0209  2 THEN
;  211          0210  3     BEGIN
;  212          0211  3     OCD = .RQCB [RQCB_L_OCD];                          ! Get the OCD address
;  213          0212  4     IF ((.OCD [OCD_B_SCOPE] EQL OPC$K_GROUP) AND       ! If GROUP OCD, the group fields
;  214          0213  5         (.OCD [$BYTEOFFSET (OCD_L_UIC),16,16,0] NEQ    ! of the UIC must be the same for both
;  215          0214  4          .RQCB [$BYTEOFFSET (RQCB_L_UIC),16,16,0]))    ! the known operator and the requestor.
;  216          0215  4     OR ((.OCD [OCD_B_SCOPE] EQL OPC$K_USER) AND        ! If USER OCD, the UICs must be equal.
;  217          0216  4         (.OCD [OCD_L_UIC] NEQ .RQCB [RQCB_L_UIC]))
;  218          0217  3     THEN
;  219          0218  4         BEGIN                                          ! Dismiss the request
;  220          0219  4         DEALLOCATE_RQCB (.RQCB);
;  221          0220  4         RETURN;
;  222          0221  3         END;
;  223          0222  2     END;
;  224          0223  2 MESSAGE_VECTOR [0] = 0;                               ! Assume no error
```

```
;  225          0224  2 MESSAGE_VECTOR [1] = 0;
;  226          0225  2 !
;  227          0226  2 !
;  228          0227  2 ! Force remote operators (DECnet remote terminals and
;  229          0228  2 ! dial-in lines) to be 'temporary' operators.
;  230          0229  2 !
;  231          0230  2 !
;  232          0231  2 IF .RQCB [OPRSTS_V_REMTRM]
;  233          0232  2 THEN
;  234          0233  2     $bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_PERMOPER] = FALSE;
;  235          0234  2 !
;  236          0235  2 !
;  237          0236  2 ! If an enable request includes SECURITY operator, make sure that the requestor
;  238          0237  2 ! has SECURITY privilege.
;  239          0238  2 !
;  240          0239  3 IF (.$bblock [RQCB [RQCB_L_ATTNMASK1]-$BYTEOFFSET (OPC$V_NM_SECURITY), OPC$V_NM_SECURITY])
;  241          0240  3   AND (NOT .$bblock [RQCB [RQCB_L_PRIVMASK1], PRV$V_SECURITY])
;  242          0241  3 !
;  243          0242  3 ! Above test prevents someone from both enabling and disabling a security operator without having
;  244          0243  3 ! SECURITY privilege.  If the following line is added, you wouldn't need privilege to disable.
;  245          0244  3 !
;  246          0245  3 !   AND (NOT .$bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_DISABLE])
;  247          0246  2 THEN
;  248          0247  3     BEGIN
;  249          0248  3     !
;  250          0249  3     ! No priv to set security operators, clear the bit in the request
;  251          0250  3     !
;  252          0251  3     $bblock [RQCB [RQCB_L_ATTNMASK1]-$BYTEOFFSET (OPC$V_NM_SECURITY), OPC$V_NM_SECURITY] = 0;
;  253          0252  3     !
;  254          0253  3     ! If no other operators requested, then it is an illegal request
;  255          0254  3     !
;  256          0255  4     IF  (.RQCB [RQCB_L_ATTNMASK1] EQL 0) AND (.RQCB [RQCB_L_ATTNMASK2] EQL 0)
;  257          0256  3     THEN
;  258          0257  3         MESSAGE_VECTOR [0] = OPC$_ILLRQST;
;  259          0258  2     END;
;  260          0259  2 !
;  261          0260  2 !
;  262          0261  2 ! Make sure that something is being enabled or disabled.
;  263          0262  2 ! If not, this is an illegal request.  Also, do not allow
;  264          0263  2 ! the user to touch the PERMOPER bit if the operator device
;  265          0264  2 ! is a remote terminal.
;  266          0265  2 !
;  267          0266  3 IF  (.RQCB [RQCB_L_ATTNMASK1] EQL 0)
;  268          0267  3 AND (.RQCB [RQCB_L_ATTNMASK2] EQL 0)
;  269          0268  3 AND (NOT .$bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_PERMOPER])
;  270          0269  3 OR  ((.RQCB [OPRSTS_V_REMTRM]) AND
;  271          0270  3     (.$bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_PERMOPER]))
;  272          0271  2 THEN
;  273          0272  2     MESSAGE_VECTOR [0] = OPC$_ILLRQST;                  ! Set request status
;  274          0273  2 !
;  275          0274  2 ! See if the user has the correct privilege(s)
;  276          0275  2 ! for the request to be granted.
;  277          0276  2 !
;  278          0277  3 IF (( RQCB [RQCB_B_SCOPE] EQL OPC$K_SYSTEM) AND                ! SYSTEM scope requires OPER privilege
;  279          0278  3     (NOT .$bblock [RQCB [RQCB_L_PRIVMASK1], PRV$V_OPER]))
;  280          0279  3 OR ((.RQCB [RQCB_B_SCOPE] EQL OPC$K_GROUP) AND                ! GROUP scope requires GROUP privilege
;  281          0280  4     ((NOT .$bblock [RQCB [RQCB_L_PRIVMASK1], PRV$V_GROUP]) OR
```

```
;  282          0281  3      (NOT .$bblock [RQCB [RQCB_L_PRIVMASK1], PRV$V_OPER])))
;  283          0282  3 OR ((.RQCB [RQCB_L_SENDERUIC] NEQ .RQCB [RQCB_L_UIC]) AND        ! Issuing a request on another's behalf
;  284          0283  3     (NOT .$bblock [RQCB [RQCB_L_PRIVMASK1], PRV$V_OPER]))       ! requires OPER privilege.
;  285          0284  2 THEN
;  286          0285  2     MESSAGE_VECTOR [0] = OPC$_ILLRQST;                          ! Set request status
;  287          0286  2 !
;  288          0287  2 ! If an error was detected, notify the
;  289          0288  2 ! would-be operator and abort the request.
;  290          0289  2 !
;  291          0290  2 IF .MESSAGE_VECTOR [0] NEQ 0
;  292          0291  2 THEN
;  293          0292  3     BEGIN
;  294          0293  3     FORMAT_MESSAGE (.RQCB, MESSAGE_VECTOR);
;  295          0294  3     NOTIFY_OPERATOR (.RQCB);
;  296          0295  3     DEALLOCATE_RQCB (.RQCB);
;  297          0296  3     RETURN;
;  298          0297  2     END;
;  299          0298  2 !
;  300          0299  2 ! Finish the request.
;  301          0300  2 !
;  302          0301  2 IF NOT .$bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_DISABLE]
;  303          0302  2 THEN
;  304          0303  3     BEGIN
;  305          0304  3     !
;  306          0305  3     ! This is an enable request.
;  307          0306  3     !
;  308          0307  3     ENABLE_FLAG = 1;                                ! Set bit value
;  309          0308  3     MESSAGE_VECTOR [0] = OPC$_TERMENAB;             ! Set message code
;  310          0309  3     IF NOT .FOUND
;  311          0310  3     THEN
;  312          0311  4         BEGIN
;  313          0312  4         !
;  314          0313  4         ! The operator is not known to OPCOM.  An operator RQCB
;  315          0314  4         ! must be created and inserted onto the appropriate OCD
;  316          0315  4         ! operator list so the generalized ENABLE logic will work.
;  317          0316  4         !
;  318          0317  4         FULL_ENABLE = TRUE;
;  319          0318  4         ALLOCATE_DS (RQCB_K_TYPE, OPER_RQCB);                   ! Clone a new RQCB
;  320          0319  4         CH$MOVE (RQCB_K_OVERLAY_SIZE,                           ! Copy old RQCB
;  321          0320  4                  RQCB [RQCB_T_OVERLAY], OPER_RQCB [RQCB_T_OVERLAY]);
;  322          0321  4         OPER_RQCB [RQCB_L_OPTIONS] = 0;                         ! Zero option mask
;  323          0322  4         OPER_RQCB [RQCB_L_RQ_OPTIONS] = 0;                      ! Zero option mask
;  324          0323  4         OPER_RQCB [RQCB_L_ATTNMASK1] = 0;                       ! Zero mask for enable
;  325          0324  4         OPER_RQCB [RQCB_L_ATTNMASK2] = 0;                       ! Zero mask for enable
;  326          0325  4         !
;  327          0326  4         ! Insert the operator RQCB into the operator list in the appropriate
;  328          0327  4         ! OCD.  If no OCD is found, then a new one must be created.
;  329          0328  4         !
;  330          0329  4         IF NOT FIND_OCD (.RQCB [RQCB_B_SCOPE], .RQCB [RQCB_L_UIC], OCD)
;  331          0330  4         THEN
;  332          0331  4             CREATE_OCD (.RQCB [RQCB_B_SCOPE], .RQCB [RQCB_L_UIC], OCD);
;  333          0332  4         INSQUE (.OPER_RQCB, OCD [OCD_L_OPERFLINK]);             ! Insert cloned RQCB onto operator list
;  334          0333  4         OPER_RQCB [RQCB_L_OCD] = .OCD;                          ! Set pointer to OCD
;  335          0334  4         OCD [OCD_W_OPERCOUNT] = .OCD [OCD_W_OPERCOUNT]+1;       ! Increment operator count
;  336          0335  4         ARG_LIST [0] = 1;                                       ! Set arguement count
;  337          0336  4         ARG_LIST [1] = OPER_RQCB [RQCB_L_OPER_LEN];             ! Set operator device name
;  338          0337  4         ARG_LIST [2] = ON;                                      ! Bit value
```

