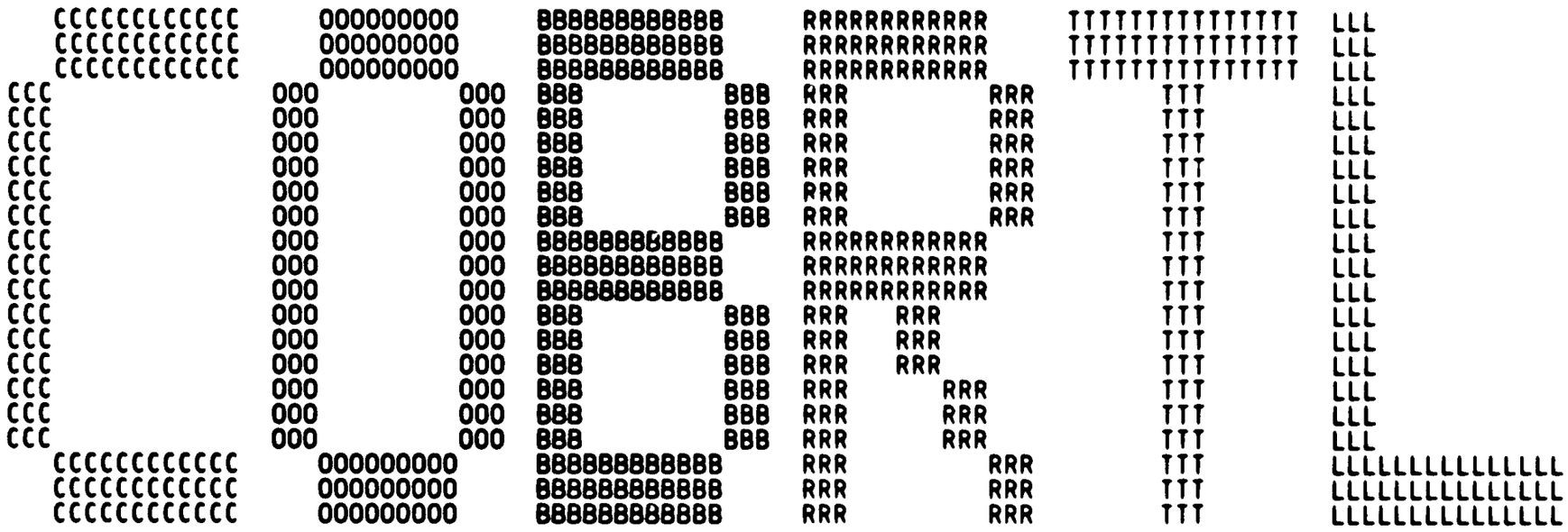
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRRRRRRRRRR TTTTTTTTTTTTTTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBBBBBBBBBB RRRRRRRRRRR TTT LLL
CCC 000 000 BBBBBBBBBBBB RRRRRRRRRRR TTT LLL
CCC 000 000 BBBBBBBBBBBB RRRRRRRRRRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCC 000 000 BBB BBB RRR RRR TTT LLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRR RRR TTT LLLLLLLLLLLLLLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRR RRR TTT LLLLLLLLLLLLLLL
CCCCCCCCCCC 000000000 BBBBBBBBBBBB RRR RRR TTT LLLLLLLLLLLLLLL

```
  CCCCCCCC    OOOOOO    BBBBBBBB    RRRRRRRR    EEEEEEEEEE    SSSSSSSS  TTTTTTTTTT  VV      VV    AAAAAA
  CCCCCCCC    OOOOOO    BBBBBBBB    RRRRRRRR    EEEEEEEEEE    SSSSSSSS  TTTTTTTTTT  VV      VV    AAAAAA
CC           OO      OO  BB      BB  RR      RR  EE          SS            TT      VV      VV  AA      AA
CC           OO      OO  BB      BB  RR      RR  EE          SS            TT      VV      VV  AA      AA
CC           OO      OO  BB      BB  RR      RR  EE          SS            TT      VV      VV  AA      AA
CC           OO      OO  BB      BB  RR      RR  EE          SS            TT      VV      VV  AA      AA
CC           OO      OO  BBBBBBBB    RRRRRRRR    EEEEEEEE      SSSSSS      TT      VV      VV  AA      AA
CC           OO      OO  BBBBBBBB    RRRRRRRR    EEEEEEEE      SSSSSS      TT      VV      VV  AA      AA
CC           OO      OO  BB      BB  RR  RR      EE                  SS    TT      VV      VV  AAAAAAAAAA
CC           OO      OO  BB      BB  RR  RR      EE                  SS    TT      VV      VV  AAAAAAAAAA
CC           OO      OO  BB      BB  RR    RR    EE                  SS    TT        VV  VV    AA      AA
CC           OO      OO  BB      BB  RR    RR    EE                  SS    TT        VV  VV    AA      AA
  CCCCCCCC    OOOOOO    BBBBBBBB    RR      RR  EEEEEEEEEE  SSSSSSSS      TT          VV      AA      AA
  CCCCCCCC    OOOOOO    BBBBBBBB    RR      RR  EEEEEEEEEE  SSSSSSSS      TT          VV      AA      AA


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 COB$$RESTVA(
;     2          0002  0                 IDENT = '1-005'                 ! file: COBRESTVA.B32
;     3          0003  0                 ) =
;     4          0004  1 BEGIN
;     5          0005  1
;     6          0006  1 !
;     7          0007  1 !*****************************************************************************
;     8          0008  1 !*                                                                           *
;     9          0009  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                  *
;    10          0010  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                    *
;    11          0011  1 !*  ALL RIGHTS RESERVED.                                                     *
;    12          0012  1 !*                                                                           *
;    13          0013  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED    *
;    14          0014  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
;    15          0015  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    16          0016  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    17          0017  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    18          0018  1 !*  TRANSFERRED.                                                             *
;    19          0019  1 !*                                                                           *
;    20          0020  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    21          0021  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    22          0022  1 !*  CORPORATION.                                                             *
;    23          0023  1 !*                                                                           *
;    24          0024  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
;    25          0025  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                  *