```
;  339     0338  4          $CMKRNL (ROUTIN=EXE$SETOPR, ARGLST=ARG_LIST);   ! Set the OPR bit in the device UCB.
;  340     0339  3          END;
;  341     0340  3      END
;  342     0341  2  ELSE
;  343     0342  2      IF NOT .FOUND
;  344     0343  2      THEN
;  345     0344  3          BEGIN
;  346     0345  3          !
;  347     0346  3          ! The user is trying to disable a nonexistent operator.
;  348     0347  3          !
;  349     0348  3          MESSAGE_VECTOR [0] = OPC$_ILLRQST;
;  350     0349  3          MESSAGE_VECTOR [1] = ;
;  351     0350  3          FORMAT_MESSAGE (.RQCB, MESSAGE_VECTOR);
;  352     0351  3          NOTIFY_OPERATOR (.RQCB);
;  353     0352  3          DEALLOCATE_RQCB (.RQCB);
;  354     0353  3          RETURN;
;  355     0354  3          END
;  356     0355  2      ELSE
;  357     0356  3          BEGIN
;  358     0357  3          !
;  359     0358  3          ! This is a legitimate DISABLE request.
;  360     0359  3          !
;  361     0360  3          ENABLE_FLAG = 0;                                ! Set for disable request
;  362     0361  3          MESSAGE_VECTOR [0] = OPC$_TERMDSBL;
;  363     0362  2          END;
;  364     0363  2  !
;  365     0364  2  ! Tell the rest of the world about the change.  We convert the RQCB into a form which contains all
;  366     0365  2  ! variable length buffers (rather than pointers to heap memory), then send the converted RQCB to
;  367     0366  2  ! all the other OPCOMs in the universe.
;  368     0367  2  !
;  369     0368  2  CLUSMSG_RQCB_SEND (-1, CLM__OPRENABLE, .RQCB);
;  370     0369  2  !
;  371     0370  2  ! Update the OCD count vector for each bit present in the bit mask.
;  372     0371  2  ! The count will be decremented for a DISABLE, incremented for an ENABLE.
;  373     0372  2  ! Also update the OCD operator intrest mask, and the corresponding intrest
;  374     0373  2  ! mask in the operator RQCB.  Note if there was a potential lose of operator
;  375     0374  2  ! coverage on any outstanding requests.
;  376     0375  2  !
;  377     0376  2  LOST_COVERAGE = UPD_OPER_CONTEXT (.$bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_DISABLE],
;  378     0377  2                                    .RQCB [RQCB_L_ATTNMASK1],
;  379     0378  2                                    .RQCB [RQCB_L_ATTNMASK2],
;  380     0379  2                                    .OPER_RQCB
;  381     0380  2                                   );
;  382     0381  2  !
;  383     0382  2  ! Set the PERMOPER bit as desired.                                                          ; R
;  384     0383  2  !
;  385     0384  2  IF .$bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_PERMOPER]
;  386     0385  2  THEN
;  387     0386  2      $bblock [OPER_RQCB [RQCB_L_RQ_OPTIONS], OPC$V_PERMOPER] = .ENABLE_FLAG;                 ;
;  388     0387  2  !                                                                                          ;
;  389     0388  2  ! If the OPER_RQCB attention mask went to 0, then                                          ;
;  390     0389  2  ! disable the operator terminal.                                                           ;
;  391     0390  2  !
;  392     0391  3  IF (.OPER_RQCB [RQCB_L_ATTNMASK1] EQL 0) AND (.OPER_RQCB [RQCB_L_ATTNMASK2] EQL 0)
;  393     0392  2  THEN
;  394     0393  3      BEGIN
;  395     0394  3      FULL_DISABLE = TRUE;                                                                   ;
```

```
;  396          0395  3     $bbblock [OPER_RQCB [RQCB_L_OPTIONS], OPC$V_NOBRD] = .$bbblock [RQCB [RQCB_L_OPTIONS], OPC$V_NOBRD];
;  397          0396  3     $bbblock [OPER_RQCB [RQCB_L_OPTIONS], OPC$V_NOLOG] = .$bbblock [RQCB [RQCB_L_OPTIONS], OPC$V_NOLOG];
;  398          0397  3     DEALLOCATE_RQCB (.RQCB);                        ! Get rid of the old RQCB
;  399          0398  3     REMQUE (.OPER_RQCB, RQCB);                      ! Remove operator RQCB from list
;  400          0399  3     RQCB = .OPER_RQCB;                              ! Copy the RQCB address
;  401          0400  3     OCD = .RQCB [RQCB_L_OCD];                       ! Get OCD address
;  402          0401  3     OCD [OCD_W_OPERCOUNT] = .OCD [OCD_W_OPERCOUNT] - 1;
;  403          0402  3     ARG_LIST [0] = 1;                               ! Set arguement count
;  404          0403  3     ARG_LIST [1] = OPER_RQCB [RQCB_L_OPER_LEN];     ! Set operator device name
;  405          0404  3     ARG_LIST [2] = OFF;                             ! Bit value
;  406          0405  3     $CMKRNL (ROUTIN=EXE$SETOPR, ARGLST=ARG_LIST);       ! Clear the OPR bit in the device UCB.
;  407          0406  2     END;
;  408          0407  2 !
;  409          0408  2 OCD = .OPER_RQCB [RQCB_L_OCD];
;  410          0409  2 RQCB [RQCB_L_OCD] = .OCD;
;  411          0410  2 IF .FULL_ENABLE OR .FULL_DISABLE
;  412          0411  2 THEN
;  413          0412  3     BEGIN
;  414          0413  3     !
;  415          0414  3     ! Send the ENABLE/DISABLE message to all CENTRAL operators
;  416          0415  3     ! connected to this OCD.  Notify the requestor even if not a CENTRAL
;  417          0416  3     ! operator.  If this is a full disable, explicitly notify this operator of
;  418          0417  3     ! the DISABLE.  This is necessary because the operator is no longer
;  419          0418  3     ! on the OCD's operator list, and would not be notified otherwise.
;  420          0419  3     ! Note that the message is sent only if this is the initial ENABLE
;  421          0420  3     ! or a full DISABLE for this operator.  Note that the request RQCB
;  422          0421  3     ! must be used instead of the operator RQCB to avoid a race condition
;  423          0422  3     ! when a user does an enable/disable and then implicitly disables the
;  424          0423  3     ! operator.  This would cause the OPER_RQCB to evaporate in the middle
;  425          0424  3     ! of the NOTIFY_LISTED_OPERATORS operation.
;  426          0425  3     !
;  427          0426  3     RQCB [RQCB_L_ATTNMASK1] = OPC$M_NM_CENTRL;
;  428          0427  3     RQCB [RQCB_L_ATTNMASK2] = 0;
;  429          0428  3     RQCB [HDR_V_BRD] = TRUE;
;  430          0429  3     MESSAGE_VECTOR [1] = 0;
;  431          0430  3     MESSAGE_VECTOR [2] = RQCB [RQCB_L_OPER_LEN];
;  432          0431  3     MESSAGE_VECTOR [3] = .RQCB [RQCB_B_USERNAMELEN];
;  433          0432  3     MESSAGE_VECTOR [4] = RQCB [RQCB_T_USERNAME];
;  434          0433  3     FORMAT_MESSAGE (.RQCB, MESSAGE_VECTOR);
;  435          0434  3     LOG_MESSAGE (.RQCB);
;  436          0435  3     NOTIFY_LISTED_OPERATORS (.RQCB);
;  437          0436  3     IF .FULL_DISABLE
;  438          0437  3     THEN
;  439          0438  3         NOTIFY_OPERATOR (.RQCB);
;  440          0439  2     END;
;  441          0440  2 !
;  442          0441  2 ! Since the operator status has changed, send an operator
;  443          0442  2 ! status message to the operator.  This does not have to
;  444          0443  2 ! be done in the case of a full DISABLE.
;  445          0444  2 !
;  446          0445  2 ! If the operator was implicitly disabled during a NOTIFY_LISTED_OPERATORS
;  447          0446  2 ! operation, then do not send the message, as it would cause the user to
;  448          0447  2 ! think that the enable operation had completed.  This does not close the
;  449          0448  2 ! timing window, but only makes it's occurrance less annoying.  A simple
;  450          0449  2 ! test for the implicit disable is to check the sequence number on the
;  451          0450  2 ! operator RQCB. If it is zero, it means that operator has been disabled,
;  452          0451  2 ! and the RQCB deallocated.
```