;    26          0026  1 !*                                                                           *
;    27          0027  1 !*                                                                           *
;    28          0028  1 !*****************************************************************************
;    29          0029  1 !
;    30          0030  1 !
;    31          0031  1 !++
;    32          0032  1 ! FACILITY:     COBOL SUPPORT
;    33          0033  1 !
;    34          0034  1 ! ABSTRACT
;    35          0035  1 !
;    36          0036  1 !
;    37          0037  1 !
;    38          0038  1 ! ENVIRONMENT:  Vax-11 User Mode
;    39          0039  1 !
;    40          0040  1 ! AUTHOR: MLJ , CREATION DATE: 05-MAY-1979
;    41          0041  1 !
;    42          0042  1 ! MODIFIED BY:
;    43          0043  1 !
;    44          0044  1 ! 1-001 - Original. MLJ  05-MAY-1979
;    45          0045  1 ! 1-002 - Added boilerplate and comments. RKR 24-AUG-1979
;    46          0046  1 ! 1-003 - Use RTLLIB for LIBRARY and define psect via macros. RKR 18-SEPT-79
;    47          0047  1 ! 1-004 - Rename to COBRESTVA from former SYSRESTVA  RKR 24-SEPT-79
;    48          0048  1 !         (Entry point becomes COB$$RESTVA)
;    49          0049  1 ! 1-005 - Make module name match entry point, cosmetic changes.
;    50          0050  1 !         RKR 18-OCT-79
;    51          0051  1 !
;    52          0052  1 !--
;    53          0053  1
;    54          0054  1 !<BLF/PAGE>
```

```
;    56          0055  1 !
;    57          0056  1 ! SWITCHES
;    58          0057  1 !
;    59          0058  1
;    60          0059  1 SWITCHES ADDRESSING_MODE (EXTERNAL = GENERAL, NONEXTERNAL = WORD_RELATIVE);
;    61          0060  1
;    62          0061  1 !
;    63          0062  1 ! LINKAGES
;    64          0063  1 !
;    65          0064  1 !       NONE
;    66          0065  1 !
;    67          0066  1 ! TABLE OF CONTENTS:
;    68          0067  1 !
;    69          0068  1 FORWARD ROUTINE
;    70          0069  1
;    71          0070  1         COB$$RESTVA ;   ! Restore virtual address from image file
;    72          0071  1 !
;    73          0072  1 ! INCLUDE FILES
;    74          0073  1 !
;    75          0074  1 REQUIRE 'RTLIN:RTLPSECT';       ! Macros for psect declarations
;    76          0169  1 LIBRARY 'RTLLIB';       ! Image header structures and sizes
;    77          0170  1
;    78          0171  1 !
;    79          0172  1 ! MACROS
;    80          0173  1 !
;    81          0174  1 !       NONE
;    82          0175  1 !
;    83          0176  1 ! EQUATED SYMBOLS
;    84          0177  1 !
;    85          0178  1 !       NONE
;    86          0179  1 !
;    87          0180  1 ! PSECT DECLARATIONS:
;    88          0181  1 !
;    89          0182  1 DECLARE_PSECTS (COB) ;  ! Psects for COB$ facility
```

```
;    91          0183  1 GLOBAL ROUTINE COB$$RESTVA (
;    92          0184  1
;    93          0185  1                          INADR,          ! Input addr.'s to be restored
;    94          0186  1                          CHAN,           ! Channel number
;    95          0187  1                          EFN             ! Event flag number
;    96          0188  1
;    97          0189  1                          )=
;    98          0190  1 !
;    99          0191  1 !++     FUNCTIONAL DESCRIPTION
;   100          0192  1 !
;   101          0193  1 !
;   102          0194  1 ! FORMAL PARAMETERS:
;   103          0195  1 !
;   104          0196  1 !
;   105          0197  1 ! INADR.ra.r
;   106          0198  1 !
;   107          0199  1 !       address of a 2-longword array containing the starting and ending
;   108          0200  1 !       virtual addresses in the process's virtual address space of the
;   109          0201  1 !       area to be restored from the image file.
;   110          0202  1 !
;   111          0203  1 ! CHAN.rl.v
;   112          0204  1 !
;   113          0205  1 !       number of the channel on which the image file has been accessed.
;   114          0206  1 !       The file must have been accessed with an RMS $OPEN macro; the
;   115          0207  1 !       file options parameter (FOP) in the FAB must indicate a user
;   116          0208  1 !       file open (UFO) keyword).  The access mode at which the channel
;   117          0209  1 !       was opened must be the same or less privileged than the access
;   118          0210  1 !       mode of the caller.
;   119          0211  1 !
;   120          0212  1 ! EFN.rl.v
;   121          0213  1 !
;   122          0214  1 !       Event flag number to use for I/O on CHAN.
;   123          0215  1 !
;   124          0216  1 !
;   125          0217  1 ! IMPLICIT INPUTS:
;   126          0218  1 !
;   127          0219  1 !       NONE
;   128          0220  1 !
;   129          0221  1 ! IMPLICIT OUTPUTS:
;   130          0222  1 !
;   131          0223  1 !       NONE
;   132          0224  1 !
;   133          0225  1 ! ROUTINE VALUE:
;   134          0226  1 ! COMPLETION CODES:
;   135          0227  1 !
;   136          0228  1 !       SS$_BADIMGHDR   ! Bad image header read
;   137          0229  1 !       SS$_INSFARG     ! Insufficient arguments to this routine
;   138          0230  1 !       SS$_NORMAL      ! Normal successful completion
;   139          0231  1 !       SS$_PAGOWNVIO   ! Page ownership violation
;   140          0232  1 !
;   141          0233  1 !       If read error occurs, RMS error status is returned.
;   142          0234  1 !
;   143          0235  1
;   144          0236  1 !
;   145          0237  1 ! SIDE EFFECTS:
;   146          0238  1 !
;   147          0239  1 !       NONE
```

```
;  148          0240  1 !--
;  149          0241  1 !--
;  150          0242  1 !
;  151          0243  1 !
;  152          0244  2     BEGIN
;  153          0245  2     MAP
;  154          0246  2         INADR:          REF VECTOR;
;  155          0247  2
;  156          0248  2     LOCAL
;  157          0249  2         STATUS,                                 ! System service status
;  158          0250  2         IHDBUF:         BLOCK[512,BYTE],        ! Block buffer for image header
;  159          0251  2         IMGBUF:         VECTOR[512,BYTE],       ! Block buffer for image file
;  160          0252  2         IOSB:           VECTOR[2],              ! I/O status block
;  161          0253  2         ISD:            REF BLOCK[,BYTE],       ! Pointer to image section descriptor
;  162          0254  2         IHDVBN,                                 ! Current VBN of image header
;  163          0255  2         MAXIHDVBN;                              ! Maximum VBN of image header
;  164          0256  2     BUILTIN
;  165          0257  2         ACTUALCOUNT;
;  166          0258  2
;  167          0259  2
;  168          0260  2
;  169          0261  2     !+
;  170          0262  2     ! Make sure enough arguments are present.
;  171          0263  2     !-
;  172          0264  2     IF ACTUALCOUNT() LSS 3 THEN RETURN SS$_INSFARG;
;  173          0265  2
;  174          0266  2
;  175          0267  2
;  176          0268  2     !+
;  177          0269  2     ! Read the first (usually, only) block of the image header.
;  178          0270  2     !-
;  179          0271  2     IHDVBN = 1;
;  180        P 0272  2     STATUS = $QIOW(
;  181        P 0273  2         EFN=.EFN,
;  182        P 0274  2         CHAN=.CHAN,
;  183        P 0275  2         FUNC=IO$_READVBLK,
;  184        P 0276  2         IOSB=IOSB,
;  185        P 0277  2         P1=IHDBUF,
;  186        P 0278  2         P2=512,
;  187          0279  2         P3=1);
;  188          0280  2     IF NOT .STATUS THEN RETURN .STATUS;
;  189          0281  2     IF NOT .IOSB THEN RETURN .IOSB<0,16>;
;  190          0282  2
;  191          0283  2
;  192          0284  2
;  193          0285  2     !+
;  194          0286  2     ! Save the number of blocks in the image header, make sure that the
;  195          0287  2     ! SIZE field is reasonable, and point to the first ISD.
;  196          0288  2     !-
;  197          0289  2     MAXIHDVBN = .IHDBUF[IHD$B_HDRBLKCNT];