```
;  453          0452  2 !
;  454          0453  2 IF (NOT .FULL_DISABLE) AND (.OPER_RQCB [RQCB_L_SEQNUM] NEQ 0)
;  455          0454  2 THEN
;  456          0455  2     BEGIN
;  457          0456  2     !
;  458          0457  2     ! Create a buffer descriptor for the status message,
;  459          0458  2     ! format the message, and log it.
;  460          0459  2     !
;  461          0460  2     STATUS_DESC [0,0,32,0] = OPC$K_MAXREAD;             ! Set buffer size
;  462          0461  2     STATUS_DESC [DSC$A_POINTER] = STATUS_BUF;           ! Set buffer address
;  463          0462  2     MESSAGE_VECTOR [0] = OPC$_OPERSTS;                  ! Set message number
;  464          0463  2     MESSAGE_VECTOR [1] = 0;                             ! Set # of message args
;  465          0464  2     MESSAGE_VECTOR [2] = RQCB [RQCB_L_OPER_LEN];        ! Operator desc. addr
;  466          0465  2     MESSAGE_VECTOR [3] = STATUS_DESC;                   ! Status line desc. addr
;  467          0466  3     IF INTERPRET_MASK (OPER_RQCB [RQCB_L_ATTNMASK1], STATUS_DESC, STATUS_DESC)
;  468          0467  3     THEN
;  469          0468  4         BEGIN
;  470          0469  4         FORMAT_MESSAGE (.RQCB, MESSAGE_VECTOR);
;  471          0470  4         LOG_MESSAGE (.RQCB);                            ! Log the status message
;  472          0471  4         NOTIFY_OPERATOR (.RQCB);                        ! Send the message to the operator
;  473          0472  3         END;
;  474          0473  2     END;
;  475          0474  2 !
;  476          0475  2 ! Deallocate the RQCB.  If this was a full enable, clear
;  477          0476  2 ! the pointer to the operator device name, as both the
;  478          0477  2 ! RQCB and the OPER_RQCB point to the same device name.
;  479          0478  2 ! Deallocating the RQCB in this case would cause the OPER_RQCB
;  480          0479  2 ! device name to evaporate.
;  481          0480  2 !
;  482          0481  2 IF .FULL_ENABLE
;  483          0482  2 THEN
;  484          0483  2     RQCB [RQCB_L_OPER_PTR] = 0;
;  485          0484  2 DEALLOCATE_RQCB (.RQCB);
;  486          0485  2 !
;  487          0486  2 ! If there was a potential loss of operator coverage,
;  488          0487  2 ! check the remaining requests queued to this OCD.
;  489          0488  2 ! The routine will cancel any requests that no longer
;  490          0489  2 ! have operator coverage.
;  491          0490  2 !
;  492          0491  2 IF .LOST_COVERAGE
;  493          0492  2 THEN
;  494          0493  2     CHECK_OPER_COVERAGE (.OCD);
;  495          0494  2
;  496          0495  1 END;                                                    ! End of OPRENABLE_HANDLER


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

                                                .EXTRN  RQCB_K_TYPE, MIN_SCOPE
                                                .EXTRN  MAX_SCOPE, ALLOCATE_DS
                                                .EXTRN  CHECK_OPER_COVERAGE
                                                .EXTRN  CHECK_REQUEST, CLUSMSG_CONV_CLM_RQCB
                                                .EXTRN  CLUSMSG_RQCB_SEND
                                                .EXTRN  CREATE_OCD, DEALLOCATE_MCB
                                                .EXTRN  DEALLOCATE_RQCB
                                                .EXTRN  DUMP_LOG_FILE, FIND_OCD
```

```
                                        .EXTRN  FIND_OPERATOR, FORMAT_MESSAGE
                                        .EXTRN  INTERPRET_MASK, LOG_MESSAGE
                                        .EXTRN  NOTIFY_LISTED_OPERATORS
                                        .EXTRN  NOTIFY_OPERATOR
                                        .EXTRN  UPD_OPER_CONTEXT
                                        .EXTRN  VALID_OPERATOR, WRITE_LOG_FILE
                                        .EXTRN  GLOBAL_STATUS, EXE$SETOPR
                                        .EXTRN  SYS$CMRRNL

                                        .PSECT  $CODE$,NOWRT,2

                            0FFC 00000          .ENTRY  OPRENABLE_HANDLER, Save R2,R3,R4,R5,R6,R7,- ; 0110
                                                        R8,R9,R10,R11
                    5B 00000000G    00  9E 00002         MOVAB   EXE$SETOPR, R11
                    5E      F5CC    CE  9E 00009         MOVAB   -2612(SP), SP
                                    59  7C 0000E         CLRQ    FULL_DISABLE                    ; 0167
                0044    8F      04  BC  B1 00010         CMPW    @BUFFER_DESC, #68               ; 0172
                                    01  1E 00016         BGEQU   1$
                                        04 00018         RET
                                    5E  DD 00019 1$:     PUSHL   SP                              ; 0179
                                    04  AC  DD 0001B     PUSHL   BUFFER_DESC
                0000G   CF              02  FB 0001E     CALLS   #2, CHECK_REQUEST
                        01              50  E8 00023     BLBS    STATUS, 2$
                                        04 00026         RET
                        56              6E  D0 00027 2$: MOVL    RQCB, R6                        ; 0188
                                        56  DD 0002A     PUSHL   R6
                                    04  AC  DD 0002C     PUSHL   BUFFER_DESC
                0000G   CF              02  FB 0002F     CALLS   #2, VALID_OPERATOR
                        03              50  E8 00034     BLBS    R0, 4$
                                  016C  31  00037 3$:    BRW     22$
                                    04  AE  9F 0003A 4$:  PUSHAB  OPER_RQCB                      ; 0200
                                        56  DD 0003D     PUSHL   R6
                    0000G   CF          02  FB 0003F     CALLS   #2, FIND_OPERATOR
                            52          50  D0 00044     MOVL    R0, FOUND
                            23          52  E9 00047     BLBC    FOUND, 6$                       ; 0208
                        08  AE      24  A6  D0 0004A     MOVL    36(R6), OCD                     ; 0211
                            50      08  AE  D0 0004F     MOVL    OCD, R0                         ; 0212
                            02      0B  A0  91 00053     CMPB    11(R0), #2
                                        07  12 00057     BNEQ    5$
                        6A  A6      26  A0  B1 00059     CMPW    38(R0), 106(R6)                 ; 0214
                                        D7  12 0005E     BNEQ    3$
                            03      0B  A0  91 00060 5$: CMPB    11(R0), #3                      ; 0215
                                        07  12 00064     BNEQ    6$
                        68  A6      24  A0  D1 00066     CMPL    36(R0), 104(R6)                 ; 0216
                                        CA  12 0006B     BNEQ    3$
                                    18  AE  7C 0006D 6$: CLRQ    MESSAGE_VECTOR                  ; 0223
            04          78  A6          01  E1 00070     BBC     #1, 120(R6), 7$                 ; 0231
                        58  A6          02  8A 00075     BICB2   #2, 88(R6)                      ; 0233
                            1E      5D  A6  E9 00079 7$: BLBC    93(R6), 8$                      ; 0239
            19          34  A6          06  E0 0007D     BBS     #6, 52(R6), 8$                  ; 0240
                        5D  A6          01  8A 00082     BICB2   #1, 93(R6)                      ; 0251
                            50          56  D0 00086     MOVL    R6, R0                          ; 0255
                                    5C  A0  D5 00089     TSTL    92(R0)
                                        0D  12 0008C     BNEQ    8$
                                    60  A6  D5 0008E     TSTL    96(R6)
                                        08  12 00091     BNEQ    8$
                        18  AE  0005807C  8F  D0 00093     MOVL    #360572, MESSAGE_VECTOR         ; 0257
```