;  198          0290  2     IF .IHDBUF[IHD$W_SIZE] GTRU 510 THEN RETURN SS$_BADIMGHDR;
;  199          0291  2     ISD = IHDBUF + .IHDBUF[IHD$W_SIZE];
;  200          0292  2
;  201          0293  2
;  202          0294  2
;  203          0295  2     !+
;  204          0296  2     ! Loop over the requested virtual pages.
```

```
;  205          0297  2      !-
;  206          0298  2      INCRA VPN FROM .(INADR[0])<9,23> TO .(INADR[1])<9,23> DO
;  207          0299  3          BEGIN
;  208          0300  3          LOCAL
;  209          0301  3              SVA,                    ! Starting virtual address
;  210          0302  3              LEN;                    ! Length
;  211          0303  3
;  212          0304  3
;  213          0305  3          !+
;  214          0306  3          ! Find the image section descriptor that describes the current
;  215          0307  3          ! virtual page.  Since these occur in ascending sequence, we
;  216          0308  3          ! just scan forward until the correct one is found, or we
;  217          0309  3          ! discover one that describes a following page, or the end of
;  218          0310  3          ! the table.  In the latter cases, the desired page is not
;  219          0311  3          ! described by any ISD.
;  220          0312  3          !-
;  221          0313  3          WHILE .VPN GEQ .ISD[ISD$V_VPN]+.ISD[ISD$W_PAGCNT] DO
;  222          0314  4              BEGIN
;  223          0315  4
;  224          0316  4
;  225          0317  4              !+
;  226          0318  4              ! Point to the next ISD, and make sure it exists.
;  227          0319  4              !-
;  228          0320  4              ISD = .ISD + .ISD[ISD$W_SIZE];
;  229          0321  4              IF .ISD GTRA IHDBUF+510 THEN RETURN SS$_BADIMGHDR;
;  230          0322  4
;  231          0323  4
;  232          0324  4              !+
;  233          0325  4              ! If the SIZE field is zero, we have reached the end of the
;  234          0326  4              ! table.  This indicates that the requested VPN is higher
;  235          0327  4              ! than any page described by the ISDs.
;  236          0328  4              !-
;  237          0329  4              IF .ISD[ISD$W_SIZE] EQL 0
;  238          0330  4              THEN
;  239          0331  4                  RETURN SS$_PAGOWNVIO;
;  240          0332  4
;  241          0333  4
;  242          0334  4              !+
;  243          0335  4              ! If the SIZE field is -1, the next ISD is in the following
;  244          0336  4              ! block.  Read it and continue searching.  However, make
;  245          0337  4              ! sure not to run off the end of the image header.
;  246          0338  4              !-
;  247          0339  4              IF .ISD[ISD$W_SIZE] EQL %X'FFFF'
;  248          0340  4              THEN
;  249          0341  5                  BEGIN
;  250          0342  5                  IF .IHDVBN EQL .MAXIHDVBN THEN RETURN SS$_BADIMGHDR;
;  251          0343  5                  IHDVBN = .IHDVBN + 1;
;  252        P 0344  5                  STATUS = $QIOW(
;  253        P 0345  5                      EFN=.EFN,
;  254        P 0346  5                      CHAN=.CHAN,
;  255        P 0347  5                      FUNC=IO$_READVBLK,
;  256        P 0348  5                      IOSB=IOSB,
;  257        P 0349  5                      P1=IHDBUF,
;  258        P 0350  5                      P2=512,
;  259          0351  5                      P3=.IHDVBN);
;  260          0352  5                  IF NOT .STATUS THEN RETURN .STATUS;
;  261          0353  5                  IF NOT .IOSB THEN RETURN .IOSB<0,16>;
```

```
;  262          0354  5                ISD = IHDBUF;
;  263          0355  4                END;
;  264          0356  4            END;
;  265          0357  4
;  266          0358  4
;  267          0359  4        !+
;  268          0360  4        ! If the current ISD is for a virtual page that follows the
;  269          0361  4        ! requested one, this indicates that no ISD describes the
;  270          0362  4        ! requested one, since the preceding ISD described pages that
;  271          0363  4        ! precede the requested one.
;  272          0364  4        !-
;  273          0365  4        IF .VPN LSS .ISD[ISD$V_VPN]
;  274          0366  4        THEN
;  275          0367  4            RETURN SS$_PAGOWNVIO;
;  276          0368  4
;  277          0369  4
;  278          0370  4        !+
;  279          0371  4        ! Make sure that the ISD does not describe a global page.  We
;  280          0372  4        ! do not support them.
;  281          0373  4        !-
;  282          0374  4        IF .ISD[ISD$V_GBL]
;  283          0375  4        THEN
;  284          0376  4            RETURN SS$_PAGOWNVIO;
;  285          0377  4
;  286          0378  4
;  287          0379  4        !+
;  288          0380  4        ! Compute the virtual address and length of the area within the
;  289          0381  4        ! current page that will be restored.
;  290          0382  4        !-
;  291          0383  4        SVA = MAXA(.VPN^9, .INADR[0]);                  ! Find base address
;  292          0384  4        LEN = MINA(.VPN^9+511, .INADR[1]) - .SVA + 1;   ! Find length
;  293          0385  4
;  294          0386  4
;  295          0387  4        !+
;  296          0388  4        ! Do the restoration.  If the page is not writable, we will not
;  297          0389  4        ! restore it, since it should not have changed (unless by
;  298          0390  4        ! debugger).  If it is demand zero, just fill the region with
;  299          0391  4        ! zero.  Otherwise, read the appropriate block from the image,
;  300          0392  4        ! and copy the region over.
;  301          0393  4        !-
;  302          0394  4        IF .ISD[ISD$V_WRT]
;  303          0395  4        THEN
;  304          0396  4            IF .ISD[ISD$V_DZRO]
;  305          0397  4            THEN
;  306          0398  4                CH$FILL(0, .LEN, .SVA)
;  307          0399  4            ELSE
;  308          0400  4                BEGIN
;  309        P 0401  4                STATUS = $QIOW(
;  310        P 0402  4                    EFN=.EFN,
;  311        P 0403  4                    CHAN=.CHAN,
;  312        P 0404  4                    FUNC=IO$_READVBLK,
;  313        P 0405  4                    IOSB=IOSB,
;  314        P 0406  4                    P1=IMGBUF,
;  315        P 0407  4                    P2=512,
;  316          0408  4                    P3=.ISD[ISD$L_VBN]+.VPN-.ISD[ISD$V_VPN]);
;  317          0409  4                IF NOT .STATUS THEN RETURN .STATUS;
;  318          0410  4                IF NOT .IOSB THEN RETURN .IOSB<0,16>;
```

```
;  319          0411  4              CH$MOVE(.LEN, IMGBUF[.SVA-(.VPN^9)], .SVA);
;  320          0412  3              END;
;  321          0413  2          END;
;  322          0414  2
;  323          0415  2
;  324          0416  2      SS$_NORMAL
;  325          0417  1      END;


                                                        .TITLE  COB$$RESTVA
                                                        .IDENT  \1-005\

                                                        .EXTRN  SYS$QIOW

                                                        .PSECT  _COB$CODE,NOWRT,  SHR,  PIC,2

                                        0FFC 00000      .ENTRY  COB$$RESTVA, Save R2,R3,R4,R5,R6,R7,R8,R9,- ; 0183
                                                                R10,R11
                                5E      FBE8    CE  9E 00002    MOVAB   -1048(SP), SP
                                03              6C  91 00007    CMPB    (AP), #3                            ; 0264
                                                06  1E 0000A    BGEQU   1$
                                50      0114    8F  3C 0000C    MOVZWL  #276, R0
                                                    04 00011    RET
                             08 AE              01  D0 00012 1$:MOVL    #1, IHDVBN                          ; 0271