```
                50              56   D0 0009B 8$:     MOVL    R6, R0                          ; 0266
                          5C    A0   D5 0009E         TSTL    92(R0)                          ;
                                0A   12 000A1         BNEQ    9$                              ;
                          60    A6   D5 000A3         TSTL    96(R6)                          ; 0267
                                05   12 000A6         BNEQ    9$                              ;
          0A    58      A6         01   E1 000A8     BBC     #1, 88(R6), 10$                 ; 0268
          0D    78      A6         01   E1 000AD 9$:  BBC     #1, 120(R6), 11$                ; 0269
          08    58      A6         01   E1 000B2     BBC     #1, 88(R6), 11$                 ; 0270
                18      AE 0005807C  8F   D0 000B7 10$:  MOVL    #360572, MESSAGE_VECTOR         ; 0272
                01            53     A6   91 000BF 11$:  CMPB    83(R6), #1                      ; 0277
                                     05   12 000C3         BNEQ    12$                             ;
          1B    32      A6         02   E1 000C5         BBC     #2, 50(R6), 14$                 ; 0278
                02            53     A6   91 000CA 12$:  CMPB    83(R6), #2                      ; 0279
                                     09   12 000CE         BNEQ    13$                             ;
                11            31     A6   E9 000D0         BLBC    49(R6), 14$                     ; 0280
          0C    32      A6         02   E1 000D4         BBC     #2, 50(R6), 14$                 ; 0281
                68      A6   38    A6   D1 000D9 13$:  CMPL    56(R6), 104(R6)                 ; 0282
                                     0D   13 000DE         BEQL    15$                             ;
          08    32      A6         02   E0 000E0         BBS     #2, 50(R6), 15$                 ; 0283
                18      AE 0005807C  8F   D0 000E5 14$:  MOVL    #360572, MESSAGE_VECTOR         ; 0285
                              18     AE   D5 000ED 15$:  TSTL    MESSAGE_VECTOR                  ; 0290
                                     03   13 000F0         BEQL    16$                             ;
                                00A0 31   00F2         BRW     21$                             ;
                50                   52   D2 000F5 16$:  MCOML   FOUND, R0                      ; 0309
                03            58     A6   E9 000F8         BLBC    88(R6), 17$                     ; 0301
                                0088 31   00FC         BRW     20$                             ;
                58                   01   D0 000FF 17$:  MOVL    #1, ENABLE_FLAG                 ; 0307
                18      AE 00058001  8F   D0 00102         MOVL    #360449, MESSAGE_VECTOR         ; 0308
                78                   50   E9 0010A         BLBC    R0, 19$                         ; 0309
                5A                   01   D0 0010D         MOVL    #1, FULL_ENABLE                 ; 0317
                              04     AE   9F 00110         PUSHAB  OPER_RQCB                       ; 0318
                         00000000G   8F   DD 00113         PUSHL   #RQCB_K_TYPE                    ;
                0000G   CF                   02   FB 00119     CALLS   #2, ALLOCATE_DS                 ;
                57            04     AE   D0 0011E         MOVL    OPER_RQCB, R7                   ; 0320
          10    A7      10    A6     0084 8F   28 00122     MOVC3   #132, 16(R6), 16(R7)            ;
                                     54   A7   7C 0012A     CLRQ    84(R7)                          ; 0321
                                     5C   A7   7C 0012D     CLRQ    92(R7)                          ; 0323
                                     08   AE   9F 00130     PUSHAB  OCD                             ; 0329
                                     68   A6   DD 00133     PUSHL   104(R6)                         ;
                7E                   53   A6   9A 00136     MOVZBL  83(R6), -(SP)                   ;
                0000G   CF                   03   FB 0013A     CALLS   #3, FIND_OCD                    ;
                0F                   50   E8 0013F         BLBS    R0, 18$                         ;
                                     08   AE   9F 00142     PUSHAB  OCD                             ; 0331
                                     68   A6   DD 00145     PUSHL   104(R6)                         ;
                7E                   53   A6   9A 00148     MOVZBL  83(R6), -(SP)                   ;
                0000G   CF                   03   FB 0014C     CALLS   #3, CREATE_OCD                  ;
          50    08      AE 00000050  8F   C1 00151 18$:  ADDL3   #80, OCD, R0                    ; 0332
                60                   67   0E 0015A         INSQUE  (R7), (R0)                      ;
                51            04     AE   D0 0015D         MOVL    OPER_RQCB, R1                   ; 0333
                50            08     AE   D0 00161         MOVL    OCD, R0                         ;
                      24      A1           50   D0 00165     MOVL    R0, 36(R1)                      ;
                                     46   A0   B6 00169     INCW    70(R0)                          ; 0334
                      0C      AE           01   D0 0016C     MOVL    #1, ARG_LIST                    ; 0335
                      10      AE     7C    A1   9E 00170     MOVAB   124(R1), ARG_LIST+4             ; 0336
                      14      AE           01   D0 00175     MOVL    #1, ARG_LIST+8                  ; 0337
                                     0C   AE   9F 00179     PUSHAB  ARG_LIST                        ; 0338
                                     5B   DD 0017C         PUSHL   R11                             ;
```

```
                                00000000G  00              02  FB 0017E           CALLS   #2, SYS$CMKRNL                                  ;
                                                           31  11 00185 19$:      BRB     24$                                             ; 0301
                                      24                   50  E9 00187 20$:      BLBC    R0, 23$                                         ; 0342
                                    18    AE 0005807C      8F  D0 0018A           MOVL    #360572, MESSAGE_VECTOR                         ; 0348
                                          AE       1C      AE  D4 00192           CLRL    MESSAGE_VECTOR+4                                ; 0349
                                          AE       18      AE  9F 00195 21$:      PUSHAB  MESSAGE_VECTOR                                  ; 0350
                                                           56  DD 00198           PUSHL   R6
                                  0000G   CF               02  FB 0019A           CALLS   #2, FORMAT_MESSAGE
                                                           56  DD 0019F           PUSHL   R6                                              ; 0351
                                  0000G   CF               01  FB 001A1           CALLS   #1, NOTIFY_OPERATOR
                                                           56  DD 001A6 22$:      PUSHL   R6                                              ; 0352
                                  0000G   CF               01  FB 001A8           CALLS   #1, DEALLOCATE_RQCB
                                                           04  001AD             RET                                                     ; 0344
                                                           58  D4 001AE 23$:      CLRL    ENABLE_FLAG                                     ; 0360
                                    18    AE 00058011      8F  D0 001B0           MOVL    #360465, MESSAGE_VECTOR                         ; 0361
                                      52                   6E  D0 001B8 24$:      MOVL    RQCB, R2                                        ; 0368
                                                           52  DD 001BB           PUSHL   R2
                                                           0B  DD 001BD           PUSHL   #11
                                      7E                   01  CE 001BF           MNEGL   #1, -(SP)
                                  0000G   CF               03  FB 001C2           CALLS   #3, CLUSMSG_RQCB_SEND
                                      53       04      AE  D0 001C7           MOVL    OPER_RQCB, R3                                   ; 0379
                                                           53  DD 001CB           PUSHL   R3
                                      7E       5C      A2  7D 001CD           MOVQ    92(R2), -(SP)                                   ; 0377
          7E        58    A2                 01  EF 001D1           EXTZV   #0, #1, 88(R2), -(SP)                                   ; 0376
                                  0000G   CF               04  FB 001D7           CALLS   #4, UPD_OPER_CONTEXT
                                      54                   50  D0 001DC           MOVL    R0, LOST_COVERAGE
                                06    58    A2             01  E1 001DF           BBC     #1, 88(R2), 25$                                 ; 0384
    58    A3                  01          01               58  F0 001E4           INSV    ENABLE_FLAG, #1, #1, 88(R3)                     ; 0386
                                          5C    A3         D5  001EA 25$:         TSTL    92(R3)                                          ; 0391
                                                           55  12 001ED           BNEQ    26$
                                          60    A3         D5  001EF             TSTL    96(R3)
                                                           50  12 001F2           BNEQ    26$
                                      59                   01  D0 001F4           MOVL    #1, FULL_DISABLE                                ; 0394
          50        54    A2             01  EF 001F7           EXTZV   #1, #1, 84(R2), R0                                      ; 0395
    54    A3                  01          01               50  F0 001FD           INSV    R0, #1, #1, 84(R3)
    54    A3                  01          00       54      A2  F0 00203           INSV    84(R2), #0, #1, 84(R3)                          ; 0396
                                                           52  DD 0020A           PUSHL   R2                                              ; 0397
                                  0000G   CF               01  FB 0020C           CALLS   #1, DEALLOCATE_RQCB
                                      6E                   63  0F 00211           REMQUE  (R3), RQCB                                      ; 0398
                                      6E       04      AE  D0 00214           MOVL    OPER_RQCB, RQCB                                 ; 0399
                                      50                   6E  D0 00218           MOVL    RQCB, R0                                        ; 0400
                                    08    AE       24      A0  D0 0021B           MOVL    36(R0), OCD
                                      50       08      AE  D0 00220           MOVL    OCD, R0                                         ; 0401
                                                       46  A0  B7 00224           DECW    70(R0)
                                    0C    AE                 01  D0 00227           MOVL    #1, ARG_LIST                                    ; 0402
                10    AE          04    AE 0000007C      8F  C1 0022B           ADDL3   #124, OPER_RQCB, ARG_LIST+4                     ; 0403
                                          14    AE         D4  00235             CLRL    ARG_LIST+8                                      ; 0404
                                          0C    AE         9F  00238             PUSHAB  ARG_LIST                                        ; 0405
                                                           5B  DD 0023B           PUSHL   R11
                                00000000G  00              02  FB 0023D           CALLS   #2, SYS$CMKRNL
                                      52                   6E  7D 00244 26$:      MOVQ    RQCB, R2                                        ; 0409
                                    08    AE       24      A3  D0 00247           MOVL    36(R3), OCD                                     ; 0408
                                    24    A2       08      AE  D0 0024C           MOVL    OCD, 36(R2)                                     ; 0409
                                      03                   5A  E8 00251           BLBS    FULL_ENABLE, 27$                                ; 0410
                                      3F                   59  E9 00254           BLBC    FULL_DISABLE, 28$
                                    5C    A2                 01  7D 00257 27$:      MOVQ    #1, 92(R2)                                      ; 0426
                                    28    A2                 02  88 0025B           BISB2   #2, 40(R2)                                      ; 0428
```