                                                7E  7C 00016    CLRQ    -(SP)                               ; 0279
                                7E              01  7D 00018    MOVQ    #1, -(SP)
                                7E      0200    8F  3C 0001B    MOVZWL  #512, -(SP)
                                        FE00    CD  9F 00020    PUSHAB  IHDBUF
                                                7E  7C 00024    CLRQ    -(SP)
                                  30 AE         9F  00026       PUSHAB  IOSB
                                  31            DD  00029       PUSHL   #49
                                  08 AC         DD  0002B       PUSHL   CHAN
                                  0C AC         DD  0002E       PUSHL   EFN
                    00000000G     00            0C  FB 00031    CALLS   #12, SYS$QIOW
                                  6E            50  D0 00038    MOVL    R0, STATUS
                                  03            6E  E8 0003B    BLBS    STATUS, 2$                          ; 0280
                                            012B  31 0003E      BRW     15$
                                  03      10 AE     E8 00041 2$:BLBS    IOSB, 3$                            ; 0281
                                            012C  31 00045      BRW     17$
                               0C AE    FE10    CD  9A 00048 3$:MOVZBL  IHDBUF+16, MAXIHDVBN                ; 0289
                            01FE 8F     FE00    CD  B1 0004E    CMPW    IHDBUF, #510                        ; 0290
                                                53  1A 00055    BGTRU   5$
                                  50    FE00    CD  9E 00057    MOVAB   IHDBUF, R0                          ; 0291
                                  56    FE00    CD  3C 0005C    MOVZWL  IHDBUF, ISD
                                  56            50  C0 00061    ADDL2   R0, ISD
                                  5A      04    AC  D0 00064    MOVL    INADR, R10                          ; 0298
04 AE       05 AA                 17            01  EF 00068    EXTZV   #1, #23, 5(R10), 4(SP)
   58          6A                 17            09  EF 0006F    EXTZV   #9, #23, (R10), VPN
                                            010E  31 00074      BRW     20$
   50       04 A6                 15            00  EF 00077 4$:EXTZV   #0, #21, 4(ISD), R0                ; 0313
                                  51      02    A6  3C 0007D    MOVZWL  2(ISD), R1
                                  50            51  C0 00081    ADDL2   R1, R0
                                  50            58  D1 00084    CMPL    VPN, R0
                                                64  19 00087    BLSS    9$
                                  50            66  3C 00089    MOVZWL  (ISD), R0                           ; 0320
                                  56            50  C0 0008C    ADDL2   R0, ISD
                                  50      FE    AD  9E 0008F    MOVAB   IHDBUF+510, R0                      ; 0321
```

```
                                 50          56  D1 00093        CMPL    ISD, R0                       ;
                                             12  1A 00096        BGTRU   5$                            ;
                                             66  B5 00098        TSTW    (ISD)                         ; 0329
                                             5D  13 0009A        BEQL    10$                           ;
                         FFFF    8F          66  B1 0009C        CMPW    (ISD), #65535                 ; 0339
                                             D4  12 000A1        BNEQ    4$                            ;
                           0C    AE     08   AE  D1 000A3        CMPL    IHDVBN, MAXIHDVBN             ; 0342
                                             05  12 000A8        BNEQ    6$                            ;
                                 50     44   8F  9A 000AA  5$:   MOVZBL  #68, R0                       ;
                                                 04 000AE        RET                                   ;
                                        08   AE  D6 000AF  6$:   INCL    IHDVBN                        ; 0343