```
                    1C   AE   D4 0025F          CLRL    MESSAGE_VECTOR+4                        ; 0429
              20    AE   7C   A2 9E 00262        MOVAB   124(R2), MESSAGE_VECTOR+8               ; 0430
              24    AE   74   A2 3C 00267        MOVZWL  116(R2), MESSAGE_VECTOR+12              ; 0431
              28    AE   3C   A2 9E 0026C        MOVAB   60(R2), MESSAGE_VECTOR+16               ; 0432
                         18   AE 9F 00271        PUSHAB  MESSAGE_VECTOR                          ; 0433
                              52 DD 00274        PUSHL   R2
                    0000G CF    02 FB 00276      CALLS   #2, FORMAT_MESSAGE
                              52 DD 0027B        PUSHL   R2                                      ; 0434
                    0000G CF    01 FB 0027D      CALLS   #1, LOG_MESSAGE
                              52 DD 00282        PUSHL   R2                                      ; 0435
                    0000G CF    01 FB 00284      CALLS   #1, NOTIFY_LISTED_OPERATORS
                         0A     59 E9 00289      BLBC    FULL_DISABLE, 28$                       ; 0436
                              52 DD 0028C        PUSHL   R2                                      ; 0438
                    0000G CF    01 FB 0028E      CALLS   #1, NOTIFY_OPERATOR
                         4E     59 E8 00293      BLBS    FULL_DISABLE, 29$                       ; 0453
                             0C   A3 D5 00296 28$: TSTL   12(R3)
                              49 13 00299        BEQL    29$
              F8    AD      0A00 8F 3C 0029B    MOVZWL  #2560, STATUS_DESC                      ; 0460
              FC    AD        2C AE 9E 002A1    MOVAB   STATUS_BUF, STATUS_DESC+4               ; 0461
              18    AE 000580C3  8F D0 002A6    MOVL    #360643, MESSAGE_VECTOR                 ; 0462
                        1C   AE   D4 002AE      CLRL    MESSAGE_VECTOR+4                        ; 0463
              20    AE   7C   A2 9E 002B1        MOVAB   124(R2), MESSAGE_VECTOR+8               ; 0464
              24    AE   F8   AD 9E 002B6        MOVAB   STATUS_DESC, MESSAGE_VECTOR+12          ; 0465
                         F8   AD 9F 002BB        PUSHAB  STATUS_DESC                             ; 0466
                         F8   AD 9F 002BE        PUSHAB  STATUS_DESC
                         5C   A3 9F 002C1        PUSHAB  92(R3)
                    0000G CF    03 FB 002C4      CALLS   #3, INTERPRET_MASK
                         18     50 E9 002C9      BLBC    R0, 29$
                         18   AE 9F 002CC        PUSHAB  MESSAGE_VECTOR                          ; 0469
                              52 DD 002CF        PUSHL   R2
                    0000G CF    02 FB 002D1      CALLS   #2, FORMAT_MESSAGE
                              52 DD 002D6        PUSHL   R2                                      ; 0470
                    0000G CF    01 FB 002D8      CALLS   #1, LOG_MESSAGE
                              52 DD 002DD        PUSHL   R2                                      ; 0471
                    0000G CF    01 FB 002DF      CALLS   #1, NOTIFY_OPERATOR
                         04     5A E9 002E4 29$: BLBC    FULL_ENABLE, 30$                        ; 0481
                             0080 C2 D4 002E7    CLRL    128(R2)                                 ; 0483
                              52 DD 002EB 30$:   PUSHL   R2                                      ; 0484
                    0000G CF    01 FB 002ED      CALLS   #1, DEALLOCATE_RQCB
                         08     54 E9 002F2      BLBC    LOST_COVERAGE, 31$                      ; 0491
                           08   AE DD 002F5      PUSHL   OCD                                     ; 0493
                    0000G CF    01 FB 002F8      CALLS   #1, CHECK_OPER_COVERAGE
                              04 002FD 31$:      RET                                             ; 0495

; Routine Size:  766 bytes,    Routine Base:  $CODE$ + 0000
```

```
;  498          0496  1 GLOBAL ROUTINE OPRENABLE_CLM_HANDLER (BUFFER_DESC : $ref_bblock, CLM : $ref_bblock, LEN) : NOVALUE =
;  499          0497  1
;  500          0498  1 !++
;  501          0499  1 ! Functional description:
;  502          0500  1 !
;  503          0501  1 !       This routine is the handler for all CLM_OPRENABLE messages received by OPCOM.  The message
;  504          0502  1 !       contains RQCB information from an OPCOM process on a remote node.  The RQCB describes an
;  505          0503  1 !       operator on the remote node which the remote OPCOM has just enabled or disabled.
;  506          0504  1 !
;  507          0505  1 !
;  508          0506  1 ! Input:
;  509          0507  1 !
;  510          0508  1 !       BUFFER_DESC -   The address of a quadword buffer descriptor that describes the buffer
;  511          0509  1 !                       containing the complete message (the $SNDOPR header plus the CLMRQCB
;  512          0510  1 !                       structure)
;  513          0511  1 !       CLM -           A pointer to start of the CLMRQCB structure embedded in the message.
;  514          0512  1 !                       The RQCB is the first entry in the message, so we can access CLMRQCB
;  515          0513  1 !                       or RQCB fields with the same pointer.
;  516          0514  1 !       LEN -           Length of the CLMRQCB structure
;  517          0515  1 !
;  518          0516  1 ! Implicit Input:
;  519          0517  1 !
;  520          0518  1 !       None.
;  521          0519  1 !
;  522          0520  1 ! Output:
;  523          0521  1 !
;  524          0522  1 !       None.
;  525          0523  1 !
;  526          0524  1 ! Implict output:
;  527          0525  1 !
;  528          0526  1 !       None.
;  529          0527  1 !
;  530          0528  1 ! Side effects:
;  531          0529  1 !
;  532          0530  1 !       None.
;  533          0531  1 !
;  534          0532  1 ! Routine value:
;  535          0533  1 !
;  536          0534  1 !       None.
;  537          0535  1 !--
;  538          0536  1
;  539          0537  2 BEGIN                                           ! Start of OPRENABLE_CLM_HANDLER
;  540          0538  2
;  541          0539  2 LOCAL
;  542          0540  2         STATUS_DESC     : $desc_block,          ! Descriptor for status message
;  543          0541  2         STATUS_BUF      : $bblock [OPC$K_MAXREAD], ! Buffer for status message
;  544          0542  2         MESSAGE_VECTOR  : VECTOR [5,LONG],      ! Message vector
;  545          0543  2         MSG             : $ref_bblock,          ! Pointer to user request text
;  546          0544  2         ARG_LIST        : VECTOR [3],           ! Arguement list
;  547          0545  2         ENABLE_FLAG     : LONG,                 ! 1 if ENABLE, 0 if DISABLE
;  548          0546  2         FULL_DISABLE    : LONG,                 ! Boolean
;  549          0547  2         FULL_ENABLE     : LONG,                 ! Boolean
;  550          0548  2         FOUND           : LONG,                 ! Boolean
;  551          0549  2         LOST_COVERAGE   : LONG,                 ! Boolean
;  552          0550  2         OCD             : $ref_bblock,          ! OCD structure
;  553          0551  2         RQCB            : $ref_bblock,          ! ReQuest Context Block
;  554          0552  2         OPER_RQCB       : $ref_bblock,          ! Known operator RQCB
```