                                             7E  7C 000B2        CLRQ    -(SP)                         ; 0351
                                             7E  D4 000B4        CLRL    -(SP)                         ;
                                        14   AE  DD 000B6        PUSHL   IHDVBN                        ;
                                 7E   0200   8F  3C 000B9        MOVZWL  #512, -(SP)                   ;
                                      FE00   CD  9F 000BE        PUSHAB  IHDBUF                        ;
                                             7E  7C 000C2        CLRQ    -(SP)                         ;
                                        30   AE  9F 000C4        PUSHAB  IOSB                          ;
                                             31  DD 000C7        PUSHL   #49                           ;
                                        08   AC  DD 000C9        PUSHL   CHAN                          ;
                                        0C   AC  DD 000CC        PUSHL   EFN                           ;
                    00000000G    00          0C  FB 000CF        CALLS   #12, SYS$QIOW                 ;
                                 6E          50  D0 000D6        MOVL    R0, STATUS                    ;
                                 03          6E  E8 000D9        BLBS    STATUS, 7$                    ; 0352
                                           008D  31 000DC        BRW     15$                           ;
                                 03     10   AE  E8 000DF  7$:   BLBS    IOSB, 8$                      ; 0353
                                           008E  31 000E3        BRW     17$                           ;
                                 56   FE00   CD  9E 000E6  8$:   MOVAB   IHDBUF, ISD                   ; 0354
                                             8A  11 000EB        BRB     4$                            ; 0313
      58         04    A6        15          00  ED 000ED  9$:   CMPZV   #0, #21, 4(ISD), VPN          ; 0365
                                             04  14 000F3        BGTR    10$                           ;
                                 06     08   A6  E9 000F5        BLBC    8(ISD), 11$                   ; 0374
                                 50   01EC   8F  3C 000F9  10$:  MOVZWL  #492, R0                      ; 0376
                                                 04 000FE        RET                                   ;
                       5B        58          09  78 000FF  11$:  ASHL    #9, VPN, R11                  ; 0383
                                 59          5B  D0 00103        MOVL    R11, R9                       ;
                                 6A          59  D1 00106        CMPL    R9, (R10)                     ;
                                             03  1E 00109        BGEQU   12$                           ;
                                 59          6A  D0 0010B        MOVL    (R10), R9                     ;
                                 57          59  D0 0010E  12$:  MOVL    R9, SVA                       ;
                                 59   01FF   CB  9E 00111        MOVAB   511(R11), R9                  ; 0384
                           04    AA          59  D1 00116        CMPL    R9, 4(R10)                    ;
                                             04  1B 0011A        BLEQU   13$                           ;
                                 59     04   AA  D0 0011C        MOVL    4(R10), R9                    ;
                                 59          57  C2 00120  13$:  SUBL2   SVA, R9                       ;
                                             59  D6 00123        INCL    LEN                           ;
                       59  08    A6          03  E1 00125        BBC     #3, 8(ISD), 19$               ; 0394
                       08  08    A6          02  E1 0012A        BBC     #2, 8(ISD), 14$               ; 0396
      59               00        6E          00  2C 0012F        MOVC5   #0, (SP), #0, LEN, (SVA)      ; 0398
                                             67     00134                                              ;
                                             4C  11 00135        BRB     19$                           ;
                                             7E  7C 00137  14$:  CLRQ    -(SP)                         ; 0408
                                             7E  D4 00139        CLRL    -(SP)                         ;
                       50        58     0C   A6  C1 0013B        ADDL3   12(ISD), VPN, R0              ;
      51         04    A6        15          00  EF 00140        EXTZV   #0, #21, 4(ISD), R1           ;
                       7E        50          51  C3 00146        SUBL3   R1, R0, -(SP)                 ;
                                 7E   0200   8F  3C 0014A        MOVZWL  #512, -(SP)                   ;
```

```
                          2C    AE  9F 0014F        PUSHAB  IMGBUF                        ;
                                7E  7C 00152        CLRQ    -(SP)                         ;
                          30    AE  9F 00154        PUSHAB  IOSB                          ;
                                31  DD 00157        PUSHL   #49                           ;
                          08    AC  DD 00159        PUSHL   CHAN                          ;
                          0C    AC  DD 0015C        PUSHL   EFN                           ;
              00000000G  00     0C  FB 0015F        CALLS   #12, SYS$QIOW                 ;
                         6E     50  D0 00166        MOVL    R0, STATUS                    ;
                         04     6E  E8 00169        BLBS    STATUS, 16$                   ; 0409
                         50     6E  D0 0016C 15$:   MOVL    STATUS, R0                    ;
                                    04 0016F        RET                                   ;
                         05  10 AE  E8 00170 16$:   BLBS    IOSB, 18$                     ; 0410
                         50  10 AE  3C 00174 17$:   MOVZWL  IOSB, R0                      ;
                                    04 00178        RET                                   ;
           50            57     5B  C3 00179 18$:   SUBL3   R11, SVA, R0                  ; 0411
           67       18 AE40     59  28 0017D        MOVC3   LEN, IMGBUF[R0], (SVA)        ;
                                58  D6 00183 19$:   INCL    VPN                           ; 0298
                   04    AE     58  D1 00185 20$:   CMPL    VPN, 4(SP)                    ;
                                03  1A 00189        BGTRU   21$                           ;
                              FEE9  31 0018B        BRW     4$                            ;
                         50     01  D0 0018E 21$:   MOVL    #1, R0                        ; 0417
                                    04 00191        RET                                   ;

; Routine Size:  402 bytes,    Routine Base:  _COB$CODE + 0000
```

```
;  327          0418  1 END
;  328          0419  0 ELUDOM



;                                   PSECT SUMMARY
;
;       Name                    Bytes                           Attributes
;
;  _COB$CODE                        402  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(2)


;                                 Library Statistics
;
;                                        -------- Symbols --------      Pages       Processing
;       File                             Total    Loaded   Percent      Mapped      Time
;
;  _$255$DUA28:[SYSLIB]LIB.L32;1          18619        17         0        1000        00:01.4



;                                   COMMAND QUALIFIERS

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

; Size:          402 code + 0 data bytes
; Run Time:        00:07.8
; Elapsed Time:    00:34.4
; Lines/CPU Min:    3231
; Lexemes/CPU-Min: 22465
; Memory Used:  148 pages
; Compilation Complete
```

COBRMSBLO
LIS

COBSPANC
LIS

COBSWITCH
LIS

COBRESTVA
LIS

CONVSHR
MAP

CONVCOMIO
LIS

COBSETSWI
LIS

CONVDEF
MDL

CONVLINK
REQ

COBSPANC2
LIS

CONV

CONVERT
MAP

CONVCALL
LIS

COBVECTOR
LIS

CONVERT
REQ

RECLAIM
MAP

ADDKEY
LIS