```
;  555          0553  2          STATUS          : LONG;
;  556          0554  2
;  557          0555  2 FULL_ENABLE = FALSE;
;  558          0556  2 FULL_DISABLE = FALSE;
;  559          0557  2 !
;  560          0558  2 ! Check the version number of the message.  If the message is from any other version,
;  561          0559  2 ! simply ignore it.
;  562          0560  2 !
;  563          0561  2 IF .CLM [CLM_B_DS_VERSION] NEQ CLMRQCB_K_DS_VERSION
;  564          0562  2 THEN
;  565          0563  2     RETURN DUMP_LOG_FILE (.BUFFER_DESC, %ASCID 'CLM__OPRENABLE mismatch');
;  566          0564  2 !
;  567          0565  2 ! Allocate an RQCB and convert the message RQCB into the new RQCB
;  568          0566  2 !
;  569          0567  2 IF NOT CLUSMSG_CONV_CLM_RQCB (.CLM, RQCB)
;  570          0568  2 THEN
;  571          0569  2     RETURN DUMP_LOG_FILE (.BUFFER_DESC, ascid_INVALIDRQCB);
;  572          0570  2 !
;  573          0571  2 ! See if the operator is already known to OPCOM.  This entails scanning down the appropriate operator
;  574          0572  2 ! list and comparing the device names for equality.  FIND_OPERATOR will set RQCB [RQCB_L_OCD] if it
;  575          0573  2 ! finds a match.
;  576          0574  2 !
;  577          0575  2 FOUND = FIND_OPERATOR (.RQCB, OPER_RQCB);
;  578          0576  2 !
;  579          0577  2 ! If the operator is known to OPCOM, then the requestor's UIC must agree with the UIC of the known
;  580          0578  2 ! operator.  If it does not, then dismiss the request without notifying the requestor, as this
;  581          0579  2 ! would only allow a user to cause an operator to be plauged with error messages for a request he
;  582          0580  2 ! did not send.
;  583          0581  2 !
;  584          0582  2 IF .FOUND
;  585          0583  2 THEN
;  586          0584  3     BEGIN
;  587          0585  3     OCD = .RQCB [RQCB_L_OCD];                           ! Get the OCD address
;  588          0586  4     IF ((.OCD [OCD_B_SCOPE] EQL OPC$K_GROUP) AND        ! If GROUP OCD, the group fields
;  589          0587  5         (.OCD [$BYTEOFFSET (OCD_L_UIC),16,16,0] NEQ     ! of the UIC must be the same for both
;  590          0588  4          .RQCB [$BYTEOFFSET (RQCB_L_UIC),16,16,0]))     ! the known operator and the requestor.
;  591          0589  4     OR ((.OCD [OCD_B_SCOPE] EQL OPC$K_USER) AND         ! If USER OCD, the UICs must be equal.
;  592          0590  4         (.OCD [OCD_L_UIC] NEQ .RQCB [RQCB_L_UIC]))
;  593          0591  3     THEN
;  594          0592  4         BEGIN                                           ! Dismiss the request
;  595          0593  4         WRITE_LOG_FILE (%ASCID 'bad uic in clmrqcb');
;  596          0594  4         DEALLOCATE_RQCB (.RQCB);
;  597          0595  4         RETURN;
;  598          0596  3         END;
;  599          0597  2     END;
;  600          0598  2 MESSAGE_VECTOR [0] = 0;                                ! Assume no error
;  601          0599  2 MESSAGE_VECTOR [1] = 0;
;  602          0600  2 !
;  603          0601  2 ! Finish the request.
;  604          0602  2 !
;  605          0603  2 IF NOT .$bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_DISABLE]
;  606          0604  2 THEN
;  607          0605  3     BEGIN
;  608          0606  3     !
;  609          0607  3     ! This is an enable request.
;  610          0608  3     !
;  611          0609  3     ENABLE_FLAG = 1;                                ! Set bit value
```

```
;  612          0610  3     MESSAGE_VECTOR [0] = OPC$_TERMENAB;          ! Set message code
;  613          0611  3     IF NOT .FOUND
;  614          0612  3     THEN
;  615          0613  4         BEGIN
;  616          0614  4         !
;  617          0615  4         ! The operator is not known to OPCOM.  An operator RQCB
;  618          0616  4         ! must be created and inserted onto the appropriate OCD
;  619          0617  4         ! operator list so the generalized ENABLE logic will work.
;  620          0618  4         !
;  621          0619  4         FULL_ENABLE = TRUE;
;  622          0620  4         ALLOCATE_DS (RQCB_K_TYPE, OPER_RQCB);                   ! Clone a new RQCB
;  623          0621  4         CH$MOVE (RQCB_K_OVERLAY_SIZE,                           ! Copy old RQCB
;  624          0622  4                  RQCB [RQCB_T_OVERLAY], OPER_RQCB [RQCB_T_OVERLAY]);
;  625          0623  4         OPER_RQCB [RQCB_L_OPTIONS] = 0;                         ! Zero option mask
;  626          0624  4         OPER_RQCB [RQCB_L_RQ_OPTIONS] = 0;                      ! Zero option mask
;  627          0625  4         OPER_RQCB [RQCB_L_ATTNMASK1] = 0;                       ! Zero mask for enable
;  628          0626  4         OPER_RQCB [RQCB_L_ATTNMASK2] = 0;                       ! Zero mask for enable
;  629          0627  4         !
;  630          0628  4         ! Insert the operator RQCB into the operator list in the appropriate
;  631          0629  4         ! OCD.  If no OCD is found, then a new one must be created.
;  632          0630  4         !
;  633          0631  4         IF NOT FIND_OCD (.RQCB [RQCB_B_SCOPE], .RQCB [RQCB_L_UIC], OCD)
;  634          0632  4         THEN
;  635          0633  4             CREATE_OCD (.RQCB [RQCB_B_SCOPE], .RQCB [RQCB_L_UIC], OCD);
;  636          0634  4         INSQUE (.OPER_RQCB, OCD [OCD_[_OPERFLINK]);     ! Insert cloned RQCB onto operator list
;  637          0635  4         OPER_RQCB [RQCB_L_OCD] = .OCD;                  ! Set pointer to OCD
;  638          0636  4         OCD [OCD_W_OPERCOUNT] = .OCD [OCD_W_OPERCOUNT]+1;! Increment operator count
;  639          0637  3         END;
;  640          0638  3     END
;  641          0639  2 ELSE
;  642          0640  2     IF NOT .FOUND
;  643          0641  2     THEN
;  644          0642  3         BEGIN
;  645          0643  3         !
;  646          0644  3         ! The user is trying to disable a nonexistent operator.
;  647          0645  3         !
;  648          0646  3         MESSAGE_VECTOR [0] = OPC$_ILLRQST;
;  649          0647  3         MESSAGE_VECTOR [1] = 0;
;  650          0648  3         FORMAT_MESSAGE (.RQCB, MESSAGE_VECTOR);
;  651          0649  3         NOTIFY_OPERATOR (.RQCB);
;  652          0650  3         DEALLOCATE_RQCB (.RQCB);
;  653          0651  3         RETURN;
;  654          0652  3         END
;  655          0653  2     ELSE
;  656          0654  3         BEGIN
;  657          0655  3         !
;  658          0656  3         ! This is a legitimate DISABLE request.
;  659          0657  3         !
;  660          0658  3         ENABLE_FLAG = 0;                                ! Set for disable request
;  661          0659  3         MESSAGE_VECTOR [0] = OPC$_TERMDSBL;
;  662          0660  2         END;
;  663          0661  2 !
;  664          0662  2 ! Update the OCD count vector for each bit present in the bit mask. The count will be decremented
;  665          0663  2 ! for a DISABLE, incremented for an ENABLE. Also update the OCD operator intrest mask, and the
;  666          0664  2 ! corresponding intrest mask in the operator RQCB.  Note if there was a potential lose of operator
;  667          0665  2 ! coverage on any outstanding requests.
;  668          0666  2 !
```

```
;  669          0667  2 LOST_COVFRAGE = UPD_OPER_CONTEXT (.$bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_DISABLE],
;  670          0668  2                                  .RQCB [RQCB_L_ATTNMASK1],
;  671          0669  2                                  .RQCB [RQCB_L_ATTNMASK2],
;  672          0670  2                                  .OPER_RQCB
;  673          0671  2                                 );
;  674          0672  2 !
;  675          0673  2 ! Set the PERMOPER bit as desired.
;  676          0674  2 !
;  677          0675  2 IF .$bblock [RQCB [RQCB_L_RQ_OPTIONS], OPC$V_PERMOPER]
;  678          0676  2 THEN
;  679          0677  2     $bblock [OPER_RQCB [RQCB_L_RQ_OPTIONS], OPC$V_PERMOPER] = .ENABLE_FLAG;
;  680          0678  2 !
;  681          0679  2 ! If the OPER_RQCB attention mask went to 0, then disable the operator terminal.
;  682          0680  2 !
;  683          0681  3 IF (.OPER_RQCB [RQCB_L_ATTNMASK1] EQL 0) AND (.OPER_RQCB [RQCB_L_ATTNMASK2] EQL 0)
;  684          0682  2 THEN
;  685          0683  3     BEGIN
;  686          0684  3     FULL_DISABLE = TRUE;
;  687          0685  3     $bblock [OPER_RQCB [RQCB_L_OPTIONS], OPC$V_NOBRD] = .$bblock [RQCB [RQCB_L_OPTIONS], OPC$V_NOBRD];
;  688          0686  3     $bblock [OPER_RQCB [RQCB_L_OPTIONS], OPC$V_NOLOG] = .$bblock [RQCB [RQCB_L_OPTIONS], OPC$V_NOLOG];
;  689          0687  3     DEALLOCATE_RQCB (.RQCB);                    ! Get rid of the old RQCB
;  690          0688  3     REMQUE (.OPER_RQCB, RQCB);                  ! Remove operator RQCB from list
;  691          0689  3     RQCB = .OPER_RQCB;                          ! Copy the RQCB address
;  692          0690  3     OCD = .RQCB [RQCB_L_OCD];                   ! Get OCD address
;  693          0691  3     OCD [OCD_W_OPERCOUNT] = .OCD [OCD_W_OPERCOUNT] - 1;
;  694          0692  2     END;
;  695          0693  2 !
;  696          0694  2 OCD = .OPER_RQCB [RQCB_L_OCD];
;  697          0695  2 RQCB [RQCB_L_OCD] = .OCD;
;  698          0696  2 IF .FULL_ENABLE OR .FULL_DISABLE
;  699          0697  2 THEN
;  700          0698  3     BEGIN
;  701          0699  3     !
;  702          0700  3     ! Send the ENABLE/DISABLE message to all CENTRAL operators connected to this OCD.  Do not notify
;  703          0701  3     ! the requestor, she is on another node.
;  704          0702  3     !
;  705          0703  3     RQCB [RQCB_L_ATTNMASK1] = OPC$M_NM_CENTRL;
;  706          0704  3     RQCB [RQCB_L_ATTNMASK2] = 0;
;  707          0705  3     RQCB [HDR_V_BRD] = TRUE;
;  708          0706  3     MESSAGE_VECTOR [1] = 0;
;  709          0707  3     MESSAGE_VECTOR [2] = RQCB [RQCB_L_OPER_LEN];
;  710          0708  3     MESSAGE_VECTOR [3] = .RQCB [RQCB_B_USERNAMELEN];
;  711          0709  3     MESSAGE_VECTOR [4] = RQCB [RQCB_T_USERNAME];
;  712          0710  3     FORMAT_MESSAGE (.RQCB, MESSAGE_VECTOR);
;  713          0711  3     LOG_MESSAGE (.RQCB);
;  714          0712  3     NOTIFY_LISTED_OPERATORS (.RQCB);
;  715          0713  2     END;
;  716          0714  2 !
;  717          0715  2 ! Deallocate the RQCB.  If this was a full enable, clear the pointer to the operator device name,
;  718          0716  2 ! as both the RQCB and the OPER_RQCB point to the same device name.  Deallocating the RQCB in this
;  719          0717  2 ! case would cause the OPER_RQCB device name to evaporate.
;  720          0718  2 !
;  721          0719  2 IF .FULL_ENABLE
;  722          0720  2 THEN
;  723          0721  2     RQCB [RQCB_L_OPER_PTR] = 0;
;  724          0722  2 DEALLOCATE_RQCB (.RQCB);
;  725          0723  2 !
```

```
;  726          0724  2 ! If there was a potential loss of operator coverage,check the remaining requests queued to this OCD.
;  727          0725  2 ! The routine will cancel any requests that no longer have operator coverage.
;  728          0726  2 !
;  729          0727  2 IF .LOST_COVERAGE
;  730          0728  2 THEN
;  731          0729  2     CHECK_OPER_COVERAGE (.OCD);
;  732          0730  2
;  733          0731  1 END;                                    ! End of OPRENABLE_CLM_HANDLER


                                                                .PSECT  $PLIT$,NOWRT,NOEXE,2

20 45 4C 42 41 4E 45 52 50 4F 5F 5F 4D 4C 43  00000 P.AAB:   .ASCII  \CLM__OPRENABLE mismatch\<0>
                  00 68 63 74 61 6D 73 69 6D  0000F
                                 010E0017     00018 P.AAA:   .LONG   17694743
                                 00000000'    000           .ADDRESS P.AAB
72 6D 6C 63 20 6E 69 20 63 69 75 20 64 61 62  00020 P.AAD:   .ASCII  \bad uic in clmrqcb\<0><0>
                              00 00 62 63 71  0002F
                                 010E0012     00034 P.AAC:   .LONG   17694738
                                 00000000'    00038          .ADDRESS P.AAD

                                                                .EXTRN  ASCID_INVALIDRQCB

                                                                .PSECT  $CODE$,NOWRT,2

                                       0FFC 00000               .ENTRY  OPRENABLE_CLM_HANDLER, Save R2,R3,R4,R5,R6,-; 0496
                                                                        R7,R8,R9,R10,R11
                                 5B    0000G   CF  9E 00002      MOVAB   DEALLOCATE_RQCB, R11
                                 5E    F5CC    CE  9E 00007      MOVAB   -2612(SP), SP
                                               59  7C 0C00C      CLRQ    FULL_ENABLE                                     ; 0555
                                 52      08    AC  D0 0000E      MOVL    CLM, R2                                         ; 0561
                                 02      02    A2  91 00012      CMPB    2(R2), #2
                                               06  13 00016      BEQL    1$
                                       0000'   CF  9F 00018      PUSHAB  P.AAA                                           ; 0563
                                               10  11 0001C      BRB     2$
                                       4004    8F  BB 0001E 1$:  PUSHR   #^M<R2,SP>                                     ; 0567
                0000G  CF                      02  FB 00022      CALLS   #2, CLUSMSG_CONV_CLM_RQCB
                       0D                      50  E8 00027      BLBS    R0, 3$
                                       0000G   CF  9F 0002A      PUSHAB  ASCID_INVALIDRQCB                               ; 0569
                                         04    AC  DD 0002E 2$:  PUSHL   BUFFER_DESC
                0000G  CF                      02  FB 00031      CALLS   #2, DUMP_LOG_FILE
                                               04  00036      RET
                                         04    AE  9F 00037 3$:  PUSHAB  OPER_RQCB                                     ; 0575
                       57                04    AE  D0 0003A      MOVL    RQCB, R7
                                               57  DD 0003E      PUSHL   R7
                0000G  CF                      02  FB 00040      CALLS   #2, FIND_OPERATOR
                       52                      50  D0 00045      MOVL    R0, FOUND
                       2F                      52  E9 00048      BLBC    FOUND, 6$                                       ; 0582
                  08   AE                24    A7  D0 0004B      MOVL    36(R7), OCD                                     ; 0585
                       50                08    AE  D0 00050      MOVL    OCD, R0                                         ; 0586
                       02                0B    A0  91 00054      CMPB    11(R0), #2
                                               07  12 00058      BNEQ    4$
                  6A   A7                26    A0  B1 0005A      CMPW    38(R0), 106(R7)                                 ; 0588
                                               0D  12 0005F      BNEQ    5$
                       03                0B    A0  91 00061 4$:  CMPB    11(R0), #3                                      ; 0589
                                               13  12 00065      BNEQ    6$
```

```
                    68   A7    24   A0  D1 00067          CMPL    36(R0), 104(R7)                                 ; 0590
                                    0C  13 0006C          BEQL    6$                                              ;
                         0000'      CF  9F 0006E 5$:      PUSHAB  P.AAC                                           ; 0593
                    0000G CF              01  FB 00072          CALLS   #1, WRITE_LOG_FILE                      ;
                                    0098  31 00077          BRW     10$                                             ; 0594
                              18    AE  7C 0007A 6$:      CLRQ    MESSAGE_VECTOR                                  ; 0598
                         50         52  D2 0007D          MCOML   FOUND, R0                                       ; 0611
                         6F    58   A7  E8 00080          BLBS    88(R7), 9$                                      ; 0603
                         58         01  D0 00084          MOVL    #1, ENABLE_FLAG                                 ; 0609
                    18   AE 00058001   8F  D0 00087          MOVL    #360449, MESSAGE_VECTOR                         ; 0610
                         5F         50  E9 0008F          BLBC    R0, 8$                                          ; 0611
                         59         01  D0 00092          MOVL    #1, FULL_ENABLE                                 ; 0619
                              04    AE  9F 00095          PUSHAB  OPER_RQCB                                       ; 0620
                         000000000G 8F  DD 00098          PUSHL   #RQCB_K_TYPE                                    ;
                    0000G CF              02  FB 0009E          CALLS   #2, ALLOCATE_DS                         ;
                         56         04  AE  D0 000A3          MOVL    OPER_RQCB, R6                                   ; 0622
               10   A6    10   A7  0084  8F  28 000A7          MOVC3   #132, 16(R7), 16(R6)                    ;
                              54   A6  7C 000AF          CLRQ    84(R6)                                          ; 0623
                              5C   A6  7C 000B2          CLRQ    92(R6)                                          ; 0625
                              08   AE  9F 000B5          PUSHAB  OCD                                             ; 0631
                              68   A7  DD 000B8          PUSHL   104(R7)                                         ;
                         7E   53   A7  9A 000BB          MOVZBL  83(R7), -(SP)                                   ;
                    0000G CF              03  FB 000BF          CALLS   #3, FIND_OCD                            ;
                         0F         50  E8 000C4          BLBS    R0, 7$                                          ;
                              08   AE  9F 000C7          PUSHAB  OCD                                             ; 0633
                              68   A7  DD 000CA          PUSHL   104(R7)                                         ;
                         7E   53   A7  9A 000CD          MOVZBL  83(R7), -(SP)                                   ;
                    0000G CF              03  FB 000D1          CALLS   #3, CREATE_OCD                          ;
                    50   08   AE 00000050   8F  C1 000D6 7$:      ADDL3   #80, OCD, R0                            ; 0634
                         60         66  0E 000DF          INSQUE  (R6), (R0)                                      ;
                         51    04   AE  D0 000E2          MOVL    OPER_RQCB, R1                                   ; 0635
                         50    08   AE  D0 000E6          MOVL    OCD, R0                                         ;
                    24   A1              50  D0 000EA          MOVL    R0, 36(R1)                                      ;
                              46   A0  B6 000EE          INCW    70(R0)                                          ; 0636
                                   2F  11 000F1 8$:      BRB     12$                                             ; 0603
                         22         50  E9 000F3 9$:      BLBC    R0, 11$                                         ; 0640
                    18   AE 0005807C   8F  D0 000F6          MOVL    #360572, MESSAGE_VECTOR                         ; 0646
                              1C   AE  D4 000FE          CLRL    MESSAGE_VECTOR+4                                ; 0647
                              18   AE  9F 00101          PUSHAB  MESSAGE_VECTOR                                  ; 0648
                                   57  DD 00104          PUSHL   R7                                              ;
                    0000G CF              02  FB 00106          CALLS   #2, FORMAT_MESSAGE                      ;
                                   57  DD 0010B          PUSHL   R7                                              ; 0649
                    0000G CF              01  FB 0010D          CALLS   #1, NOTIFY_OPERATOR                     ;
                                   57  DD 00112 10$:      PUSHL   R7                                              ; 0650
                         6B              01  FB 00114          CALLS   #1, DEALLOCATE_RQCB                     ;
                                   04  00117          RET                                                     ; 0642
                                   58  D4 00118 11$:      CLRL    ENABLE_FLAG                                     ; 0658
                    18   AE 00058011   8F  D0 0011A          MOVL    #360465, MESSAGE_VECTOR                         ; 0659
                         53    04   AE  D0 00122 12$:      MOVL    OPER_RQCB, R3                                   ; 0670
                                   53  DD 00126          PUSHL   R3                                              ;
                         52    04   AE  D0 00128          MOVL    RQCB, R2                                        ; 0669
                         7E    5C   A2  7D 0012C          MOVQ    92(R2), -(SP)                                   ; 0668
               7E        58   A2              01  EF 00130          EXTZV   #0, #1, 88(R2), -(SP)                   ; 0667
                    0000G CF              04  FB 00136          CALLS   #4, UPD_OPER_CONTEXT                    ;
                         54              50  D0 0013B          MOVL    R0, LOST_COVERAGE                       ;
                              06   58   A2              01  E1 0013E          BBC     #1, 88(R2), 13$                 ; 0675
               58   A3              01              01  58  F0 00143          INSV    ENABLE_FLAG, #1, #1, 88(R3)     ; 0677
```

```
                                                 5C  A3  D5 00149 13$:    TSTL    92(R3)                                  ; 0681
                                                     36  12 0014C        BNEQ    14$                                     ;
                                                 60  A3  D5 0014E        TSTL    96(R3)                                  ;
                                                     31  12 00151        BNEQ    14$                                     ;
                                           5A        01  D0 00153        MOVL    #1, FULL_DISABLE                        ; 0684
        50      54  A2                     01        01  EF 00156        EXTZV   #1, #1, 84(R2), R0                      ; 0685
    54  A3          01                     01        50  F0 0015C        INSV    R0, #1, #1, 84(R3)                      ;
    54  A3          01                     00    54  A2  F0 00162        INSV    84(R2), #0, #1, 84(R3)                  ; 0686
                                                     52  DD 00169        PUSHL   R2                                      ; 0687
                                           6B        01  FB 0016B        CALLS   #1, DEALLOCATE_RQCB                     ;
                                           6E        63  0F 0016E        REMQUE  (R3), RQCB                              ; 0688
                                           6E    04  AE  D0 00171        MOVL    OPER_RQCB, RQCB                         ; 0689
                                           50        6E  D0 00175        MOVL    RQCB, R0                                ; 0690
                                     08    AE    24  A0  D0 00178        MOVL    36(R0), OCD                             ;
                                           50    08  AE  D0 0017D        MOVL    OCD, R0                                 ; 0691
                                           46  A0  B7 00181              DECW    70(R0)                                  ;
                                           50    04  AE  D0 00184 14$:    MOVL    OPER_RQCB, R0                           ; 0694
                                     08    AE    24  A0  D0 00188        MOVL    36(R0), OCD                             ;
                                           52        6E  D0 0018D        MOVL    RQCB, R2                                ; 0695
                                     24    A2    08  AE  D0 00190        MOVL    OCD, 36(R2)                             ;
                                           03        59  E8 00195        BLBS    FULL_ENABLE, 15$                        ; 0696
                                           32        5A  E9 00198        BLBC    FULL_DISABLE, 16$                       ;
                                     5C    A2        01  7D 0019B 15$:    MOVQ    #1, 92(R2)                              ; 0703
                                     28    A2        02  88 0019F        BISB2   #2, 40(R2)                              ; 0705
                                                 1C  AE  D4 001A3        CLRL    MESSAGE_VECTOR+4                        ; 0706
                                     20    AE    7C  A2  9E 001A6        MOVAB   124(R2), MESSAGE_VECTOR+8               ; 0707
                                     24    AE    74  A2  3C 001AB        MOVZWL  116(R2), MESSAGE_VECTOR+12              ; 0708
                                     28    AE    3C  A2  9E 001B0        MOVAB   60(R2), MESSAGE_VECTOR+16               ; 0709
                                                 18  AE  9F 001B5        PUSHAB  MESSAGE_VECTOR                          ; 0710
                                                     52  DD 001B8        PUSHL   R2                                      ;
                                   0000G   CF            02  FB 001BA    CALLS   #2, FORMAT_MESSAGE                      ;
                                                         52  DD 001BF    PUSHL   R2                                      ; 0711
                                   0000G   CF            01  FB 001C1    CALLS   #1, LOG_MESSAGE                         ;
                                                         52  DD 001C6    PUSHL   R2                                      ; 0712
                                   0000G   CF            01  FB 001C8    CALLS   #1, NOTIFY_LISTED_OPERATORS             ;
                                           04            59  E9 001CD 16$:    BLBC    FULL_ENABLE, 17$                    ; 0719
                                                 0080    C2  D4 001D0        CLRL    128(R2)                             ; 0721
                                                         52  DD 001D4 17$:    PUSHL   R2                                  ; 0722
                                           6B            01  FB 001D6        CALLS   #1, DEALLOCATE_RQCB                 ;
                                           08            54  E9 001D9        BLBC    LOST_COVERAGE, 18$                  ; 0727
                                             08      AE  DD 001DC            PUSHL   OCD                                 ; 0729
                                   0000G   CF            01  FB 001DF        CALLS   #1, CHECK_OPER_COVERAGE             ;
                                                         04  001E4 18$:      RET                                         ; 0731

; Routine Size:  485 bytes,    Routine Base:  $CODE$ + 02FE

;  734          0732  1
;  735          0733  1 END                                                 ! End of OPRENABLE
;  736          0734  0 ELUDOM



;                                  PSECT SUMMARY
```

```
;       Name                    Bytes                   Attributes

;  $CODE$                       1251  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                         60  NOVEC,NOWRT,  RD ,NOEXE,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        17         0        1000        00:01.8
;  _$255$DUA28:[OPCOM.OBJ]OPCOMLIB.L32;1  633        43         6          43        00:00.9



;                                   COMMAND QUALIFIERS

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

; Size:         1251 code + 60 data bytes
; Run Time:       00:26.0
; Elapsed Time.   01:22.7
; Lines/CPU Min:   1695
; Lexemes/CPU-Min: 18820
; Memory Used:  244 pages
; Compilation Complete
```

0291 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

REPLYBRD
LIS

SECURITY
LIS

OPRENABLE
LIS

REPLYMAIN
LIS

RQSTMAIN
LIS